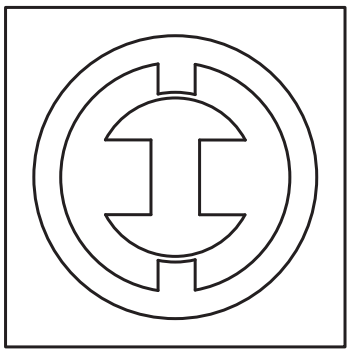
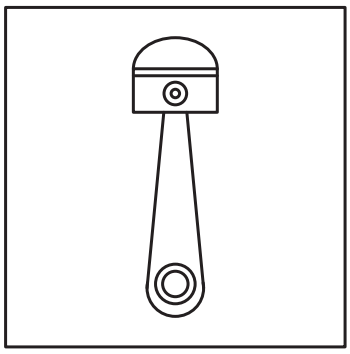
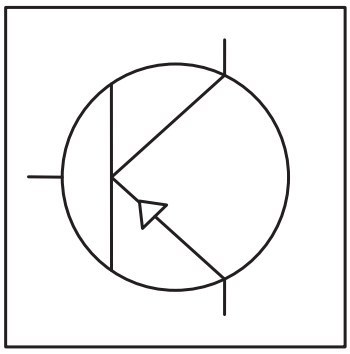
Monitoring Delayed Systems with Spatial Distribution in Presence of Realtime Requirements

Synchronization and Prediction

Bachelor Thesis
by Jan Maximilian Montenbruck

University Park, PA, January 2011

Scientific Supervision:

Prof. Dr.-Ing. Dieter Schramm
Prof. Dr. Sean Brennan
Dipl.-Ing. Gregor Hiesgen
Dipl.-Ing. Benjamin Hesse

Abstract

Keywords. Nonlinear Dynamics; Control Theory; Synchronizability; Stability; Model Predictive Control; Internal Model Principle; Smith Prediction; Delay Differential Equations; Transcendental Modal Analysis; Lambert W Function; Mori's Theorems; Block Matrix Analysis; Williamson Theorem; Elemental Symmetric Polynomials; Newton Identities; Lagrange Multipliers; Real Zero Bounds; Zero Row Sum; Nonlinear Convex Constraint Optimization; Delay Measurement; Fault Tolerance; Network Topology; Graph Theory

This thesis addresses the matter of monitoring delayed systems. The considered systems include nonlinear systems with linear input and output functions. Since the approach to the granting of a realtime monitoring is the utilization of a model, the theories of model predictive control and synchronization are introduced. Particularly, delays induced by network structures will be analyzed. The first approach to the monitoring problem is an alterated Smith Predictor with a plant model output instead of a reference function. This structure will be analyzed according to its modes and its stability criteria. While the Lambert W function is used for the transcendental modal analysis of the delay differential equations of the coupling terms, Mori's Theorems on the stability of delay differendital equations lay the groundwork for the stability criteria. For solving these criteria, the Williamson theorem is used to solve the block matrix representation, Newton's identities are used to find its roots in dependence of the systems input and output functions, and the method of Lagrange multipliers find the extremes in these roots. The second approach creates a new framework for the problem of model-plant synchronization based on a simple mechanical reference system. For the analytical solution to this issue, modern methods of nonlinear convex constraint optimization are applied. The results are then interpreted to posit a lemma on the form of the coupling matrices, that is then again proven by a modal analysis with the Lambert W function. In last instance, the problem of accurate delay measurement is investigated. While the IEEE 1588 protocol is implemented continuously, a method for its fault-tolerant interpretation is designed, as well. Therefore, principles of graph theory are discussed. In the last part of the thesis, crucial results of the synthesis are proven by an experimental setup based on coupled realtime simulation systems. Simple dynamical systems are numerically simulated on spatially distributed computing systems, so that criteria and algortithms can be tested and evaluated.

Acknowledgements

It is a pleasure to thank all the people that have made this thesis possible.
First of all I would like to thank my mother for her intelligent and loving education and my friends for their support, amongst them I would especially like to thank my best friend Pascal Matthess for calming me down when I got too ambitious.
I owe my deepest gratitude to my supervisors Prof. Dr.-Ing. Dieter Schramm and Prof. Dr. Eng. Sean Brennan. Thank you for the endless hours of reading, correcting, understanding, inspiring, and forcing me to rest. I am indebted to Dipl.-Ing. Benjamin Hesse and Dipl.-Ing. Gregor Hiesgen for the years of subject-specific mentoring and teaching and their final support of my thesis. I would like to thank Jesse Pentzer for the huge amount of time he spent helping me with english, engineering, and surviving the cold. I would also like to show my gratitude to all the graduate students at the Intelligent Vehicles and Systems group at the Pennsylvania State University and the chair of mechatronics at the University of Duisburg-Essen. Thank you Pramod Vemulapalli and Kshitij Jerath for reading my concoction and Philipp Mucha for everything LaTeX-related and years of studying together.
Thank you Prof. Dr.-Ing. Thorsten Brandt for recommending me to the Pennsylvania State University and thanks to the graduate program for the tuition waiver.
I am grateful for the generous scholarship of the Dr. Jürgen and Irmgard Ulderup foundation that made it possible for me to afford travelling and living for my stay in the USA. I want to emphasize at that point what great chance it is for a young engineering student to write his Bachelor thesis abroad at a prestigious university and with the support of so many skilled scientists. I also thank the German National Academic Foundation that supports me with a scholarship for some years and always provided a perfect network with financing and mentoring that also approached the Dr. Jürgen and Irmgard Ulderup foundation.
Another thankyou goes to the Mathworks, that finance the distributed systems program at the Pennsylvania State University, and that provided the best possible software framework possible for engineering together with Maple and their collaborative toolbox.

Contents

List of Figures

List of Tables

List of Algorithms

Notation

Variables are formatted italic. Well defined mathematic functions like rank, diag, or tr are formatted non-italic. The real and complex parts of variables are denoted by the Fraktur letters $\Re$ and $\Im$, respectively. Physical values are described by calligraphic letters like $\mathcal{C}$. Vectors are underlined and matrices are bold, as $\underline{c}$ and $\boldsymbol{C}$, respectively. The upper right corner of a variable generally denotes exponentials, but denotes the operators transpose and the adjoint in case of T and $*$, respectively. The lower right corner defines the system or the time domain that the variable refers to. A vector or matrix element is implied by the index of the lower right corner. The first and the second partial derivative are denoted by the nabla operator $\nabla(\cdot)$ and the upside-down nabla operator $\Delta(\cdot)$, respectively. Single vertical bars on both sides of a variable define determinants and double vertical bars on both sides norms of vector spaces. Bold white variables denote well-defined sets, as $\mathbb{C}$ all complex numbers, $\mathbb{R}$ all real numbers, and $\mathbb{Z}$ all integers. The hash symbol # is defined by the cardinality of a set. Time derivatives are represented by the number of dots on top of a variable. The denotation of variables is listed below.

A ..... set of arcs
$\boldsymbol{A}$ ..... state-matrix
B ..... set of bifurcations
$\boldsymbol{B}$ ..... input matrix
C ..... controller
$\boldsymbol{C}$ ..... output matrix
$\mathcal{C}$ ..... differential coefficient
$\mathcal{D}$ ..... differential coefficient
E ..... node
$\mathcal{E}$ ..... differential coefficient
$\mathcal{F}$ ..... differential coefficient
H ..... Hardy space
$\boldsymbol{I}$ ..... unit matrix
K ..... proportional gain
$\boldsymbol{K}$ ..... inner coupling matrix
L ..... Lyapunov exponent
L ..... set of leafs
$\boldsymbol{L}$ ..... oscillator coupling matrix
M ..... master
$\boldsymbol{M}$ ..... arbitrary matrix
$\boldsymbol{\mathcal{M}}$ ..... block matrix
N ..... number of networking systems
P ..... plant
R ..... reference signal
S ..... slave
T ..... tournament
U ..... input signal

$\boldsymbol{U}$ ..... unitary matrix
V ..... set of vertices
$\boldsymbol{V}$ ..... unitary matrix
W ..... walk
Y ..... output signal
a ..... polynomial coefficient
a ..... arc
b ..... bifurcation
c ..... coupling factor
d ..... disturbance
$d(\cdot,\cdot)$ ..... distance
$d(\cdot)$ ..... degree
e ..... eccentricity
$\underline{f}$ ..... state function
$\underline{g}$ ..... coupling function
$\underline{h}$ ..... input function
i ..... consecutive variable
j ..... complex number
j ..... consecutive variable
l ..... leaf
$l(\cdot)$ ..... walk length
m ..... tournament size
m ..... arbitrary matrix element
n ..... degree of freedom
p ..... port
s ..... frequency-proportional complex number
t ..... time
v ..... vertex
$\underline{x}$ ..... state-space vector of respective node
$\boldsymbol{\Gamma}$ ..... coupling configuration matrix
Δt_D ..... delay
Δt_O ..... offset
$\boldsymbol{\Xi}$ ..... singular value decomposition matrix
Σ ..... set of singular values
$\boldsymbol{\Psi}$ ..... incidence matrix
Ω ..... Lyapunov condition
β ..... Lagrange constraint
γ ..... Lagrange constraint
ϵ ..... iteration step
ζ ..... overall stability bound
$\underline{\zeta}$ ..... first variation of the all-state vector
$\underline{\theta}$ ..... limit
$\varkappa$ ..... clock drift
λ ..... eigenvalue
μ ..... matrix measure
$\underline{\xi}$ ..... first variation of the all-state vector
ρ ..... model
ρ ..... singular value
σ ..... set of eigenvalues
ς ..... stability bound
ϕ ..... Lagrange multiplier
$\underline{\chi}$ ..... all-state vector
$\overline{\psi}$ ..... incidence
ω ..... frequency
$\sim$ ..... approximation
$\mathbf{0}$ ..... zero matrix

1

Motivation

In essence, things are composed of networks. Nature, society, and mankind have intrinsic couplings and interactions among their subsystems. These elaborate structures comprise a natural functionality and elemental stability that artificial systems often lack. In spite of that, technical networks tend to be more intricate, spatially distributed, and heterogenous. These factors enhance delays and instabilities that have to be compensated by means of system design. These compensation methods are the focus of this thesis.
To be more precise, systems that work together need to be coordinated. When one of the systems tries to transfer a task to another one without the second system being prepared for this event, delays or instabilities may occur. A common reference or understanding between the systems enables them to communicate in an appropriate way. Synchronization can be the primary means of coordinating events. But even if it was possible to start the systems exactly at the same time, there could be drifts, delays, or different initial conditions between their respective dynamics, making it impossible to use this method without modification. Since networks and distributed systems become larger and more complex, the task in coordinating and controlling them increases in complexity, as well. This is especially apparent when different mechanical, electrical, and informatic components are joined, each of them introducing another dimension of delay, response time, latency or offset. In an ideal case, it may be possible to make all the systems run synchronously, or at least be able to predict how each of them will react, schedule their tasks, compensate delays, or know the delays. These and further tasks are the goals of control in distributed systems.

1.1 Problem Statement

In this thesis, realtime is defined as the delay treshold that is small enough to be ignored in the analysis of performance and stability. Realtime requirements are therefore performance criteria dependent on a response time defining that treshold. When a system with such requirements receives its response from a delayed subsystem, those requirements can hardly be met unless it is possible to reduce the delay up to the given treshold. When the delayed system additionally receives an input from the realtime system so that both systems are coupled in a loop it is even possible that unstable behaviour occurs. A schematic representation of such a coupling is depicted in Figure 1.1.

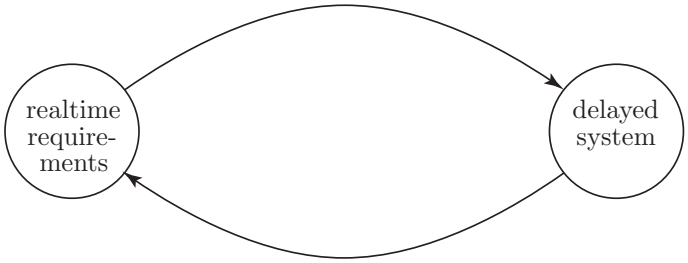


Figure 1.1: Schematic Problem Statement

Any system that is sufficiently distributed will comprise such delays due to transportation of matter, energy, or information. The aim in designing such systems is conceptualizing appropriate topologies, couplings, and controls to be able to grant prompt and stable system behavior. This thesis should provide criteria for optimizing control and coupling parameters within networked systems and algorithms for overall network design and delay measurement. Those procedures are intended to be as generic as possible. The developed methods should therefore be applicable to linear and nonlinear systems equally, where the degrees of freedom are variable. The mathematical results are proven and demonstrated by technical examples. This thesis approaches the stated problems by attempting to both, predict system responses, and synchronize the system with a model so that the model can be used to ensure the described realtime requirements. A schematic representation of this solution is depicted in Figure 1.2.

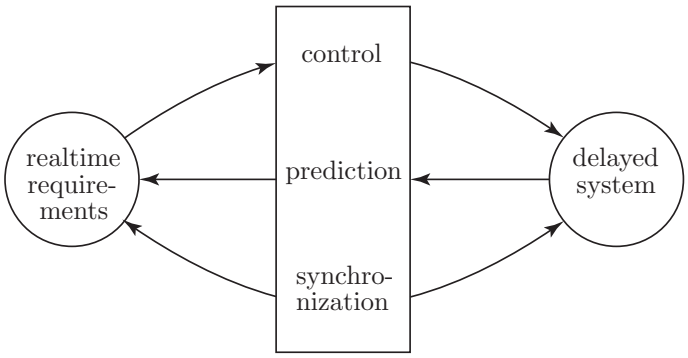


Figure 1.2: Schematic Problem Solution

A special case that will be discussed in this thesis is a delay that is induced by signal transmission in a network, as depicted in Figure 1.3. Those delays may vary due to physical phenomena, so that algorithms for an adequate delay measuring have to be developed. Those algorithms will usually be strongly dependent on network topology, so that optimal types of network architecture have to be designed.

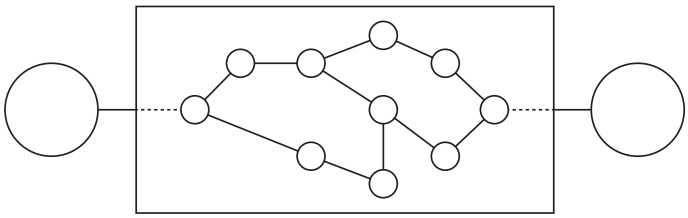


Figure 1.3: Signal Delay Induced by a Network Structure

1.2 Applications

As previously stated, systems with performance criteria dependent on a response time that is a function of transportational factors are examples for systems with realtime requirements. Most hardware-in-the-loop systems have such performance requirements. Components are tested most realistically when they are ran at their designated running speed. When a test sequence has to be run at such a speed, requiring state information from another component that is providing this information with a certain delay, the first component will run the whole test sequence using incorrect information. It would be desirable to decrease the delay to receive correct testing results. When human beings are elements of a testing process, they will notice, whenever a response, that they receive, is delayed, especially when this response is not delayed in the surrounding that the test is emulating. An optimal system would respond to any human input with a delay that is small enough not to affect futher inputs.
In fact, most monitoring systems of delayed processes can be optimized by such means. Such systems are often specified as human-machine interface or human control systems. Particularly, that means that the time between a critical event within the controlled system and the respective reaction by the human controller is the crucial performance criterion of such systems. The faster events are displayed to the monitoring system, the faster it is possible for the human controller to decide about further actions. Again, the system requires to reduce the signal delay to a point where controller actions are fast enough to compensate critical events.

1.3 Scientific Preamble

As described above, the main aim of this research is to overcome delay. In essence, a Smith[a] predictor will be used. The predictor is a model-based approach of automatically feeding a delayed process with adequate signals. Because the described problems require monitoring of the delayed process, this research will attempt to synchronize the states between a reference model and the delayed process. This will allow to monitor the behaviour of the model and therefore meet real-time requirements. The given structure of the delayed process, the reference model, and the Smith predictor will induce a coupling of systems so that coupling matrices and the theory of synchronizing dynamic systems will be studied. Furthermore, since the sensitivities of the principles used to delay uncertainties will be discovered, a method for precise delay measurement based on IEEE 1588 will be described that can additionally synchronize the inner clocks of each node to grant an optimal benefit of each computational step. To additionally enhance the fault-tolerance of the overall system, an adequate network topology will be characterized.
The final concept will allow for structuring an existing network by a special topology, coupling real-time models with delayed processes via Smith predictors, tuning controllers corresponding to synchronizability criteria to enhance promptness of monitoring response time, and measure delays between those systems.

[a] Otto J. M. Smith *1917†2009, American educator and electrical engineer

2

State of the Art

The phenomena of synchronization and prediction have been investigated in fields such as mathematics, physics, engineering, and informatics for years. This chapter tries to give an overview of historical and recent advances in these fields.

2.1 Synchronization

Synchronization is a field of science that has been mainly discussed in mathematics and physics, especially in the fields of nonlinear dynamics, chaotic systems, and neural networks. Most of the basic investigations on synchronization are based on Christiaan Huygens[a] discovery on the antiphase movement of coupled pendulums on ships in 1650 [9] and published in 1669 as *Instructions Concerning the Use of Pendulum-Watches for Finding the Longitude at Sea* [46]. Though Huygens gave suggestions for the reason of asynchronous movements, appropriate conditions for synchronous movement were yet missing. The first paper using the term synchronization in a meaning comparable to the usage within this thesis is *On Asynchronous Action* by N. Minorsky[b] [71]. Minorsky described phase differences between asynchronous oscillators and defined conditions that excitate or quench those. The latter can formally be called synchronization. Though not the major interest of this article, the asynchronous quenching is within already named an asymptotic phenomenon. Minorsky then concretized those results on synchronization and presented them at the International Symposium on Nonlinear Oscillations in 1961 as *O Sinkhronizatsii (engl.: On Synchronization).* This particular topic was not addressed again in literature, before I. Blekhman[c] wrote *The Problem of Synchronization of Dynamical Systems* in 1964. Blekhman used a small variational parameter substituting the difference between osciallations of technical systems and deduced that the damping in this parameter is the synchronizability condition. In his conclusions, he mentioned that the study of this parameter is the same as the study of the signs of the roots of a certain matrix using the Kronecker symbol. Though he did not investigate that fact exactly, this is yet the first paper stating this condition. This is especially notable because R. Nagaev denoted terms that are very similar to the coupling matrices used today in the next issue of the same journal with *The Synchronization of Nearly-Similar Dynamic Systems Close to Liapunov Systems* . Nagaev has since then published various papers in the field of synchronization [76, 78, 77]. Later, in 1974, A. Gurtovnik and I. Neimark[d] referred to the previous achievments of Nagaev and Blekhman in *On Synchronization of Dynamic Systems* [37]. Though this paper introduced the term of synchronization manifold and extended previous definitions to n dimensions, it did not introduce new findings.

[a] Christiaan Huygens *1629†1695, Dutch mathematician, astronomer, physicist, horologist and novelist
[b] Nicolas Minorsky *1885†1970, Russian physicist
[c] Ilya I. Blekhman *1928, Russian physicist and mathematician
[d] Yurii Isaakovich Neimark *1920, Russian physicist and mathematician

Nevertheless, the paper reintroduced the topic so that applications of the given techniques became a focus of research in the fields of biology and chemistry around 1975 [105, 103, 69, 104]. Those papers introduced terms such as spatial extent, diffuse couplings, uncertainties, and other problems in the synchronization research of today. It was a bit later, mainly around 1984, that there were numerous papers published about the synchronization of clocks [99, 56, 65, 68, 40, 64]. Although clocks can be modelled as oscillators from the system theory point of view, this research focused on a purely algorithmic point of view. From that point onwards it was again the field of physics that started todays interest in synchronization. It started with T. Carroll, L. Pecora, and F. Rachford, who published several papers on chaotic systems in 1989 [13, 15, 14] and then *Synchronization in Chaotic Systems* in 1990 [81]. The latter has been cited numerous times. Though this paper was outstanding in the field of physics, it was no use to the problems discussed in this thesis. It was a paper called *Synchronous Chaos in Coupled Osciallator Systems* that gave the notation of coupling equations used today. Between 1990 and 2002 there were numerous papers describing experimental results of special cases of synchronization of nonlinear or chaotic systems [89, 50, 91, 90, 51, 11, 2], but none of them gave such a generic and precise approach as M. Barahona gave in 2002 in *Synchronization in Small-World Systems* [7]. He was the first to give bounds for synchronizability and optimization criteria to enhance the damping in the difference between state vectors. This article laid the foundation for recent research in the synchronization of networks [3, 61, 80, 119] and nonlinear coupling types, especially delayed and nonsymmetric ones [119, 102, 22, 116, 62, 47, 60, 115]. These methods are useful in technical control systems.

2.2 Prediction

Seeing the future has always been an interest of mankind. As a result, natural sciences have tried to improve the knowledge about prediction of predetermined causal events. One of the first sciences yielding considerable advances in this field of research was meteorology. Though Otto von Guericke[e], 1660, was the first one to predict severe weather [107] following decreasing barometric pressure in *Experimenta nova Magdeburgica de vacuo spatio (engl.: New Magdeburg Experiments on Vacuum)*, it was Sir Gilbert Walker[f] who first tried to deduce mathematical models based on measurements in *Seasonal Weather and its Prediction* in 1933 [94]. The first acknowledgement prediction in mathematics was by A. Copeland in *Predictions and Probabilities* in 1936, where he tried to give stochastical conditions for successful predictions [20]. Within the following years, papers about prediction as a statistical matter were published [45, 112], and 1947 brought the first publication using a recursive method for signal predictive filter design in *The Wiener Root Mean Square Error Criterion in Filter Design and Prediction* by N. Levinson[g] [59]. Thirteen years later, R. Kalman[h] published his well known paper *A New Approach to Linear Filtering and Prediction Problems* [48], using a similar approach to Levinson. Before Kalman, in 1957, Otto Smith developed his groundbreaking non-statistical prediction principle in *Closer Control of Loops with Dead Time* [97] and refined it in *A Controller to Overcome Dead Time* in 1959 [98]. It was the first principle that could apply existing knowledge about a process and combine it in a mathematical model to predict the process behaviour. The Smith predictor also implemented the internal model principle, even though that principle was not discovered until 1974 [28, 29, 30] by B. Francis[i] in *Synthesis of Multivariable Regulators* , later refined in *The Internal Model Principle for Linear Multivariable Regulators* and generalized in*The Internal Model Princinple of Control Theory.* The next decades were spent on self-tuning / adaptive control, estimation, optimization and uncertainties [17, 18] in general, but particular emphasis was placed on model-based and model-predictive control theory. Within the 1990s, a noteable number of papers were published on alterations of the known Smith predictor [92, 58, 54, 57], trying to make it stable in the presence of uncertainties in model and delay. In recent years, several articles were published on the field of research that is primarily relevant to this thesis: Using models for enhancing real-time capabilities

[e] Otto von Guericke *1602†1686, German politician, jurisprudent, physicist, and veterinarian
[f] Sir Gilbert Thomas Walker *1868†1958, British physicist and statistician
[g] Norman Levinson *1912†1975, American mathematician
[h] Rudolf Emil Kalman *1930, Ungarian mathematician
[i] Bruce Allen Francis *1947, Canadian mechanical and electrical engineer

of delayed applications. One of the major applications in this area is spatially distributed operating systems [34, 75, 96, 95, 110, 16], in which the aim is to synchronize the movements of a robot and an operator. A new principle of integrating models into real-time processes was also found in a method called dynamic substructuring [55, 108]. This method introduces delays due to the coupling of slow, real systems with fast, simulated systems. The given papers investigate Hopf[j] bifurcations of one-dimensional delayed couplings. Distributed real-time hardware-in-the-loop test beds, which are similar to that method, are facing problems of predictive control due to signal delays [26, 27, 25]. In this context, the term transparency is used, to describe the synchronization error between individual nodes.

[j] Eberhard Frederich Ferdinand Hopf *1902†1983, Austrian mathematician and astronomer

3

Principles

This chapter will explore the three very basic principles that are utilized in the following chapters. As described in 1.3, knowledge of the synchronizability of coupled dynamic systems, predictive control, and delay as well as offset measurement will be necessary. These fundamentals will be explained in the given order. Subsequently, some known problems of the given principles will be named and discussed.

3.1 Synchronization of Coupled Systems

Several sciences have been addressing the definition of time. In a mathematical sense, time is a one-dimensional, oriented, Euclidean[a] space [93]. The term synchronous is less discussed, but is no less important. In a common sense, the understanding of synchronous behavior is that N systems show the same behavior at the same time. In an engineering sense, the understanding of synchronous behavior is that these N systems have equal state vectors at the same time [82, 62].

Definition 3.1. *N systems are called synchronized, when*

$$\left\|\underline{x}_i - \underline{x}_j\right\| = 0 \,\forall\, i, j = 1, 2, \cdots, N \tag{3.1}$$

where $\underline{x}_i = [x_{i_1} \cdots x_{i_n}] \in \mathbb{R}^n$ is the state vector of system i with n degrees of freedom, and N is the number of identical interacting systems. The dynamics of these states are assumed to be in the nonlinear form of

$$\dot{\underline{x}}_i = \underline{f}_i\left(\underline{x}_i\right) + \underline{h}_i\left(\underline{u}_i\right) \;\forall\, i = 1, 2, \cdots, N \tag{3.2}$$

where $\dot{\underline{x}}_i = \frac{d\underline{x}_i}{dt} \in \mathbb{R}^n$ is the time-derivative of $\underline{x}_i$, $\underline{u}_i \in \mathbb{R}^n$ is the input vector of system i and $\underline{f}_i : \mathbb{R}^n \to \mathbb{R}^n$, $\underline{h}_i : \mathbb{R}^n \to \mathbb{R}^n$ are vector-valued functions expressing the dependency of $\dot{\underline{x}}_i$ on $\underline{x}_i$ and $\underline{u}_i$, and that may as well be functions of time. Now, in order for states to be synchronized, each state has to be made a function of the other states within the network [62], so that a coupling is induced. In general, the inputs of the systems can then be described as a function of the states of other systems.

$$\underline{u}_i = \underline{g}_i\left(\underline{x}_j, \cdots, \underline{x}_N\right) \;\forall\, j \neq i \tag{3.3}$$

It is therefore possible to describe the dynamical behaviour of each system as a function of the states of all systems within the network. Assuming $\underline{h}_i$ to be a linear function $\boldsymbol{K}_i$, the system dynamics can be written as

$$\dot{\underline{x}}_i = \underline{f}_i\left(\underline{x}_i\right) + \boldsymbol{K}_i \underline{g}_i\left(\underline{x}_j, \cdots, \underline{x}_N\right) \;\forall\, j \neq i \tag{3.4}$$

[a] Euclid of Alexandria fl. 300BC, Greek mathematician

Hereafter it is assumed that control systems couplings are induced by summations, subtractions, and gains, and thus the function $\underline{g}_i$ has to be a linear combination resulting from these operations. These linear operations are expressed by a factor γ_{ij} [67] so that $\underline{g}_i = \sum_{j=1}^{N} \gamma_{ij} \underline{x}_j$ holds. In addition, it is well known [116] that synchronization can only be achieved among systems with identical structure, parameters, and states. Therefore, it is required that $\underline{f}_i = \underline{f}_j \; \forall \, i, j = 1, 2, \cdots N$ and $\boldsymbol{K}_i = \boldsymbol{K}_j \; \forall \, i, j = 1, 2, \cdots N$ hold. Under these assumptions (3.4) can be rewritten as

$$\dot{\underline{x}}_i = \underline{f}\left(\underline{x}_i\right) + \sum_{j=1}^{N} \gamma_{ij} \boldsymbol{K} \, \underline{x}_j \; \forall \, i, j = 1, 2, \cdots, N \tag{3.5}$$

where $\boldsymbol{K}$ is called the inner coupling matrix and all γ_{ij} compose a matrix $\boldsymbol{\Gamma} = [\gamma_{ij}] \in \mathbb{R}^{N \times N}$ that is called the coupling configuration matrix [116] which is a very compact expression of the topology of the described network.

Example. Figure 3.1 depicts three typical examples for possible cases of couplings that occur in control systems. The picture shows a summation point with positive and nonpositive elements and a gain.

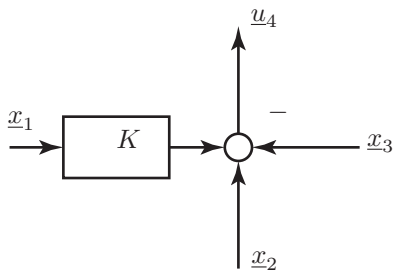


Figure 3.1: Couplings in Control Systems

In this case, the coupling is unidirectional for simplicity. Obviously, the input of system 4 is given by $\underline{u}_4 = K\underline{x}_1 + \underline{x}_2 - \underline{x}_3$, so that $\gamma_{41} = K$, $\gamma_{42} = 1$ and $\gamma_{43} = -1$. The coupling configuration matrix is therefore defined by

$$\boldsymbol{\Gamma} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ K & 1 & -1 & 0 \end{bmatrix} \tag{3.6}$$

The coupling configuration matrix is only a substitution, so that all the following methods can be applied generally without its specific definition, likewise. Using this notation, the steps are easier to illustrate and apply. The main purpose of the coupling configuration matrix is the notation [115] of (3.5) as

$$\dot{\underline{\chi}} = \underline{F}\left(\underline{\chi}\right) + \left[\boldsymbol{\Gamma} \otimes \boldsymbol{K}\right] \underline{\chi} \tag{3.7}$$

which is an algebraic representation of the whole network, where $\underline{\chi} = \left[\underline{x}_1 \underline{x}_2 \cdots \underline{x}_N\right]^T \in \mathbb{R}^{Nn}$ and $\underline{F}\left(\underline{\chi}\right) = \left[\underline{f}\left(\underline{x}_1\right) \underline{f}\left(\underline{x}_2\right) \cdots \underline{f}\left(\underline{x}_N\right)\right]^T$.

Definition 3.2. $\underline{\chi} = \left[\underline{x}_1, \underline{x}_2, \cdots, \underline{x}_N\right]^T \in \mathbb{R}^{Nn}$ *is the all-state vector of* N *nodes*

For the analysis of the nonlinear dynamics of the network, it is possible to calculate the first variation [101] of $\underline{\chi}$ and call it $\underline{\xi}$.

Definition 3.3. *The first variation* $\underline{\xi}$ *of* $\underline{\chi}$ *is the limit in* θ *of*

$$\underline{\xi} = \int \left. \frac{d\dot{\underline{\chi}}\left(\underline{\chi} + \theta \underline{p}\right)}{d\theta} \right|_{\theta=0} dt \tag{3.8}$$

with $\underline{\chi}$ *all solutions to* $\dot{\underline{\chi}}$ *and* $\underline{p}$ *a specific solution.*

By this definition and the substitution of the special case $\underline{p} = \underline{\xi}$, the dynamical behaviour of $\dot{\underline{\chi}}$ may be rewritten using $\underline{\xi}$ so that the network representation is denoted as

$$\dot{\underline{\xi}} = \left[\boldsymbol{I}^N \otimes \frac{\partial \underline{f}}{\partial \underline{x}} + \boldsymbol{\Gamma} \otimes \frac{\partial \underline{h}}{\partial \underline{x}} \right] \underline{\xi} \tag{3.9}$$

or, in a shorter form,

$$\dot{\underline{\xi}} = \left[\boldsymbol{I}^N \otimes \boldsymbol{\nabla} \left(\underline{f} \right) + \boldsymbol{\Gamma} \otimes \boldsymbol{K} \right] \underline{\xi} \tag{3.10}$$

where $\boldsymbol{I}^N = \text{diag}\{1, 1, \cdots, 1\} \in \mathbb{R}^N$. The eigenvectors of $\boldsymbol{\Gamma}$ now represent the modes of system $\underline{\xi}$, that is therefore a combination of modes of the networking systems. Since a synchronized behaviour is demanded, the matrix should have at least one eigenvector $\begin{bmatrix} 1_1 & \cdots & 1_N \end{bmatrix}$, that represents uniform movement of all systems. This eigenvector has to be the eigenvector of a purely imaginary latent root, so that the synchronized movement is not damped out, and the other movements have to be stable so that they do not dominate the synchronous movement. These criteria enable synchronizability.

Example. The system in Figure 3.2 depicts a simple mechanical system that has modes that satisfy synchronizability criteria.

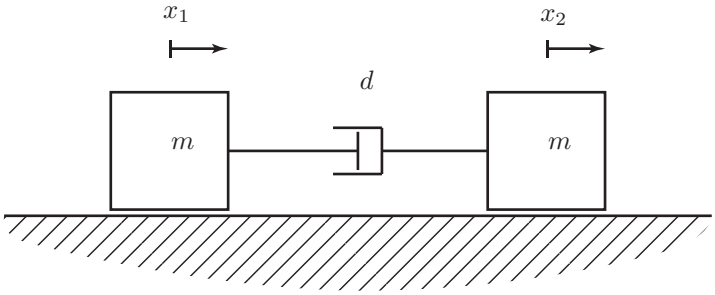


Figure 3.2: Typical Synchronizable System

For simplicity, let $m = d = 1$. Then the dynamic equations of the masses are $\dot{x}_1 + x_1 - x_2 = 0$ and $\dot{x}_2 + x_2 - x_1 = 0$. Substituting $\underline{x} = [x_1 \; x_2]^T$, the differential equation $\dot{\underline{x}} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \underline{x} = \boldsymbol{A}\underline{x}$ holds. Applying the similarity transform $\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V} = \boldsymbol{D}$, the modal matrix $\boldsymbol{V}$ yields the synchronous mode $[1 \; 1]^T$ corresponding to the latent root 0 and the asynchronous mode $[-1 \; 1]^T$ with the respective latent root -2. This example is very intuitive because it is physically that the system damps asynchronous movements and synchronous movements dominate.

Lemma 3.1. *System $\underline{\xi}$ is said to have at least one synchronized state when at least one eigenvector of $\boldsymbol{\Gamma}$ has the form $\begin{bmatrix} 1_1 & \cdots & 1_N \end{bmatrix}$*

Lemma 3.2. *System $\underline{\xi}$ is said to be synchronizable if the synchronized state is the eigenvector of a purely imaginary latent root of $\boldsymbol{\Gamma}$ and all other eigenvalues have negative real parts.*

All other eigenvalues should be negative [85] and as negative as possible [63] to have a strong damping between the systems so that synchronization can be achieved as soon as possible. Figure 3.3 depicts the desired set of eigenvalues in the s-plane.

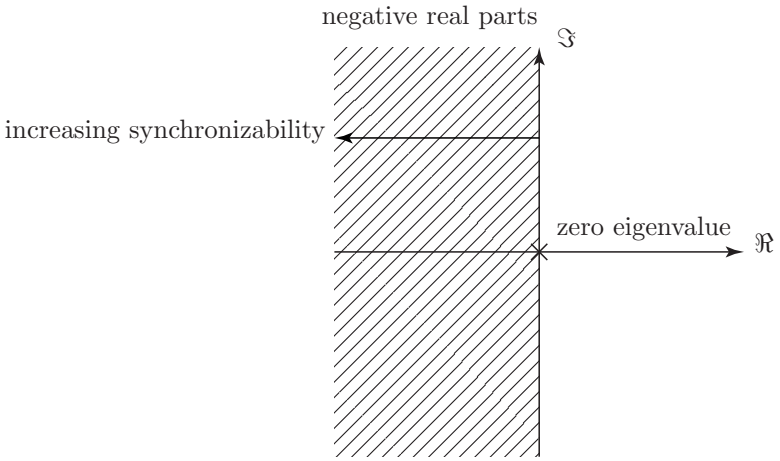


Figure 3.3: Desired Eigenvalue Properties of Coupling Matrix

That is why the eigenratio is a good measure for the synchronizability [7]. Now, if there is at least one synchronous state, there is yet no information about its stability and determinism. It is therefore necessary to compute the set of Lyapunov[b] exponents $L\left(\underline{\xi}\right)$ [111] .

