Czech Technical University in Prague
Faculty of Mechanical Engineering

Ph.D. Thesis

Analysis and Synthesis of Time Delay System Spectrum

Ing. Tomáš Vyhlídal

Supervisor
Prof. Ing. Pavel Zítek, DrSc.

Branch of science: Control and Systems Engineering

2003

ACKNOWLEDGEMENTS

First of all, I would like to thank my parents for their support during my studies.

I would like to express my gratitude to my supervisor Professor Pavel Zítek for his inspiration and encouragement during my work at this thesis.

I would like to acknowledge the financial support of the research, the results of which are presented in this thesis, provided by the Ministry of Education of the Czech Republic under the Project LN 00 B096.

Also, I would like to thank Hlávka's Foundation for the financial support during my postgraduate studies.

ANOTACE

Použití lineárních modelů pro popis reálných systémů (jejichž dynamika je většinou nelineární) je konvenčním přístupem používaným v oboru automatického řízení. Pomocí klasického lineárního modelu můžeme popsat pouze část dynamiky systému (dynamiku v okolí tzv. pracovního bodu). Výhoda lineárních modelů spočívá v jejich snadné analýze a snadném návrhu řízení pomocí metod dostupných pro tuto třídu modelů. Hlavní nevýhoda těchto modelů, kromě jejich omezené platnosti, je dána tím, že jejich jedinými dynamickými elementy jsou integrátory aproximující soustředěné akumulace. Z toho vyplývá, že popis systémů s rozloženými parametry (včetně časových zpoždění), se kterými se setkáváme v technické praxi, je pomocí tohoto přístupu poněkud problematický. Mnohem lepších výsledků při modelování takovýchto systémů lze dosáhnout zahrnutím dopravních zpoždění do struktury lineárního modelu. Takto získaný popis daný soustavou lineárních funkcionálních diferenciální rovnic nabízí větší variabilitu struktury modelu než klasický popis, což umožňuje přesněji popsat dynamiku systému. Na druhou stranu, analýza modelů s dopravním zpožděním a syntéza řízení je zpravidla složitější než u klasického přístupu. Typickou vlastností modelů s dopravním zpožděním je nekonečné spektrum vlastních hodnot (pólů a nul) systému. V této práci je navržena metoda analýzy dynamiky systémů s dopravním zpožděním na základě znalosti spekter pólů a nul systému. Pro výpočet pólů a nul je navržen původní algoritmus založený na mapování charakteristických funkcí systému. Důležitost jednotlivých pólů je posuzována na základě váhových funkcí přenosů prvního a druhého řádu které jsou získány rozložením přenosu systému použitím zobecněné Heavisideovy věty o rozkladu. Tímto způsobem je možné definovat skupinu pólů (z nekonečné množiny pólů) které jsou rozhodující v dynamice systému. Určení dominantních pólů systému umožňuje nejenom analyzovat módy systému, ale také dynamiku systému výhodně změnit přesunutím těchto dominantních pólů. Tak jako u klasických systémů, i u systémů s dopravním zpožděním je možné dynamiku změnit zavedením zpětných vazeb od stavových proměnných systému. Na druhou stranu je nutné podotknout, že touto metodou můžeme umístit pouze malou část spektra pólů. V této práci je proveden rozbor této metody řízení při aplikaci na systémy s dopravním zpožděním a je předložena metoda návrhu zpětných vazeb. Snadná aplikovatelnost metody pro analýzu dynamiky systému s dopravním zpožděním a efektivnost návrhu koeficientů zpětných vazeb od stavových veličin přesunutím dominantních pólů jsou ukázány v aplikačním příkladu, kde analyzovaným systémem je laboratorní tepelná soustava.