Definition 3.4. *The set of Lyapunov exponents* $L = (L_1, \cdots, L_{Nn})$ *of* $\underline{\chi}$ *is the set of elements of* $\underline{L} = [L_1 \cdots L_{Nn}]^T$ *satisfiying*

$$\underline{\xi} \propto e^{\underline{L}t}\underline{\xi}_0 \tag{3.11}$$

It can be understood as an eigenvalue of the perturbation. Positive Lyapunov exponents represent orbital divergence and low predictability due to chaos, whereas negative Lyapunov exponents represent orbital convergence, high predictability, and decay of pertubations [44], each with respect to initial values. The difference is shown in Figure 3.4.

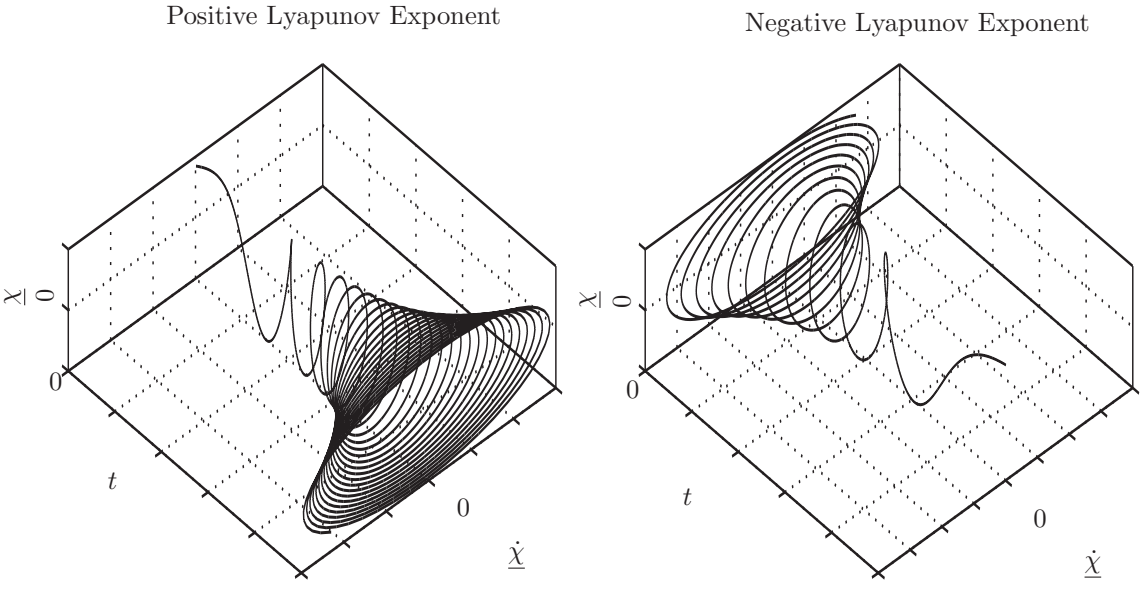


Figure 3.4: Effects of the Signs of Lyapunov Exponents

These properties show that a negative exponent is desirable. Since the first variation of $\underline{\chi}$ has already been computed, the Lyapunov exponents are easy to compute [22] by finding

$$L = \sigma\left(\boldsymbol{\nabla}\left(\underline{f}\right) + \Omega\boldsymbol{K}\right) \tag{3.12}$$

[b] Aleksandr Mikhailovich Lyapunov *1857†1918, Russian mathematician and physicist

where σ is the set of latent roots and Ω a gain that scales the coupling in the same way as the latent roots of $\mathbf{\Gamma}$.

Definition 3.5. *The set σ of a matrix $\boldsymbol{M}$ is the set of solutions to $|\boldsymbol{M} - \boldsymbol{I}^n\lambda| = 0 \; \forall \; \boldsymbol{M} \in \mathbb{R}^{n\times n}$.*

Hence, the the Lyapunov exponent is a function of Ω, which represents the latent roots of $\mathbf{\Gamma}$. Both conditions bound the stable synchronized region.

Lemma 3.3. *A System $\underline{\chi}$ has a conditionally stable synchronized region if its coupling configuration matrix $\mathbf{\Gamma}$ satisfies*

$$\sigma\left(\mathbf{\Gamma}\right) \in \Omega \; \forall \; \max\left(L\right) < 0 \tag{3.13}$$

By that, the given coupling coefficients correspond to a stable trajectory in the state-space, because substitution into (3.12) for Ω induces a negative Lyapunov exponent.

3.2 Model Based Control of Systems with Delay

When attempting to design a control, it is required to first program an adequate mathematical model of the system. The system and control parameters can then be investigated on the model. If a proper model of the system already exists, it may be possible to use it for the control itself, not only for its design. This possibility is a crucial topic in Internal Model Control [66]. Though a regular approach to control design may be sufficient to ensure the stability of the closed loop, it does not have to ensure its stability for any uncertainties in the entire model or in the disturbances that occur [43]. Closed loop synthesis using an internal model can be designed to have these characteristics. In general, any element of the closed loop, whether it is the plant, disturbance, or reference, that is known and embedded into the controller as a proper model, enhances the stability of the closed loop in the presence of uncertainties or variations. There is also a class of Internal Model Controls that use a model of the delay of the plant, and perhaps the best known of these is the Smith Predictor. As described above, the basic idea of this control is to use a model of the plant and a model of its delay for the closed loop. To develop the theory of the Smith Predictor, transfer functions are utilized as system representation.

Definition 3.6. *A transfer function of a system is the ratio between its output and its input.*

Let $P : \mathbb{C} \to \mathbb{C}$ be the transfer function of the plant containing a frequency-proportional phase difference $e^{-s\Delta t_D} : \mathbb{C} \to \mathbb{C}$ where $s = j\omega \in \mathbb{C}$ is a frequency-proportional variable with $\omega \in \mathbb{R}$ the frequency, and $j = \sqrt{-1}$, Δt_D is called delay, and $\rho = \frac{P}{e^{-s\Delta t_D}} : \mathbb{C} \to \mathbb{C}$ the transfer function of the model.

Definition 3.7. *A transfer function is said to be delayed when its phase difference is frequency-proportional.*

Then it is possible to measure the output of ρ before the output of P. So if there is a controller $C : \mathbb{C} \to \mathbb{C}$ providing the argument for the plant to be controlled, it is possible to give that argument to a mathematic model of ρ and to P at the same time [109], as depicted in Figure 3.5.

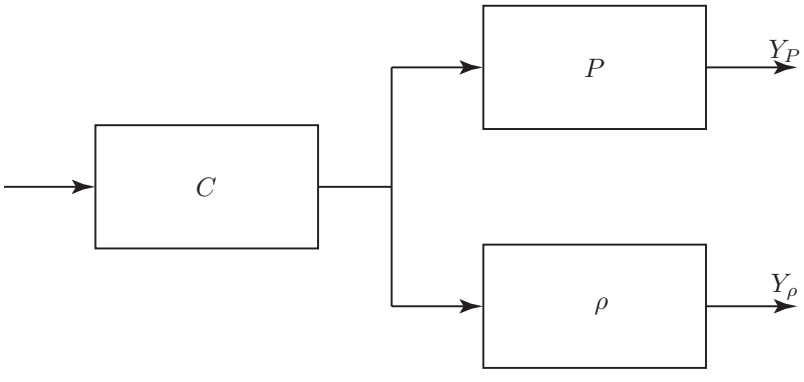


Figure 3.5: Principle of Prediction

Y_ρ is the output of ρ, and Y_P is the output of P. Then $Y_\rho e^{-s\Delta t_D} = Y_P$, so that those signals are identical at different times and the result is the basic principle of prediction. To make this more clear, it is easy to transpose it to the time-domain as

$$Y_\rho = \int_\infty^0 e^{-st} y_\rho dt = \int_\infty^0 e^{-st} y_P \left(t + \Delta t_D\right) dt = Y_P e^{s\Delta t_D} \tag{3.14}$$

It is now possible to close this open loop as if ρ was the plant to be controlled, like Figure 3.6 suggests:

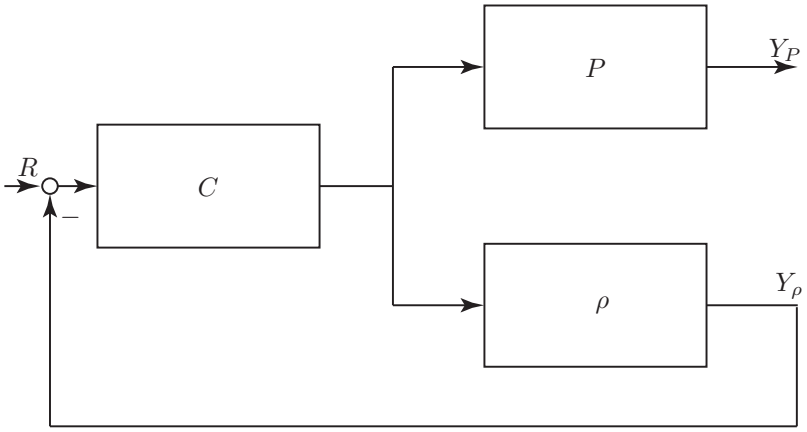


Figure 3.6: Principle of Model Predictive Control

Here R is the reference, that is the desired output signal of P. This closed loop now controls P as if it had no delay, only that P passes on the desired results with delay, but that fact no longer affects the stability of the closed loop with C and ρ. The transfer functions of the system depicted are

$$\frac{Y_\rho}{R} = \frac{C\rho}{1 + C\rho} \tag{3.15}$$

and

$$\frac{Y_P}{R} = PC\left(1 - \frac{C\rho}{1 + C\rho}\right) \tag{3.16}$$

It is plain to see that (3.15) has the same structure as a classical control loop whereas (3.16) is a typical steering or open-loop system [53]. This may already explain one of the crucial properties of prediction: It only works on stable plants because unless there is a response of the system that is used in any way to create a proper input, it is not possible to know whether the given inputs are suitable or not. Though, for the moment, the problem of controller design is now linear and easier to stabilize and that is what is called model predictive control [35]. Now if there was a disturbance in P using the control of Figure 3.6, C would never notice it and the plant could even collapse without any reaction of the control [31]. This is to be expected, because disturbances would only affect the real P and not the model ρ. Hence, it is possible to let the output of C be both the argument of a model of P and the argument of P with disturbances. This setup is shown in Figure 3.7, and is the basic principle of disturbance identification [87].

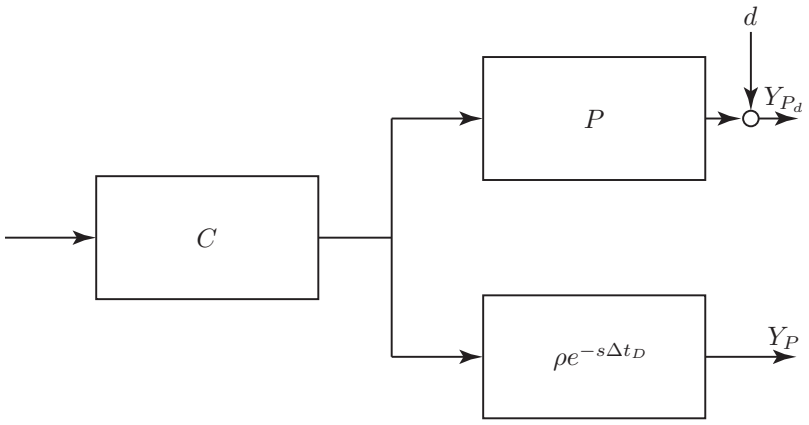


Figure 3.7: Principle of Disturbance Identification

Here d is the disturbance, Y_{P_d} is the output of the real system and Y_P is the output of the model. The outputs of the system are

$$Y_P = Y_{P_d} - d \tag{3.17}$$

and

$$Y_{P_d} = Y_P + d \tag{3.18}$$

The difference between their output signals would therefore be a representation of the disturbance and could be used to regulate the system input in such a way that the disturbance is identified and regulated. For most cases, in which disturbances are influencing the plant in a negative way, that means damping their effect out.
If this difference is used to close the loop, the result is the basic principle of the Internal Model Control, that is shown completely in Figure 3.8.

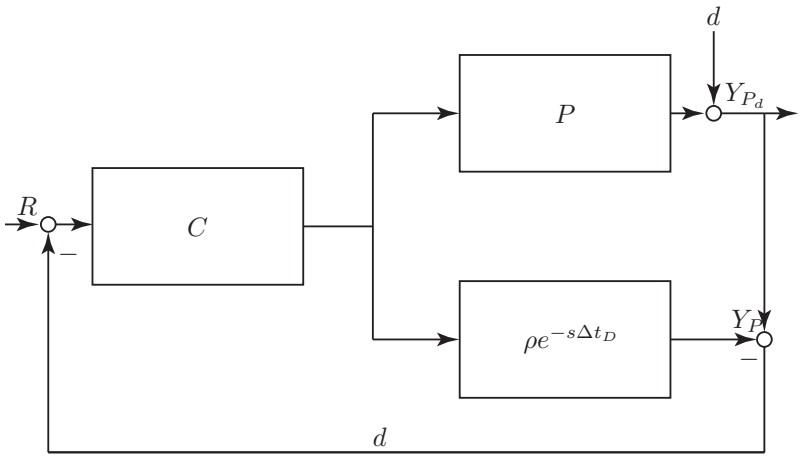


Figure 3.8: Principle of Internal Model Control

This impressive principle is able to identify disturbances due to its transfer functions

$$\frac{Y_{P_d}}{R} = PC \;\forall\; d = 0 \tag{3.19}$$

and

$$\frac{Y_{P_d}}{d} = 1 - PC \;\forall\; R = 0 \tag{3.20}$$

only because $d = Y_{P_d} - Y_P$, if and only if $P = \rho e^{-s\Delta t_D}$. Though, it shows the same problematic property as the prediction principle, that is the characteristic of a steering or open-loop control.

Therefore, this system is only stable for stable plants, too [88]. Though, it is in the case of stable transfer functions very easy to design C to eliminate or damp d - An ability that is missing for regular controllers without internal models [66].
If the plant is stable, it is nevertheless possible to use both principles and even combine them to maintain a synergy, that is finally called the Smith Predictor [79], see Figure 3.9.

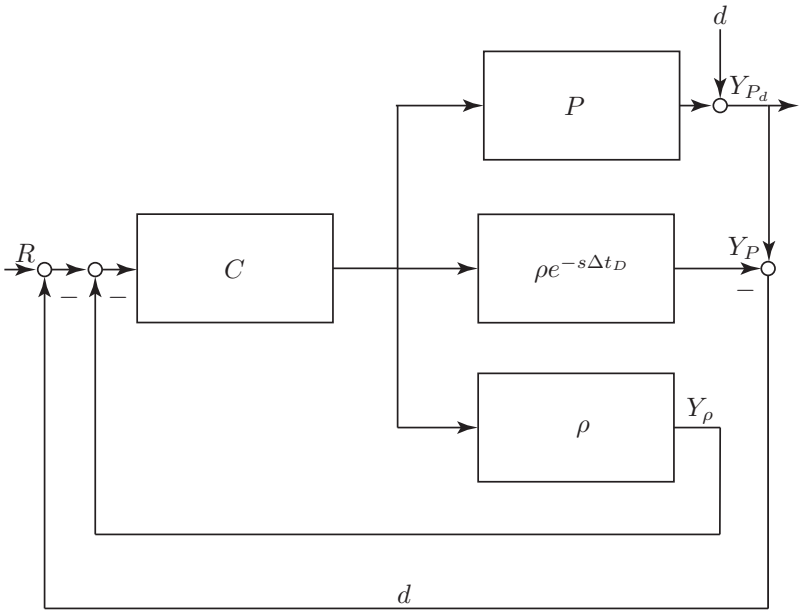


Figure 3.9: Principle of the Smith Predictor

it is easy to deduce its transfer functions, that are

$$\frac{Y_{P_d}}{R} = \frac{PC}{1+C\rho} = \frac{C\rho}{1+C\rho}e^{-s\Delta t_D} \tag{3.21}$$

and

$$\frac{Y_{P_d}}{d} = \frac{1-PC+C\rho}{1+C\rho} = \frac{1+C\left(\rho-\rho e^{-s\Delta t_D}\right)}{1+C\rho} \tag{3.22}$$

Now, the control is able to overcome delay [98], damp disturbances and additionally, the controller design problem has become linear so that it may be designed as if there was no delay [120]. But there is one significant problem that has been ignored. All equations used the relation $\rho e^{-s\Delta t_D} = P$. Literally, that means that the model $\rho e^{-s\Delta t_D}$ must match the plant P exactly. In fact, uncertainties may occur, that are within ρ or within Δt_D. The real system may be so complicated, that it is not trivial to model it mathematically. The delay may vary, too. Both problems are model uncertainties. An additional aim in controller design is to make the system stable, even in presence of uncertainties. Such a goal is called robust controller design [8]. Of course, there are also possibilities of identifying uncertainties and including them in the feedback signal. Examining the block diagram of the internal model control, one may observe that a similar correction is occuring. Model uncertainties are comparable to errors in their effect on the feedback signal. Using Figure 3.9, $Y_{P_d} - Y_P = d + U\left(P - \rho e^{-s\Delta t_D}\right) \neq d$, where U is the output of C, seems trivial. That means, that the model uncertainties are already included within the feedback. Ignoring d, whenever P and $\rho e^{-s\Delta t_D}$ differ, it will raise the control gain. This principle, of course, may not only be used for control, but also for controller optimization and even model optimization. In optimization theory, there is usually a cost function to minimize. Table 3.1 lists the obvious criteria for the optimizations named:

	Cost function
Model optimization	$\|Y_{P_d} - Y_P\|_i = \sqrt[i]{\int_0^\infty \|Y_{P_d} - Y_P\|^{i\alpha} \, dt}$
Controller optimization	$\|R + Y_P - Y_{P_d}\|_i = \sqrt[i]{\int_0^\infty \|R + Y_P - Y_{P_d}\|^{i\alpha} \, dt}$

Table 3.1: Optimization Options

Where $i \in \{1, 2, \cdots, \infty\}$ is any of the named so called H_i-norms, according to the Hardy[c] Space [100] and α an additional weighting.

Definition 3.8. *The Hardy Space H is the set all functions $\hbar$ that fulfill*

$$\max\left(\frac{1}{2\pi}\int_0^{2\pi} \left|\hbar\left(re^{i\theta}\right)\right|^p d\theta\right)^{1/p} < \infty \; \forall \; 0 < r < 1, 0 < p < \infty \tag{3.23}$$

The left term denotes the Hardy measure or Hardy norm, that that was originally meant to denote the mean value of the modulus of a complex function [41], which is why it is used in optimization. To visualize this abstract measure, Figure 3.10 reflects the movement of the Hardy measure for the sin function in the s-plane.

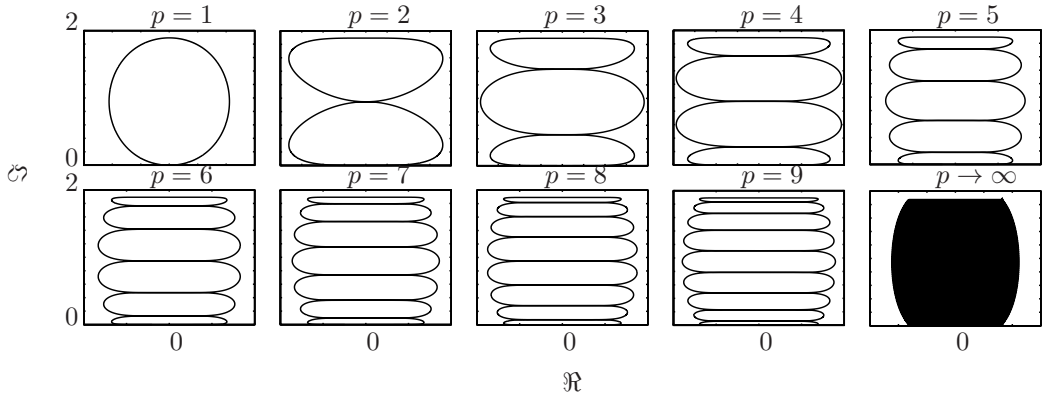


Figure 3.10: Hardy Measures of the sin Function

Since the measure uses the modulus of the complex value, the resulting values are always the distance of the curve to $0 + 0j$ at the argument 2π. Easily, the terms of Tabular 3.1 result of the Hardy measure, plugging in the respective Laplacian[d]. There are several established values for α and i, for the given problem especially $\alpha = i = 2$ [33], the so called H_2-optimization according to the H_2-norm. Obviously, the various optimizations still need an input that is then evaluated versus time. Again, there are several possibilities, as Table 3.2 shows:

Reference $\frac{1}{2\pi j}\int_{\delta-j\infty}^{\delta+j\infty} Re^{st} ds$	Disturbance $\frac{1}{2\pi j}\int_{\delta-j\infty}^{\delta+j\infty} de^{st} ds$	Initial Condition $\underline{x}\,(t=0)$
0	0	$\underline{x}_0$
δ	0	0
σ	0	0
0	δ	0
0	σ	0

Table 3.2: Optimization Test Functions

[c] Godfrey Harold Hardy *1877†1947, English mathematician and philosopher
[d] Pierre-Simon marquis de Laplace *1749†1827, French mathematician and astronomer

where $\delta = 0 \; \forall \; t \neq 0$ while $\int_{-\infty}^{\infty} \delta dt = 1$ and $\sigma = 0 \; \forall \; t < 0, \sigma = 1 \; \forall \; t \geq 0$. So a simple example for a basic optimization procedure is explained in Algorithm 1:

Algorithm 1 Optimization Procedure

```
chose optimization parameter p
chose adequate costfunction c
i = 0
while Σ c_{i+2} ≠ Σ c_i do
   for j = 1;j = 5;j++ do
      use testfunction_j
      measure c
   end for
   p = p + ϵ
   if Σ c_{i+1} < Σ c_i then
      p = p + ϵ
   else
      p = p − 2ϵ
   end if
   i++
end while
```

where ϵ is a variation of the optimization parameter, often denoted as stepsize and p is a characteristic value of the model or controller that the cost function is a function of. To get a better understanding of the proocedure, the process is additionally depicted in Figure 3.11:

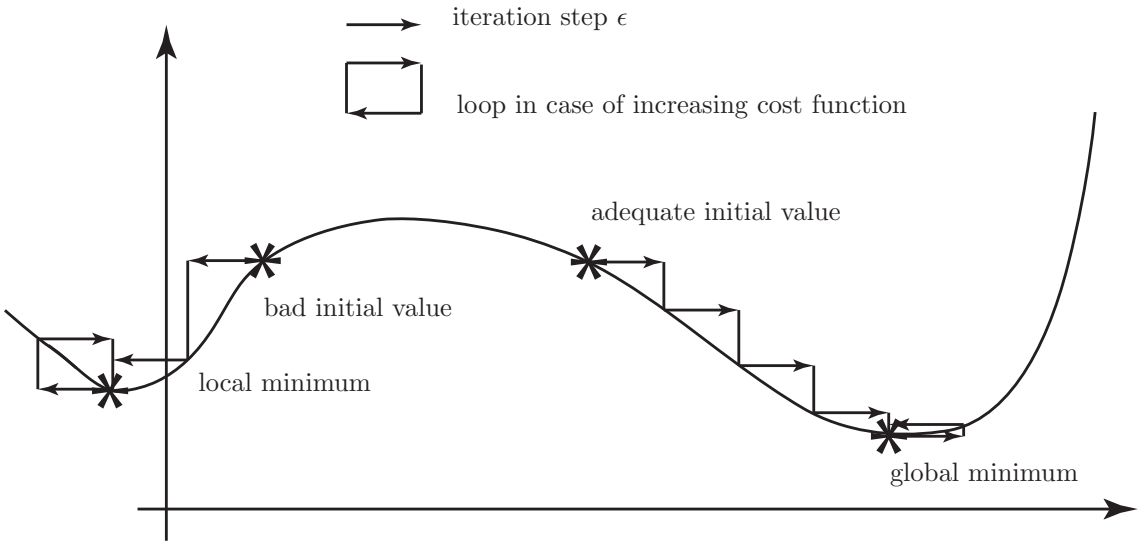


Figure 3.11: Local Optimization

Obviously, this simple procedure will only converge to the global minimum of $c : P \rightarrow \mathbb{R}$, that is $c(p_{opt}) \leq c(p) \; \forall \; p \in P$, if the initial condition $p(i = 0)$ satisfies $p_- < p(i = 0) < p_+$ with $\frac{dc}{dp}(p_-) = \frac{dc}{dp}(p_+) = 0, \frac{d^2c}{dp^2}(p_-) < 0 > \frac{d^2c}{dp^2}(p_p)$ and $p_- < p_{opt} < p_+$. That means that finding the initial value is often the crucial point in optimization.

3.3 Delay and Offset Measurement

One of the crucial weaknesses of the Smith predictor is its sensitivity to delay uncertainty, as further described in 3.4. A protocol to establish a synchronization of clocks can provide the network with

precise delay measurements, but it also provides additional advantages. Once networking elements are using a shared time domain, systems can interprete data from other systems much better because they are able to interprete the timestamp contained in this data in the same way that other systems do. In some cases, it might be even possible to schedule tasks and share them among the components of the network. In this problem, various solutions have been established [23] and are listed in Table 3.3.

	IEEE 1588	NTP	GPS	TTP
spacial extent	subnets	wide area	wide area	bus
communication	network	internet	satellite	bus
accuracy	microsecond	millisecond	microsecond	microsecond
style	master-slave	peer ensemble	client-server	distributed
latency correction	1	1	1	0
administration	0	1	1	1

Table 3.3: Clock Synchronization Protocols

IEEE 1588 is best suited for the clock synchronization of an autonomous, time-delayed network, because it includes latency correction, needs no administration and is made for networks, as can be seen in Table 3.3. This method requires a master-slave architecture in the network, first of all. Since the aim is not to implement one global, but only a mutual time on all nodes, the network time can be chosen to have a shared time that is the time of any of the nodes, though it is more practical to chose a node that is located centrally [24]. This node is called the grandmaster clock. In the problem of structuring the network, as all other nodes shall have a dependency from this node, a tree structure is an efficient implementation [86], marking each side of a link between nodes either as master or as slave. To realize this structure, a spanning tree algorithm [83] similar to Algorithm 2 is needed:

Algorithm 2 Spanning Tree Algorithm

```
1:  start at node E_a
2:  set a = #(p_a)
3:  for i = 1, i = a; i++ do
4:      set port p_ai = M
5:      cross link
6:      arrive at node E_b
7:      set arrival port p_b1 = S
8:      set b = #(p_b)
9:      if b > 1 then
10:         for j = 2, j = b; j++ do
11:             set connection p_bj = M
12:             cross link
13:             arrive at node E_c
14:             set connection p_c1 = S
15:             set c = #(p_c)
16:             if c > 1 then
17:                 ⋮
18:             end if
19:         end for
20:     end if
21: end for
```

where E is an element of the network, p is a port connected to another p via a link, and every p can either be $= M$ or $= S$ where M means master and S means slave. Using Algorithm 2, every link has one side denoted M and one S, as Algorithm 2, lines 4, 7, 11, 14 implies. Furthermore, every node except for the grandmaster has only just one S port and all other ports $= M$, as to be

seen in Algorithm 2, lines 3, 10. According to [1], a network element $E_{ordinary}$ that has only one port $\#(p_{ordinary}) = 1$ is called an ordinary clock, compare Algorithm 2, lines 9, 16, and all other network elements $E_{boundary}$ with $\#(p_{boundary}) > 1$ are called boundary clocks. In Figure 3.12 a tree spanning algorithm strucutred network is depicted.

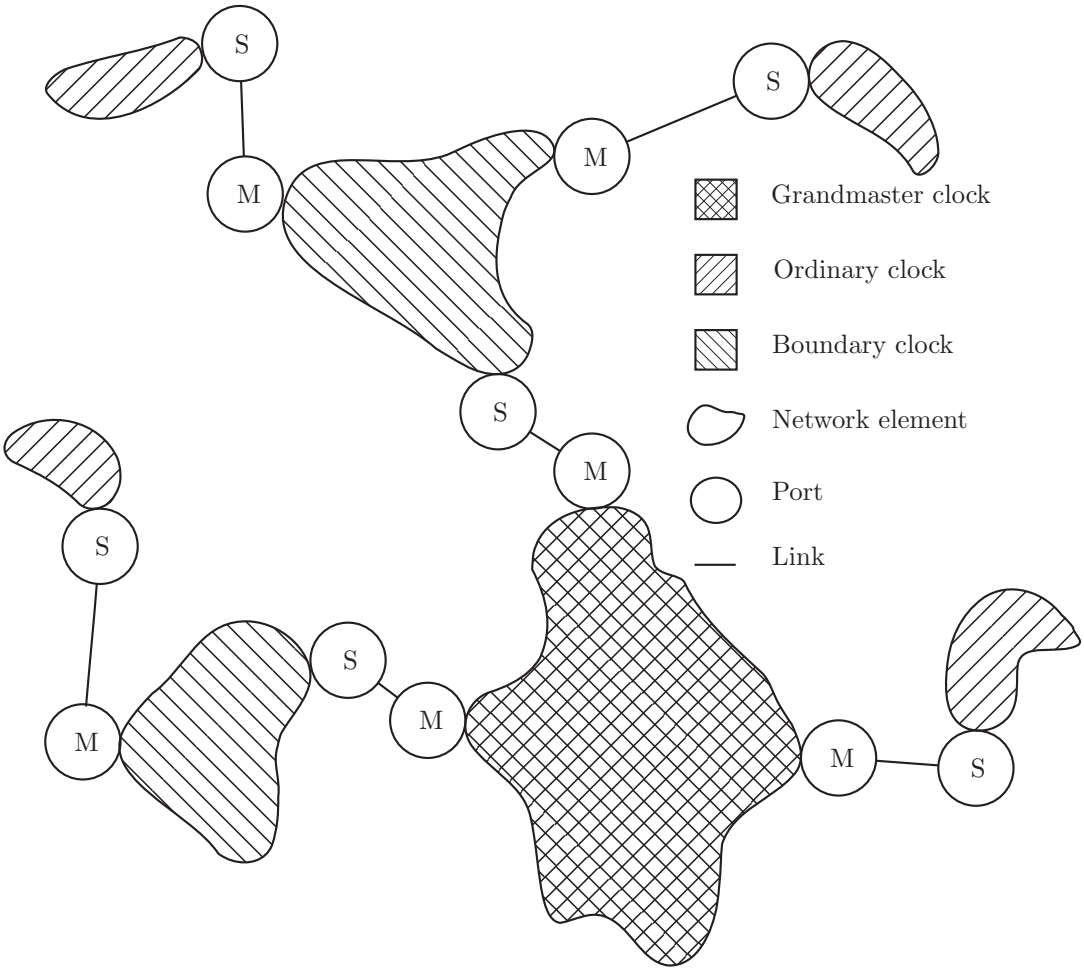


Figure 3.12: Network Structured by the Tree Spanning Algorithm

The Figure illustrates that the loop within the algorithm breaks when it reaches an ordinary clock. The algorithm then returns to the last boundary clock, structures all its ports, and afterwards goes on with the remaining ports of the boundary clock. This continues until the algorithm reaches the grandmaster and proceeds structuring its ports. When the initial network is finally structured by Algorithm 2 and the network looks like Figure 3.12, every clock is synchronized by its only master. Now a protocol has to be implemented on every port that provides values for the offset Δt_O and the delay Δt_D. Therefore, Algorithms 3 and 4 are implemented at master and slave.

Algorithm 3 Synchronization of Master Clocks

1: Δt_O, Δt_D (t_1)
2: **if** $t_M = t_1$ **then**
3: send t_1
4: **end if**
5: **if** receive t_3 **then**
6: $t_4 = t_M$
7: send t_4
8: **end if**

Where the input variable t_1 is the time the algorithm starts working, Δt_O is the offset, that is the time difference between their inner clocks, Δt_D is the delay, that is the latency or trip-time of a message from one clock to another and t_M is obviously the inner clock of the master port. Logically, because there is yet no output and the algorithm receives data, there is still the slave algorithm Algorithm 4 left to define.

Algorithm 4 Synchronization of Slave Clocks

1: **if** receive t_1 **then**
2: $t_2 = t_S$
3: $t_3 = t_S$
4: send t_3
5: **end if**
6: **if** receive t_4 **then**
7: $\Delta t_D = \frac{t_2 - t_1 + t_4 - t_3}{2}$
8: $\Delta t_O = \frac{t_2 - t_1 - t_4 + t_3}{2}$
9: **end if**
10: **return** Δt_D, Δt_O

Where t_S is the inner clock of the slave port. Since the real effects of that algorithm and the reason for its accuracy and efficiency cannot easily be read out of the Algorithms 4 and 3, Figure 3.13 tries to give a livelier depiction of its functional principle.

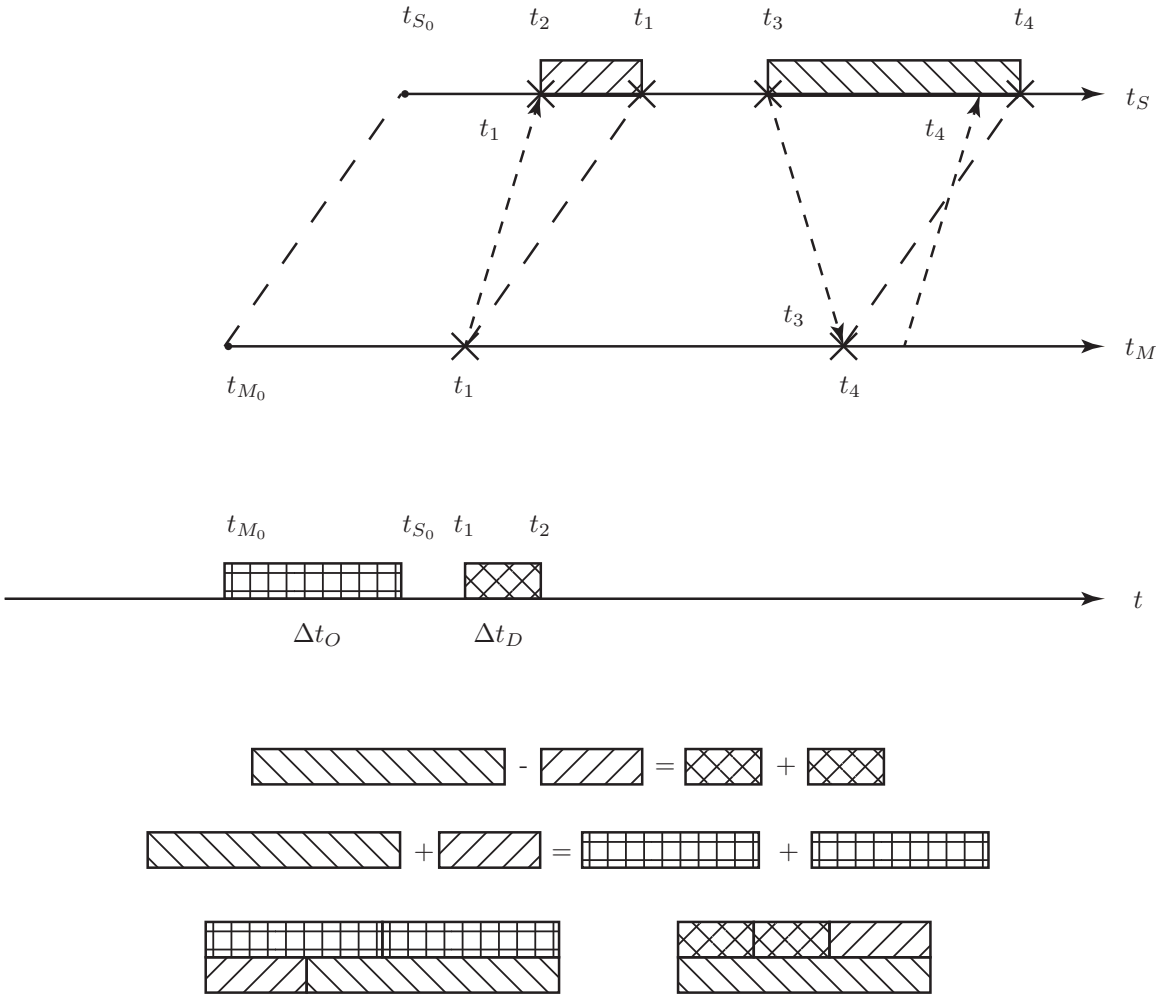


Figure 3.13: Functional Principle of Network Time Synchronization

Where t is the absolute, global time, t_S, t_M are the inner clocks of S and M, t_{S_0}, t_{M_0} are the absolute times for $t_S = 0$, $t_M = 0$, the dashed lines depicting delays, the dashed arrows depicting messages and the variables next to them the content of those. It is important that t_S and t_M are different dimensions, therefore $t_1 \in t_S \neq t_1 \in t_M$, because every clock interpretes the time values in a different way due to their distinct values of t_0. The patterned rectangles in Figure 3.13 represent the equations that the protocol is based on [24].

$$(t_4 \in t_S) - t_3 = \Delta t_D + \Delta t_O \tag{3.24}$$

$$(t_1 \in t_S) - t_2 = \Delta t_O - \Delta t_D \tag{3.25}$$

The results of (3.24)−(3.25) and (3.24)+(3.25) are

$$(t_4 \in t_S) - t_3 + t_2 - (t_1 \in t_S) = 2\Delta t_D \tag{3.26}$$

$$(t_4 \in t_S) - t_3 + (t_1 \in t_S) - t_2 = 2\Delta t_O \tag{3.27}$$

Plugging (3.26) and (3.27) into Algorithm 4 l.7,8, the result for Δt_O according to Algorithm 4 and [1] is the negative value of the Δt_O depicted in Figure 3.13. Since there are only two types of offset possible, $t_{S_0} > t_{M_0}$, and $t_{S_0} < t_{M_0}$, this fact proves the correctness of the functional principle, because the entire literature uses examples with $t_{S_0} > t_{M_0}$ while Figure 3.13 uses $t_{S_0} < t_{M_0}$. Therefore, using the official formulas, to synchronize the slave clock, the network element has to set $t_S + \Delta t_O = t_M$. In addition, the unsigned latency Δt_D is known.