ABSTRACT

In the field of control engineering, the classical approach used in modelling of the real plant dynamics is based on the linear model given by a set of ordinary linear differential equations. Since the dynamics of the real plants are non-linear as a rule, the linear model fits the dynamics of the plant only in a vicinity of the operational point at which the system has been identified. The linear model is easy to handle and the control design can be easily performed using a method available for this class of systems (models). Its main drawback, besides the restricted validity, is given by the fact that the only dynamical elements of the model are the integrators representing the point accumulations. Thus, using this modeling approach, it is difficult to fit the dynamics of the plants with distributed parameters or with the transportation phenomenon involved. Much better results in modeling of this class of systems are achieved involving time delays in the structure of the model. The model obtained in this way consists of a set of linear functional differential equations. Such a model with more variable structure (called time delay system) provides the opportunity to fit better the plant dynamics than delay free linear model. On the other hand, the analysis of the dynamics of a time delay system and its control synthesis is more complicated as a rule. The typical features of time delay systems are the infinite spectra of poles and zeros. In this thesis, the methodology for analyzing the dynamics of time delay system is introduced based on the knowledge of the decisive sets of spectra of poles and zeros. An original algorithm for computing poles and zeros of the system with delays is designed based on the mapping the characteristic functions of the system. The significance of the poles is evaluated on the basis of the weighting functions corresponding to the first and second order transfer functions resulting from applying the generalized Heaviside expansion to the transfer function of the time delay system. In this way, it is possible to define a group of the dynamics determining poles (from the infinite set of poles). Assessing the group of the most significant poles allows not only the modes to be analyzed but also the dynamics to be positively changed by shifting the most significant poles into more favorable positions. In the same way as in the case of classical delay free systems, this shifting of the poles can be accomplished using the coefficient feedback loops from the state variables. However, it should be noted, that using this pole placement method, only few poles can be prescribed while the rest of infinitely many poles is placed spontaneously. In this thesis, the features of the pole placement method using the coefficient feedback loops from the state variables applied to time delay systems are investigated and an effective method for feedback design is presented. In order to demonstrate that the method for analyzing the dynamics of systems with time delays and the extension of pole placement method described in this thesis are easily applicable, an application example, in which the system being analyzed is a laboratory heating system, is included in the end of this thesis.

CONTENTS

LIST OF SYMBOLS

Abreviations

IMC	internal model control
LMS	linear multi-step (numerical method)
MIMO	multi-input multi-output (system)
NFDE	neutral functional differential equation
RFDE	retarded functional differential equation
SISO	single-input single-output (system)
TDS	time delay system(s)
i.g.s	infinitesimal generator of the semigroup
s.o.	solution operator

Symbols

$\mathcal{A}$	infinitesimal generator of the semigroup of autonomous TDS
$\mathcal{A}$	matrix from Butcher tableau corresponding to the numerical method
$\mathbf{A}(\tau)$	functional matrix of system dynamics
$\mathbf{A}(s)$	Laplace transform of $\mathbf{A}(\tau)$
$\mathcal{A}_h$	discrete approximation of $\mathcal{A}$ for h
$\mathbf{B}(\tau)$	system input functional matrix
$\mathbf{B}(s)$	Laplace transform of $\mathbf{B}(\tau)$
$\mathcal{C}$	Banach space
$\mathbb{C}$	complex space
$\mathbf{C}$	matrix of system outputs
$\mathcal{D}$	suspect region of the complex plane in which the function roots are computed
$D(s)$	exponential polynomial, Laplace transform of the difference function
$F(s)$	first order anisochronic filter
$G(s)$	transfer function of the model approximated real plant dynamics
$g(t)$	weighting function of $G(s)$
$G_{wy}(s)$	transfer function of the closed loop system
h	sampling period, step of the discretization
$\mathbf{H}_i$	matrices of the difference equation associated with the neutral system
$H_i(s)$	transfer functions of the system modes resulting from the Heaviside expansion
$h_i(t)$	weighting functions of $H_i(s)$
h_{e_i}	pole significance evaluating criterion based on evaluating the weighting functions of the transfer functions corresponding to the modes of TDS
int(·)	largest integer less than ·
$\mathbf{I}$	identity matrix
$I(\beta,\omega)$	imaginary part of $M(\beta+\mathrm{j}\omega)$ ($N(\beta+\mathrm{j}\omega)$, $D(\beta+\mathrm{j}\omega)$)
$\mathbf{I}^0$	matrix with the values of β and ω approximating $I(\beta,\omega)=0$
$\mathbf{I}^d$	matrix of the values of $I(\beta,\omega)$ on the grid of nodes on $\mathcal{D}$
Im(·)	imag part of ·
$\mathbf{K}(s)$, $\mathbf{K}$	feedback matrix
$\mathcal{L}$, $\mathcal{D}$	linear mapping from $\mathcal{C}$ to $\mathbb{R}$
$M(s)$	denominator of $G(s)$, characteristic function of the system (quasi)polynomial
$N(s)$	numerator of $G(s)$, (quasi)polynomial