3.4 Problems

The principles that were explained in this chapter are all established and have been studied and optimized in various ways for years. Though, because these principles are rarely used together, it may be possible to alter them in several ways, and, by doing that, open new applications. The most common problems are the lack of robustness and fault tolerance. For application in networks, the best-known problem of the smith-controller is its sensitivity to delay errors [58]. In traditional controller design, the basic aim is to have a denominator of the transfer function that is < 0. Using (3.21), the stability criterion for the smith-controller is

$$|1 + C\rho| > 0 \tag{3.28}$$

Now, that there is a uncertainty in delay, that is $\left|e^{-s\Delta t_{D\sim}} - e^{-s\Delta t_D}\right|$, where Δt_D is the real and $\Delta t_{D_\sim}$ the estimated delay. The open loop including uncertainties must still be within that stable region [117], that is a robust behavior:

$$|1 + C\rho| > |C\rho| \left|e^{-s\Delta t_{D\sim}} - e^{-s\Delta t_D}\right| \tag{3.29}$$

Hence, the smith-predictor is robust until the error in delay estimation or delay measurement exceeds $|\Delta t_{D_\sim} - \Delta t_D|$ for a value of $\Delta t_{D_\sim}$ that satisfies (3.29). The major problem about the sensitivity described above is that delays may vary very fast, especially in networks. If there are effect like jitter, delay variation or package loss [106] due to fluctuation in the signal delay because of electromagnetic interference, crosstalk with other signals, signal clock variation, or insufficient bandwidth, then istabilities may occur. Most strategies have used probabilistic approaches for the prediction of $\Delta t_D\,(t)$ [4]. These methods therefore introduce additional uncertainties. Figure 3.14 [21] depicts the measurement of a two-way packet delay variation, that means that packets are sent in a fixed frequency in both directions and the frequency of arrival is measured in both directions, where the graph shows the absolute frequency of the appearance of differences of delay values from the average [21].

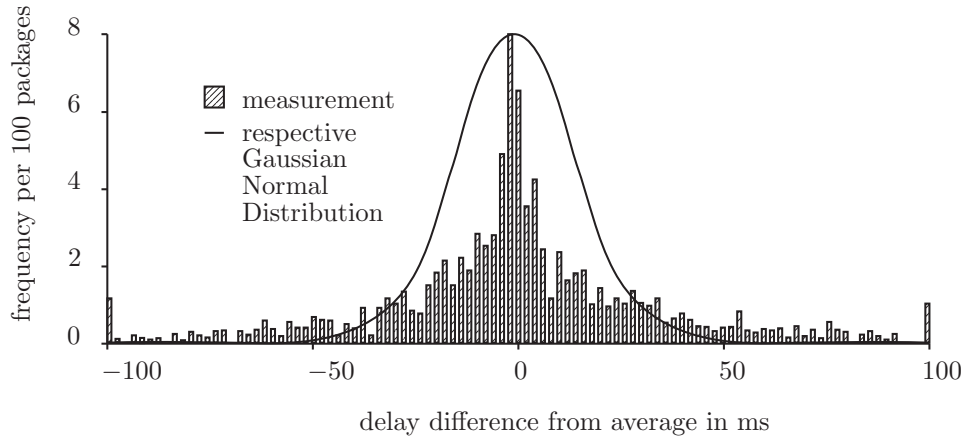


Figure 3.14: Two-Way Packet Delay Variation

Obviously, the Gaussian Normal Distribution, that is often used for probabalistic delay estimation, can only describe the appearance of delay variation in an approximate and possibly inadequate way. The effect looks rather like a mixture of a Gaussian Normal Distribution and another distribution possibly representing electronic noise.

Also, there is a numeric problem when two elements of the network, that are usually solving equations discrete, have offset computation sequences. This is easy to understand if two solvers use a step size of one hour, where one of them was started directly after the other. Then the message will arrive at time, but the content will not be used before one hour has passed.

Another problem is fault-intolerance of the IEEE 1588 protocol due to its grandmaster-dependent architecture. It is easy to see, that if the grandmaster, or only one of its direct connections to a boundary clock collapses, the whole system or a whole branch of the tree will have no delay or offset measurement at all. Figure 3.15 shows those examples:

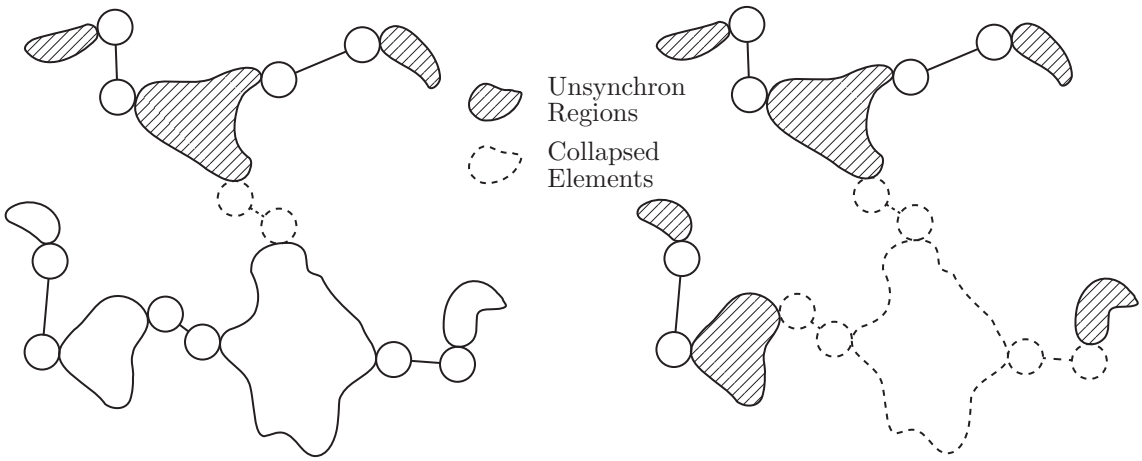
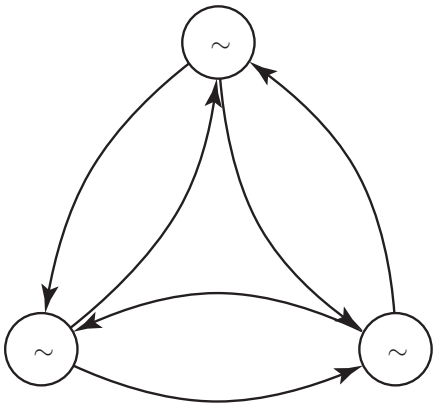
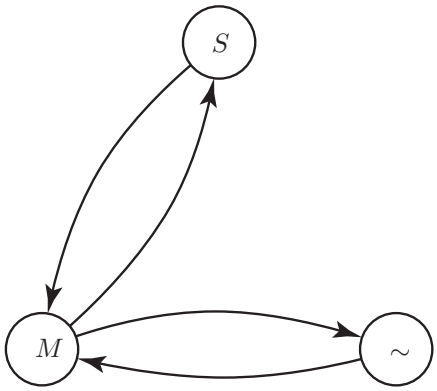


Figure 3.15: Failure in Tree structures

It may be dire, if the unsynchronized regions of the network have a control that is heavily dependent on delay measurement or solvers that are heavily dependent on offset measurement, as described above. This is especially true when instabilities in those network elements can also cause instabilities in physical systems. The fourth and last of the named problems is that most approaches dealing with the phenomenon of system synchronization [39] in coupled systems require symmetric couplings [19], in most cases even a bilateral coupling in a ring [74] and a structure with no delays, that the given network structures lack. Figure 3.16 outlines two examples:



3 Bidirectional Coupled Oscillators in a Ring



Coupling of a Smith Predictor

Figure 3.16: Coupling Types

Though other couplings and also systems with delay have been discussed [116], these solutions are not applyable to control systems because there is no reference signal. The solutions to those systems tend to be very complex and are often not feasible for real-time systems, therefore an extremely high understanding of the system is required, to simplify those methods to the utmost. The named problems are the ones were known previous to this work and that will occur while solving the initial problem. All of those will have to be overcome within the synthesis.

4

Synthesis

This chapter attempts to provide a generic method for the monitoring of delayed systems in networks. In the beginning, an alternated Smith Predictor will be introduced, that uses a realtime model as a reference. A modal analysis will be conducted on this control using the Lambert W function. The coupling terms of this framework will furthermore be analyzed according to its latent roots, where theorems on block matrix analysis, symmetric polynomials, and root optimization will be utilized. The results will be evaluated critically, giving reasons for the development of a new framework called the Zero Row Sum Negative Trace Control. In the following sections the measuring of delay and the analysis of the network causing the delay will be focused. First, the basic delay measurement protocol will be generalized to a continuous implementation. Afterwards, the master / slave hierarchy will be altered to a more fault tolerant structure. Therefore, the network topology will have to be changed, as well, using implications from graph theory.

4.1 Smith Synchronization

The basis of Smith Synchronization uses a Smith Predictor. Classical controllers are expected to make a plant follow a reference. Now, the idea is to replace the reference by a model of the plant, so that the plant follows the model and the model can be monitored instead of the plant, as Figure 4.1 shows.

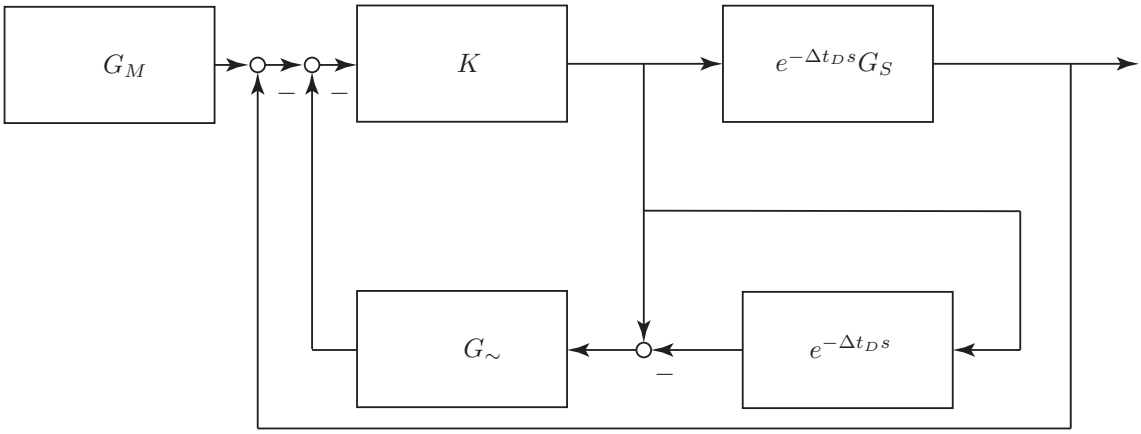


Figure 4.1: Smith Synchronization

Definition 4.1. *The structure in Figure 4.1 is called Smith Synchronization.*

Let

$$\dot{\underline{x}}_i = \underline{f}\left(\underline{x}_i\right) + \boldsymbol{B}\,\underline{u}_i \tag{4.1}$$

$$\underline{y}_i = \boldsymbol{C}\,\underline{x}_i \tag{4.2}$$

be the state-space equation of $G_i \;\forall\; i = M, S, \sim$, where $\underline{x}_i \in \mathbb{R}^n$ is the state vector, $\dot{\underline{x}}_i = \frac{d\underline{x}_i}{dt} \in \mathbb{R}^n$ its time-derivative, $\underline{u} \in \mathbb{R}^n$ is the input signal, $\underline{f} : \mathbb{R}^n \to \mathbb{R}^n$ is a vector-valued function describing the sensitivity of $\dot{\underline{x}}$ from $\underline{x}$, $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ is a matrix containing sensitivities of $\dot{\underline{x}}_i$ for $\underline{u}_i$ and $\mathbf{C}$ is the output matrix, describing how the states of i leave the sytem. From Figure 4.1, let

$$\underline{u}_M = \underline{0} \tag{4.3}$$

$$\underline{u}_S = \boldsymbol{C}K\left[\underline{x}_M\left(t-\Delta t_D\right) - \underline{x}_\sim\left(t-\Delta t_D\right) - \underline{x}_S\left(t-\Delta t_D\right)\right] \tag{4.4}$$

$$\underline{u}_\sim = \boldsymbol{C}K\left[\underline{x}_M - \underline{x}_\sim - \underline{x}_S - \underline{x}_M\left(t-\Delta t_D\right) + \underline{x}_\sim\left(t-\Delta t_D\right) + \underline{x}_S\left(t-\Delta t_D\right)\right] \tag{4.5}$$

so that (4.1) looks like

$$\dot{\underline{x}}_M = \underline{f}\left(\underline{x}_M\right) \tag{4.6}$$

$$\dot{\underline{x}}_S = \underline{f}\left(\underline{x}_S\right) + \boldsymbol{B}\,\boldsymbol{C}K\left[\underline{x}_M\left(t-\Delta t_D\right) - \underline{x}_\sim\left(t-\Delta t_D\right) - \underline{x}_S\left(t-\Delta t_D\right)\right] \tag{4.7}$$

$$\dot{\underline{x}}_\sim = \underline{f}\left(\underline{x}_\sim\right) + \boldsymbol{B}\,\boldsymbol{C}K\left[\underline{x}_M - \underline{x}_\sim - \underline{x}_S - \underline{x}_M\left(t-\Delta t_D\right) + \underline{x}_\sim\left(t-\Delta t_D\right) + \underline{x}_S\left(t-\Delta t_D\right)\right] \tag{4.8}$$

Now let $\underline{\chi} = \begin{bmatrix}\underline{x}_S & \underline{x}_M & \underline{x}_\sim\end{bmatrix}^T \in \mathbb{R}^{3n}$ and $\dot{\underline{\chi}} = \frac{d\underline{\chi}}{dt}$ its time-derivative, then

$$\dot{\underline{\chi}} = \begin{bmatrix}\underline{f}\left(\underline{x}_S\right)\\ \underline{f}\left(\underline{x}_M\right)\\ \underline{f}\left(\underline{x}_\sim\right)\end{bmatrix} + \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ -\boldsymbol{B}\,\boldsymbol{C}K & \boldsymbol{B}\,\boldsymbol{C}K & -\boldsymbol{B}\,\boldsymbol{C}K\end{bmatrix}\underline{\chi} + \tag{4.9}$$

$$+ \begin{bmatrix}-\boldsymbol{B}\,\boldsymbol{C}K & \boldsymbol{B}\,\boldsymbol{C}K & -\boldsymbol{B}\,\boldsymbol{C}K\\ 0 & 0 & 0\\ \boldsymbol{B}\,\boldsymbol{C}K & -\boldsymbol{B}\,\boldsymbol{C}K & \boldsymbol{B}\,\boldsymbol{C}K\end{bmatrix}\underline{\chi}\left(t-\Delta t_D\right) \tag{4.10}$$

It is now possible to compute the first variation of $\underline{\chi}$, which will be refferred to as $\underline{\xi} \in \mathbb{R}^{3n}$, with its time-derivative $\dot{\underline{\xi}} = \frac{d\underline{\xi}}{dt} \in \mathbb{R}^{3n}$.

$$\dot{\underline{\xi}} = \left(\boldsymbol{I}^3 \otimes \frac{\partial \underline{f}}{\partial \underline{x}} + \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ -1 & 1 & -1\end{bmatrix} \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\underline{\xi} + \left(\begin{bmatrix}-1 & 1 & -1\\ 0 & 0 & 0\\ 1 & -1 & 1\end{bmatrix} \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\underline{\xi}\left(t-\Delta t_D\right) \tag{4.11}$$

With a shorter expression for the Jacobian[a] and substitutions for the coupling matrices, (4.11) becomes

$$\dot{\underline{\xi}} = \left(\boldsymbol{I}^3 \otimes \boldsymbol{\nabla}\left(\underline{f}\right) + \boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\underline{\xi} + \left(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\underline{\xi}\left(t-\Delta t_D\right) \tag{4.12}$$

Definition 4.2. *The coupling configuration matrices of the Smith Synchronization are*

$$\boldsymbol{\Gamma_t} = \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ -1 & 1 & -1\end{bmatrix} \wedge \boldsymbol{\Gamma}_{t-\Delta t_D} = \begin{bmatrix}-1 & 1 & -1\\ 0 & 0 & 0\\ 1 & -1 & 1\end{bmatrix} \tag{4.13}$$

In this form it is obvious that the whole differential equation has to be analyzed for statements about system state stability, whereas $\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K$ and $\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K$ are to be subjected to critical scrutiny in terms of synchronization.

Therefore, the coupling dynamics $\dot{\underline{\psi}} = \boldsymbol{\Gamma}_t\underline{\psi} + \boldsymbol{\Gamma}_{t-\Delta t_D}\underline{\psi}\left(t-\Delta t_D\right)$ will be analyzed, where $\underline{\psi} \in \mathbb{R}^3$ is a variable that has the same modes in its dynamics as the smith synchronization, except for the inner dynamics of each system that are still described by $\underline{f}$. The first step is to conduct a modal analysis of $\underline{\psi}$. The Smith synchronization is said to be able to synchronize M and S.

Lemma 4.1. *The Smith synchronization induces a synchronized state for M and S.*

[a] Carl Gustav Jacob Jacobi *1804†1851, German mathematician

Proof. The roots of the state-space-equation of $\underline{\psi}$ are the solutions in s to

$$s\boldsymbol{I} = \boldsymbol{\Gamma}_{t-\Delta t_D} e^{-s\Delta t_D} + \boldsymbol{\Gamma}_t \tag{4.14}$$

Expanding by $\Delta t_D e^{(s\boldsymbol{I}-\boldsymbol{\Gamma}_t)\Delta t_D}$, this transcendental characteristic equation can be rewritten as

$$(s\boldsymbol{I} - \boldsymbol{\Gamma}_t)\,\Delta t_D e^{(s\boldsymbol{I}-\boldsymbol{\Gamma}_t)\Delta t_D} = \boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D} \tag{4.15}$$

The equation now has the form to apply the Lambert[b] W function.

Definition 4.3. *The Lambert W function is the solution in W to $z = We^W$.*

Obviously, z is not injective, so that W is multivalued.

Definition 4.4. *The k sets of solutions to W are called the branches of W. $k = \{k \in \mathbb{Z} | -\infty \leq k \leq \infty\}$. The kth solution of W is denoted as W_k. W_0 is called the principal branch.*

The Lambert W function can be used to find the infinite set of solutions in transcendental algebraic equations.

Obviously, $W\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) = (s\boldsymbol{I} - \boldsymbol{\Gamma}_t)\,\Delta t_D$.

Lemma 4.2 ([5])**.** *The roots of $\underline{\dot{x}} = \boldsymbol{\Gamma}_{t-\Delta t_D}\,\underline{x}\,(t-\Delta t_D) + \boldsymbol{\Gamma}_t\,\underline{x}$ are the elements of $\sigma\left(\frac{1}{\Delta t_D} W\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) + \boldsymbol{\Gamma}_t\right)$.*

Using Lemma 4.2, the eigenvalues of $\underline{\psi}$ are the elements of the set $\sigma\left(\frac{1}{\Delta t_D} W\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) + \boldsymbol{\Gamma}_t\right)$. Let $\boldsymbol{\Lambda} = \operatorname{diag}\{\sigma\}$, then

$$\boldsymbol{V}^{-1}\left(\frac{1}{\Delta t_D} W\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) + \boldsymbol{\Gamma}_t\right)\boldsymbol{V} = \boldsymbol{\Lambda} \tag{4.16}$$

denotes the well known similarity transform where $\boldsymbol{V}$ is the modal matrix and the modes of $\underline{\psi}$ its columns. Using Sylvester's[c] formula,

$$e^{\boldsymbol{\Gamma}_t\Delta t_D} = \boldsymbol{V}\left(\boldsymbol{\Gamma}_t\Delta t_D\right)\boldsymbol{\Lambda}\left(\boldsymbol{\Gamma}_t\Delta t_D\right)\boldsymbol{V}^{-1}\left(\boldsymbol{\Gamma}_t\Delta t_D\right) \tag{4.17}$$

Because $\sigma\left(\boldsymbol{\Gamma}_t\Delta t_D\right) = \{0, 0, -\Delta t_D K\}$, its eigenvalues build the matrix

$$\boldsymbol{V}\left(\boldsymbol{\Gamma}_t\Delta t_D\right) = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \tag{4.18}$$

so that the matrix exponential results in

$$e^{\boldsymbol{\Gamma}_t\Delta t_D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ e^{-\Delta t_D K} - 1 & -e^{-\Delta t_D K} + 1 & e^{-\Delta t_D K} \end{bmatrix} \tag{4.19}$$

The argument of the Lambert W function is the matrix

$$\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D} = \begin{bmatrix} -TKe^{-TK} & TKe^{-TK} & -TKe^{-TK} \\ 0 & 0 & 0 \\ TKe^{-TK} & -TKe^{-TK} & TKe^{-TK} \end{bmatrix} \tag{4.20}$$

Obviously, $\operatorname{rank}\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) = 1$ and $\sigma\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right) = \{0, 0, 0\}$. The eigenvalue zero has the algebraic multiplicity 3 but only a geometric multiplicity of 2. There is no similarity transform to this matrix, so that is is not possible to apply Sylvester's formula. Since the matrix is nilpotent with $\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\Delta t_D e^{\boldsymbol{\Gamma}_t\Delta t_D}\right)^{\varphi} = \boldsymbol{0}\ \forall\ \varphi > 1$, the matrix Lambert W function can instead

[b] Johann Heinrich Lambert *1728†1777, Alsatian mathematician, logician, physicist and philosopher
[c] James Joseph Sylvester, *1814†1897, English mathematician

be expressed as a fast converging sum. Since the principal branch of the Lambert W function is defined as

$$W_0(z) = \sum_{n=1}^{\infty} \frac{(-n)^{n-1}}{n!} z,$$

it converges after the first element in this case, so that

$$W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) = \mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D} \tag{4.21}$$

The matrix that defines the latent roots and modes of the Smith synchronization has the form

$$\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t = \tag{4.22}$$

$$= \begin{bmatrix} -K\mathrm{e}^{-\Delta t_D K} & K\mathrm{e}^{-\Delta t_D K} & -K\mathrm{e}^{-\Delta t_D K} \\ 0 & 0 & 0 \\ K\left(\mathrm{e}^{-\Delta t_D K} - 1\right) & -K\left(\mathrm{e}^{-\Delta t_D K} - 1\right) & K\left(\mathrm{e}^{-\Delta t_D K} - 1\right) \end{bmatrix}$$

with $\sigma\left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t\right) = \{-K, 0, 0\}$ and the respective modal matrix

$$\boldsymbol{V} = \begin{bmatrix} -\frac{e^{-\Delta t_D K}}{e^{-\Delta t_D K} - 1} & -1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{4.23}$$

The mode corresponding to the third eigenvector is a synchronized state of S and M that is not damped. All other modes have eigenvalues with non-positive real parts. The $k \neq 0$ branches of the Lambert W function can be ignored since the real parts of the roots are strictly monotonically decreasing with increasing k. □

Therefore, the Smith Synchronization has a synchronized state, but it not yet said to be synchronizable. The problem is the nature of the second mode of this system. This undamped mode is proportional to the behaviour of system $\sim$ and is added to the inner motion f of system S and the synchronized motion.

Lemma 4.3. *The Smith Synchronization is not synchronizable for all $\Delta t_D > 0$.*

Proof. The second mode of the Smith Synchronization is undamped with the eigenvector $[1\ 0\ -1]$. Without that mode, systems S and M are synchronizable. Now there is an additional motion of S antiproportional to the motion of $\sim$, so that $|\underline{x}_S - \underline{x}_M| \propto \underline{x}_\sim$. The input signal of $\sim$ is proportional to $1 - e^{-\Delta t_D s}$, that is $> 0 \; \forall \; \Delta t_D > 0$. □

Definition 4.5. *A system with a synchronized state, nonpositive latent roots and at least one more latent root with real part zero is said to achieve ragged synchronization. The remaining error $\left|\underline{x}_i - \underline{x}_j\right| = e$ is called synchronization error.*

Lemma 4.4. *The Smith synchronization is said to achieve ragged synchronization with a synchronization error bounded by and proportional to Δt_D.*

At this point it is possible to show the convergence of the modes in the sense of attraction in the state space. By initiating the systems with different conditions and a stable differential equation $\dot{x} = -x + u$ each, the developments of the modes can be plotted versus time. These modal attractors are depicted in Figure 4.2.

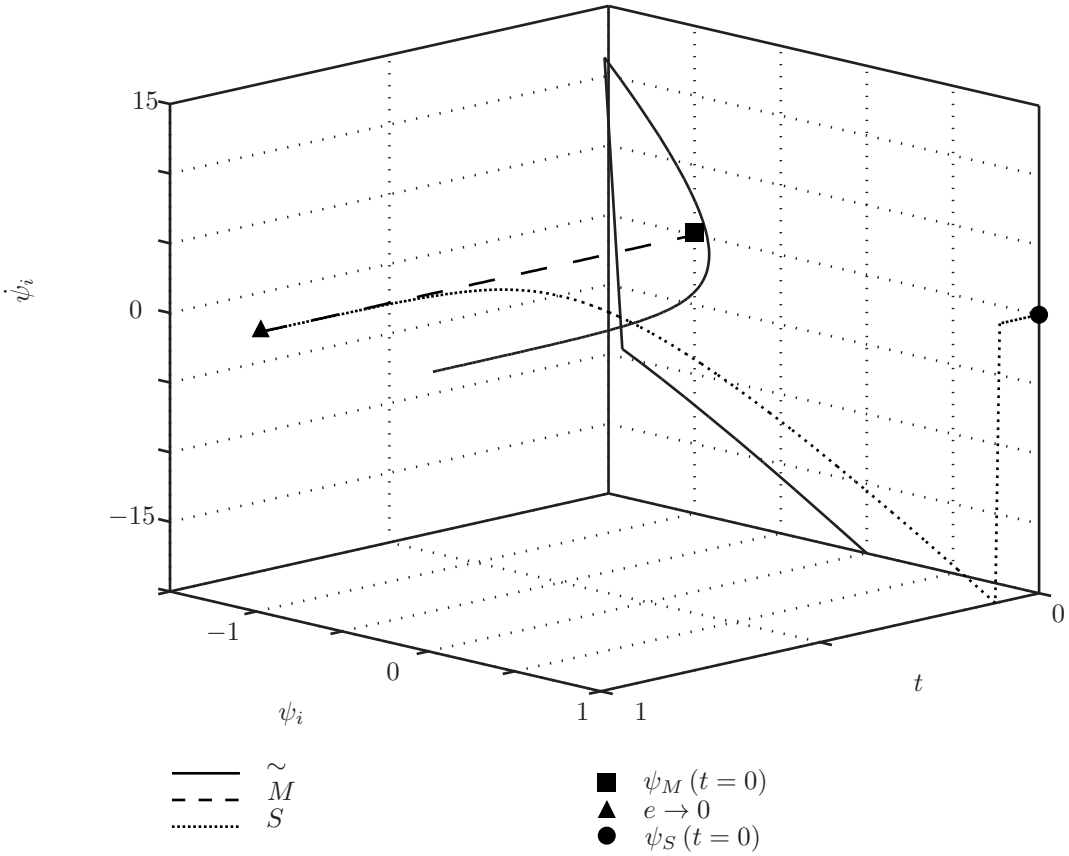


Figure 4.2: Modal Attractors of the Smith Synchronization

It is possible to see that the modes of S and M are attracted to the same point for any initial offset. The offset decreases exponentially. To investigate this function closer, the dynamics of the coupling are started with different delays for the test function $\dot{x} = -x + u$. This time, the results are not depicted in the state space, but as the synchronization error versus time as shown in Figure 4.3.

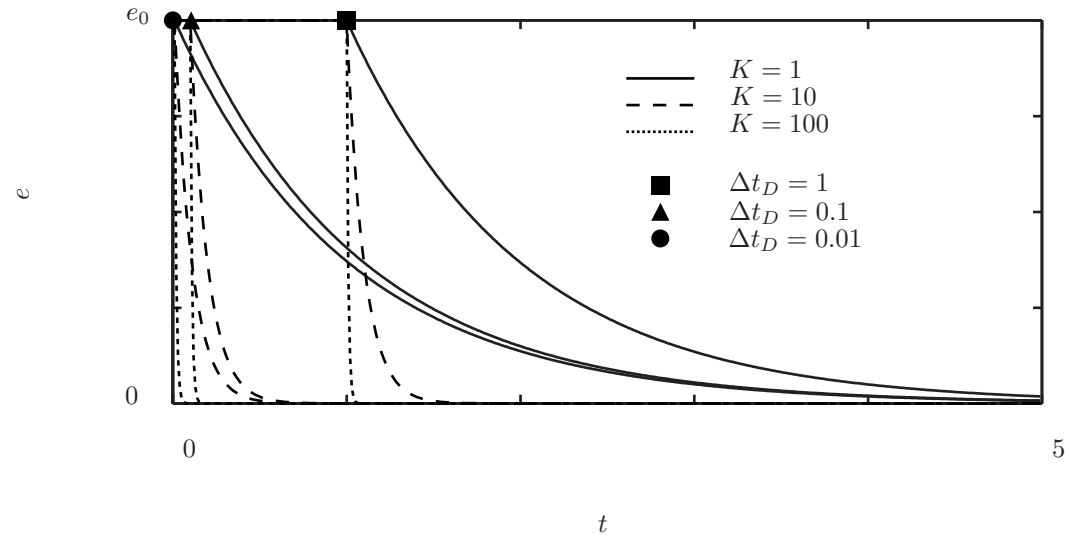


Figure 4.3: Synchronization Errors of the Smith Synchronization

With these results, the error convergence function can be stated as $e = e_0 e^{-K(t-\Delta t_D)/100}$ for that case. However, though the modes converge when there is no motion in each system, there is still the second mode that bounds the final synchronization error for any system motion. Because the exact process cannot be generalized, the worst case will be shown in an example.

Example. A simple nonlinear dynamic system is considered in a Smith synchronization. The system is said to be described by the state space equation

$$\underline{\dot{x}}_i = \begin{bmatrix} x_{i_2}^2 x_{i_1} \\ \cos x_{i_2} \end{bmatrix} + \boldsymbol{I}^2 \underline{u}_i \; \forall \; i = M, S, \sim \tag{4.24}$$

This system has a strong dependence on the system $\sim$ since the cosine function already has a certain dynamic behavior, although the input might be delayed. Such systems are usually not very efficient in a Smith synchronization. The reference model of the Smith synchronization has an input function

$$\underline{u}_M = \underline{\sin t} \tag{4.25}$$

that is therefore also the input function for the Smith synchronization and for system M, respectively. The attractors of systems M and S for this input are depicted in Figure 4.4.

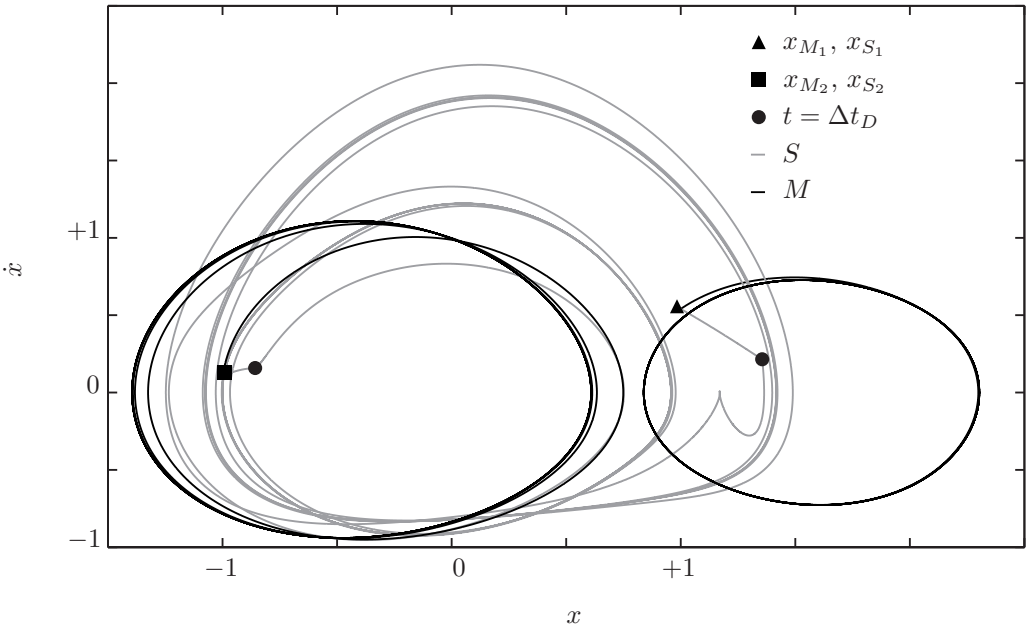


Figure 4.4: Worst Case of Smith Synchronization

The impact of the strong dependence on the delay is obvious. The systems have a similar behavior, but do not converge at all. Both enter a limit cycle at a different place in the state space.

The crucial property of the discussed control is, of course, the system delay. In terms of synchronization, its nonsymmetrical coupling is noteworthy as well. It is not longer possible to compute the stability of a single matrix that is symmetric and zero row sum. The problem here requires the analysis of two matrices, that are connected nonlinearly, neither of which is symmetric nor zero row sum. A zero eigenvalue of a coupling matrix represents a Hopf bifurcation. Though both coupling matrices are singular, it is not trivial to find out wether the coupling equations undergo a Hopf bifurcation when analyzing both of them. An approach used to guarantee stability in general, so that a system contains only eigenvalues smaller than zero, is to be found in the theorem of Mori [72]. In the given problem, the aim is to have one eigenvalue of the coupling terms be zero and all others negative. Mori's theorem utilizes the matrix measure $\mu(\boldsymbol{M})$ [118].