$N_{\mathcal{D}}$	number of zeros of the function being analysed in $\mathcal{D}$
$P(s)$	transfer function of the model describing true real plant dynamics
$P_{\mathcal{D}}$	number of poles of the function being analysed in $\mathcal{D}$
$\mathbb{R}$	real space
$R(\beta,\omega)$	real part of $M(\beta+\mathrm{j}\omega)$ ($N(\beta+\mathrm{j}\omega)$, $D(\beta+\mathrm{j}\omega)$)
$R(\lambda_i)$	residues corresponding to λ_i
$\mathbf{R}^0$	matrix with the values of β and ω approximating $R(\beta,\omega)=0$
$\mathbf{R}^{\mathrm{d}}$	matrix of the values of $R(\beta,\omega)$ on the grid of nodes on $\mathcal{D}$
$\mathrm{Re}(\cdot)$	real part of $\cdot$
s	complex variable, operator of Laplace transform
s	stage of the numerical method
s_i	root of the function with argument s
$\mathbf{S}^+$	Moore-Penrose inverse of $\mathbf{S}$
$\mathcal{T}^{\circ}(t)$	solution operator of autonomous TDS for t
t	time
$\mathbf{u}(t)$	vector of system input variables
$\mathbf{u}(s)$	Laplace transform of $\mathbf{u}(t)$
$\mathbf{x}(t)$	vector of system state variables
$\mathbf{x}(s)$	Laplace transform of $\mathbf{x}(t)$
$\boldsymbol{x}_k$	state of the discrete system corresponding to TDS at discrete time k
$\boldsymbol{x}_t$	state of TDS at time t
$\mathbf{y}(t)$	vector of system output variables
$\mathbf{y}(s)$	Laplace transform of $\mathbf{y}(t)$
z	operator of Z transform, complex variable
z_i	poles of discretized system
η	time delay
$\Delta\sigma$	pole shifting increment in the continuous pole placement
$\sigma(\cdot)$	eigenvalue spectrum of $\cdot$
$\Delta(\lambda)$	characteristic matrix of the system
$\Phi(s)$	argument variable
$\boldsymbol{\Phi}$	matrix of discrete approximation of $\mathcal{T}^{\circ}(h)$
δ	operator of Delta transform, complex variable
δ_i	poles of δ-model
υ_i	eigenvalues of the essential spectrum of neutral system
λ	complex variable
λ_i	poles of the system
σ_i	prescribed spectrum of poles
μ_i	zeros of the system
ε_N	absolute value of the difference of two successive approximations of Newton's method
Δ_s	increment of the grid of nodes on $\mathcal{D}$
τ	delay variable, time delay

Remark If the physical units (dimensions) of the variables are not given with the values of the variables, these are considered dimensionless, i.e., in a range of a certain elementary units. In this way, the variables are presented in the examples in Chapters 3 and 4.

1. INTRODUCTION, THE ACTUAL STATE OF RESEARCH AND THE THEORETICAL BACKGROUND OF TIME DELAY SYSTEMS

Time delay systems (TDS) provide an alternative way for building the models of real plants. Beside the integrators, involving the delays as the other dynamical elements brings about favourable features in fitting the real plant dynamics. On the other hand, the functionality of the system matrices (considering the linear models) results in infinite spectra of the system poles and zeros. This inconvenient feature of time delay systems causes difficulties in the analysis and control design of TDS.

The main topics of this thesis are the analysis of TDS dynamics based on locating the distribution of the dominant poles and zeros and the control design of TDS based on the pole placement using the coefficient feedback from the state variables. In chapter 1, I am going to make an overview of the key literature sources dealing with the time delay systems. Also the necessary background of the theory of TDS will be introduced in chapter 1. The stress will be laid on the concept of poles and zeros of TDS and on the methods available for their computing. Also the overview of the available methods of the pole placement based control of TDS will be made in this chapter.