Definition 4.6. *The function μ_i of a matrix $\boldsymbol{M}$ is its matrix measure*

$$\mu_i(\boldsymbol{M}) = \lim_{\theta \to 0} \frac{\|1 + \theta \boldsymbol{M}\|_i - 1}{\theta} \; \forall \; i = 1, 2, \infty \tag{4.26}$$

Theorem 4.1. *The coupling equation achieves asymptotic stability if any* $i \in \{1, 2, \infty\}$ *fulfills*

$$\mu_i \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right) + \left\| \boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right\|_i = 0 \tag{4.27}$$

The matrix measure can be simplified to

$$\mu_1 \left(\boldsymbol{M} \right) = \max_j \left(\Re \left(m_{jj} \right) + \sum_{\substack{i=1 \\ i \neq j}}^{n} \left| m_{ij} \right| \right) \tag{4.28}$$

$$\mu_2 \left(\boldsymbol{M} \right) = \max_j \left(\frac{\lambda_j \left(\boldsymbol{M} + \boldsymbol{M}^* \right)}{2} \right) \tag{4.29}$$

$$\mu_\infty \left(\boldsymbol{M} \right) = \max_j \left(\Re \left(m_{jj} \right) + \sum_{\substack{i=1 \\ i \neq j}}^{n} \left| m_{ji} \right| \right) \tag{4.30}$$

Where $\boldsymbol{M} = \{m_{ij}\} \in \mathbb{C}^{n \times n}$, $\sigma \left(\boldsymbol{M} \right) = \left(\lambda_1, \cdots, \lambda n \right)$ is the set of eigenvalues, that is the set of solutions of the latent roots of $\left| \boldsymbol{M} - \boldsymbol{I}^n \lambda \right| = 0$ and $\boldsymbol{M}^* = \bar{\boldsymbol{M}}^T$ is the adjoint. Furthermore, $\| \boldsymbol{M} \|$ is the operator norm that is defined as

$$\left\| \boldsymbol{M} \right\|_i = \max_{\underline{z} \neq \underline{0}} \left(\frac{\left\| \boldsymbol{M}\, \underline{z} \right\|_i}{\left\| \underline{z} \right\|_i} \right) \; \forall \, i = 1, 2, \infty \tag{4.31}$$

each induced by the respective vector norm

$$\left\| \underline{z} \right\|_i = \sqrt[i]{\sum_{j=1}^{n} \left| z_j \right|^i} \; \forall \, i = 1, 2, \infty \tag{4.32}$$

where $\underline{z} = \begin{bmatrix} z_1 & \cdots & z_n \end{bmatrix} \in \mathbb{R}^n$ so that they simplify to

$$\left\| \boldsymbol{M} \right\|_1 = \max_j \left(\sum_{i=1}^{n} \left| m_{ij} \right| \right) \tag{4.33}$$

$$\left\| \boldsymbol{M} \right\|_2 = \max_j \left(\Sigma_j \left(\boldsymbol{M} \right) \right) \tag{4.34}$$

$$\left\| \boldsymbol{M} \right\|_\infty = \max_j \left(\sum_{i=1}^{n} \left| m_{ji} \right| \right) \tag{4.35}$$

where $\Sigma \left(\boldsymbol{M} \right) = \left(\rho_1, \cdots, \rho_{r(\boldsymbol{M})} \right)$ is the set of singular values.

Definition 4.7. *The set* Σ *of a matrix* $\boldsymbol{M}$ *is the set of the solutions to* $\boldsymbol{M} = \boldsymbol{U}\, \boldsymbol{\Xi}\, \boldsymbol{V}^*$ *where* $\boldsymbol{U} \in \mathbb{R}^{n \times n}$, $\boldsymbol{V} \in \mathbb{R}^{n \times n}$ *are matrices with* $\boldsymbol{U}^* \boldsymbol{U} = \boldsymbol{I}^n$, $\boldsymbol{V}^* \boldsymbol{V} = \boldsymbol{I}^n$ *and* $\boldsymbol{\Xi} = \begin{bmatrix} \operatorname{diag} \left\{ \Sigma \left(\boldsymbol{M} \right) \right\} & \boldsymbol{0}^{\operatorname{rank}(A) \times n - \operatorname{rank}(A)} \\ \boldsymbol{0}^{n - \operatorname{rank}(A) \times \operatorname{rank}(A)} & \boldsymbol{0}^{n - \operatorname{rank}(A) \times n - \operatorname{rank}(A)} \end{bmatrix}$ *where* $\boldsymbol{0}$ *is the zero matrix of the appropriate dimension.*

Computing of all $\mu_i \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right)$ and $\left\| \boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right\|_i$ tends to be complex or impossible for $\dim \left(\boldsymbol{B}\,\boldsymbol{C}K \right) > 1$. Since $\dim \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right) = \dim \left(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right) = 3 \dim \left(\boldsymbol{B}\,\boldsymbol{C}K \right)$ and (4.27) tends to be untrue for most $i = 1, \infty$. Thus, this thesis will focus on a symbolic computation for $i = 2$.
The major problem is the special eigenvalue problem for $\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right)$ and $\left(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right)^* \left(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right)$. Because the Kronecker notation allows a block representation of the matrices, this thesis will use the Williamson[d] theorem [113].

[d] John Williamson *1901†1949, Scottish mathematician

Theorem 4.2. *The set σ of a matrix $\boldsymbol{M}$ of the form*

$$\boldsymbol{M} = \boldsymbol{f}\left(\boldsymbol{\mathcal{M}}_1, \cdots, \boldsymbol{\mathcal{M}}_k\right) \in \mathbb{R}^{n\times n} \wedge \boldsymbol{\mathcal{M}}_i \in \mathbb{R}^{\frac{n}{\sqrt{k}}\times\frac{n}{\sqrt{k}}}\ \forall\, i = 1, \cdots, k \tag{4.36}$$

where $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ is a matrix-valued function, holds

$$\sigma\left(\boldsymbol{M}\right) = \sigma\left(\boldsymbol{f}\left(\operatorname{diag}\left\{\sigma\left(\boldsymbol{\mathcal{M}}_1\right)\right\}, \cdots, \operatorname{diag}\left\{\sigma\left(\boldsymbol{\mathcal{M}}_k\right)\right\}\right)\right) \tag{4.37}$$

Applied to the given problem, it means that

$$\left|\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) - \boldsymbol{I}^{3n}\lambda\right| = \left|\boldsymbol{\Gamma}_t^T \otimes \boldsymbol{C}^T\,\boldsymbol{B}^T K + \boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{I}^{3n}\lambda\right| = \tag{4.38}$$

$$= (-\lambda)^{2n}\prod_{i=1}^{n}\left(\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K\right) - \lambda\right) -$$

$$-\,(-\lambda)^{n}\prod_{i=1}^{n}\left(\lambda_i\left(-\boldsymbol{C}^T\,\boldsymbol{B}^T K\right)\right)^2 - (-\lambda)^{n}\prod_{i=1}^{n}\left(\lambda_i\left(\boldsymbol{C}^T\,\boldsymbol{B}^T K\right)\right)^2 =$$

$$= (-\lambda)^{2n}\prod_{i=1}^{n}\left(\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K\right) - \lambda\right) - 2\,(-\lambda)^{n}\prod_{i=1}^{n}\left(\lambda_i\left(-\boldsymbol{C}^T\,\boldsymbol{B}^T K\right)\right)^2 \tag{4.39}$$

Since $\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) = \left[\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right]^T$, the roots of (4.39) must be real and because the polynomials major power is built from products, and so the polynomial must be monic. Also, its span is $3n - n = 2n$. Since it is monic with $\Im\left(\lambda_i\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right)\right) = 0$ it can also be rewritten as

$$\prod_{i=1}^{3n}\left(\lambda - \left(\lambda_i\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right)\right)\right) = \sum_{i=0}^{3n}\left(-1\right)^i a_{3n-i}\lambda^{3n-i} \tag{4.40}$$

Then, the Newton[e] identities [70] can be applied.

Theorem 4.3. *Let $e_k\left(a_1, \cdots, a_n\right)$ denote the elementary symmetric polynomical $\sum_{1\le j_1<j_2<\cdots\le j_k} a_{j_1}\cdots a_{j_k}$ and $p_k\left(a_1, \cdots, a_n\right)$ the k-th power sum $\sum_{i=1}^{n} a_i^k$, then the Newton identities are stated*

$$k e_k = \sum_{i=1}^{k}\left(-1\right)^{i-1} e_{k-i} p_i \tag{4.41}$$

Now in reverse, every ith power sum of the latent roots can be expressed as the elementary symmetrix polynomoal of the respective a_i:

$$\sum_{i=1}^{3n}\lambda_i^0\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right) = a_{3n} = 1 \tag{4.42}$$

$$\sum_{i=1}^{3n}\lambda_i^1\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right) = a_{3n-1} \tag{4.43}$$

$$\sum_{i=1}^{3n}\lambda_i^2\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right) = a_{3n-1}^2 - 2a_{3n-2} \tag{4.44}$$

$$\sum_{i=1}^{3n}\lambda_i^3\left(\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)^* + \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right) = a_{3n-1}^3 - 3a_{3n-1}a_{3n-2} + 3a_{n-3} \tag{4.45}$$

$$\vdots$$

[e] Isaac Newton *1643†1727, English physicist, mathematician, astronomer, philosopher, alchemist and theologian

Where (4.43) would also be the solution to the Law of Vieta[f] [32] with substitution of $a_{3n} = 1$. Now that both the matrix norm (4.34) and the matrix measure (4.29) are only functions of maximums among the set of latent roots, there is a possibility to set bounds for the extreme values among the solutions of (4.39), subject to the constraints (4.43), (4.44). Hence, the method of Lagrange[g] multipliers [84] will be used. The extremes in the function

$$\alpha\left(\lambda_2\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right), \cdots, \lambda_{3n}\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right)\right) = \tag{4.46}$$

$$= \lambda_1\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right) = const$$

have to be found, subject to the constraints (4.43), (4.44)

$$\beta\left(\lambda_2\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right), \cdots, \lambda_{3n}\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right)\right) = \tag{4.47}$$

$$= \sum_{i=1}^{3n} \lambda_i^1\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right) - a_{3n-1} = 0$$

$$\gamma\left(\lambda_2\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right), \cdots, \lambda_{3n}\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right)\right) = \tag{4.48}$$

$$= \sum_{i=1}^{3n} \lambda_i^2\left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)\right) - a_{3n-1}^2 + 2a_{3n-2} = 0$$

So that the Lagrange function

$$\ell = \alpha - \varphi_1\beta - \varphi_2\gamma \tag{4.49}$$

where φ_1, φ_2 are the Lagrange multipliers, solves the extremes of α for

$$\nabla\ell = 0 \tag{4.50}$$

Out of the system of (4.50), $3n+1$ equations are obtained, two of those representing (4.43) and (4.44). All other $3n-1$ conditions are

$$\lambda_1 + \varphi_1 + 2\varphi_2\lambda_i = 0 \;\forall\; i = 2, \cdots, 3n \tag{4.51}$$

The system (4.51) is now fully determined and its solution is

$$\lambda_i = \lambda_j \;\forall\; i, j = 2, \cdots, 3n \tag{4.52}$$

in the extremes of any λ_1. Plugging this condition for an extreme in λ_1 in (4.43) and (4.44), the results are

$$\lambda_1 + (3n-1)\,\lambda_2 = a_{3n-1} \tag{4.53}$$

$$\lambda_1^2 + (3n-1)\,\lambda_2^2 = a_{3n-1}^2 - 2a_{3n-2} \tag{4.54}$$

Plugging (4.53) into (4.54) to obtain a single expression for λ_1,

$$\lambda_1^2 + (3n-1)\left(\frac{a_{3n-1} - \lambda_1}{3n-1}\right)^2 = a_{3n-1}^2 - 2a_{3n-2} \tag{4.55}$$

$$\lambda_1^2 - \frac{2a_{3n-1}}{3n}\lambda_1 + a_{3n-1}^2\left(\frac{2-3n}{3n}\right) + a_{3n-2}\left(\frac{6n-2}{3n}\right) = 0 \tag{4.56}$$

Lemma 4.5. *The bounds of the latent roots of* (4.39) *are within the interval*

$$\left[\frac{a_{3n-1}}{3n} + \sqrt{a_{3n-1}^2\left(\frac{9n^2-6n+1}{9n^2}\right) - a_{3n-2}\left(\frac{6n-2}{3n}\right)}, \right. \tag{4.57}$$

$$\left. \frac{a_{3n-1}}{3n} - \sqrt{a_{3n-1}^2\left(\frac{9n^2-6n+1}{9n^2}\right) - a_{3n-2}\left(\frac{6n-2}{3n}\right)}\right] =$$

[f] Francois Viete *1540†1603, French mathematician

[g] Joseph-Louis Lagrange *1736†1813, Italian mathematician and astronomer

$$= \left\{ \lambda \left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K) \right) \in \mathbb{R} \left| \frac{a_{3n-1}}{3n} + \sqrt{a_{3n-1}^2 \left(\frac{9n^2-6n+1}{9n^2} \right) - a_{3n-2} \left(\frac{6n-2}{3n} \right)} > \right. \right.$$

$$\left. \left. > \lambda \left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K) \right) > \frac{a_{3n-1}}{3n} - \sqrt{a_{3n-1}^2 \left(\frac{9n^2-6n+1}{9n^2} \right) - a_{3n-2} \left(\frac{6n-2}{3n} \right)} \right\}$$

It is, however, not useful to know the bounds of the zeros that are to be found as a function of the polynomial coefficients of (4.39) without knowing the coefficients. Having a close look at the products, some of the terms may be generalized [32], such as

$$\lambda^{3n} + \lambda^{3n-1} \left(-\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right) + \tag{4.58}$$

$$+\lambda^{3n-2} \left(\frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right)^2 - \frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i^2 \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right) \right) +$$

$$+ \cdots \lambda^{3n-3} + \cdots$$

Looking at the definition of the coefficients in (4.40), one may note that the Newton identity uses alternating signs [70]. Therefore, whenever using (4.57), the following relations hold:

$$a_{3n-1} = \sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \tag{4.59}$$

$$a_{3n-2} = \frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right)^2 - \frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i^2 \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right) \tag{4.60}$$

Now the matrix measure induced by the 2-norm can be expressed as

$$\mu_2 \left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K \right) = \max_j \left(\frac{\lambda_j \left((\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K) + (\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* \right)}{2} \right) = \tag{4.61}$$

$$= \frac{1}{2} \left[\frac{\left(\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right)}{3n} + \right.$$

$$+ \left[\left(\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right)^2 \left(\frac{9n^2-6n+1}{9n^2} \right) - \left(\frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right)^2 - \right. \right.$$

$$\left. \left. \left. - \frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i^2 \left(-\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) \right) \right) \left(\frac{6n-2}{3n} \right) \right]^{1/2} \right]$$

To complete the computation of (4.27), a representation of $\left\| \boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K \right\|_2$ must be found. It is well known that a numeric solution for the singular values is possible, as stated above, but the given method will approach it through the symbolic solution of the eigenvalues of $(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K)^*$ $(\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K)$, very similar to the technique used above. First of all, the special eigenvalue problem can be rewritten as

$$\left| (\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K)^* (\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K) - \boldsymbol{I}^{3n}\lambda \right| = \tag{4.62}$$

$$= \left| \left(\boldsymbol{\Gamma}_{t-\Delta t_D}^T \otimes \boldsymbol{C}^T\,\boldsymbol{B}^T K \right) (\boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}K) - \boldsymbol{I}^{3n}\lambda \right| =$$

$$= \left| \left(\boldsymbol{\Gamma}_{t-\Delta t_D}^T \boldsymbol{\Gamma}_{t-\Delta t_D} \right) \otimes \boldsymbol{C}^T\,\boldsymbol{B}^T \boldsymbol{C}\,\boldsymbol{B}K^2 - \boldsymbol{I}^{3n}\lambda \right| =$$

$$= \prod_{i=1}^{n} \left(\lambda_i \left(2\boldsymbol{C}^T\,\boldsymbol{B}^T \boldsymbol{C}\,\boldsymbol{B}K^2 \right) - \lambda \right)^3 + 2 \prod_{i=1}^{n} \lambda_i^3 \left(2\boldsymbol{C}^T\,\boldsymbol{B}^T \boldsymbol{C}\,\boldsymbol{B}K^2 \right) -$$

$$-3\prod_{i=1}^{n}\lambda_i^2\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\left(\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)-\lambda\right)$$

So that, obviously, in this case

$$a_{3n-1}=\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right) \tag{4.63}$$

$$a_{3n-2}=\frac{1}{2}\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)^2-\frac{1}{2}\sum_{i=1}^{n}3\lambda_i^2\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right) \tag{4.64}$$

Again, because $\left[\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)^*\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)\right]^T=\left[\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)^*\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)\right]$ holds and because every singular value is a square root of the eigenvalue of the self-adjoint operator, it is possible to denote

$$\left\|\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right\|_2=\max_j\left(\sigma_j\left(\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)\right)= \tag{4.65}$$

$$=\left[\frac{\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)}{3n}+\left[\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)^2\left(\frac{9n^2-6n+1}{9n^2}\right)-\right.\right.$$

$$\left.\left.-\left(\frac{1}{2}\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)^2-\frac{1}{2}\sum_{i=1}^{n}3\lambda_i^2\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)\left(\frac{6n-2}{3n}\right)\right]^{1/2}\right]^{1/2}$$

Lemma 4.6. *The criterion for a zero eigenvalue of the coupling term and a Hopf bifurcation in the differential equations of Smith Synchronization is*

$$\Re\left(\varsigma\right)=\frac{1}{2}\left[\frac{\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K-\boldsymbol{C}^T\,\boldsymbol{B}^TK\right)\right)}{3n}+\right. \tag{4.66}$$

$$+\left[\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K-\boldsymbol{C}^T\,\boldsymbol{B}^TK\right)\right)^2\left(\frac{9n^2-6n+1}{9n^2}\right)-\left(\frac{1}{2}\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K-\boldsymbol{C}^T\,\boldsymbol{B}^TK\right)\right)^2-\right.\right.$$

$$\left.\left.\left.-\frac{1}{2}\left(\sum_{i=1}^{n}\lambda_i^2\left(-\boldsymbol{B}\,\boldsymbol{C}K-\boldsymbol{C}^T\,\boldsymbol{B}^TK\right)\right)\right)\left(\frac{6n-2}{3n}\right)\right]^{1/2}\right]+$$

$$+\left[\frac{\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)}{3n}+\left[\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)^2\left(\frac{9n^2-6n+1}{9n^2}\right)-\right.\right.$$

$$\left.\left.-\left(\left(\frac{1}{2}\left(\sum_{i=1}^{n}3\lambda_i\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)^2-\frac{1}{2}\sum_{i=1}^{n}3\lambda_i^2\left(2\boldsymbol{C}^T\,\boldsymbol{B}^T\boldsymbol{C}\,\boldsymbol{B}K^2\right)\right)\left(\frac{6n-2}{3n}\right)\right]^{1/2}\right]^{1/2}=0$$

The criterion holds for the trivial case of $K=0$. However, there is no possibility of realizing a control, then. If the elements of the coupling matrices hold the same condition, that is the trivial solution of setting all of them zero for inducing a synchronized state, there is no possibility of granting synchronization for arbitrary delays. There is, nevertheless, a possibility of spanning a parameter space of the coupling terms and the delay that still satisfies the boundary stability for the coupling differential equation so that synchronization is still possible. The condition is stated in the second theorem of Mori [73].

Theorem 4.4. *When* (4.66) *does not hold, the coupling terms of* (4.12) *are instable, whenever the roots of*

$$\left|\left[\boldsymbol{I}^{3n}\lambda-\boldsymbol{\Gamma}_t\otimes\boldsymbol{B}\,\boldsymbol{C}K-\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D\lambda}\right]\right|\in\left[0+0j,\varsigma\right] \tag{4.67}$$

where $[0+0j,\varsigma]$ *is a rectangular parameter plane in the s-plane, that is bound in* $\Re$ *by* $[0,\Re\left(\varsigma\right)]=\{\Re\left(\lambda\right)\in\mathbb{R}|0\leq\Re\left(\lambda\right)\leq\Re\left(\varsigma\right)\}$. $\Re\left(\varsigma\right)$ *is defined by the criterion from 4.66, and in* $\Im$ *by* $[\pm\Im\left(\varsigma\right)j]=\{\Im\left(\lambda\right)\in\mathbb{C}|-\Im\left(\varsigma\right)j\leq\Im\left(\lambda\right)\leq+\Im\left(\varsigma\right)j\}$ *where* $\Im\left(\varsigma\right)$ *is defined by*

$$\Im\left(\varsigma\right)=\mu_2\left(-j\boldsymbol{\Gamma}_t\otimes\boldsymbol{B}\,\boldsymbol{C}K\right)+\left\|\boldsymbol{\Gamma}_{t-\Delta t_D}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right\|_2 \tag{4.68}$$

In fact, $\Im(\varsigma)$ is very similar to (4.27), which is why only the first part has to be computed. Therefore, $\frac{1}{2}\max\left(\sigma\left(\left[\boldsymbol{\Gamma}_t \otimes -\boldsymbol{B}\,\boldsymbol{C}Kj\right] + \left[\boldsymbol{\Gamma}_t \otimes -\boldsymbol{B}\,\boldsymbol{C}Kj\right]^*\right)\right)$ is needed. The term simplifies to

$$\mu_2\left(-j\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) = \frac{1}{2}j\max\left(\sigma\left(\left[\boldsymbol{\Gamma}_t^T - \boldsymbol{\Gamma}_t\right] \otimes \boldsymbol{B}\,\boldsymbol{C}K\right)\right) \tag{4.69}$$

Here it is important to note that any real matrix $\boldsymbol{M} - \boldsymbol{M}^T$ with $\{m_{ij}\} \in \mathbb{R}$ has purely imaginary latent roots. Therefore, (4.57) would not hold. Expanding by j, the latent roots become purely real and symmetrical to the imaginary axis. Hence it is well known, that

$$\sum_{i=1}^{n} \lambda_i\left(\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K\right) = 0 \tag{4.70}$$

so that the second Newton identity simplifies to

$$a_{3n-2} = -\frac{1}{2}\sum_{i=1}^{n}\left(j\lambda_i\right)^2\left(\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K\right) \tag{4.71}$$

and the first one to zero. That is why

$$\mu_2\left(-j\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) = \sqrt{\frac{3n-1}{12n}\sum_{i=1}^{n}\left(j\lambda_i\right)^2\left(\boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{C}^T\,\boldsymbol{B}^T K\right)} \tag{4.72}$$

The decision whether the roots of (4.67) are located in the instable rectangle or not turns out to be hard. It is more convenient to instead find
$\sigma\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K - \boldsymbol{\Gamma}_{t-\Delta t_D} \otimes \boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right)$ and ensure its stability. It is plain to see that this matrix is always singular. The characteristic polynomial has the leading terms

$$\prod_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right) - \lambda\prod_{i=1}^{n}\lambda_i\left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]} - \boldsymbol{B}\,\boldsymbol{C}K\right) - \lambda \tag{4.73}$$

So that Newton's first identity is

$$a_{n-1} = \sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right) + \lambda_i\left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]} - \boldsymbol{B}\,\boldsymbol{C}K\right) \tag{4.74}$$

Now, that $\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right) = \left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right)\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)$ is satisfied,
$\sigma\left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]} - \boldsymbol{B}\,\boldsymbol{C}K\right) \subseteq \left\{\sigma\left(-\boldsymbol{B}\,\boldsymbol{C}K\right) + \sigma\left(\boldsymbol{B}\,\boldsymbol{C}Ke^{-\Delta t_D[0+0j,\varsigma]}\right)\right\}$ holds. By that, a_{n-1} simplifies to $a_{n-1} = \sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)$ and a_{n-2} to

$$a_{n-2} = \frac{1}{2}\Bigg(\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\right)^2 - \tag{4.75}$$

$$-\left(\left(2e^{-2\Delta t_D[0+0j,\varsigma]} - 2e^{-\Delta t_D[0+0j,\varsigma]} - 1\right)\sum_{i=1}^{n}\lambda_i^2\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\right)\Bigg)$$

Furthermore, the order of this particular equation is reduced due to its singularities.

Lemma 4.7. *In every case that* (4.66) *does not hold, stability can be granted when*

$$\Re\Bigg[\frac{\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)}{2n} + \Bigg(\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\right)^2\frac{4n^2-4n+1}{4n^2} - \Bigg(\left(\sum_{i=1}^{n}\lambda_i\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\right)^2 - \tag{4.76}$$

$$-\left(\left(2e^{-2\Delta t_D[0+0j,\varsigma]} - 2e^{-\Delta t_D[0+0j,\varsigma]} - 1\right)\sum_{i=1}^{n}\lambda_i^2\left(-\boldsymbol{B}\,\boldsymbol{C}K\right)\right)\Bigg)\frac{2n-1}{2n}\Bigg)^{1/2}\Bigg] = \zeta \leq 0$$

4.2 Zero Row Sum Negative Trace Control

The Weakness of the Smith Synchronization is its second mode. This mode sets an upper and lower bound to the asymptotic synchronization error. A control that does not induce such a mode is desirable. Since the Lambert W function allows an analytic solution of time-delayed couplings, this design issue is simplified. Referring to the example of the two-mass-damper system again, there already is an example for an optimal type of coupling matrix, that is of form $\begin{bmatrix} -m & m \\ m & -m \end{bmatrix}\ \forall\, m > 0$. Since the modes of transcendental coupling equations are defined by the modal matrix of the matrix $\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t$, the idea is to bring this matrix to a form that holds

$$\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t \propto \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \tag{4.77}$$

$$\mathrm{tr}\left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t\right) < 0 \tag{4.78}$$

$$\Im\left(\mathrm{tr}\left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t\right)\right) = 0 \tag{4.79}$$

With $\mathbf{\Gamma}_{t-\Delta t_D} = \left[\gamma_{t-\Delta t_{D}ij}\right] \in \mathbb{R}^{2\times 2}$, $\mathbf{\Gamma}_t = \left[\gamma_{t_{ij}}\right] \in \mathbb{R}^{2\times 2}$, Δt_D, the stated problem is a function with a set of nine unknowns when only considering the principle branch of the Lambert W function and a function with a infinite set of unknowns when considering all branches of the function. For the sake of well-posedness of the problem, only the principle branch will be considered at first. Furthermore, since engineering systems are investigated, the real case is that the delayed system cannot be accessed in the nondelayed domain, so that

$$\gamma_{t_{1j}} = 0\ \forall\, j \vee \gamma_{t_{2j}} = 0\ \forall\, j \tag{4.80}$$

W.l.o.g., the second case $\gamma_{t_{2j}} = 0\ \forall\ j$ can be considered. The relation (4.77) can be expressed through the use of notation $\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t = [a_{ij}] \in \mathbb{R}^{2\times 2}$ as

$$a_{11} + a_{12} = a_{11} + a_{21} = a_{21} + a_{22} = 0 \tag{4.81}$$

The fourth condition, $a_{12} + a_{22} = 0$, follows from the first three. The problem is now stated with the cost function $\underline{f} = \begin{bmatrix} a_{11} + a_{12} \\ a_{11} + a_{21} \\ a_{21} + a_{22} \end{bmatrix}$, and the nonlinear constraints

$$\beta = \Im\left(\mathrm{tr}\left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t\right)\right) \tag{4.82}$$

$$\beta' = \mathrm{tr}\left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\mathbf{\Gamma}_t \Delta t_D}\right) + \mathbf{\Gamma}_t\right) \tag{4.83}$$

The solution to this problem is the solution in

$$\boldsymbol{L} = \nabla\left(\left|\underline{f}\right|\right) + \varphi \nabla\left(\beta\right) + \varphi' \nabla\left(\beta'\right) = \mathbf{0} \tag{4.84}$$

that satisfies the Karush[h]-Kuhn[i]-Tucker[j] conditions $\beta' < 0$, $\beta = 0$, $\varphi' > 0$, and $\varphi'\beta' = 0$ [52, 49]. Since the problem is too complex to solve analytically, a procedure has to be used that was invented by M.C. Biggs [10] and that uses a recursive algorithm that executes three subsections for every step, as to be seen in Algorithm 5.

[h] William Karush *1917†1997, American mathematician
[i] Harold William Kuhn *1925, American mathematician
[j] Albert William Tucker *1905†1995, Canadian mathematician

Algorithm 5 Bigg's Algorithm

1: **while** Value of function is not smaller than threshold **do**
2: Compute second gradient of the cost function using quasi-Newton
3: Find the optimal step direction using quadratic programming
4: forward one step in that direction
5: **end while**

The second gradient of the cost function is computed using the Broyden-Fletcher-Goldfarb-Shanno method [114], that is a quasi-Newton subclass.

$$\Delta\left(\left|\underline{f}_{k+1}\right|\right) = \Delta\left(\left|\underline{f}_{k}\right|\right) + \frac{\left(\boldsymbol{L}_{k+1} - \boldsymbol{L}_{k}\right)^{2}}{\left(\boldsymbol{L}_{k+1} - \boldsymbol{L}_{k}\right) h} - \frac{\Delta\left(\left|\underline{f}_{k}\right|\right) h^{2} \Delta\left(\left|\underline{f}_{k}\right|\right)}{h\left(\left|\underline{f}_{k}\right|\right) h} \tag{4.85}$$

where h is the stepsize. Then, locally, the problem of finding the direction for the next step has to be solved. The problem can be stated as

$$\boldsymbol{q} = \min\left(\frac{1}{2} h^{2} \Delta\left(\boldsymbol{L}_{k}\right) + \nabla\left(\left|\underline{f}_{k}\right|\right) h\right) \tag{4.86}$$

subject to

$$\nabla\left(\beta_{k}\right) h + \beta_{k} = 0 \tag{4.87}$$

$$\nabla\left(\beta_{k}'\right) h + \beta_{k}' \leq 0 \tag{4.88}$$

To solve this subproblem, the projection Method is used [36]. The name of the method origins from the use of a projection matrix $\boldsymbol{Q}$ that ensures that the minimum found for (4.86) is within the space that fulfills the constraints.

$$\boldsymbol{Q}^{T}\begin{bmatrix}\nabla\left(\beta_{k}\right) \\ \nabla\left(\beta_{k}'\right)\end{bmatrix}^{T} = \begin{bmatrix}\boldsymbol{R} \\ 0\end{bmatrix} \tag{4.89}$$

where $\boldsymbol{Q}^{*}\boldsymbol{Q} = \boldsymbol{I}$. By that, $\boldsymbol{Q}$ is orthogonal to the constraints and every vector mapped by $\boldsymbol{Q}$ is mapped to the bounds of the constraints. Now $\boldsymbol{Q}$ can be used to project $\boldsymbol{q}$.

$$\boldsymbol{q} = \min\left(\frac{1}{2} h \boldsymbol{Q}^{T} \Delta\left(\boldsymbol{Q}\right) h + \nabla\left(\underline{f}\right) \boldsymbol{Q} h\right) \tag{4.90}$$

with

$$\nabla\left(\frac{1}{2} h \boldsymbol{Q}^{T} \Delta\left(\boldsymbol{Q}\right) h + \nabla\left(\underline{f}\right) \boldsymbol{Q} h\right) = \boldsymbol{Q}^{T} \Delta\left(\boldsymbol{Q}\right) h + \boldsymbol{Q}^{T} \nabla\left(\underline{f}\right) = 0 \tag{4.91}$$

This local solution for h is linear, but requires a initial value that fulfills the constraints. Since the values $[\gamma_{t-\Delta t_{D11}}\ \gamma_{t-\Delta t_{D12}}\ \gamma_{t-\Delta t_{D21}}\ \gamma_{t-\Delta t_{D22}}\ \gamma_{t_{11}}\ \gamma_{t_{12}}] = [-1\ 0.1\ 0.1\ -1\ -1\ 0.1]$ fulfill $\beta = 0$, $\beta' \approx -2\ \forall\ \Delta t_{D} = 0.1\upsilon\ \forall\ \upsilon \in Y$, those values were chosen for the initiation of the optimization, and Bigg's Algorithm was run with $\Delta t_{D} = 0.1\upsilon\ \forall\ \upsilon \in Y$ and a minimization threshold of 10^{-8}, where $Y = \left\{\upsilon \in \mathbb{Z} \middle| 1 \leq \upsilon \leq 20\right\}$. The optimization yields the results in Table 4.1.

Δt_D	$\gamma_{t-\Delta t_{D11}}$	$\gamma_{t-\Delta t_{D12}}$	$\gamma_{t-\Delta t_{D21}}$	$\gamma_{t-\Delta t_{D22}}$	$\gamma_{t_{11}}$	$\gamma_{t_{12}}$	
0.1	−0.36234	0.819565	0.791364	−0.713088	−0.422024	−0.072129	✓
0.2	0.118086	0.175844	1.33857	0.0903052	−0.0883816	−0.385849	
0.3	1.21343	0.861509	0.246101	−0.0914255	−0.880339	−0.444115	
0.4	1.40948	1.54927	1.03617	−0.0141146	−1.11163	−0.605053	
0.5	−1.35794	1.35777	1.35784	−1.35782	$9.467\,10^{-6}$	−0.000118	✓
0.6	2.03034	0.589139	0.242523	−0.051150	−0.85327	−0.18014	
0.7	−0.751768	0.755260	0.766101	−0.734042	−0.0016186	−0.014351	✓
0.8	−0.972775	0.9371	0.980372	−0.936509	−0.0017998	−0.014660	✓
0.9	−0.0259161	1.09202	0.811053	−0.096633	−0.0105447	−0.61642	
1.0	−0.769295	0.774049	0.7782	−0.769046	0.00091	−0.006322	✓
1.1	0.531440	1.08483	$1.901\,10^{-5}$	$3.795\,10^{-6}$	−0.28431	−0.58037	
1.2	−0.647457	0.65366	0.661625	−0.667978	0.00021385	0.0119947	✓
1.3	−0.603301	0.603616	0.603296	−0.603396	$3.0017\,10^{-5}$	−0.0001424	✓
1.4	1.54954	1.50354	0.518355	0.23299	−0.512083	−0.41772	
1.5	−0.471261	0.467458	0.473025	−0.4585	0.0001782	−0.003697	✓
1.6	−0.475596	0.47463	0.478619	−0.47141	0.000223	−0.003184	✓
1.7	−0.447427	0.44744	0.447381	−0.44736	$1.9024\,10^{-5}$	$-6.36\,10^{-5}$	✓
1.8	0.051978	0.049176	0.0213	−0.015262	−0.06014	−0.02644	✓
1.9	−0.361329	0.3622	0.364007	−0.36056	$3.289\,10^{-5}$	−0.002324	✓
2.0	0.684542	−0.187827	0.241006	−0.1315	−0.32421	0.13228	✓

Table 4.1: Optimization Results

The last column indicates whether the optimization converged or not where the convergence criterion is

$$|f| \overset{?}{\leq} 10^{-8} \tag{4.92}$$

As a first verification of these results, every converged coupling function implemented using the test function $\dot{x} = -x + u + u'$, where u is the input resulting from the coupling matrices and $u' = \sin t$. The attractors of both systems can be seen in Figure 4.5.

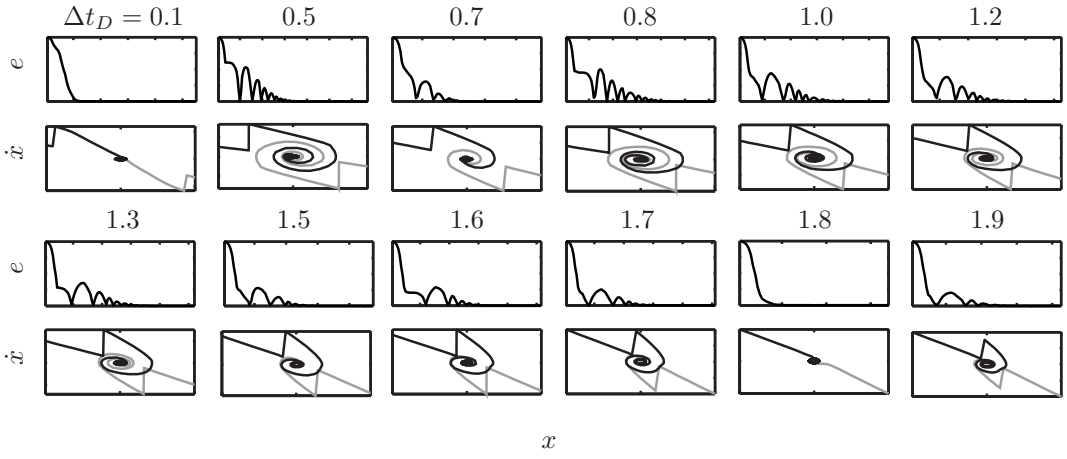


Figure 4.5: Attractors for the Test Function Based on Optimization

Though all attractors converge $e \to 0$, because there is no longer an undamped antiphase mode, yet both systems require the input function in realtime for this convergence. An optimal solution would damp e so heavily, that a delayed input of the delayed system would lead to convergence, too. The reason for that behavior is that the real parts of the damped eigenvalue are relatively small. Furthermore, there is no closed form solution to the problem yet. Nevertheless, comparing the eigenvalues of the solutions to the coupling matrix elements, there are some regularities. The

eigenvalues of the coupling are approximately the first-row first-column element of the non-delayed coupling matrix. Furthermore, the delayed coupling matrix is of type

$$\sum_{j=1}^{2} \gamma_{t-\Delta t_{Dij}} = 0 \; \forall \; i \tag{4.93}$$

$$\operatorname{tr}\left(\Gamma_{t-\Delta t_{Dij}}\right) < 0 \tag{4.94}$$

$$\gamma_{t-\Delta t_{Dii}} = \gamma_{t-\Delta t_{Djj}} \; \forall \; i, j \tag{4.95}$$

Using these conditions, a closed form solution to the modes of the coupling is possible.

Lemma 4.8. *Coupling matrices of type* $\Gamma_{t-\Delta t_D} = \begin{bmatrix} -\gamma_{t-\Delta t_D} & \gamma_{t-\Delta t_D} \\ \gamma_{t-\Delta t_D} & -\gamma_{t-\Delta t_D} \end{bmatrix}$ *in the delayed domain and* $\Gamma_t = \begin{bmatrix} -\gamma_t & \gamma_t \\ 0 & 0 \end{bmatrix}$ *in the non-delayed domain induce synchronizability for two coupled systems.*

Proof. The solution to the matrix exponential of $\mathbf{\Gamma}_t$ turns out simple. Though the matrix is not nilpotent, its squares are always of form

$$\mathbf{\Gamma}_t^i = \begin{bmatrix} (-\gamma_t)^i & -(-\gamma_t)^i \\ 0 & 0 \end{bmatrix} \tag{4.96}$$

So that the matrix exponential simplifies to

$$e^{\Delta t_D \mathbf{\Gamma}_t} = \begin{bmatrix} e^{-\Delta t_D \gamma_t} & 1 - e^{-\Delta t_D \gamma_t} \\ 0 & 1 \end{bmatrix} \tag{4.97}$$

The next step, that is the argument of the Lambert W function, holds an interesting result.

$$\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\Delta t_D \mathbf{\Gamma}_t} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t} \tag{4.98}$$

This matrix is of the initially desired form. Therefore, logically, its Lambert W function is computed using Sylvester's formula.

$$W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\Delta t_D \mathbf{\Gamma}_t}\right) = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) \end{bmatrix} \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} = \tag{4.99}$$

$$= \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \frac{1}{2} W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)$$

Finally, the matrix that has the eigenbehavior of the coupling can be computed.

$$\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\Delta t_D \mathbf{\Gamma}_t}\right) + \mathbf{\Gamma}_t = \tag{4.100}$$

$$= \begin{bmatrix} \frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)}{2\Delta t_D} - \gamma_t & -\frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)}{2\Delta t_D} + \gamma_t \\ -\frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)}{2\Delta t_D} & \frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)}{2\Delta t_D} \end{bmatrix}$$

This resulting matrix may not be exactly of form $\begin{bmatrix} -m & m \\ m & -m \end{bmatrix}$, but nevertheless of form $\begin{bmatrix} -m_1 & m_1 \\ m_2 & -m_2 \end{bmatrix}$. Such matrices do not exactly have the modes $[1\ 1]$ and $[1\ -1]$, but they do always have the first mode, that is the synchronized state. The given matrix can be diagonalized to

$$\boldsymbol{V}^{-1} \left(\frac{1}{\Delta t_D} W\left(\mathbf{\Gamma}_{t-\Delta t_D} \Delta t_D e^{\Delta t_D \mathbf{\Gamma}_t}\right) + \mathbf{\Gamma}_t \right) \boldsymbol{V} = \begin{bmatrix} 0 & 0 \\ 0 & \frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) - \Delta t_D \gamma_t}{\Delta t_D} \end{bmatrix} \tag{4.101}$$

where

$$\boldsymbol{V} = \begin{bmatrix} 1 & \frac{2\Delta t_D \gamma_t - W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)}{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right)} \\ 1 & 1 \end{bmatrix} \tag{4.102}$$

□

The only task in designing the systems is now to keep the second eigenvalue negative, and for every given case in the region of a negative largest Lyapunov exponent, respectively. Therefore, the hopf bifurcations of that eigenvalue will be analyzed. The eigenvalue undergoes a Hopf bifurcation for all solutions to

$$W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) = \Delta t_D \gamma_t \tag{4.103}$$

First of all, it is noteable that alle branches of the Lambert W function $k \neq 0$ can be neglected, since the solutions to the function have a strictly monotonically decreasing real part for increasing $k \,\forall\, k > 0$ and for decreasing $k \,\forall\, k < 0$. Figure 4.6 depicts some branches of the solution to (4.103).

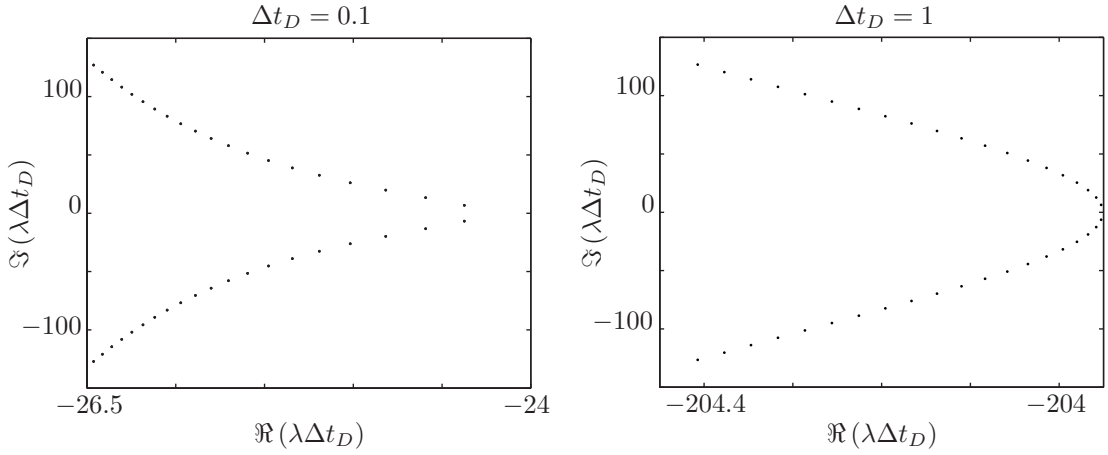


Figure 4.6: Monotonical Behavior of Real Part of Antiphase Latent Root

Therefore, the bifurcation analysis for the principal branch is sufficient. However, the function can undergo a Hopf bifurcation for the principal solution. Furthermore, another type of bifurcation will be of special interest. As parameters change, the solution to the Lambert W function may become imaginary. This solution may cause unsynchronized and even critical behavior. A simple bifurcation analysis in a multiparameter system can usually be conducted by holding one parameter constant and varying the other one. The eigenvalue is approximately the value of $-\gamma_t$ and exactly the value of $-\gamma_t$ for sufficiently large Δt_D.

$$\frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) - \Delta t_D \gamma_t}{\Delta t_D} \approx -\gamma_t \tag{4.104}$$

$$\lim_{\Delta t_D \to \infty} \frac{W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) - \Delta t_D \gamma_t}{\Delta t_D} = -\gamma_t \tag{4.105}$$

That means, that the Lambert W function converges to zero as Δt_D gets bigger. The simple reason is the asymptotic behavior of the function te^{-t}. According to these results, it is very likely, that large values will be used for γ_t to enhance synchronizability, whereas $\gamma_{t-\Delta t_D}$ will be used without any gain, that is a simple summation point in a control interpretation. Therefore, the bifurcation analysis is split into two cases. First, $\gamma_{t-\Delta t_D}$ is treated as a constant of value one, while $W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) - \Delta t_D \gamma_t$ is evaluated on the interval $\left\{\gamma_t \in \mathbb{R} \middle| -100 < \gamma_t < 100\right\}$ for the cases $\Delta t_D = \{0.1, 1, 10\}$. Then, γ_t is held constant with a value of 10. For larger values, the influence of γ_t is to big for a proper analysis. The same is true for Δt_D. $\gamma_{t-\Delta t_D}$ only influences the latent roots for sufficiently small $\gamma_{t-\Delta t_D}$, so that it is set to 0.1. For these constants, $W\left(-2\Delta t_D \gamma_{t-\Delta t_D} e^{-\Delta t_D \gamma_t}\right) - \Delta t_D \gamma_t$ is evaluated on the interval $\left\{\gamma_{t-\Delta t_D} \in \mathbb{R} \middle| -100 < \gamma_{t-\Delta t_D} < 100\right\}$. The values for the latent root are depicted in Figure 4.7.

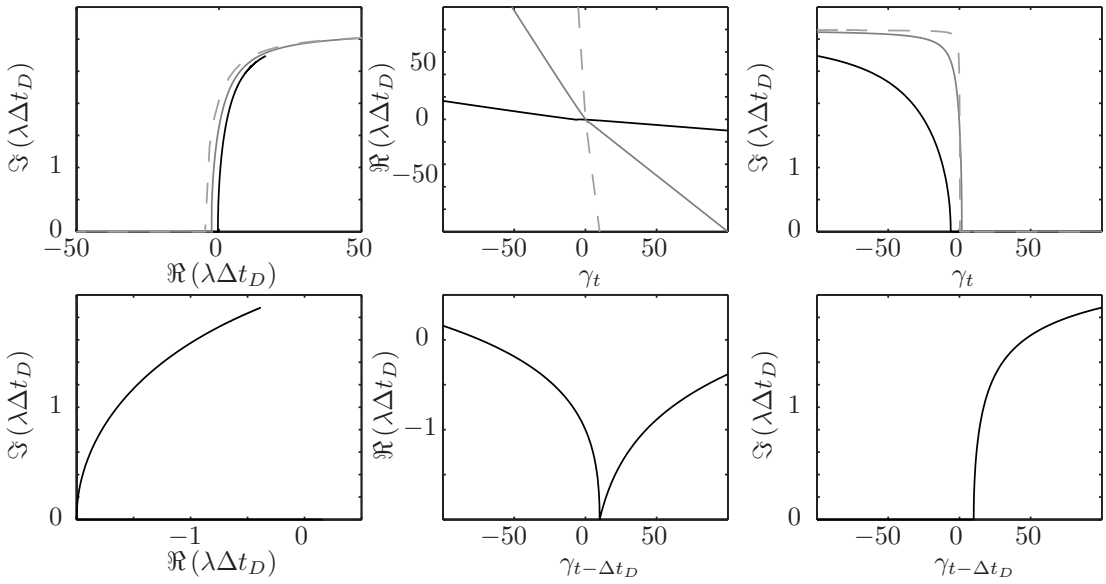


Figure 4.7: Bifurcations of the Antiphase Eigenvalue

Luckily, the root clearly undergoes a Hopf bifurcation for $\gamma_t = 0$ and remains unstable for $\gamma_t < 0$ but also remains strictly stable for $\gamma_t > 0$, no matter what Δt_D is considered for analysis. Also the imaginary part strictly converges to a numerical value. The imaginary part clearly disappears for all $\gamma_t > 2$. Analyzing the behavior of the sign of the latent root for $\gamma_{t-\Delta t_D}$, the real part of the root is strictly negative for positive $\gamma_{t-\Delta t_D}$, again. Imaginary values only occur for $\gamma_{t-\Delta t_D} > \gamma_t$. Following these design criteria, it is possible to turn the developed coupling into a control scheme. Now, appropriate design rules have been deduced for the synchronization. The coupling matrix in the delayed domain is chosen to be $\begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}$ and the coupling matrix in the non-delayed domain with a gain $\begin{bmatrix} -K & K \\ 0 & 0 \end{bmatrix}$ where K is chosen as high as possible, while the second eigenvalue of the coupling, that is mostly defined by K of course still has to be within the region of a negative largest Lyapunov exponent. With these statements, for the first time, the coupling scheme can be expressed as a control scheme by a block diagramm, that is depicted in Figure 4.8.

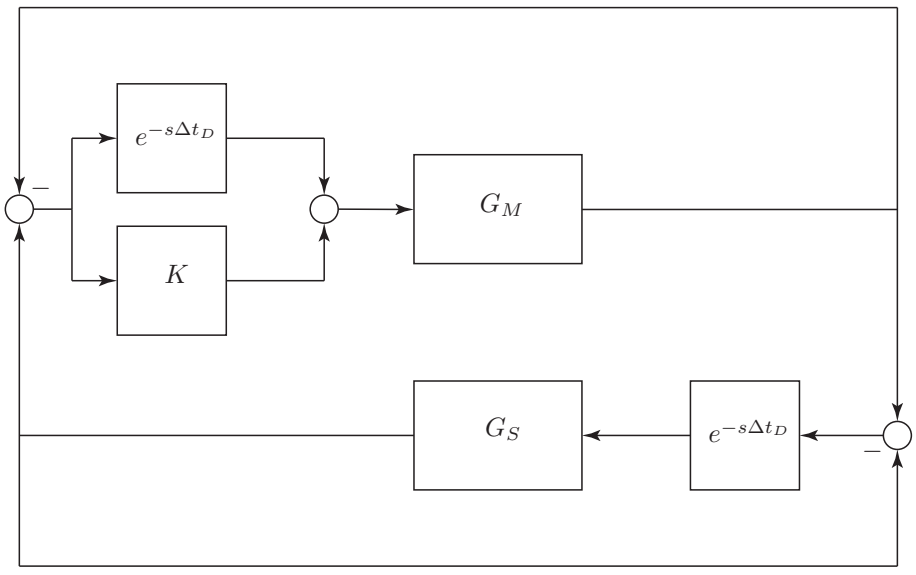


Figure 4.8: Zero Row Sum Negative Trace Control

Definition 4.8. *The control shown in Figure 4.8 is called zero row sum negative trace control.*

Example. The simple nonlinear dynamic system already considered to test the Smith synchronization is considered. The system is said to be described by the state space equation

$$\dot{\underline{x}}_i = \begin{bmatrix} x_{i_2}^2 x_{i_1} \\ \cos x_{i_2} \end{bmatrix} + \boldsymbol{I}^2 \underline{u}_i \; \forall \; i = M, S \tag{4.106}$$

The input function is chosen to be

$$\underline{u} = \underline{\sin t} \tag{4.107}$$

as an example for a continuous input function, and for another example

$$\underline{u} = \sqcap = \begin{cases} 0 & |t| > \frac{1}{2} \\ \frac{1}{2} & |t| = \frac{1}{2} \\ 1 & |t| < \frac{1}{2} \end{cases} \tag{4.108}$$

as an example for a discontinuous input function. This function is supposed to be very hard to synchronized due to its hard discontinuities. For both input function, to emphasize its input, the coupling factor K was set first 10 and then 100. The attractors of both systems, as they react to the inputs are depicted in Figure 4.9.

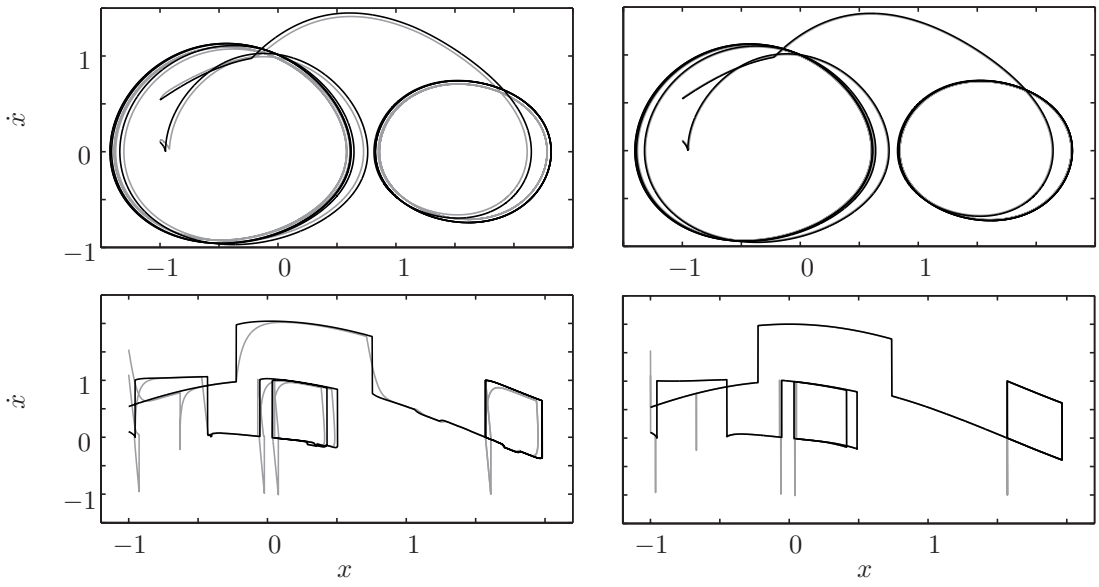


Figure 4.9: Attractors of the Zero Row Sum Negative Trace Control

For the $\sin t$ input function, the attractors of both system are hard to distinguish for $K = 100$. Even the $\sqcap$ input function, that causes high divergence of the state spaces of both systems for $K = 10$, matches both attractors with a high accuracy for $K = 100$.

4.3 Delay and Offset Measurement

The traditional IEEE 1588 protocol is event-triggered. The approach used in this thesis modifies this to be a continuous implementation, which makes the protocol a control problem rather than a programming problem. For flexibility and fault-tolerance, the implementation is supposed to measure delay and offset on both ports, master and slave. The block diagram of the principle is shown in Figure 4.10.

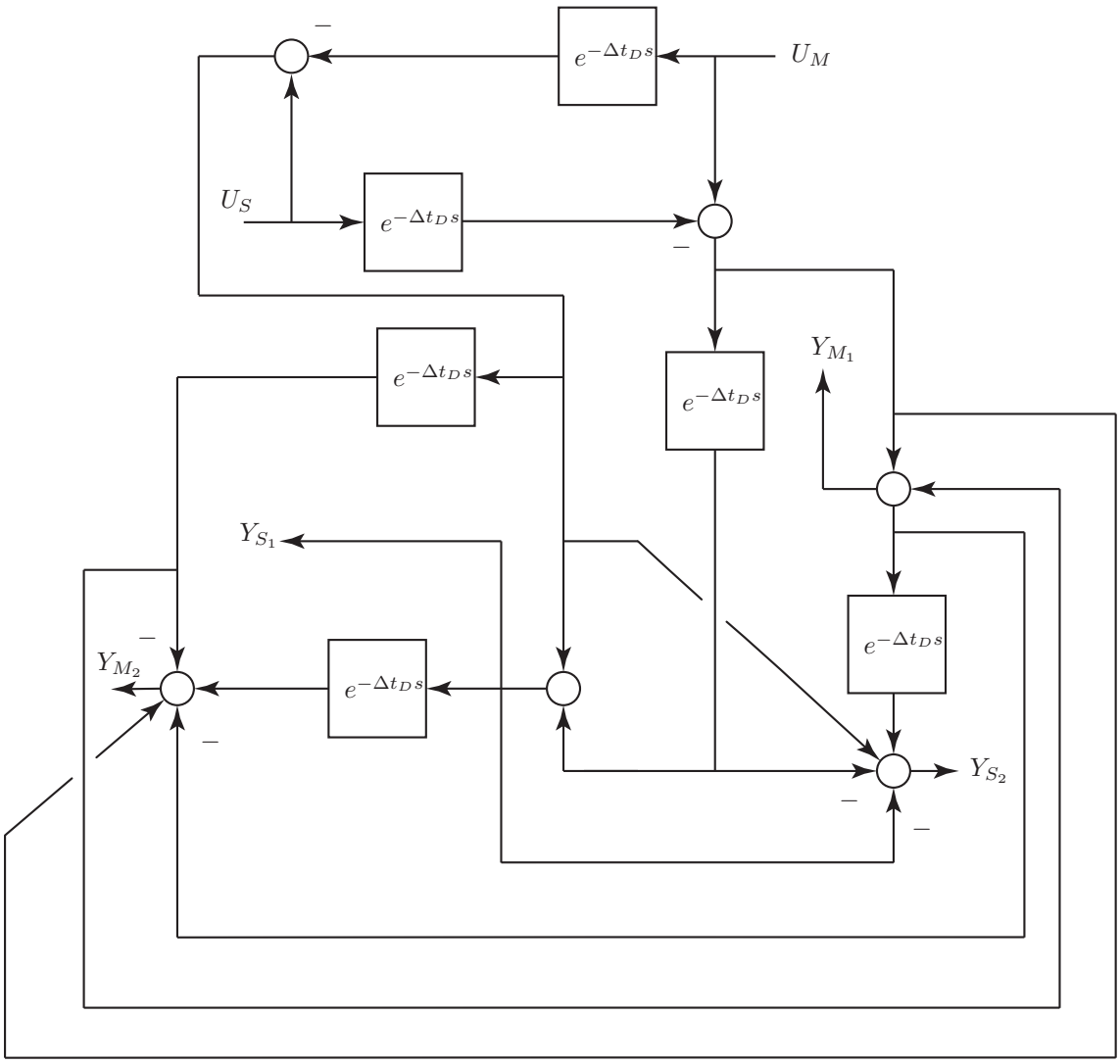


Figure 4.10: Continuous Clock Synchronization

Here u_S is the inner clock of the slave, u_M is the inner clock of the master, y_{M_1}, y_{M_2} are the time values measured at the master node and y_{S_1}, y_{S_2} are the values measured at the slave node. Let Δt_{O_M} be the offset of the master clock, Δt_{O_S} the offset of the slave clock, t_M the time dimension of the master clock, t_S the time dimension of the slave clock and $\varkappa_M$, $\varkappa_S$ the drift of those clocks, which may or may not be linear. Then

$$u_M = \Delta t_{O_M} + t_M = \Delta t_{O_M} + \varkappa_M t \tag{4.109}$$

$$u_S = \Delta t_{O_S} + t_S = \Delta t_{O_S} + \varkappa_S t \tag{4.110}$$

and

$$\int_0^\infty e^{-st} u_M dt = \frac{\Delta t_{O_M}}{s} + \frac{\varkappa_M}{s^2} = U_M \tag{4.111}$$

$$\int_0^\infty e^{-st} u_S dt = \frac{\Delta t_{O_S}}{s} + \frac{\varkappa_S}{s^2} = U_S \tag{4.112}$$

Definition 4.9. *The structure in Figure 4.10 is called continuous clock synchronization.*

Lemma 4.9. *The continuous clock synchronization puts out the isolated variables Δt_D, Δt_O, with respect to $\varkappa$.*

Proof. From Figure 4.10, the transfer functions of the system are

$$Y_{M_1} = e^{-\Delta t_D s}\left(U_S - e^{-\Delta t_D s} U_M\right) + U_M - U_S e^{-\Delta t_D s} = U_M\left(1 - e^{-2\Delta t_D s}\right) \tag{4.113}$$

$$Y_{S_1} = e^{-\Delta t_D s}\left(U_M - e^{-\Delta t_D s} U_S\right) + U_S - e^{-\Delta t_D s} U_M = U_S\left(1 - e^{-2\Delta t_D s}\right) \tag{4.114}$$

$$\begin{aligned} Y_{M_2} &= e^{-\Delta t_D s}\left(U_M e^{-\Delta t_D s} - U_S\right) + U_M - U_S e^{-\Delta t_D s} - Y_{M_1} + Y_{S_1} e^{-\Delta t_D s} = \\ &= U_M\left(1 + e^{-2\Delta t_D s}\right) - 2U_S e^{-\Delta t_D s} - Y_{M_1} + Y_{S_1} e^{-\Delta t_D s} \end{aligned} \tag{4.115}$$

$$\begin{aligned} Y_{S_2} &= U_S - U_M e^{-\Delta t_D s} - e^{-\Delta t_D s}\left(U_M - U_S e^{-\Delta t_D s}\right) + Y_{M_1} e^{-\Delta t_D s} - Y_{S_1} = \\ &= U_S\left(1 + e^{-2\Delta t_D s}\right) - 2U_M e^{-\Delta t_D s} + Y_{M_1} e^{-\Delta t_D s} - Y_{S_1} \end{aligned} \tag{4.116}$$

Plugging (4.111) and (4.111) into the four output signals, the result is

$$Y_{M_1} = \frac{\Delta t_{O_M}}{s} + \frac{\varkappa_M}{s^2} - \frac{\Delta t_{O_M}}{s} e^{-2\Delta t_D s} - \frac{\varkappa_M}{s^2} e^{-2\Delta t_D s} \tag{4.117}$$

$$Y_{S_1} = \frac{\Delta t_{O_S}}{s} + \frac{\varkappa_S}{s^2} - \frac{\Delta t_{O_S}}{s} e^{-2\Delta t_D s} s - \frac{\varkappa_S}{s^2} e^{-2\Delta t_D s} s \tag{4.118}$$

$$\begin{aligned} Y_{M_2} &= \frac{\Delta t_{O_M}}{s} + \frac{\varkappa_M}{s^2} + \frac{\Delta t_{O_M}}{s} e^{-2\Delta t_D s} + \frac{\varkappa_M}{s^2} e^{-2\Delta t_D s} s - \\ &- 2\frac{\Delta t_{O_S}}{s} e^{-\Delta t_D s} s - 2\frac{\varkappa_S}{s^2} e^{-\Delta t_D s} s - Y_{M_1} + Y_{S_1} e^{-\Delta t_D s} \end{aligned} \tag{4.119}$$

$$\begin{aligned} Y_{S_2} &= \frac{\Delta t_{O_M}}{s} + \frac{\varkappa_M}{s^2} + \frac{\Delta t_{O_M}}{s} e^{-2\Delta t_D s} + \frac{\varkappa_M}{s^2} e^{-2\Delta t_D s} s - \\ &- 2\frac{\Delta t_{O_S}}{s} e^{-\Delta t_D s} s - 2\frac{\varkappa_S}{s^2} e^{-\Delta t_D s} s + Y_{M_1} e^{-\Delta t_D s} - Y_{S_1} \end{aligned} \tag{4.120}$$

If these equations are transferred into the real domain again, they look like

$$y_{M_1} = \frac{1}{2\pi j}\int_{\delta - j\infty}^{\delta + j\infty} e^{st} Y_{M_1} ds = \Delta t_{O_M} + \varkappa_M t - \Delta t_{O_M} - \varkappa_M\left(t + 2\Delta t_D\right) = -2\varkappa_M \Delta t_D \tag{4.121}$$

$$y_{S_1} = \frac{1}{2\pi j}\int_{\delta - j\infty}^{\delta + j\infty} e^{st} Y_{S_1} ds = \Delta t_{O_S} + \varkappa_S t - \Delta t_{O_S} - \varkappa_S\left(t + 2\Delta t_D\right) = -2\varkappa_S \Delta t_D \tag{4.122}$$

$$\begin{aligned} y_{M_2} &= \frac{1}{2\pi j}\int_{\delta - j\infty}^{\delta + j\infty} e^{st} Y_{M_2} ds = \\ &= \Delta t_{O_M} + \varkappa_M t + \Delta t_{O_M} + \varkappa_M\left(t + 2\Delta t_D\right) - 2\Delta t_{O_S} - 2\varkappa_S\left(t + \Delta t_D\right) - 2\varkappa_M \Delta t_D + 2\varkappa_S \Delta t_D = \\ &= 2\left(\left(\varkappa_M - \varkappa_S\right) t + \Delta t_{O_M} - \Delta t_{O_S}\right) \end{aligned} \tag{4.123}$$

$$\begin{aligned} y_{S_2} &= \frac{1}{2\pi j}\int_{\delta - j\infty}^{\delta + j\infty} e^{st} Y_{S_2} ds = \\ &= \Delta t_{O_S} + \varkappa_S t + \Delta t_{O_S} + \varkappa_S\left(t + 2\Delta t_D\right) - 2\Delta t_{O_M} - 2\varkappa_M\left(t + \Delta t_D\right) + 2\varkappa_M \Delta t_D - 2\varkappa_S \Delta t_D = \\ &= 2\left(\left(\varkappa_S - \varkappa_M\right) t + \Delta t_{O_S} - \Delta t_{O_M}\right) \end{aligned} \tag{4.124}$$

□

These equations clearly show, that the scheme in Figure 4.10 is able to measure delay for both nodes. Even if the clocks drift, the measurement of offset and delay will always adapt to this drift. The value for Δt_D will usually not be the same at each node, and it is always the delay obtained by the node that measures it with respect to the other node. This is the first step to abandon the master/slave principle. In the given control, both nodes are equal and even use exactly the same equations.

4.4 Network Structure

The problem of network structuring is a problem of graph theory. While structuring a network for clock synchronization, the starting point is a undirected unweighted connected graph N, that is the given network, with a set of $\#V$ vertices $V(N) = \{v_i : i \in [\#V]\}$ and a set of $\#A$ arcs $A(N) = \{a_i : i \in [\#A]\}$ that has to be directed and structured. Some typical graphs in N will be defined [6]:

Definition 4.10. *A walk W with $V(W) = \{v_i : i \in [l]\}$ and $A(W) = \{a_i : i \in [l-1]\}$ is a sequence $v_1 a_1 \cdots a_{l-1} v_l$ that is called closed if $v_1 = v_l$ and open if $v_1 \neq v_l$. The walk is called trail if $a_j \neq a_i \ \forall\ i, j \in [l-1]$ and path if $v_i \neq v_j \ \forall\ i, j \in [l]$. A closed path is called cycle and a loop is a cycle with $l = 2$. A walk is called Hamiltonian if $V(W) = V(N)$ and a Hamiltonian[k] walk with $A(W) = A(N)$ is called Eulerian[l]. A tournament T with $V(T) = \{v_{ij} : i, j \in [m]\}$ and $A(T) = \left\{a_i : i \in \left[\binom{m}{2}\right]\right\}$ is a graph containing one arc $a_{ij} \vee a_{ji}$ between $v_i \wedge v_j \ \forall\ i, j \in \left[\binom{m}{2}\right]$. A tree is a graph containing no cycles and only exact one vertex with an outgoing or only exact one vertex with an incoming arc.*

For a better understanding, the named graphs are additionally depicted in Figure 4.11:

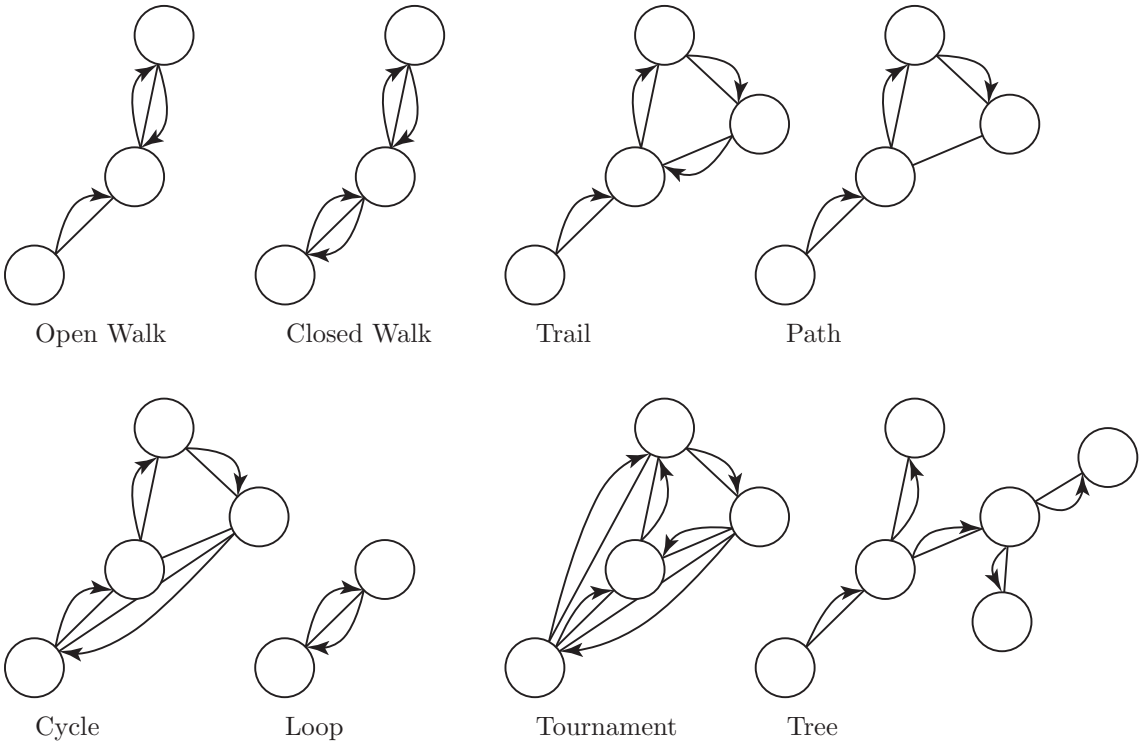


Figure 4.11: Types of Graphs

The given graphs are mathematical structures and none of them is made for network design by means of fault-tolerance or special algorithms working between nodes. Therefore none of them fits those purposes in every way. To conclude optimal properties among them, Table 4.2 compares them according to individual problems and advantages.

[k] Sir William Rowan Hamilton *1805†1865, Irish physicist, astronomer and mathematician
[l] Leonhard Euler *1707†1783, Swiss mathematician and physicist

Type	Disadvantage	Advantage
Open Walk	No crucial property to base algorithm on	Possible for every graph
Closed Walk	Loses crucial property for certain failures	Circeling algorithms possible
Trail	Not possible for every graph	Low dependence on single arcs, crucial property always remains
Path	Not possible for every graph	Low dependence on single vertices, crucial property always remains
Cycle	Loses crucial property for any given failure	Circeling algorithms possible
Loop	Strong dependence on single arcs	Makes nearly every graph possible
Tournament	Not possible for every graph	Highest fault-tolerance, crucial property always remains
Tree	Failure-intolerant, many vertices cut off in case of failure	Loopless, works in every network, low dependence on single arcs
Hamiltonian	Construction is not always possible and NP-complete	Every node can achieve synchronization
Eulerian	Construction is not always possible	Every node can achieve synchronization, low dependence on single arcs

Table 4.2: Disadvantages and Advantages of Graph Types for Network Structuring

The main problems are that some structures do not apply to every network, and that some graphs lose their structure in the case of failure of one arc or vertex. Algorithms are usually based on the graph type and will therefore not work in case of such a failure. Some of these cases are depicted in Figure 4.12.