1.1 Concept of state in classical state space description, system poles and zeros

1.1.1 Description of classical (delay free) linear system

Before outlining the theoretical background of time delay systems let us briefly mention the concept of classical linear model description. Since the very beginning of the modern control engineering, see, e.g., Zadeh and Desoer, (1963), linear state space description has been very valuable and frequently used tool in modelling. There are several reasons why this description has become common in the field of control engineering. To begin with, the model written in the linear state space description

$$\begin{aligned} \frac{d\mathbf{x}(t)}{dt} &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) \end{aligned} \tag{1.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of state variables, $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^p$ are the vectors of system inputs and outputs, respectively, and $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{p \times n}$ are coefficient matrices, provides the possibility to analyse easily the basic features of the system, i.e., stability, the modes of dynamics, controllability and observability (Ogata, 1997). Description (1.1) is also convenient from the point of control design. In many cases, model (1.1) is the result of linearization of a non-linear model with the right-hand side given by a general function $f(\mathbf{x}, \mathbf{u}, t)$. Such a linear model is valid in the vicinity of the operational point in which the linearization has been accomplished. In fact, this limited validity of the linear model has to be always taken into consideration, even if the model has been built directly as the linear model. It is given by the fact that the dynamics of the real plants are nonlinear as a rule. On the other hand, many real processes retain or are kept by the controllers close to a certain system state. The description by means of linear model (1.1) is mostly adequate and sufficient for these processes. State space description (1.1) is also convenient with respect to the practical realization of the model. The model consisting of a set of first order differential equations can be easily built in a software modelling tool, e.g., Matlab, Simulink. There have been developed many powerful numerical methods for simulation of the system in the state space description (Hairer, et al, 1990), (Hairer, and Wanner, 1999). Another convenient feature of the linear system description is that the Laplace transform of (1.1), which follows, can be obtained very easily

$$s\mathbf{x}(s) = \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s) + \mathbf{x}_0$$
$$\mathbf{y}(s) = \mathbf{C}\mathbf{x}(s) \tag{1.2}$$

where s (the complex variable) is the Laplace operator, $\mathbf{x}(s)$, $\mathbf{u}(s)$ and $\mathbf{y}(s)$ are the Laplace transforms of $\mathbf{x}(t)$, $\mathbf{u}(t)$ and $\mathbf{y}(t)$ and $\mathbf{x}_0$ is the vector of initial conditions.

State description (1.1) is so familiar in the field of control engineering that it does not need more comments. The objective of this chapter is to break ground for pointing out the analogies and differences of classical model (1.1) and the model of a time delay system, which will be introduced in section 1.2. Before turning the attention to the description of time delay systems, let us mention the concepts of system state, system poles and zeros corresponding to (1.1), which will be investigated more deeply for the class of time delay systems in the further sectionss.

1.1.2 State of the system

It is well known that the state of model (1.1) at time t_0 is given by the vector of state variables $\mathbf{x}(t_0)$. If the system state $\mathbf{x}(t_0)$ is known, any plant output can be computed at all future times, i.e., $t>t_0$ as a function of the state variables and the present and future values of inputs. Therefore, the initial conditions of model (1.1) are only given by the value of state variable vector at t=0, i.e., $\mathbf{x}(0)$. This fact of dealing only with the actual values of the variables brings about difficulties in describing systems with delay effects and distributed parameters involved. The impossibility to spread the variables over the time segment of the system last history is compensated by increasing the order of the system, i.e., introducing additional state variables.