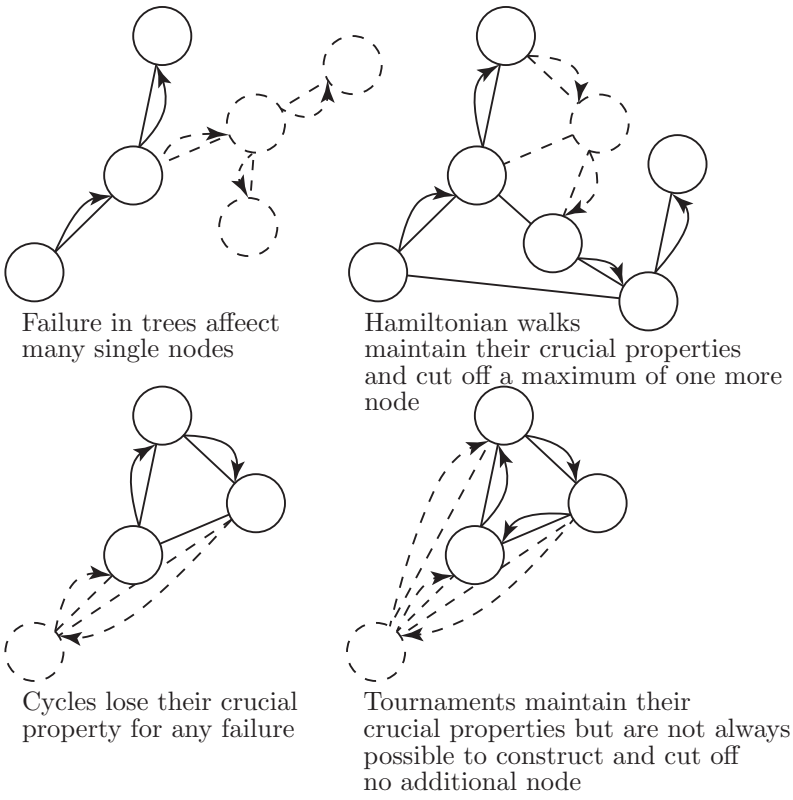


Figure 4.12: Effects of Failures in Graph Types

There is, though, a possibility to achieve fault-tolerance. If the network structure is chosen to be a Hamiltonian Walk while allowing loops, every graph may be structured that way [42].

Lemma 4.10. *There exists at least one Hamiltonian Walk for every connected graph if it is allowed to contain loops.*

The Hamiltonian walks do exclude one system maximum in case of failure and all other failures will produce further Hamiltonian walks. The best Hamiltonian walk, now, would be the one that is using as few arcs as possible, to be less dependent from single arcs and to achieve synchronization in shortest possible time. Finding the shortest walk through a graph is usually a NP-complete problem [38]. However, in the given network, the length of the arcs does not even matter, but only the number of arcs used. This is why it differs significantly from the Travelling Salesman Problem. Two find the shortest Hamiltonian walk, the nomenclature within the graph has to be defined. The definitions are given in Figure 4.13

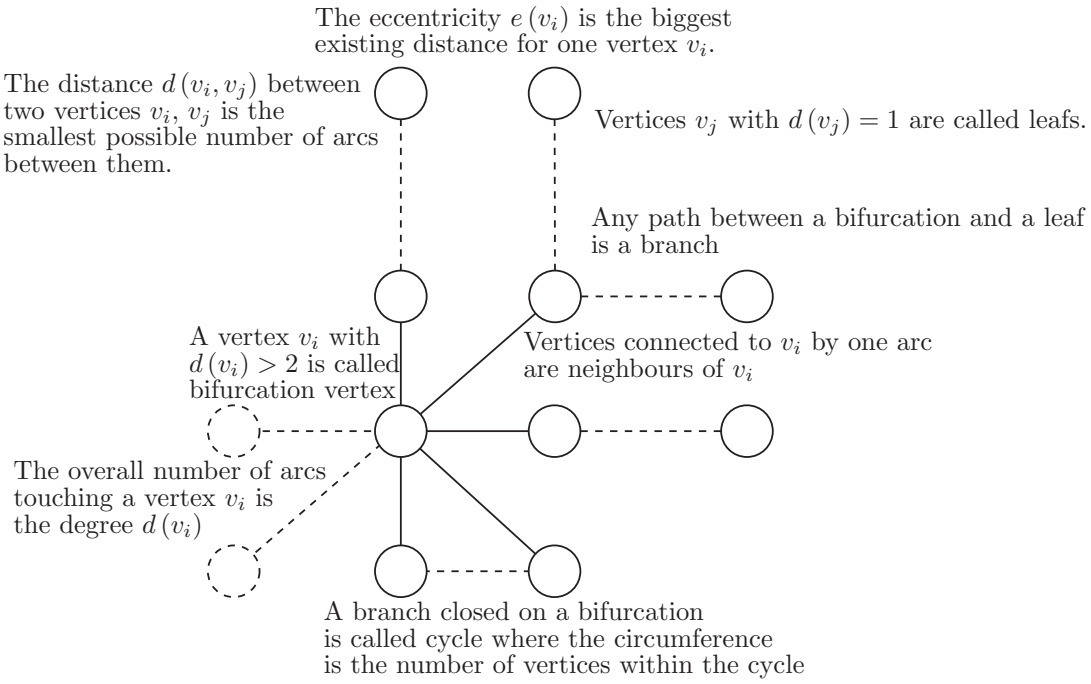


Figure 4.13: Nomenclature in Graphs

Now, the problem of finding the optimum Hamiltonian walk can be approached very logically. A graph without a bifurcation is a Hamiltonian. According to a bifurcation, every branch has to be walked twice, up to its leaf and back to the bifurcation to proceed to the next branch, unless it is the starting brach of the walk that is the source, the ending branch of the walk that is the sink, or a circle on the bifurcation. So these three cases have to be analyzed accurately in order to chose the most efficient walk. It seems that the main decision is about what the central bifurcation vertex is. Because the aim is to walk the longest branches only once, it seems reasonable to define the central bifurcation as the one with the highest sum of its eccentricity and the second largest distance to any leaf of the graph.

Definition 4.11. *The central bifurcation b_c of a graph is the vertex that fulfills*

$$\max_{i,j} \left(e\left(b_i\right) + d\left(b_i, l_j\right) \right) = e\left(b_c\right) + d\left(b_c, l_e\right) \;\; \forall \, i \in B, j \in L \tag{4.125}$$

where B is the set of bifurcations and L is the set of leafs.

Definition 4.12. *The leaf that defines the eccentricity of the bifurcation vertex is called l_s.*

$$e\left(b_c\right) = d\left(b_c, l_s\right) \tag{4.126}$$

Then, obviously, l_s is the designated starting vertex for the shortest walk and l_e the ending vertex. It is additionally important to define them as leafs of different branches, as Figure 4.14 explains:

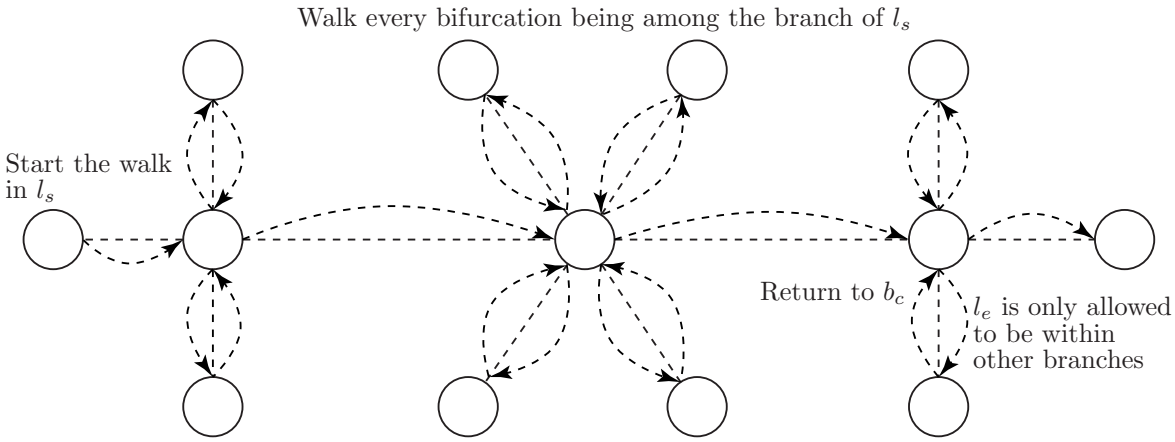


Figure 4.14: Handling of Multiple Eccentricities in the Same Branch

The simple reason is that every other choice would induce an unnecessary return to those branches, later. The order for walking the branches of the bifurcations among this branch can be chosen arbitrarily. There is yet one undiscussed problem. Cycles will be optimal for the given purposes because they only have to be walked once. But very long cycles may be chosen as the ending branch, therefore orienting them unidirectional. Those circles may therefore be cut off on one of the end arcs. Cycles containing two neighbors of a bifurcation may therefore be denoted as a branch of that bifurcation, where one of the two neighbors can arbitrarily be defined as its leaf. A case of a bifurcation with a longest cycle and a case of a bifurcation without a longest cycle are depicted in Figure 4.15.

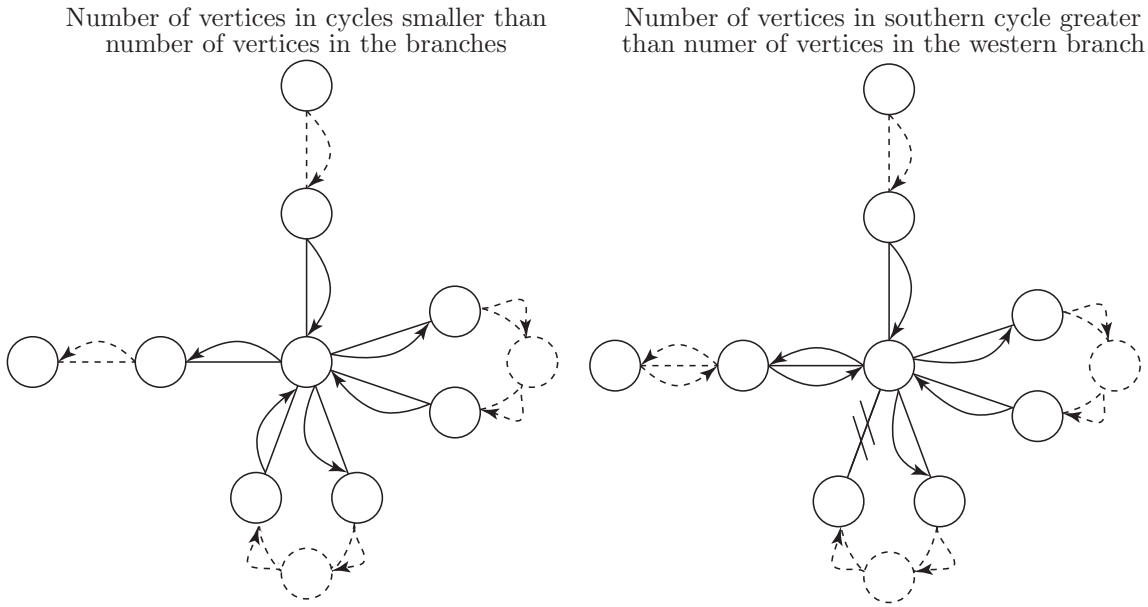


Figure 4.15: Cases of Cycles on Bifurcations

It seems that nearly all requirements for determining the shortest walk have been gathered. To automate the optimal solution of the problem, or at least to solve it from an engineering point of

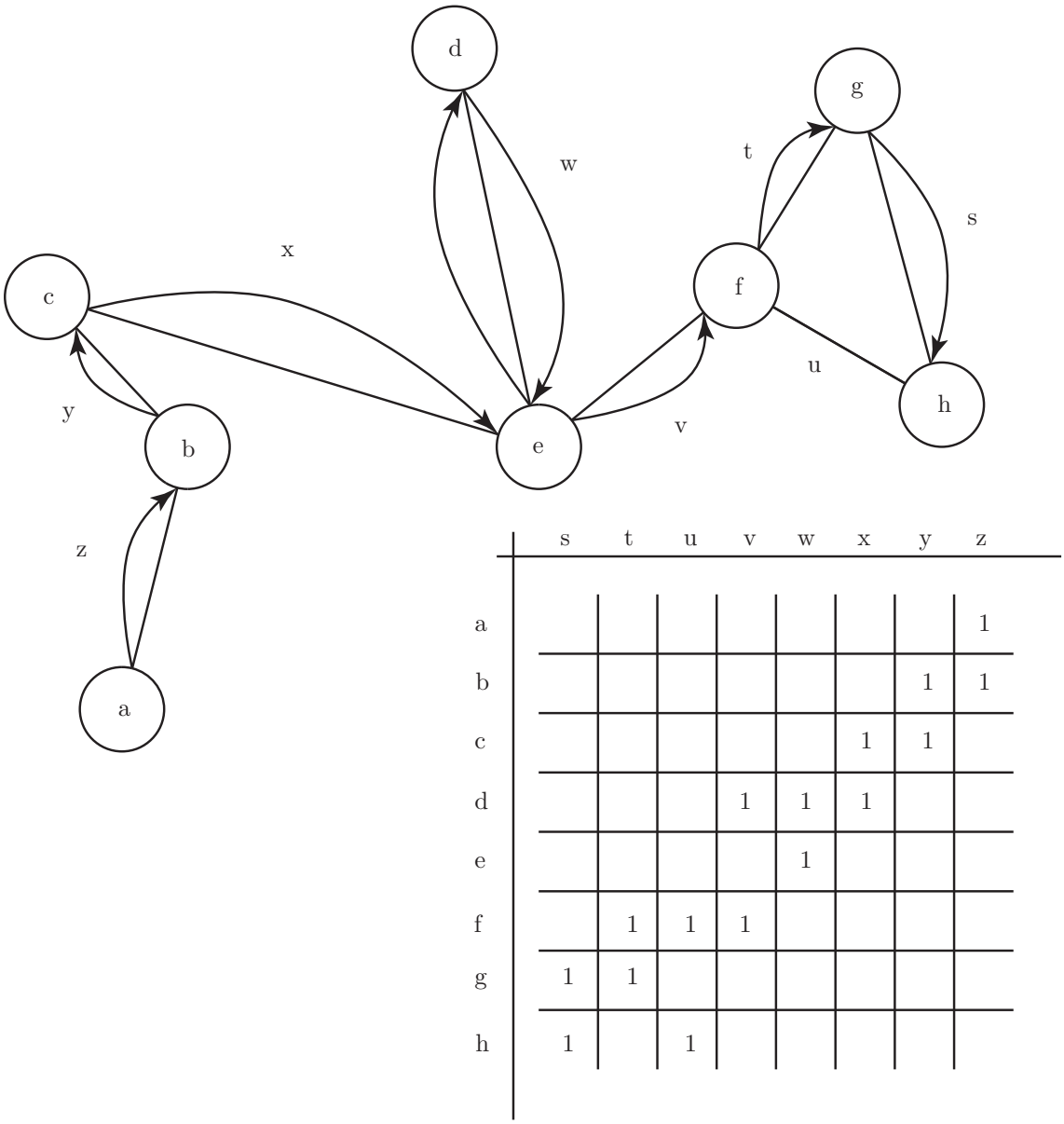
view, an invariant of the network graph is needed first, because the planar projection of a graph can rarely be used for algorithmic programming.

Definition 4.13. *An invariant of a graph is a property that remains unchanged when homomorphisms are applied.*

Definition 4.14. *A homomorphism is a graph type preserving mapping between two graphs.*

For the given purposes, the incidence matrix [12] is the best graph invariant to deal with. The incidence matrix is best to understand by using an example.

Example. The graph in Figure 4.16 is considered.



	s	t	u	v	w	x	y	z
a								1
b							1	1
c						1	1	
d				1	1	1		
e					1			
f		1	1	1				
g	1	1						
h	1		1					

Figure 4.16: Designing an Hamiltonian Walk via Incidence Matrix

The figure already contains the shortest Hamiltonian walk $azbycxewdwvftgsh$, that is trivial, in this case. As depicted, the rows of the matrix represent vertices whereas the columns represent arcs. The entries of the matrix represent the number of touches between one vertex and the

respective arc. For any given technical network, because no vertex will be connected to itself, the incidence matrix $\mathbf{\Psi} = \{\psi_{ij}\} \in \mathbb{R}^{\#V \times \#A}$ will always fulfill

$$\psi_{ij} = 0 \vee \psi_{ij} = 1 \; \forall \; i = 1, \cdots, \#V, j = 1, \cdots, \#A \tag{4.127}$$

That is why the incidence matrix is, in essence, a planar isotopy of the graph, when all elements within a row and all elements within a column are connected. This is very useful to the given purposes because distances, branches and cycles can be found algorithmic / automatic, as Figure 4.17 suggests.

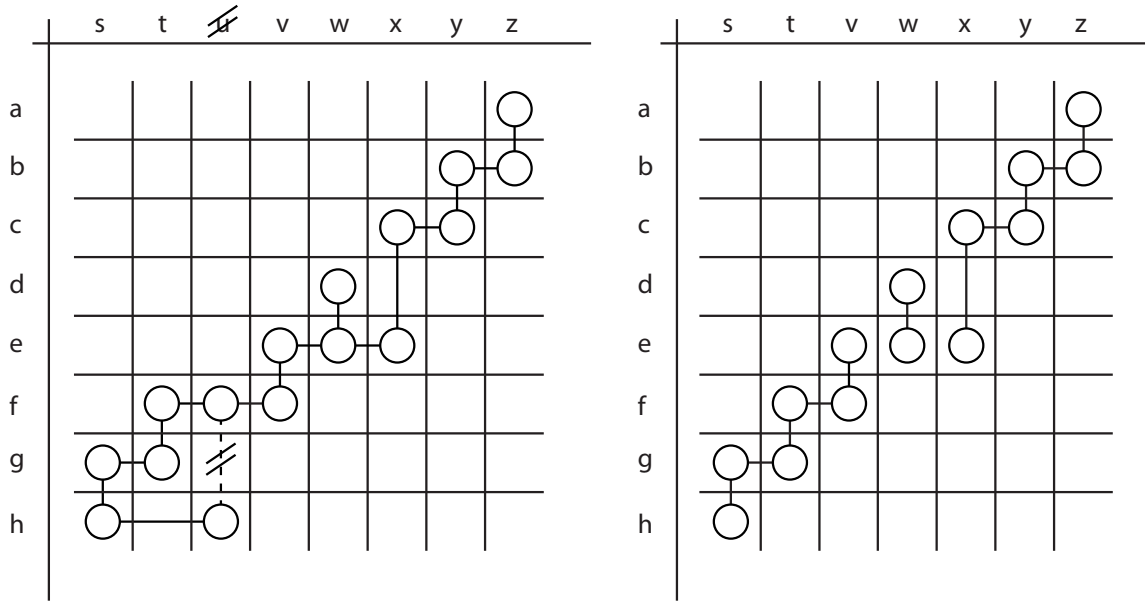


Figure 4.17: Detection of Loops and Length of Branches via Incidence Matrix

The example in the figure is the incidence matrix out of Figure 4.16 and is one example of a cycle that is long enough to be converted into a branch by cutting off one bifurcation arc. It is plain to see that if the branch of w on the bifurcation e has a length of more than two vertices, the cycle will be used instead of cut off. The figure also shows that for purposes of system design, the whole column may be deleted from the matrix, so it is easier to see the distance to the leafs.

So, obviously, the incidence matrix can easily detect cycles, that is a closed cycle in the incidence matrix, leafs, that is a row l with

$$\sum_{j=1}^{\#A} \psi_{lj} = 1 \tag{4.128}$$

and bifurcations, that is a row b with

$$\sum_{j=1}^{\#A} \psi_{bj} > 2 \tag{4.129}$$

It is now possible to write the walking procedure in an algorithm. The according pseudocode is to be seen in Algorithm 6.

Algorithm 6 Construction of Shortest Hamiltonian Walk Allowing Loops

Define $\mathbf{\Psi}(N)$
for all $i \in V$ **do**
  Compute $\sum_{j=1}^{\#A} \psi_{ij}$
  if $\sum_{j=1}^{\#A} \psi_{ij} = 1$ **then**
    $v_i = l_k$
    $k++$
  else
    if $\sum_{j=1}^{\#A} \psi_{ij} > 2$ **then**
      $v_i = b_m$
      $m++$
    end if
  end if
  $\#B = m$
  $B(N) = \{b_i : i \in [\#B]\}$
  $\#L = k$
  $L(N) = \{l_i : i \in [\#L]\}$
  $i++$
end for
Connect all $\psi_{ij} = 1$
Detect cycles
for all $b_i \in B$ **do**
  for all $l_j \in B$ **do**
    $d_n = d(b_i, l_j)$
    $n++$
    $j++$
  end for
  $i++$
end for
for all $p \in B$ **do**
  $p = p - 1$
  for $i = 1$, $i++$, $i = \#L$ **do**
    if $d_{i+p} > d_{i-1+p}$ **then**
      $e_1(b_{p+1}) = d_{i+p}$
      $l_i = l_s(b_{p+1})$
      $Z =$branchof(l_i)
    end if
  end for
  for $i = 1$, $i++$, $i = \#L$ **do**
    if $d_{i+p} > d_{i-1+p}, d_{i+p} \neq e_1(b_{p+1}), l_i \notin Z$ **then**
      $e_2(b_{p+1}) = d_{i+p}$
      $l_i = l_e(b_{p+1})$
    end if
  end for
  $s(b_{p+1}) = e_1(b_{p+1}) + e_2(b_{p+1})$
  $p++$
end for
$S(N) = \{s_i : i \in [\#B]\}$
$\max(S) = s_c$
$b_c = b_c$
$e_1(b_c) = d(b_c, l_s)$
$e_2(b_c) = d(b_c, l_e)$
w=l_s,branchof(l_s),branchof(b_c),branchof(l_e),l_e

When the original network is finally structured using this approach, the synchronization scheme working for this graph has to be found, since there is no grandmaster. The principle has to be the same at every edge, even in case of failure, so that the network becomes fault-tolerant. Let there be an oriented Hamiltonian walk that has edges numbered $e_i = e_1, \cdots, e_N$ and that has local times at each edge $t_i = t_1, \cdots, t_N$. Let everey edge contain the continuous clock synchronization algorithm in both directions, and let every edge i measure the delay to the edge $i-1$, receive the delay that $i-1$ has measured to $i-2$, sum them up and send them to $i+1$, that measures its delay to i, adds it to those of i to $i-1$ and $i-1$ to $i-2$ and so on. Every edge is doing exactly the same process and afterwards sets its local time to $t_i - \sum_{i=1}^{i} \Delta t_{O_i}$. The respective algorithhm looks like the sequence in Algorithm 7:

Algorithm 7 Fault-Tolerant Synchronization

```
1:  for all i do
2:    if i = i_min − 1 then
3:      receive ∑Δt_O at port i_min − 1
4:      measure Δt_O at port i_min − 1
5:      ∑Δt_O = ∑Δt_O + Δt_O
6:      set local time t = t − ∑Δt_O
7:      send ∑Δt_O to port i_min + 1
8:    else
9:      receive ∑Δt_O at port i − 1
10:     measure Δt_O at port i − 1
11:     ∑Δt_O = ∑Δt_O + Δt_O
12:     send ∑Δt_O to port i + 1
13:   end if
14: end for
```

Only edges that had their rows met more than one time during the design so that they have multiple i, means they touch at least two heads and two tails, set their clocks only accoring to the edge with the lowest i, this does not have to be done online, it can be programmed like that during system startup, so that each node has its i locally saved.

Example. A numerical example, as Table 4.3 is one, may be more descriptive. It refers to the network of Figure 4.16 and introduces random offsets for each vertex.

letter	number i	local time t_i	receive $\sum \Delta t_O$	measure Δt_O	send $\sum \Delta t_O$	set t_i
a	1	$+5$	0	0	0	$+5-0=+5$
b	2	-1	0	-6	-6	$-1+6=+5$
c	3	-3	-6	-2	-8	$-3+8=+5$
d	4	$+8$	-8	$+11$	$+3$	$+8-3=+5$
e	5	$+10$	$+3$	$+2$	$+5$	$+10-5=+5$
d	6	$+5$	$+5$	-5	$+0$	$+5-0=+5$
f	7	0	0	-5	-5	$0+5=+5$
g	8	-7	-5	-7	-12	$-7+12=+5$
h	9	$+13$	-12	$+20$	$+8$	$+13-8=+5$

Table 4.3: Synchronization in a Hamiltonian Walk

After setting the time in the last column, the local time at each vertex is identical. The computation of the synchronization needs one step at each vertex.

The amazing thing is that this approach is fault-tolerant. This is because every edge that is cut off has the same effect as if there was another edge with $i = 1$, that is a edge that receives a zero as the sum of previous delays. This is even valid for the edges meeting multiple head and toes, they become the $i = 1$ edge when its $i-1$ collapses or just send to an empty edge when any of its direct $i+x$ collapses. For system failures, the Hamiltonian walks may become smaller, but they are still working, and they are synchronized after only one stepsize.

5

Experiments

To illustrate the methods and principles deduced in 4, they are now applied to real systems and networks. This chapter will give some numerical experiments, measurements and applications of the given techniques. The particular aim is to enhance understanding and make application of Smith Synchronization easier.

5.1 Experimental Setup

The graphs and examples shown in this thesis result from an easy experimental setup. Two computers are connected via a direct cable or wireless connection. Each of them can send an receive on any port. One of them is defined to be the master system, the other one is the slave system. Each of them contains the simple dynamic model of a mass-spring-damper

$$\ddot{x} + \mathcal{D}\dot{x} + \mathcal{C}x = \mathcal{F}\dot{u}' + \mathcal{E}u' \tag{5.1}$$

where $\mathcal{D}$ is the damping, $\mathcal{C}$ the spring constant, $\mathcal{F}$ the derivative gain on the central input, and $\mathcal{E}$ is the direct gain. The space-state can be defined as

$$\underline{\dot{x}} = \begin{bmatrix} 0 & 1 \\ -\mathcal{C} & -\mathcal{D} \end{bmatrix} \underline{x} + \begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix} \underline{u}' \tag{5.2}$$

$$\underline{y} = \begin{bmatrix} 1 & 1 \end{bmatrix} \underline{x} \tag{5.3}$$

where $\underline{\dot{x}} = \begin{bmatrix} \dot{x} & \ddot{x} \end{bmatrix}^T = \frac{d\underline{x}}{dt}$ is the time-derivative of $\underline{x} = \begin{bmatrix} x & \dot{x} \end{bmatrix}^T$, the input $\underline{u} = \begin{bmatrix} u & \dot{u} \end{bmatrix}^T$ and $\underline{y} = \underline{x}$ is the output. The system of course may also be expressed as the transfer function

$$\frac{Y}{U'} = \frac{\mathcal{F}s + \mathcal{E}}{s^2 + \mathcal{D}s + \mathcal{C}} \tag{5.4}$$

where $Y = \int_0^\infty e^{-st} y dt$ and $U' = \int_0^\infty e^{-st} u' dt$. If the origin of u' is now situated at the slave for the master input and the other way round, it will be delayed, so that

$$\underline{\dot{x}} = \begin{bmatrix} 0 & 1 \\ -\mathcal{C} & -\mathcal{D} \end{bmatrix} \underline{x} + \begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix} \underline{u}\left(t - \Delta t_D\right) \tag{5.5}$$

and

$$\frac{Y}{U} = \frac{\mathcal{F}s + \mathcal{E}}{s^2 + \mathcal{D}s + \mathcal{C}} e^{-s\Delta t_D} \tag{5.6}$$

The aim is to synchronize the state vectors of the master $\underline{x}_M$ and the slave $\underline{x}_S$ at both computers for $x_M\left(t=0\right) \neq x_M\left(t=0\right)$ and to develop a network and controller design procedure that enables the synchronization of n systems in a distributed network with delay.

5.2 Smith Synchronization

The simple dynamic system shortly presented in 5.1 will now be embedded in the Smith synchronization shown in Figure 4.1. To depict that, Figure 5.1 presents a model abstraction of the system with mixed control and mechanic symbols, that are similar to the form of Figure 4.1.

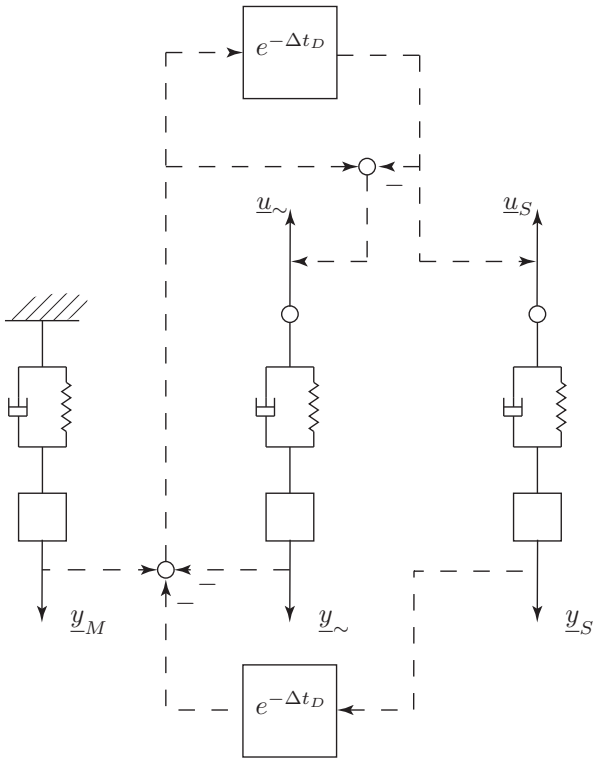


Figure 5.1: Simple Dynamic Systems in Smith Synchronization

Here the solid arrows represent mechanic dimensions and the dotted arrows control signals. Referring to this graphic, the aim of the control would be to synchronize the outer oscillators. To guarantee a synchonized state for the systems, only the coupling matrix $\boldsymbol{L} = \begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix}$ has to be considered.

$$-\begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix} - \begin{bmatrix} 0 & \mathcal{E} \\ 0 & \mathcal{F} \end{bmatrix} = -\begin{bmatrix} 0 & \mathcal{E} \\ \mathcal{E} & \mathcal{F} \end{bmatrix} \tag{5.7}$$

With the set of eigenvalues

$$\sigma\left(-\boldsymbol{L}^* - \boldsymbol{L}\right) = \left\{-\mathcal{F} + \sqrt{\mathcal{E}^2 + \mathcal{F}^2}, -\mathcal{F} - \sqrt{\mathcal{E}^2 + \mathcal{F}^2}\right\} \tag{5.8}$$

So that the respective sums are

$$\lambda_1 + \lambda_2 = -2\mathcal{F} \tag{5.9}$$

$$\left(\lambda_1 + \lambda_2\right)^2 = 4\mathcal{F}^2 \tag{5.10}$$

$$\lambda_1^2 + \lambda_2^2 = 4\mathcal{F}^2 + 2\mathcal{E}^2 \tag{5.11}$$

So that the needed matrix measure is

$$\mu_2\left(\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) = \frac{1}{2}\left(\frac{-\mathcal{F}}{3} + \sqrt{2.\bar{7}\mathcal{F}^2 + 1.\bar{6}\mathcal{E}^2}\right) \tag{5.12}$$

Now, the second term is a function of

$$2\begin{bmatrix} 0 & \mathcal{E} \\ 0 & \mathcal{F} \end{bmatrix}\begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix} = 2\begin{bmatrix} \mathcal{E}^2 & \mathcal{E}\mathcal{F} \\ \mathcal{E}\mathcal{F} & \mathcal{F}^2 \end{bmatrix} \tag{5.13}$$

and the corresponding set of eigenvalues

$$\sigma\left(2\boldsymbol{L}^{*}\boldsymbol{L}\right)=\left\{0,2\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)\right\} \tag{5.14}$$

and again

$$\lambda_{1}+\lambda_{2}=2\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right) \tag{5.15}$$

$$\left(\lambda_{1}+\lambda_{2}=\right)^{2}=4\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)^{2} \tag{5.16}$$

$$\lambda_{1}^{2}+\lambda_{2}^{2}=4\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)^{2} \tag{5.17}$$

By that, the appropriate norm for this term is

$$\left\|\boldsymbol{\Gamma}_{t-\Delta t_{D}}\otimes\boldsymbol{B}\,\boldsymbol{C}K\right\|_{2}=\sqrt{\mathcal{E}^{2}+\mathcal{F}^{2}+\sqrt{5\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)^{2}}}=1.8\sqrt{\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)} \tag{5.18}$$

The aim in tuning the coupling terms will therefore be

$$\frac{1}{2}\left(\frac{-\mathcal{F}}{3}+\sqrt{2.\bar{7}\mathcal{F}^{2}+1.\bar{6}\mathcal{E}^{2}}\right)+\sqrt{\mathcal{E}^{2}+\mathcal{F}^{2}+\sqrt{5\left(\mathcal{E}^{2}+\mathcal{F}^{2}\right)^{2}}}=\Re\left(\varsigma\right)\approx0 \tag{5.19}$$

Where $\Re\left(\varsigma\right)$ is the optimization cost function. Fulfilling this condition, it is possible to visualize the optimization effect. In this case, only the trivial solutions $\mathcal{F}=\mathcal{E}=0\vee K=0$ enable $\varsigma=0$. Those solutions do not enforce a coupling at all and do not control the slave system. Therefore the synchronization state is to be understood as the equilibrium of the master system, so that both of them are finally synchronized in a sense of no movement. Obviously, the aim is to minimize the cost function $\Re\left(\varsigma\right)$, while controlling the system, though. Figure 5.2 depicts the cost function $Re\left(\varsigma\right)$ versus $\mathcal{F}$ and $\mathcal{E}$.

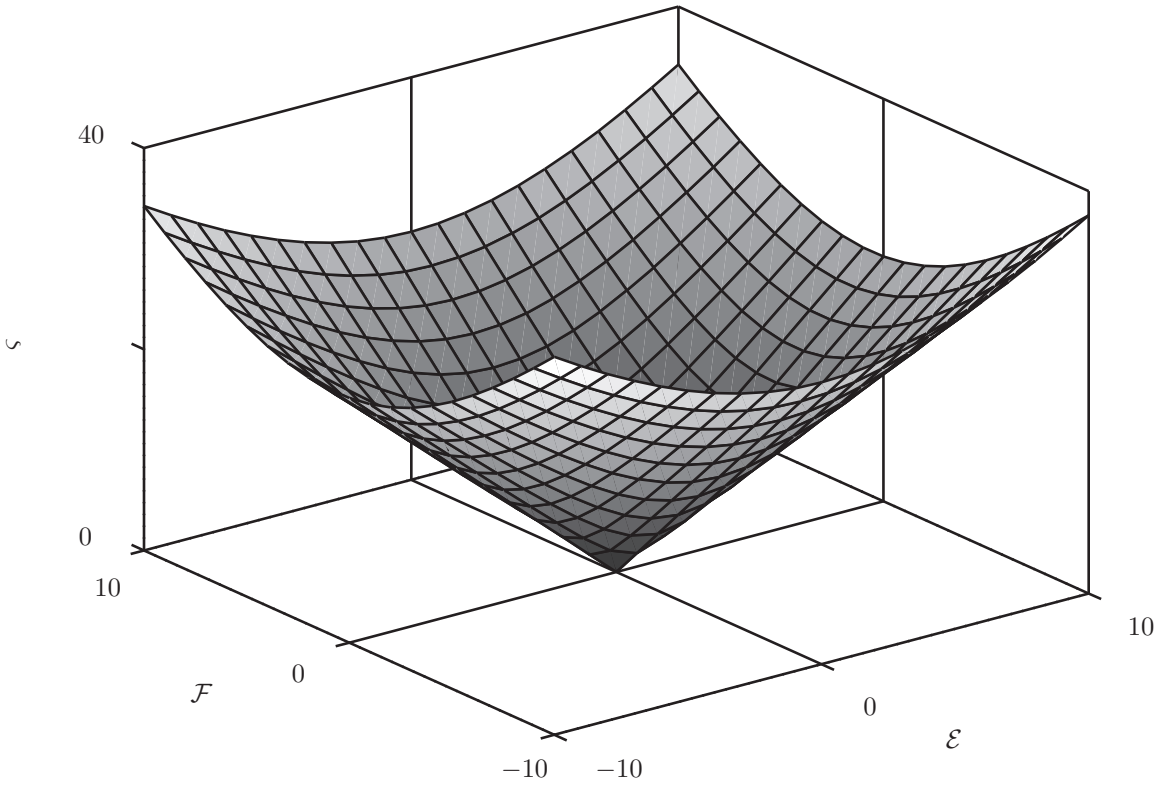


Figure 5.2: Optimization Function for Largest Eigenvalue

Now, various values for sufficient small $\Re\left(\varsigma\right)$, $\mathcal{F}$ will be considered. Then, the synchronization error $e=\|\underline{x}_{M}-\underline{x}_{S}\|$ may be measured versus time, in different values of $\mathcal{E}$ in the neighborhood

of the global equilibrium of the synchronizability condition. These examples will use a fixed $\mathcal{F}$ to show the effects of the cost function, that is in turn a representation for the largest eigenvalue. The following tests where done with the numerical values out of Table 5.1

Symbolic Variable	Numerical Value
K	10
$\mathcal{C}$	1
$\mathcal{D}$	0.1
$\mathcal{E}$	alternating
$\mathcal{F}$	0.1
Δt_D	0.2
$\underline{x}_{M_0}$	$\begin{bmatrix}10 & 0\end{bmatrix}^T$
$\underline{x}_{S_0}$	$\begin{bmatrix}0 & 0\end{bmatrix}^T$

Table 5.1: Numerical Values for Measurements of the Neighborhood of $\mathcal{E}_{opt}$

With the given values, measurements of x_M, x_S, $\|\underline{x}_M - \underline{x}_S\|$ and $\int \|\underline{x}_M - \underline{x}_S\|$ are depicted versus varying $\mathcal{E}$ in the neigborhood of $\mathcal{E} = 0$ in Figure 5.3

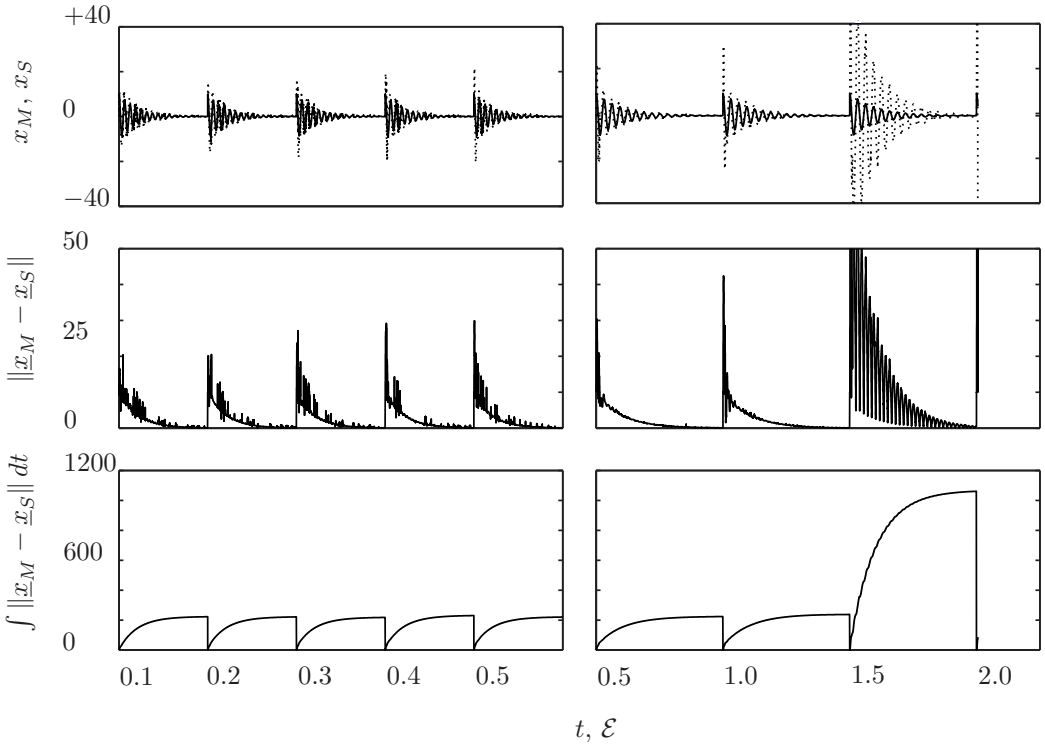


Figure 5.3: Synchronization Process with Different Approximations of the Largest Eigenvalue

Where the dotted lines represent the behaviour of x_S and the non-dotted the one of x_M. Most of the depicted functions seem to achieve complete synchronization, though the estimation of the largest eigenvalue is greater than zero. The reason for this phenomenon is that the estimated eigenvalue is an upper bound, and so the actual eigenvalue may be smaller than the estimated one. Nearly all of the given values of $\mathcal{E} \approx 0$ enforce a zero eigenvalue. Measurements like these may be used for parameter optimization, but obviously, the analytical optimization minimizing the largest eigenvalue results in the same behaviour as the H^2 experimental optimization, that is the $\int e$ experiment in Figure 5.3. Some of the functions additionally seem to oscillate whereas some seem to be damped, and others are nearly instable. Further measurements in bigger scales and additional output values are needed for exact conclusions. The meaning of the cost function

depicted in Figure 5.2 is now even more explicit. While low values of $\mathcal{E}$ induce a nearly complete synchronization, overshooting increases with increasing $\mathcal{E}$ and is even instable for $\mathcal{E} = 2$. The synchronization error shows that a H_2 optimization gives the same conclusion as the cost function does. Yet, only synchronizability has been considered and means of stability have been neglected, although it is well known that instabilities occur for $\mathcal{E} < 0$. Therefore, the Lyapunov exponents of the system have to be computed, again as a function of $\mathcal{E}$ to optimze the control further. Since the system is linear and its first variation is also linear, the set of L equals

$$L = \sigma\left(\begin{bmatrix} 0 & 1 \\ -\mathcal{C} & -\mathcal{D} \end{bmatrix} + \Omega \begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix}\right) \tag{5.20}$$

And because only $\max(L)$ is relevant to means of stability,

$$\max(L) = \frac{\Omega\mathcal{F} - \mathcal{D}}{2} + \sqrt{\Omega^2\left(\frac{\mathcal{F}^2}{4}\right) + \Omega\left(\mathcal{E} - \frac{\mathcal{F}\mathcal{D}}{2}\right) + \frac{\mathcal{D}^2}{4} - \mathcal{C}} \tag{5.21}$$

The values for $\max(L) < 0$ as function of Ω, $\mathcal{E}$ are to be analyzed. The corresponding plot is shown in Figure 5.4:

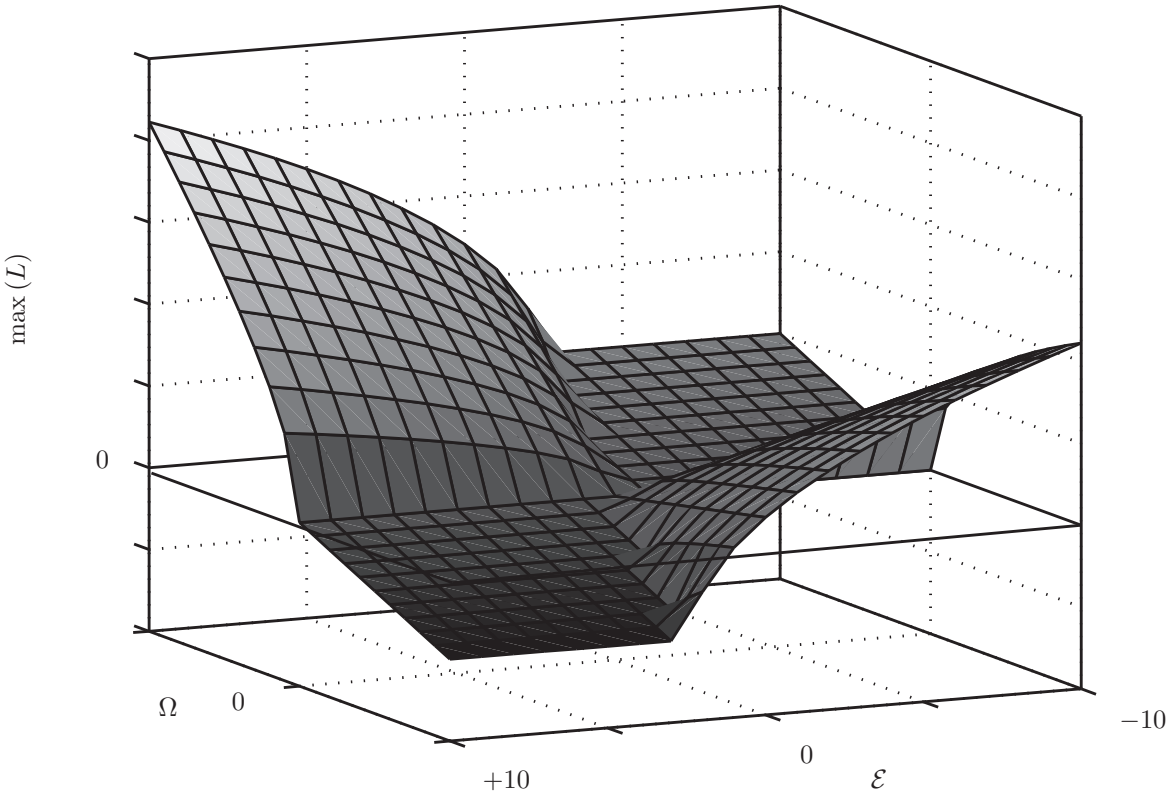


Figure 5.4: Optimization Function of the Lyapunov Exponent

The figure shows that the synchronized region is unbounded to the negative s-plane. This fact simplifies the optimization, since a bounded region for negative $\max(L)$ would have enforced an analyzation of the smallest eigenvalue. Nevertheless, the smallest eigenvalue could also be computed out of the interval optimization in 4.1. Now the interpretation of this equation is that the eigenvalues of the coupling matrix have to be non-positive and $\mathcal{E}$ has to be positive. In essence, this information only means that the feedback provided in the loop of the Smith Predictor has to be negative. Or, in other words, that K has to be positive, since K could as well be used for tuning of coupling parameters. In general, it is important that the whole coupling tuning demonstrated in this section can be applied equally to the tuning of controllers, imagining that K

is a vector that amplifies $\mathcal{E}$ and $\mathcal{F}$ seperately. Summarizing the optimization results, $K\mathcal{E}$ and $K\mathcal{F}$ have to be positive to ensure stability and small enough to ensure synchronization, that is also a matter of stability since the master system is stable. Though the given criteria are sufficient to replace experimental optimization, the largest eigenvalue is still an approximation and the exact synchronized and deeply stable region has to be found experimentally. Now, using the second theorem of Mori, a precise bound may be found without the need of experiments. Since $\Re(\varsigma)$ is known, $\Im(\varsigma)$ is needed. The eigenvalues of

$$\begin{bmatrix} 0 & 0 \\ \mathcal{E} & \mathcal{F} \end{bmatrix} - \begin{bmatrix} 0 & \mathcal{E} \\ 0 & \mathcal{F} \end{bmatrix} = \begin{bmatrix} 0 & -C \\ C & 0 \end{bmatrix} \tag{5.22}$$

are $\sigma\left(\boldsymbol{L} - \boldsymbol{L}^T\right) = \{+j\mathcal{E}, -j\mathcal{E}\}$. By that, the matrix measure is $\mu_2\left(-j\boldsymbol{\Gamma}_t \otimes \boldsymbol{B}\,\boldsymbol{C}K\right) = \frac{1}{2}\left(\frac{-\mathcal{F}}{3} + \sqrt{2.\overline{7}\mathcal{F}^2 + 1.\overline{6}\mathcal{E}^2}\right) = \sqrt{\frac{5}{12}}\mathcal{E}$. Hence, it is trivial to denote $\Im(\varsigma) = \sqrt{\frac{5}{12}}\mathcal{E} + 1.8\sqrt{(\mathcal{E}^2 + \mathcal{F}^2)}$. Now that the interval $[0 + 0j, \varsigma]$ is completely defined, the substituted matrix eigenvalues are needed. The set of eigenvalues of $\boldsymbol{L}$ is $\sigma(\boldsymbol{L}) = \{0, \mathcal{F}\}$ so that the final synchronizability criterion (4.76) holds, if

$$\Re\left(\mathcal{F}\left(-\frac{1}{4} + \sqrt{-\frac{3}{8} + \frac{3}{2}\left(e^{-2\Delta t_D[0+0j,\varsigma]} - e^{-\Delta t_D[0+0j,\varsigma]}\right)}\right)\right) = \zeta < 0 \tag{5.23}$$

This condition can be investigated by plotting the parameter plane $[0 + 0j, \varsigma]$ versus ζ and comparing the examples to the synchronization bounds of Figure 5.3. Therefore, ζ is computed for all values within $[0 + 0j, \varsigma]$ in a matrix versus varying Δt_D and $\mathcal{E}$ in Figure 5.5.

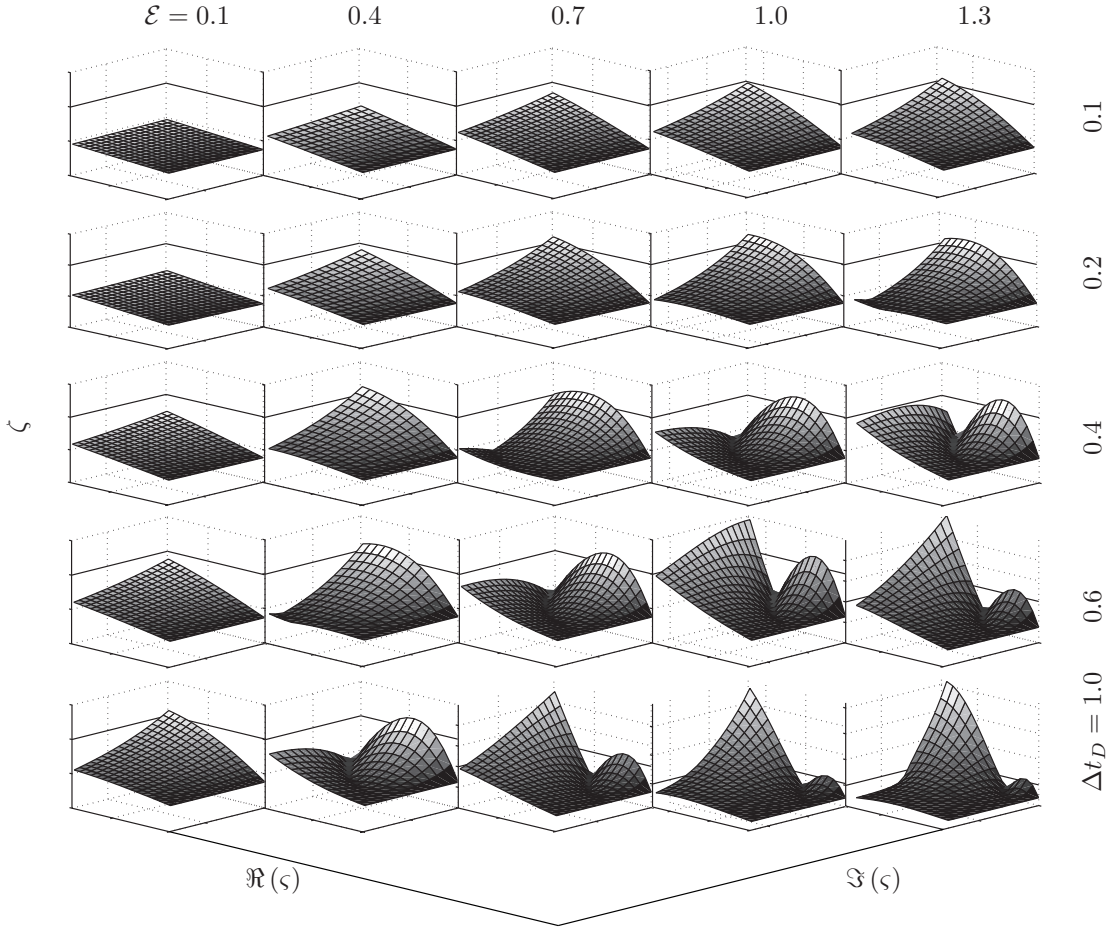


Figure 5.5: Eigenvalues within the Instable Plane in a Matrix of Delay and Coupling Parameter

In Figure 5.5, the zero plane in ζ is marked with a black line on each projection. Obviously there are two maxima arising as Δt_D and $\mathcal{E}$ increase. The first one is local, and looks rather soft compared to the second one. This does not mean unsynchronized or instable behaviour, compared to the results of Figure 5.3. However, there is no doubt about instabilities occuring for growing Δt_D and $\mathcal{E}$. Another notable fact is that the first, soft maximum, decreases again as $\mathcal{E}$ grows on, whereas the harder maximum, that is hereafter called global for all Δt_D and $\mathcal{E}$, whereas the single maxima are called global for all ς, keeps growing in each direction of increasing Δt_D and $\mathcal{E}$. It may be possible to consider only the global maxima for all ς for all Δt_D and $\mathcal{E}$ and span a parameter plane to find global maxima of ζ in all Δt_D and $\mathcal{E}$. The result is an analytical function of boundary synchronizability / stability $\Delta t_D(\mathcal{E})$ or $\mathcal{E}(\Delta t_D)$, respectively, that is depicted in Figure 5.6.

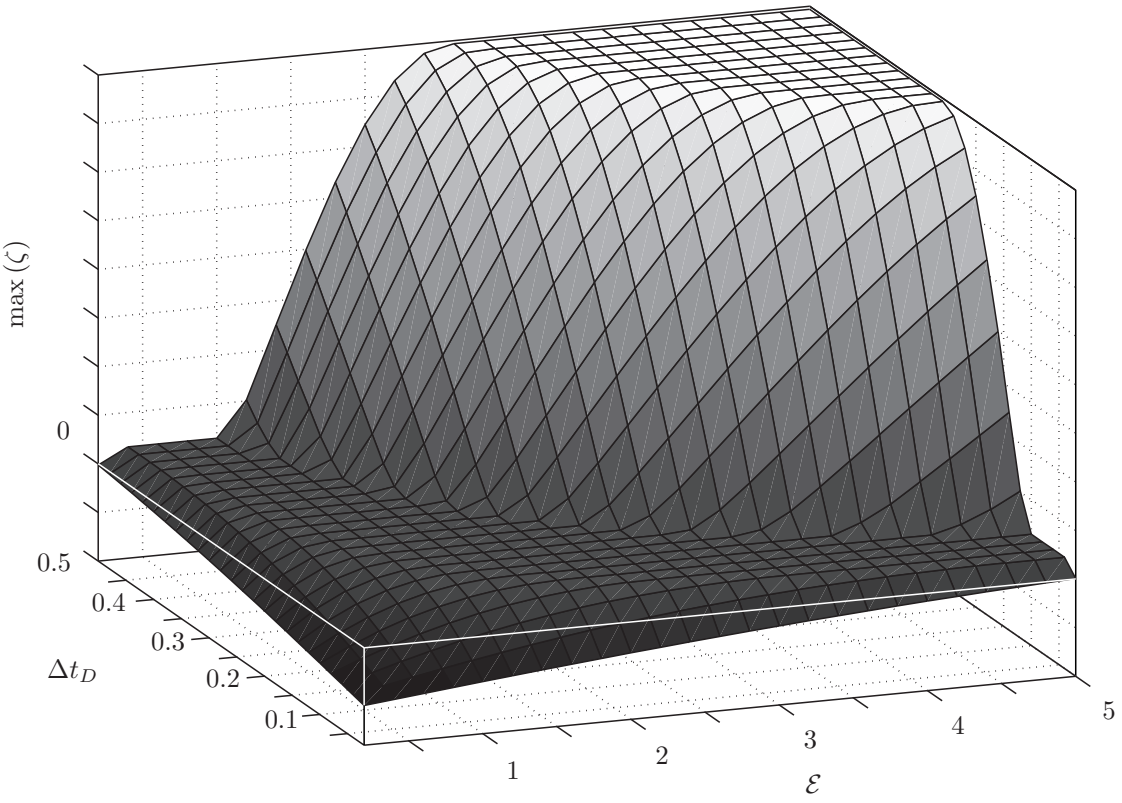


Figure 5.6: Global Maxima in ζ for all ς versus $\mathcal{E}$ and Δt_D

The soft maxima are the ones arising directly at zero whereas the hard maxima are those aventually leading to the global maximum at the top. Though the soft maxima are already defined to be unstable by Mori, the subjective analysis of the measurements in Figure 5.3 leads to the conclusion that real unstable, unsynchronized states occur for the hard maximum. For example, there are hard maxima for $\Delta t_D = 0.2$ in the region of $\mathcal{E} \approx 2$, and the experiment in Figure 5.3 shows unstable and unsynchronized behaviour for theser values. To prove this conclusion, that yet only appears in one point of the plot, further measurements with different delays have to be made. The results of these measurements are shown in Figure 5.7.

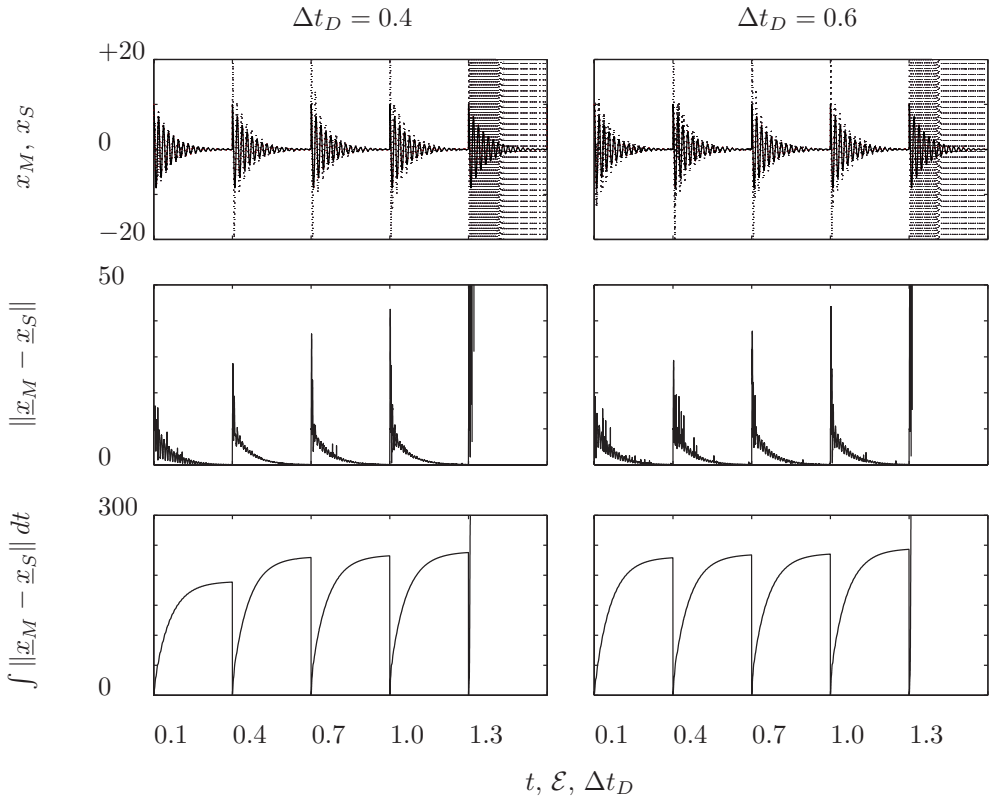


Figure 5.7: Synchronization Process with Different Delays

It seems that both cases of Δt_D tend to be unsynchronized for the case $\mathcal{E} = 1.3$. This statement agrees with the analytical results of Figure 5.6, because both cases are clearly within the region of hard maxima. In the given case, clearly, the curve of hard maxima arising in the Δt_D, $\mathcal{E}$ plane may be used as design boundary. This curve, in addition, is simple enough to be used for online parameter tuning for varying Δt_D, in case that bigger $\mathcal{E}$ grant higher efficiency for the system in any special way.

5.3 Clock Synchronization

A simple Hamiltonian walk of four components is considered. Let all components have distinct inner clocks according to offset and drift. The components will be connected one after the other and in the end, one connection will be cut off. The measurement uses numerical values for the inner clocks, that are given in Tabular 5.2:

i	$\varkappa_i$	Δt_{O_i}	connection time	disconnection time
1	1.1	7	0	none
2	1.2	13	6.104	none
3	1.3	5	10.27	20.35
4	1.4	1	14.59	none

Table 5.2: Numerical Values for Clock Synchronization Measurement

The best way to depict the results of this simple experiment is evaluating inner clocks versus a neutral clock. The respective curve is shown in Figure 5.8:

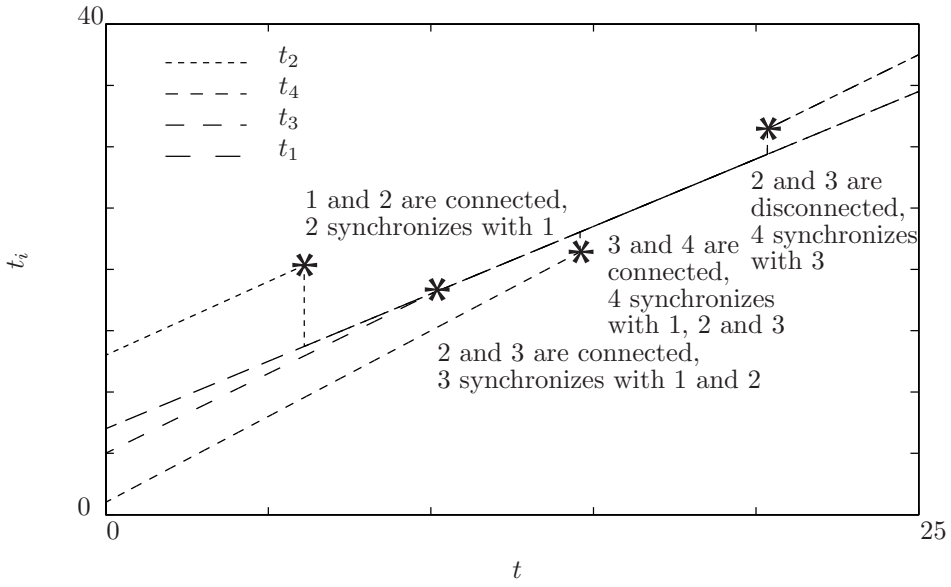


Figure 5.8: Inner Clocks Versus a Neutral Clock in Measurement of Clock Synchronization

One can see that all inner clocks always adopt to the clock of node 1 and that it takes only one step each to achieve synchronization. As soon as one of the connections is cut off, smaller Eulerian walks are formed. Each of them as its own 1 node and its own drift and offset, but all elements within one walk are completely synchronized.

6

Conclusions

The thesis has reproduced the fundamentals of predictive control and the theory of synchronizability. A new interpretation of the delay measurement protocol has been introduced, its principle has been proven using a continuous representation of its algorithms, and its network topology has been altered to grant functionality in the case of certain failures. These Hamiltonian walks can be computed in a new algorithm. The structure composed of network topology and the delay measurement protocol can be used to provide numerical values for delay length at any time to compensate for the fragility of the Smith Predictor to inaccuracies in delay modelling. The Smith Predictor was then altered in its reference by using another model that was designated to be monitored in realtime instead of the delayed plant. The solutions to the transcendental dynamic equations of the resulting control have been analyzed by using the Lambert W function. The results of the modal analysis declared the Smith Synchronization inapplicable for the monitoring of systems which dynamics are fast compared to the length of the delay. However, stability criteria for the coupling terms have been derivated and simplified using the stability theorems by Mori, the theorem on the latent roots of block matrices by Williamson, the theorem on constraint optimization by Lagrange, and the theorem on symmetric polynomials by Newton, also known as the Newton identities. The failing of the Smith Synchronization was then used as an opportunity to optimize coupling coefficients to enforce synchronizability for the coupling matrices. Therefore, Bigg's Algorithm was used, that is a Quasi-Newton method with an optimization for the first local derivative using quadratic programming. The numerical results were interpreted to conjecture a generic lemma on the form of the coupling matrices that was then again proved using the Lambert W function. The results on the coupling coefficients were then used to shape a block diagram for a control that was named the Zero Row Sum Negative Trace Control, due to the lemma on its coupling matrices. The control was nevertheless only suited for autonomous systems. The failings of the Smith Synchronization and the Zero Row Sum Negative Trace Control lead to the formulation of a conjecture.

Conjecture. *A matrix* $\in \mathbb{R}^3$ *that holds synchronizability criteria for two rows but uses one auxiliary row can be found. These constraints can be applied to the optimization of the Zero Row Sum Negative Trace Control, revealing a structure that grants synchronizability for two of its subsystems and a possibility to acces it with an input. The result is a control that synchronizes two non-autonomous systems.*

The theory on Smith Synchronization was proven in experiments using spatially distributed computer systems with real delays and simple dynamic systems on every local computer. The experimental results verified the theory that was deduced in chapter 4. But there are also undiscussed topics, that could be investigated in further work. On the one hand, only proportional controllers have been analyzed, while integral and derivative controllers have been neglected. On the other hand, the entire question of robustness, that is the question of models that do not fit the plant perfectly, has been left out in the discussion on model-plant synchronizability.

Bibliography

[1] IEEE Standard 1588-2002. *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Maeasurement and Control Systems*. The Institute of Electrical and Electronics Engineers, Inc., New York, 2002.

[2] H. Agiza and M. Yassen. Synchronization of rossler and chen chaotic dynamical systems using active control. *Physics Letters A*, 278:191–197, 2001.

[3] A. Arenas, A. Diaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou. Synchronization in complex networks. *Physics Reports*, 469:93–153, 2008.

[4] K. Arvind. Probabilistic clock synchronization in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, 5:474–487, 1994.

[5] Farshid Maghami Asl and A. Galip Ulsoy. Analysis of a system of linear differential equations. *Journal of Dynamic Systems, Measurement, and Control*, 125:215–223, 2003.

[6] Jorgen Bang-Jensen and Gregory Gutin. *Digraphs*. Springer, 2009.

[7] Mauricio Barahona and Louis M. Pecora. Synchronization in small-world systems. *Physical Review Letters*, 89:1–4, 2002.

[8] Aziz Belmiloudi. *Stabilization, Optimal and Robust Control*. Springer, 2008.

[9] M. Bennett, M. Schatz, H. Rockwood, and K. Wiesenfeld. Huygen's clocks. *Proceedings of the Royal Society*, 458:563–579, 2002.

[10] M.C Biggs. *Towards Global Optimization*. North-Holland, 1975.

[11] B. Blasius, A. Huppert, and L. Stone. Complex dynamics and phase synchronization in spatially extended ecological systems. *Nature*, 399:354–359, 1999.

[12] J. A. Bondy and U. S. R. Murty. *Graph Theory*. Springer, 2008.

[13] T. Carroll, L. Pecora, and F. Rachford. Chaos and chaotic transients in an yttrium iron garnet sphere. *Physical Review Letters A*, 40:377–386, 1989.

[14] T. Carroll, L. Pecora, and F. Rachford. Effect of surface roughening on chaos in yttrium-iron-garnet spheres. *Physical Review Letters B*, 40:2327–2331, 1989.

[15] T. Carroll, L. Pecora, and F. Rachford. Lyapunov exponents near a crisis in a spin-wave experiment. *Physical Review Letters A*, 40:4149–4152, 1989.

[16] J. Cheong, S. Niculescu, A. Annaswamy, and M. Srinivasan. Motion synchronization in virtual environments with shared haptics and large time delays. In *Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems 2005*, 2005.

[17] D. Clarke, C. Mohtadi, and P. Tuffs. Generalized predictive control 1 - the basic algorithm. *Automatica*, 23:137–148, 1987.

[18] D. Clarke, C. Mohtadi, and P. Tuffs. Generalized predictive control 2 - extensions and interpretations. *Automatica*, 23:149–160, 1987.

[19] J. J. Collins and I. Stewart. A group-theoretic approach to rings of coupled biological oscilators. *Biological Cybernetics*, 71:95–103, 1994.

[20] A. Copeland. Predictions and probabilities. *Erkenntnis*, 6:189–203, 1936.

[21] Les Cottrell, Warren Matthews, and Connie Logg. Internet monitoring & pinger at slac. 2007.

[22] Zhisheng Duan, Guanrong Chen, and Lin Huang. Synchronization of weighted networks and complex synchronized regions. *Phsics Letters A*, 372:3741–3751, 2008.

[23] J. Eidson, M. C. Fischer, and J. White. Ieee 1588 standard for a precision clock synchronization protocol for network measurement and control systems. In *34th Annucal Precise Time and Time Interval Meeting*, 2002.

[24] John C. Eidson. *Measurement, control, and communication using IEEE 1588*. Springer, 2006.

[25] T. Ersal, M. Brudnak, A. Salvi, J. Stein, Z. Filipi, and H. Fathy. Development of an internet-distributed hardware-in-the-loop simulation platform for an automotive application. In *ASME 2009 Dynamic Systems and Control Conference*, 2009.

[26] T. Ersal, M. Brudnak, A. Salvi, J. Stein, Z. Filipi, and H. Fathy. Development and model-based transparency analysis of an internet-distributed hardware-in-the-loop simulation platform. *Mechatronics*, —:—, 2010.

[27] T. Ersal, M. Brudnak, J. Stein, and H. Fathy. Statistical transparency analysis in internet-distributed hardware-in-the-loop simulation. *IEEE Transactions on Mechatronics*, —:—, 2010.

[28] B. Francis. Synthesis of multivariable regulators. *Applied Mathematics & Optimization*, 1:64–86, 1974.

[29] B. Francis. The internal model principle for linear multivariable regulators. *Applied Mathematics & Optimization*, 2:170–194, 1975.

[30] B. Francis. The internal model principle of control theory. *Automatica*, 12:457–465, 1976.

[31] Gene Franklin, David Powell, and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems*. Pearson, 2006.

[32] H. Gray Funkhouser. A short account of the history of symmetric functions of roots of equations. *The American Mathematical Monthly*, 37:357–365, 1930.

[33] C. Ganesh and J. B. Pearson. h^2-optimization with stable controllers. *Automatica*, 25:629–634, 1989.

[34] S. Ganjefar, H. Momeni, and F. Janabi-Sharifi. Teleoperation systems design using augmented wave-variables and smith predictor method for reducing time-delay effects. In *Proceedings of the International Symposium on Intelligent Control*, 2002.

[35] Carlos Garcia and Manfred Morari. Internal model control. a unifying review and some new results. *Industrial & Engineering Chemistry Process Design and Development*, 21:308–323, 1982.

[36] P.E. Gill, W. Murray, M.A. Saunders, and M.H. Wright. Procedures for optimization problems with a mixture of bounds and general linear constraints. *ASM Transactions on Mathematical Software*, 10:282–298, 1984.

[37] A. Gurtovnik and I. Neimark. On synchronization of dynamic systems. *Journal of Applied Methematics and Mechanics*, 38:749–758, 1974.

[38] Gregory Gutin and Abbraham Punnen. *The Traveling Salesman Problem and its Variations*. Springer, 2004.

[39] Jack K. Hale. Diffusive coupling, dissipation, and synchronization. *Journal of Dynamics and Differential Equations*, 9:1–52, 1997.

[40] J. Halpern, B. Simons, R. Strong, and D. Doley. Fault-tolerant clock synchronization. In *Proceedings of the third Annual ACM Symposium on Principles of Distributed Computing*, 1984.