1.1.3 System poles and zeros

Let description (1.2) be transformed into input-output relation given by the transfer matrix

$$\mathbf{G}(s) = \frac{\mathbf{y}(s)}{\mathbf{u}(s)} = \mathbf{C}\left[s\mathbf{I} - \mathbf{A}\right]^{-1}\mathbf{B} = \frac{1}{\det\left[s\mathbf{I} - \mathbf{A}\right]}\mathbf{C}\,\mathrm{adj}\left[s\mathbf{I} - \mathbf{A}\right]\mathbf{B} \tag{1.3}$$

Each element of the transfer matrix is given as a ratio of two polynomials

$$G_{kl}(s) = \frac{y_k(s)}{u_l(s)} = \frac{N_{kl}(s)}{M(s)}, \quad k = 1..p,\ l = 1..m \tag{1.4}$$

The system poles λ_i, i=1..n are defined as the solutions of the equation

$$M(s) = \det(s\mathbf{I} - \mathbf{A}) = 0 \tag{1.5}$$

which is called the characteristic equation of the system. It is well known, that the solutions of (1.5), i.e., the system poles, are equivalent to the eigenvalues of matrix $\mathbf{A}$. If λ is an eigenvalue of $\mathbf{A}$ and $\mathbf{v}$ is the eigenvector corresponding to the eigenvalue λ, then the following equation holds

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0 \tag{1.6}$$

Equation (1.6) is the set of homogeneous linear algebraic equations that acquires a non-zero solution $\mathbf{v}$ if the characteristic matrix $\Delta(\lambda) = (\mathbf{A} - \lambda\mathbf{I})$ is singular, i.e.,

$P(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}) = 0$, where $P(\lambda)$ is the characteristic polynomial of the matrix **A**. Let us compare the polynomials $P(\lambda)$ and $M(s)$. The first difference between $P(\lambda)$ and $M(s)$ is in their arguments. Obviously, this difference is only formal, because the operator s as well as λ are the complex variables. The second difference is in the signs inside the brackets which is also formal, because $\det(\mathbf{A} - \lambda\mathbf{I}) = -\det(\lambda\mathbf{I} - \mathbf{A})$ and both polynomials have the same roots. With respect to this equivalence of the system poles and eigenvalues of **A**, the poles of the system are also called the system eigenvalues.

As regards the physical meaning of the system poles, provided that no zero-pole cancellation occurs, every pole generates a natural mode in the system response to any given input. As the fast poles are referred to the poles that are farther away from the stability boundary than the other poles. The opposite to the fast poles are the slow poles, referred to as the dominant poles that are conversely closer to the stability boundary than the other poles (Goodwin, et al., 2001). The poles can be either real or complex conjugate. Real poles correspond to the non-oscillatory modes with time constants $T_i = 1/\lambda_i$, while complex poles $\lambda_{i,i+1} = \beta_i \pm j\omega_i$ correspond to the oscillatory modes represented, e.g., by the frequency ω_i and damping ratio $\xi_i = |\beta_i|/\omega_i$. The lower the value of the ratio is, the less damped is the response component corresponding to the particular mode of the system dynamics. From the stability point of view, the stable poles are located to the left from the stability boundary, imaginary axis, i.e., the real parts of the poles are negative. If one or more poles are located in the right half of the complex plane, i.e., the real parts of the poles are positive, the system is unstable.

The system zeros $\mu_{kl,i}$, $i = 0,1,..$ are the solutions of the equations

$$N_{kl}(s) = \mathbf{C}_k \,\mathrm{adj}[s\mathbf{I} - \mathbf{A}]\mathbf{B}_l = 0,\ k = 1..p,\ l = 1..m \tag{1.7}$$

where $\mathbf{C}_k$ is the row sub-matrix of **C** corresponding to the k^{th} output and $\mathbf{B}_l$ is the column sub-matrix (vector) of **B** corresponding to the l^{th} input. Unlike the system poles, which are common for all transfer functions (1.4), each of transfer functions (1.4) has its own set of zeros. The number of system zeros corresponding to $G_{kl}(s)$ is given by the degree of polynomial $N_{kl}(s)$. Although the location of the system poles determines the basic modes of the system dynamics, it is the location of the system zeros which determines the proportion in which these modes are combined. Analogously to the system poles, it is also possible to define fast and slow zeros. The fast and slow zeros are defined with respect to the dominant system poles. Fast zeros are much farther away from the stability boundary than the dominant poles and the slow zeros are those which are closer to the boundary than the dominant poles. The fast zeros influence the system responses only slightly. On the other hand, the slow zeros have a distinct effect on the system responses. In case of the step response, the slow zeros cause its overshoot as a rule. The sign of the zeros is also important, however, not from the stability point of view. The system zeros with the positive real parts are called non-minimum phase zeros. These zeros are responsible for the undershoot of the step response of the system. The influence of the system zeros to the system response has been studied, e.g., in Goodwin, et al, (2001).

Finally, let us explain the concept of system modes and their roles in the system response. In case of single poles, transfer functions (1.4) can be expanded using the Heaviside expansion theorem

$$G_{kl}(s) = \sum_{i=1}^{n} \mathrm{Res}_{s=\lambda_i}\{G_{kl}(s)\}\frac{1}{s-\lambda_i} = \sum_{i=1}^{n} \frac{N_{kl}(\lambda_i)}{M'(\lambda_i)}\frac{1}{s-\lambda_i} \tag{1.8}$$

where $M'(s) = dM(s)/ds$. The expressions

$$R_{kl}(\lambda_i) = \frac{N_{kl}(\lambda_i)}{M'(\lambda_i)} \tag{1.9}$$

are called residues. The residues are the weighting factors of the modes $\exp(\lambda_i t)$ in the system responses. In case of multiple poles the residue is given by more complicated formulas that involves higher order derivatives of $M(s)$, see Angot, (1952). The transfer function $G_{kl}(s)$ is the Laplace transform of the weighting functions $g_{kl}(t)$, which are the response to the Dirac's delta function in case of zero initial conditions. Hence, if the impulse (Dirac's delta) is applied to the input u_l of the system with zero initial conditions, the response of the output y_k is equal to the weighting function

$$g_{kl}(t) = \sum_{i=1}^{n} R_{kl}(\lambda_i)\exp(\lambda_i) \tag{1.10}$$

Obviously, the values of the residues define the roles which are played by the corresponding modes $\exp(\lambda_i t)$ in the particular input-output relation.

1.1.4 Computing the system poles and zeros as the polynomial roots

The problem, which should also be mentioned in connection with the concept of poles and zeros of system (1.1), is the way of their computing. If the system is of the second order, the characteristic equation has the form of the quadratic equation the solution of which is well known[1]. There are available analytical based algorithms for solving the task for third and fourth order polynomials, but the formulas are much more complicated. It has been proved that there are no such analytical formulas for polynomials of order higher than fourth. Therefore, a numerical method has to be used to compute the roots of these higher order polynomials. There exist many algorithms that can be used to solve this problem (from the recently appeared ones let us mention MP-solve (Bini and Fiorentino, (1999, 2000) and Eigensolve (Fortune, 2001)), but none of them can be said to be complete from both the theoretical and the practical point of view. Computing polynomial roots becomes more difficult with increasing order of the polynomial. Although there are many open problems that should be solved in this research field, nowadays it is possible to compute the roots of polynomials of considerably high orders by means of available algorithms. In case of the high order polynomial, special numerical iteration methods based on high precision computing has to be used. Some of the practical aspects of solving polynomial equations are discussed in Peters and Wilkinson, (1971) or Wilkinson (1963). The overview of the huge literature dealing with the problem of computing the roots of polynomials can be found in Pan, (1997).

Alternative way of computing the system poles is based on the fact that the concept of system poles is equivalent to the concept of matrix **A** eigenvalues. Especially for higher order systems (1.1) (from the numerical point of view), it is not convenient to compute the determinant of the system characteristic matrix, and then to find the roots of the obtained polynomial. Most of the algorithms for evaluating the determinant do not use the determinant definition formula (Rektorys, 1994), which consists of $n!$ summations of products of n matrix coefficients. Such a determinant evaluation is rather computationally demanding already for $n>3$. The more favourable approach consists in the transformation of the matrix into the diagonal matrix (for example using the Gaussian elimination). Consequently, the determinant is given by the product of the diagonal elements of the diagonal matrix. Although not so many