[41] G. H. Hardy. The mean value of the modulus of an analytic function. *Proceedings of the London Mathematical Society*, 14:269–277, 1915.

[42] John Harris, Jeffry Hirst, and Michael Mossinghoff. *Combinatorics and Graph Theory*. Springer, 2008.

[43] Peter Hippe and Joachim Deutscher. *Design of Observer-based Compensators*. Springer, 2009.

[44] Arun V. Holden. *Chaos*. Manchester University Press, 1986.

[45] Harold Hotelling. The selection of variates for use in prediction with some comments on the general problem of nuisance parameters. *The Annals of Mathematical Statistics*, 11:271–283, 1940.

[46] Christiaan Huygens. Instructions concerning the use of pendulum-watches for finding the longitude at sea. *Philosophical Transactions of the Royal Society of London*, 4:937, 1669.

[47] M. Inoue, T. Kawazoe, Y. Nishi., and M. Nagadome. Generalized synchronization and partial synchronization in coupled maps. *Physics Letters A*, 249:69–73, 1998.

[48] R. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, 1960.

[49] W. Karush. Minima of functions of several variables with inequalities as side constraints. Master's thesis, Dept. of Mathematics, Univ. of Chicago, Chicago, Illinois, 1939.

[50] L. Kocarev and U. Parlitz. General approach for chaotic synchronization with application to communication. *Physical Review Letters*, 74:5028–5031, 1995.

[51] L. Kocarev and U. Parlitz. Generalized synchronization, predictability and equivalence of unidirectionally coupled dynamical systems. *Physical Review Letters*, 76:1816–1819, 1996.

[52] H. W. Kuhn and A. W Tucker. Nonlinear programming. In *Proceedings of 2nd Berkeley Symposium*, 1951.

[53] Benjamin Kuo and Farid Golnaraghi. *Automatic Control Systems*. Wiley, 2003.

[54] H. Kwak, S. Sung, and I. Lee. A modified smith predictor with a new structure for unstable processes. *Industrial & Engineering Chemistry Research*, 38:405–411, 1999.

[55] Y. Kyrychko, K. Blyuss, A. Gonzalez-Buelga, S. Hogan, and D. Wagg. Real-time dynamic substructuring in a coupled oscillator-pendulum system. *Proceeding of the Royal Society A*, 462:1271–1294, 2006.

[56] L. Lamport and P. Melliar-Smith. Synchronizing clocks in the presence of faults. *Journal of the Association for Computing Machinery*, 32:52–78, 1985.

[57] D. Lee, M. Lee, S. Sung, and I. Lee. Robust pid tuning for smith predictor in the presence of model uncertainty. *Journal of Process Control*, 9:79–85, 1999.

[58] T. H. Lee, Q. G. Wang, and K. K. Tan. Robust smith-predictor controller for uncertain delay systems. *AlChE*, 42:1033–1040, 1996.

[59] N. Levinson. The wiener root mean square error criterion in filter design and prediction. *Journal of Mathematical Physics*, 25:261–278, 1947.

[60] X. Liu and T. Chen. Exponential synchronization of nonlinear coupled dynamical networks with a delayed coupling. *Physica A*, 381:82–92, 2007.

[61] H. Lu and C. van Leeuwen. Synchronization of chaotic neural networs via output or state coupling. *Chaos, Solitons & Fractals*, 30:166–176, 2006.

[62] Jinhu Lü and Guanrong Chen. A time-varying complex dynamical network model and its controlled synchronization criteria. In *IEEE Transactions on Automatic Control*, 2004.

[63] Jinhu Lü, Xinghuo Yu, Guanrong Chen, and Daizhan Cheng. Characterizing the synchronizability of small-world dynamical networks. *IEEE Transactions on Circuits and Systems*, 51:787–796, 2004.

[64] J. Lundelius and N. Lynch. A new fault-tolerant algorithm for clock synchronization. In *Proceedings of the third Annual ACM Symposium on Principles of Distributed Computing*, 1984.

[65] J. Lundelius and N. Lynch. An upper and lower bound for clock synchronization. *Information and Control*, 62:190–204, 1984.

[66] Jan Lunze. *Regelungstechnik 1.* Springer, 2008.

[67] Zhongjun Maa, Zengrong Liub, and Gang Zhang. A new method to realize cluster synchronization in connected chaotic networks. *CHAOS*, 16:1–9, 2006.

[68] R. Mansouri and R. Sexl. A test theory of special relativity. *General Relativity and Gravitation*, 8:497–513, 1977.

[69] M. Marek and I. Stuchl. Synchronization in two interactinc oscillatory systems. *Biophysical Chemistry*, 3:241–248, 1975.

[70] D. G. Mead. Newton's identities. *The American Mathematical Monthly*, 99:749–751, 1992.

[71] N. Minorsky. On asynchronous action. *Journal of the Franklin Institute*, 259:209–219, 1955.

[72] T. Mori. On an estimate of the decay rate for stable linear delay systems. *International Journal of Control*, 36:95–97, 1982.

[73] T. Mori and H. Kokame. Stability of $\dot{x}(t) = ax(t) + bx(t-t)$. *IEEE Transactions on Automatic Control*, 34:460–462, 1989.

[74] A. E. Motter, C.S. Zhou, and J. Kurths. Enhancing complex-network synchronization. *Europhysics Letters*, 69:334–340, 2005.

[75] S. Munir and W. Book. Internet based teleoperation using wave variables with prediction. In *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2001.

[76] R. Nagaev. Intrinsic synchronization of nearly equivalent dynamic systems under effect of weak linear coupling. *Journal of Applied Methematics and Mechanics*, 28:266–271, 1964.

[77] R. Nagaev. General problem of synchronization in an almost conservative system. *Journal of Applied Methematics and Mechanics*, 29:801–809, 1965.

[78] R. Nagaev. Synchronization in a system of essentially nonlinear objects with a single degree of freedom. *Journal of Applied Methematics and Mechanics*, 29:209–217, 1965.

[79] J. Normey-Rico and E. Camacho. *Control of Dead-Time Processes.* Springer, 2007.

[80] L. Pecora. Synchronization of oscillators in complex networks. *Pramana*, 70:1175–1198, 2008.

[81] L. Pecora and T. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64:821–824, 1990.

[82] Louis M. Pecora and Thomas L. Carroll. Master stability functions for synchronized coupled systems. *Physical Review Letters*, 80:2109–2112, 1998.

[83] Radia Perlman. *Interconnections: Bridges, Routers, Switches, and Internetworking Protocols*. Addison Wesley Longman, 2000.

[84] Murray H. Protter. *Basic Elements of Real Analysis*. Springer, 1998.

[85] Zhihua Qu. *Cooperative Control of Dynamical Systems*. Springer, 2009.

[86] Perlman Radia. An algorithm for distributed computation of a spanningtree in an extended lan. *Computer Communication Review*, 15:44–53, 1985.

[87] Jacques Richalet and Donal O'Donovan. *Predictive Functional Control*. Springer, 2009.

[88] Daniel Rivera, Manfred Morari, and Sigurd Skogestad. Internal model control. pid controller design. *Industrial & Engineering Chemistry Process Design and Development*, 25:252–265, 1986.

[89] M. Rosenblum, A. Pikovsky, and J. Kurths. Phase synchronization of chaotic oscillators. *Physical Review Letters*, 76:1804–1807, 1996.

[90] M. Rosenblum, A. Pikovsky, and J. Kurths. From phase to lag synchronization in coupled chaotic oscillators. *Physical Review Letters*, 78:4193–4196, 1997.

[91] N. Rulkov, M. Sushchik, and L. Tsimring. Generalized synchronization of chaos in directionally coupled chaotic systems. *Physical Review E*, 51:980–994, 1995.

[92] C. Santacesaria and R. Scattolini. Easy tuning of smith predictor in presence of delay uncertainty. *Automatica*, 29:1595–1597, 1993.

[93] Martin Schottenloher. *Geometrie und Symmetrie in der Physik*. Vieweg & Teubner, 1995.

[94] Gilbert Sir Walker. Seasonal weather and its prediction. *Nature*, 132:805–808, 1933.

[95] A. Smith and K. Hashtrudi-Zaad. Neural network-based teleoperation using smith predictors. In *2005 IEEE International Conference on Mechatronics and Automation*, 2005.

[96] A. Smith and K. Hashtrudi-Zaad. Smith predictor type control architectures for time delayed teleoperation. *The International Journal of Robotics Research*, 25:797–818, 2006.

[97] Otto J. Smith. Closer control of loops with dead time. *Chemical Engineering Progress*, 53:217–219, 1957.

[98] Otto J. Smith. A controller to overcome dead time. *ISA J*, 6:28–33, 1959.

[99] T. Srikanth and S. Toueg. Optimal clock synchronization. *Journal of the Association for Computing Machinery*, 34:626–645, 1987.

[100] Jan-Olov Strömberg and Alberto Torchinsky. *Weighted Hardy Spaces*. Springer, 1989.

[101] M. I. Sumin. The first variation and pontryagin's maximum principle in optimal control for partial differential equations. *Computational Mathematics and Mathematical Physics*, 49:958–978, 2009.

[102] M. Sun, C. Zeng, and L. Tian. Projective synchronization in drive-response dynamical networks of partially linear systems with time-varying coupling delay. *Physics Letters A*, 46:6904–6908, 2008.

[103] V. Torre. Synchronization of nonlinear biochemical oscillators coupled by diffusion. *Biological Cybernetics*, 17:137–144, 1975.

[104] V. Torre. A theory of synchronization of heart pace-maker cells. *Journal of Theoretical Biology*, 61:55–71, 1976.

[105] J. Tyson and S. Kauffman. Control of mitosis by a continuous biochemical oscillation: Synchronization, spatially inhomogeneous oscillations. *Journal of Mathematical Biology*, 1:289–310, 1975.

[106] Spyros Tzafestas. *Web-Based Control and Robotics Education*. Springer, 2009.

[107] Otto von Guericke. *Experimenta nova (ut vocantur) Magdeburgica de vacuo spatio*. Polytechnische Anstalt Nü rnberg, 1672.

[108] M. Wallace, J. Sieber, S. Neild, D. Wagg, and B. Krauskopf. Stability analysis of real-time dynamic substructuring using delay differential equations. *Earthquake Engineering & Structural Dynamics*, 34:1817–1840, 2005.

[109] Liuping Wang. *Model Predictive Control System Design and Implementation Using MATLAB*. Springer, 2009.

[110] S. Wang, B. Xu, and Q. Wang. Delay analysis for teleoperation over internet and smith predictor with adaptive time-delay control. In *2005 IEEE International Conference on Robotics and Biomimetics*, 2005.

[111] Panos M. Pardalos Wanpracha Chaovalitwongse and Petros Xanthopoulos. *Springer Optimization and its Applications - Volume 38: Computational Neuroscience*. Springer, 2010.

[112] S. Wilks. Statistical prediction with special reference to the problem of tolerance limits. *The Annals of Mathematical Statistics*, 13:400–409, 1942.

[113] John Williamson. The latent roots of a matrix of special type. *Bulletin of the American Mathematical Society*, 37:585–590, 1931.

[114] Stephen J. Wright and Jorge Nocedal. *Numerical Optimization*. Springer, 2000.

[115] Jianshe Wu and Licheng Jiao. Synchronization in complex delayed dynamical networks with nonsymmetric coupling. *Physica A*, 386:513–530, 2007.

[116] Jianshe Wu and Licheng Jiao. Synchronization in dynamic networks with nonsymmetrical time-delay coupling based on linear feedback controllers. *Physica A*, 387:21112119, 2008.

[117] Kazuo Yamanaka and Etsujiro Shimenura. Effects of mismatched smith controller on stability in systems with time-delay. *Automatica*, 23:787–791, 1987.

[118] Ziad Zahreddine. Matrix measure and application to stability of matrices and interval dynamical systems. *International Journal of Mathematics and Mathematical Sciences*, 2:75–85, 2003.

[119] S. Zheng and G. Bi, Q.; Cai. Adaptive projective synchronization in complex networks with time-varying coupling delay. *Physics Letters A*, 373:1553–1559, 2009.

[120] Qing-Chang Zhong. *Robust Control of Time-Delay Systems*. Springer, 2006.

A

Proofs

Proof of Mori's First Theorem. Consider the systems described by the following linear differential difference equations:

$$\dot{\underline{x}} = \boldsymbol{A}\,\underline{x} + \boldsymbol{A}_0\,\underline{x}\,(t-\tau)\,\forall t \geq 0$$

$$\underline{x} = \phi \forall 0 > t \geq -\tau$$

The solution $\underline{x}$ for $t \geq 0$ can be expressed as

$$\underline{x} = \exp(\boldsymbol{A}t)\,\underline{x}_0 + \int_0^t \exp(\boldsymbol{A}(t-s))\,\boldsymbol{A}_0\,\underline{x}\,(s-\tau)\,ds$$

Evaluating the norm $|?|_?$ of both sides of this equation yields

$$|\underline{x}_?| \leq \|\exp(\boldsymbol{A}t)\|_?\,|\underline{x}_0|_? + \int_0^t \|\exp(\boldsymbol{A}(t-s))\|_?\,\|\boldsymbol{A}_0\|_?\,|\underline{x}\,(s-\tau)|_?\,ds$$

Using the well known inequality $\|\exp(\boldsymbol{A}t)\|_? \leq \exp(\mu_?(\boldsymbol{A})\,t)\,\forall t \geq 0$ and subsituting $|\underline{x}|_? = v$, the resulting inequality is

$$v \leq \exp(\mu_?(\boldsymbol{A})\,t)\,v_0 + \|\boldsymbol{A}_0\|_? \int_0^t \exp(\mu_?(\boldsymbol{A})(t-s))\,v\,(s-\tau)\,ds \forall t \geq 0$$

Now let the integral equation with deviating argument corresponding to the above inequality of the form be z. This gives the solution to the following scalar differential difference equation:

$$\dot{z} = \mu_?(\boldsymbol{A})\,z + \|\boldsymbol{A}_0\|_?\,z\,(t-\tau)\,\forall t \geq 0$$

Using the comparison theorem with $m = 1$, the result is

$$|\underline{x}|_? = v \leq z \forall t \geq 0$$

Hence, asymptotic stability of z implies that of $\underline{x}$. In general, the solution to the scalar differential difference equation of the form

$$\dot{x} + ax + bx\,(t-\tau) = 0$$

uniquely exists and is asymptotically stable for $b \leq 0$ if and only if $a > -b \geq 0$. By applying this result to the well known inequality $\|\exp(\boldsymbol{A}t)\|_? \leq \exp(\mu_?(\boldsymbol{A})\,t)\,\forall t \geq 0$,

$$\mu_?(\boldsymbol{A}) < -\|\boldsymbol{A}_0\|_? \leq 0$$

ensures asymptotic stability for the solution to z and therefore to $\underline{x}$. □

Proof of Mori's Second Theorem. Consider linear time-delay systems described by a differential difference equation of the form

$$\underline{\dot{x}} = \boldsymbol{A}\,\underline{x} + \boldsymbol{b}\,\underline{x}\,(t-\tau)\,\forall\tau > 0$$

where $\boldsymbol{A}$,$\boldsymbol{B} \in \mathbb{R}^{n\times n}$. Assume $l_1 = \mu\,(\boldsymbol{A}) + \|\boldsymbol{B}\| \geq 0$ and $l_2 = \mu\,(-j\boldsymbol{A}) + \|\boldsymbol{B}\|$. First, it should be noted that any norm is readily verified, and thus $l_2 \geq 0$. Next, the characteristic equation of the system is rewritten as $s = \lambda\,(\boldsymbol{A} + \boldsymbol{B}e^{-\tau s})$. Assume that there exists a solution with $\Re\,(s) \geq 0$ of the characteristic equation. Owing to the properties of the matrix measure,

$$\Re\,(s) = \Re\left(\lambda\left(\boldsymbol{A} + \boldsymbol{B}e^{-\tau s}\right)\right) \leq \mu\,(\boldsymbol{A}) + \mu\left(\boldsymbol{B}e^{-\tau s}\right) \leq$$

$$\leq \mu\,(\boldsymbol{A}) + \left\|\boldsymbol{B}e^{-\tau s}\right\| \leq \mu\,(\boldsymbol{A}) + \|\boldsymbol{B}\| = l_1$$

And in a similar manner,

$$\Im\,(s) = \Im\left(\lambda\left(\boldsymbol{A} + \boldsymbol{B}e^{-\tau s}\right)\right) \leq \mu\,(-j\boldsymbol{A}) + \mu\left(-j\boldsymbol{B}e^{-\tau s}\right) \leq$$

$$\leq \mu\,(-j\boldsymbol{A}) + \left\|\boldsymbol{B}e^{-\tau s}\right\| \leq \mu\,(-j\boldsymbol{A}) + |\boldsymbol{B}\| = l_2$$

It should be noticed here that the solutions in s are distributed symmetrically on the complex plane with respect to the real axis. Thus, all the unstable solutions exist within the rectangular region $[0, l_1 + jl_2]$ and its conjugate complex $[0, l_1 - jl_2]$. On account of this statement, it is enough to prove that there are no solutions to s in this region. For this, it can be show that $Re\,(s) < 0$ is satisfied at all points in this region. Substituting $z = e^{-\tau s}$, is is possible to rewrite s as

$$s = \lambda\,(\boldsymbol{A} + \boldsymbol{B}z)$$

It is known that the right-hand side of this equation is an algebraic function of z and analytic except at the finite points where the equation in s

$$f\,(s, z) = \det\,(s\boldsymbol{I} - \boldsymbol{A} - \boldsymbol{B}z) = 0$$

has multiple solutions. However, by regarding its domain as a Riemann surface, it is possible to consider functions $\lambda\,(\boldsymbol{A} + \boldsymbol{B}z)$ to be analytic in a connected bounded region on the z-plane. No, the real and imaginary parts of an analytic function in some domain D are harmonic functions, which are characterized by satisfying Laplace's equation. The other key result to prove the theorem is the maximum principle for harmonic functions. The maximum value of a harmonic function on a closed bounded set D is taken on the boundary of D. The region $[0, l_1 + jl_2]$ on the s-plane corresponds to a closed region bounded by a sector and two concentric circles with radii $e^{-l_1\tau}$ and 1. Evidently, the boundaries of the region in the s-plane also correspond to those in the z-plane. The maximum values of the region in the s-plane are taken out of its edges. Snce the complex conjugate of an eigenvalue of a complex matrix is an eigenvalue of the complex conjugate of the matrix, it is enough to investigate three edges, to check the maximum value of $\Re\,(\lambda\,(\boldsymbol{A} + \boldsymbol{B}e^{-\tau s}))$ in the regions in the s-plane. The assumtion of the theorem assert that the real part of $\lambda\,(\boldsymbol{A} + \boldsymbol{B}e^{-\tau s})$ is negative in this region, while $\Re\,(s)$ is nonnegative in the same region. This indicates that the characteristic equation has no solutions within the instable region. □

Proof of the Newton identities. Consider the elementary symmetric function of $x_1, \cdots, x_n$

$$s_k = \sum_{1\leq i_1 < i_2 < \cdots < i_k \leq n} x_{i_1} x_{i_2} \cdots x_{i_k} \forall k = 1, 2, \cdots, n$$

and the Newton functions are

$$p_k = \sum_{i=1}^{n} (x_i)^k \,\forall k = 1, 2, 3, \cdots$$

The Newton identities are

$$p_k + \sum_{i=1}^{k-1} (-1)^i\, p_{k-i} s_i + (-1)^k\, k s_k = 0 \forall 1 \leq k \leq n$$

and

$$p_k + \sum_{i=1}^{n} (-1)^i p_{k-i} s_i = 0 \forall k > n$$

With

$$f(x) = \prod_{i=1}^{n} (x - x_i) = x^n + \sum_{i=1}^{n} (-1)^i s_i x^{n-i}$$

since

$$0 = x_j^{k-n} f(x_j) = x_j^k + \prod_{i=1}^{n} (-1)^i s_i x_j^{k-i}$$

it is possible to state

$$\sum_{j=1}^{n} x_j^{k-n} f(x_j) = 0 = \sum_{j=1}^{n} x_j^k + \sum_{j=1}^{n} \sum_{i=1}^{n} (-1)^i s_i x_j^{k-i} = p_k + \sum_{i=1}^{n} (-1)^i s_i p_{k-i}$$

which are the relations for $k > n$. For $k \leq n$, before describing the suggested derivation, the notations to be employed have to be defined. Let $(a_1, \cdots, a_n)$ where the a_i are nonnegative integers and $a_i \geq a_{i+1}$, represent $\sum x_{i_1}^{a_1} x_{i_2}^{a_2} \cdots x_{i_n}^{a_n}$ where the sum is over all permutations $(i_1, \cdots, i_n)$ of $(1, 2, \cdots, n)$ which yield distinct terms. If $a_i = 0 \forall i > t$, there will be no ambiguity if $(a_1, \cdots, a_t)$ is written instead of $(a_1, \cdots, a_n)$. To make the notation simpler, let $s_i = (1_i)$, a sequence of i ones, and if $t \geq 1$, let $(t, 1_i) = (c_1, \cdots, c_{i+1})$, where $c_1 = t$ and $c_j = 1 \forall j > 1$. To obtain the Newton identity involving $p_1, \cdots, p_k$, write t equations, where $t = \min(k-1, n)$:

$$(k-i)(1_i) = (k-i+1, 1_{i-1}) + (k-i, 1_i) \forall i = 1, \cdots, t$$

If $n \geq k = t + 1$, the last equation is

$$(1)(1_{k-1}) = (2, 1_{k-2}) + k(1_k)$$

while, if $k > n = t$, the last equation is

$$(k-n)(1_n) = (k - nm + 1, 1_{n-1})$$

since the symbol $(k-n, 1_n)$, having $n+1$ entries, represents the polynomial zero. By multiplying the ith equation by $(-1)^{i-1}$ and adding the equation, the Newton identities are obtained. □

Proof of the Williamson Theorem. Let $\boldsymbol{A}$ be a square matrix of n rowns and n columns and let $\lambda_1, \lambda_2, \cdots, \lambda_n$ be the latent roots of $\boldsymbol{A}$, where the λ_i need not be distinct. Then by a well known theorem on matrices there exists a non-singular matrix $\boldsymbol{X}$, which transforms $\boldsymbol{A}$ into a matrix whose elements in the leading diagonal are the latent roots of $\boldsymbol{A}$, while all the elements to the left of the leading diagonal are zero. This may be expressed by the matrix equation

$$\boldsymbol{X}\boldsymbol{A}\boldsymbol{X}^{-1} = \begin{bmatrix} \lambda_1 & r_{12} & \cdots & r_{1n} \\ 0 & \lambda_2 & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} = \boldsymbol{\Lambda}$$

It follows from this relation, that if $f(\boldsymbol{A})$ is a rational function of the matrix $\boldsymbol{A}$, of which the denominatior is non-singular, the matrix $\boldsymbol{X}$ transforms the matrix $f(\boldsymbol{A})$ into a matrix of the same type as $\boldsymbol{\Lambda}$, where the elements in the leading diagonal are $f(\lambda_1), f(\lambda_2), \cdots, f(\lambda_n)$. Or

$$\boldsymbol{X} f(\boldsymbol{A}) \boldsymbol{X}^{-1} = f(\boldsymbol{\Lambda})$$

where the elements to the left of the leading diagonal in $f(\boldsymbol{\Lambda})$ are all zero. If now

$$\boldsymbol{B} = (\boldsymbol{A}_{ij}) \forall i, j = 1, 2, \cdots, m$$

denotes an m-rowed, square matrix all of whose elements are n-rowed square matrices $\boldsymbol{A}_{ij}$, $\boldsymbol{B}$ is a square matrix of order mn in the elements of the matrices $\boldsymbol{A}_{ij}$. If

$$\boldsymbol{Y} = (\boldsymbol{X}_{ij}) \forall i, j = 1, 2, \cdots, m$$

is a matrix of the same type as $\boldsymbol{B}$, where $\boldsymbol{X}_{ij} = 0$, if $i \neq j$, and $\boldsymbol{X}_{ii} = \boldsymbol{X}$, then by simple multiplication

$$\boldsymbol{Y}\boldsymbol{B}\boldsymbol{Y}^{-1} = \left(\boldsymbol{X}\boldsymbol{A}_{ij}\boldsymbol{X}^{-1}\right)$$

In particular, if $\boldsymbol{A}_{ij} = f_{ij}(\boldsymbol{A})$ be rational functions, whose denominators are non-singular, of the original matrix $\boldsymbol{A}$, then

$$\boldsymbol{C} = (f_{ij}(\boldsymbol{A})) = \boldsymbol{M}[\boldsymbol{A}]$$

is an nm-rowed square matrix of the same type as $\boldsymbol{B}$. It follows that

$$\boldsymbol{Y}\boldsymbol{C}\boldsymbol{Y}^{-1} = \left(\boldsymbol{X} f_{ij}(\boldsymbol{A}) \boldsymbol{X}^{-1}\right) = (f_{ij}(\boldsymbol{\Lambda})) = \boldsymbol{D}$$

Accordingly the latent roots of $\boldsymbol{C}$ are the same as the latent roots of $\boldsymbol{D}$. But the latent roots of $\boldsymbol{D}$ are the roots ofthe equation $|\boldsymbol{D} - \lambda\boldsymbol{I}| = 0$, obtained by equating the determinant of $\boldsymbol{D} - \lambda\boldsymbol{I}$ to zero. If Δ denote the determinant of $\boldsymbol{D} - \lambda\boldsymbol{I}$ and if, for brevity, the element in the ith row and the jth column of Δ is indicated by (i, j), it follows that

$$(sn + a, rn + i) = 0 \forall a > ir, s = 0, 1, \cdots, m - 1$$

$$(sn + i, rn + i) = f_{s+1,r+1}(\lambda_i) - \delta_s^r \lambda \forall i = 1, 2, \cdots, n$$

From $\boldsymbol{\Delta}$ a new determinant Δ' is formed by taking for the rows and columns of Δ' in succession the

$$\begin{array}{ccccc} 1\text{st} & (n+1)\text{th} & (2n+1)\text{th} & \cdots & (n(m-1)+1)\text{th} \\ 2\text{nd} & (n+2)\text{th} & (2n+2)\text{th} & \cdots & (n(m-1)+2)\text{th} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ n\text{th} & 2n\text{th} & tn\text{th} & \cdots & mn\text{th} \end{array}$$

rows and columnds of Δ. Thus, Δ' is a determinant in which all the elements in the first m columnds vanish except those in the first m rows. All the elements in the second m columns vanish except those in the first $2m$ rows, and in general all the elements in the ith set of m columns vanish except those in the first im rows. Therefore Δ' is the product of the n determinants formed fromthe first m rows and the first m columns, from the second m rows and the second m columns, etc. But the determinant formed from the kth set of m columns, and the kth set of mR rows of Δ' is

$$\begin{vmatrix} f_{11}(\lambda_k) - \lambda & f_{12}(\lambda_k) & \cdots & f_{1m}(\lambda_k) \\ f_{21}(\lambda_k) & f_{22}(\lambda_k) - \lambda & \cdots & f_{2m}(\lambda_k) \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1}(\lambda_k) & f_{m2}(\lambda_k) & \cdots & f_{mm}(\lambda_k) - \lambda \end{vmatrix}$$

or shorter $|f_{rs}(\lambda_k) - \delta_r^s \lambda| \forall r, s = 1, 2, \cdots, m$. Hence

$$\Delta' = \prod_{k=1}^{n} |\boldsymbol{M}[\lambda_k] - \lambda\boldsymbol{I}|$$

But, as Δ differs from Δ' by at most a sign, $\Delta' = 0$ is equivalent to $\Delta = 0$ and accordingly the latent roots of $\boldsymbol{D}$ and therefore the latent roots of $\boldsymbol{C}$ are the nm roots of the n equations obtained by equating the n factors $|\boldsymbol{M}[\lambda] - \lambda\boldsymbol{I}|$ in turn to zero. □

B

Source Codes

Symbolic Computation of the Modal Matrix

```
1  %This algorithm computes the modes of a smith synchronization
2  %afterwards the matrix of the differences of the master and slave modes
       is
3  %computed.
4
5  clear workspace
6  clear all
7  clc
8  syms w1 w2 K T t n
9  E=[0 0 0; 0 w1 0; 0 0 w2];
10 W=[1 1 1 ; 1 0 0 ; 1 ((K*T)/(2))+sqrt((((K^2)*(T^2))/(4))-1) ((K*T)/(2)
       )-sqrt((((K^2)*(T^2))/(4))-1)];
11 X=inv(W);
12 A=W*E*X;
13 B=A*((1)/(T));
14 DELT=[0 0 0 ; 0 0 0 ; -K K -K];
15 C=B-DELT;
16 L=C*t;
17 [V,D]=eig(L);
18 F=exp(D);
19 F(1,2)=0;
20 F(1,3)=0;
21 F(2,1)=0;
22 F(2,3)=0;
23 F(3,1)=0;
24 F(3,2)=0;
25 ZZ=V*F/V;%Matrix of the time solution of form x(t)=ZZ*xT0 where [0,T]
       is the preshape interval
26 d=ZZ(1,:)-ZZ(2,:);%mode difference between master and slave. these
       elements are the ones we would like to transform to the nullspace
       by multiplication with the preshape vector. the goal is therefore d
       *xT0=0
27 d=subs(d,w1,lambertw(2,sqrt((((K^4)*(T^4))/(4))-(K^2)*(T^2))-(((K^2)*(T
       ^2))/(2))));%substitute the lambert function with the series for
       the primary branch
```

```
28 d=subs(d,w2,lambertw(2,-sqrt((((K^4)*(T^4))/(4))-(K^2)*(T^2))-(((K^2)*(
       T^2))/(2))));
```

Remaining Error for a Zero Row Sum Negative Trace Representation

```
1  function [ deltoid ] = errdelta( parm )
2  %function [ deltoid ] = errdelta( parm )
3  %   computes the error for a zero row sum negative trace matrix.
4  %   The algorithm receives the 6 variable matrix elements as input and
       puts
5  %   out the overall error remaining for both matrices resulting in a
       zero
6  %   row sum negative trace coupling
7  a=parm(1);
8  b=parm(2);
9  c=parm(3);
10 d=parm(4);
11 e=parm(5);
12 f=parm(6);
13 global Tglob
14
15 syms a11 a12 a22 a21 b11 b12 T
16 A=[a11 a12 ; a21 a22];
17 B=[b11 b12 ; 0 0];
18 eB=expm(B*T);
19 [VargW,DargW]=eig(A*T*eB);
20 W=VargW*lambertw(DargW)/VargW;
21 U=(W/T)+B;
22 d1=abs(U(1,1)+U(1,2));
23 d2=abs(U(1,1)+U(2,1));
24 d3=abs(U(2,1)+U(2,2));
25 d4=abs(U(2,2)+U(1,2));
26 err=d1+d2+d3+d4;
27 deltoid=double(subs(subs(subs(subs(subs(subs(subs(err,a11,a),a12,b),a21
       ,c),a22,d),b11,e),b12,f),T,Tglob))
28 end
```

Stability Test Algorithm for 2×2 Matrices

```
1  function [ c,ceq ] = stable( parm )
2  %function [ c,ceq ] = stable( parm )
3  %   puts out the stability (trace-stability) of the coupling terms as
4  %   function of both matrices
5  a=parm(1);
6  b=parm(2);
7  g=parm(3);
8  d=parm(4);
9  e=parm(5);
10 f=parm(6);
11 global Tglob
12
13 syms a11 a12 a22 a21 b11 b12 T
14 A=[a11 a12 ; a21 a22];
15 B=[b11 b12 ; 0 0];
16 eB=expm(B*T);
17 [VargW,DargW]=eig(A*T*eB);
```

```
W=VargW*lambertw(DargW)/VargW;
U=(W/T)+B;
d1=abs(U(1,1)+U(1,2));
d2=abs(U(1,1)+U(2,1));
d3=abs(U(2,1)+U(2,2));
d4=abs(U(2,2)+U(1,2));
err=d1+d2+d3+d4;
c=real(double(subs(subs(subs(subs(subs(subs(subs(U(1,1),a11,a),a12,b),
    a21,g),a22,d),b11,e),b12,f),T,Tglob)));
ceq=0;
end
```

Modal Test Algorithm for 2×2 Matrices

```
function [ Ud ] = utest( parm )
%function [ Ud ] = utest( parm )
%   Puts out the matrix representation of the coupling terms as a
    function
%   of the coupling matrices
a=parm(1);
b=parm(2);
g=parm(3);
d=parm(4);
e=parm(5);
f=parm(6);
global Tglob

syms a11 a12 a22 a21 b11 b12 T
A=[a11 a12 ; a21 a22];
B=[b11 b12 ; 0 0];
eB=expm(B*T);
[VargW,DargW]=eig(A*T*eB);
WDargW=[lambertw(0,DargW(1,1)) 0 ; 0 lambertw(0,DargW(2,2))];
W=VargW*WDargW/VargW;
U=(W/T)+B;
Ud=real(double(subs(subs(subs(subs(subs(subs(subs(U,a11,a),a12,b),a21,g
    ),a22,d),b11,e),b12,f),T,Tglob)));
end
```

Cost Function for Synchronizability of Mass Spring Damper Systems

```
%This algorithm computes the parameters and costs of the unstable
    parameter
%field
clc
clear workspace

f=0.1;
e=2;
T=0.2;

l1max=-(f/6)+sqrt(0.7044444*f^2+0.4166666*e^2)+1.8*sqrt(f^2+e^2);
l2max=0.6455*e+1.8*sqrt(f^2+e^2);

l1=0:0.01:l1max;
sub1=l1max/0.01
```

```
sub2=l2max/sub1
l2=0:sub2:l2max;

L1=meshgrid(l1);
L2=meshgrid(l2);

costfunc=f*(-0.25+((3/8)+2*exp(-2*T*(L1'+i*L2))-2*exp(-T*(L1'+i*L2))-1).^(0.5));
costfunc1=f*(-0.25+((3/8)+2*exp(-2*T*(i*l2))-2*exp(-T*(i*l2))-1).^(0.5));
costfunc2=f*(-0.25+((3/8)+2*exp(-2*T*(l1max+i*l2))-2*exp(-T*(l1max+i*l2))-1).^(0.5));
costfunc3=f*(-0.25+((3/8)+2*exp(-2*T*(l1+i*l2max))-2*exp(-T*(l1+i*l2max))-1).^(0.5));
c=real(costfunc);
c1=real(costfunc1);
c2=real(costfunc2);
c3=real(costfunc3);
mesh(c)
```

Search Algorithm for Maxima in Mori's Parameter Plane

```
function [ maxcost ] = maximalcost( e, T )
%function [ maxcost ] = maximalcost( e, T )
%computes the maximal cost value in the unstable parameter field for the
%experimental systems setup
f=0.1; l1max=-(f/6)+sqrt(0.7044444*f^2+0.4166666*e^2)+1.8*sqrt(f^2+e^2); l2max=0.6455*e+1.8*sqrt(f^2+e^2);
steps=0.02; l1=0:steps:l1max; sub1=l1max/steps; sub2=l2max/sub1; l2=0:sub2:l2max;
costfunc=f*(-0.25+((3/8)+2*exp(-2*T*l1)'*(cos(-2*T*l2)+i*sin(-2*T*l2))-2*exp(-T*l1)'*(cos(-T*l2)+i*sin(-T*l2))-1).^(0.5));
c=real(costfunc);
maxcost=max(max(c));
end
```

Affidavit

I hereby declare that the present Bachelor thesis *Monitoring Delayed Systems with Spatial Distribution in Presence of Realtime Requirements* has been written only by the undersigned and without any assistance from third parties.

Furthermore, I confirm that no sources have been used in the preparation of this thesis other than those indicated in the thesis itself.

Duisburg, 23/02/11 Jan Maximilian Montenbruck