[1] Already Babylonians and Egyptians (about 2000 B.C.) used the familiar formula for solving the quadratic equation (Rhind or Ahmes papyrus)

arithmetic operations have to be done as in the case of using the determinant evaluation formula, their number is high anyway. Accumulation of the truncation errors throughout this computation can result in non-negligible errors in resultant spectrum of the roots. Moreover, the higher order polynomials are likely to be ill-conditioned, i.e., even very small changes in their coefficients can result in obtaining a considerably changed spectrum of the roots, see (Wilkinson, 1984). To avoid this failure, a method for computing matrix eigenvalues of **A** is to be used, e.g., LR, QR, methods for smaller rather dense matrices or an iterative method for possibly very large and sparse matrices, see (Wilkinson, 1965), Demmel, J., (1997) and the practical guide (Bai, 2000). From the available software packages for computing matrix eigenvalues, let us mention LAPACK (Anderson, et. al., 1999), whose subroutines are used in the commonly used software tool Matlab. It should be noted that many algorithms for computing the roots of polynomials are based on computing the eigenvalus of the polynomial companyon matrices (e.g., the function *roots* in Matlab).

1.2 Models of linear TDS, motivation, development overview and introductory remarks

Classical system description (1.1) consists of a set of state and output equations. The only dynamical components of the description are the integrators on the left-hand side of the state equations. There are not any dynamical relations on the right hand side of the state equations. This fact is rather restrictive in formulating the model of a plant. All the system dynamics, which are time-distributed as a rule, has to be modelled by point accumulations in the state equations. This approach often results in a model in which some or all the state variables are artificial without a physical meaning. On the other hand, description (1.1) is convenient for building the comprehensive mathematical or control theories. There has been published a great deal of papers and books in the field of control engineering, in which model (1.1) is used as the fundamental model for describing real plants. The question is whether model (1.1) is so widely used because of its good features in describing the plants or whether it is because the model is easy to deal with. Taking into account the rather restricted potentials of model (1.1) in describing real plant dynamics. However the model is often chosen for the plant description because it is easy to handle. From the engineering point of view, the aim should not be to obtain nice results, but to achieve a progress in practical control applications. Therefore a more complex model should be used in practice.

The alternative linear system description that involves besides the integrals also other dynamical elements, time delays, shows much better potentials in describing real plants. Using the lumped and distributed delays provide the possibility to choose the state variables as the available (measured) system outputs, see, e.g., Zítek, (1998). Using the delays in modelling allows us to separate the plant dynamics into the parts and each part to describe by a few (preferably one) first order functional state equations. On the other hand the theory of the systems with time delays is more complicated.

First, let us briefly outline the historical development of the theory of TDS. The fundamental theories and the mathematical formulations were developed in twentieth century starting with the research of Voltera, who formulated differential equations with the past states of the system. Voltera, used this new concept of modelling to describe the predator-prey and viscoelaticity phenomena. In the study of ship stabilization, Minorski (1942) pointed out the importance of the consideration of the delay in the feedback mechanism. The basic theory of the stability of delay differential equations was developed by Pontryagin (1942). The increased interest in control theory during forties and fifties of the last century, contributed significantly to the rapid development of the theory of differential equations with time delays. Consequently, several books appeared providing the comprehensive introduction to the theory. In his book, Myshkis (1955, in German, translation of the 1951 Russian edition),

(1972), introduced a general class of differential equations with delayed arguments. The main ideas of the theory had already been published by Mishkis in (1949) (in Russian, translation to English (1951)). This publication can be considered as the pilot-work in the field of time delay systems providing the compact basis of the theory. From the further books, the monographs of Bellman and Dankis (1954) and Bellman and Cooke (1963) can not be omitted. The first monograph deals with the theory of linear equations, the stability theory and the application areas are extended to biology and economy. The second monograph belongs to the class of monographs that can be considered as the fundamental source of the knowledge in the field of time delay systems. In this book, the more extensive development of the theory, based on the frequency domain approach, can be found. The problem of stability was further developed and presented by Krasovski (1959) (in Russian, translation to English (1963)). In his monograph the extension of the theory of Liapunov functionals to functional differential equations was presented. To finish this overview, let us only mention the other important monographs in the field of time delay systems written by Pinney (1958), Halanay (1966) (qualitative apects), El'sgol'c and Norkin (1971) (time-varying delays, in Russian, translation to English 1973), Yanushevski (1978) (control, in Russian), Hale (1977) (general theory, neutral systems), Kolmanovskii and Nosov (1986) (stability, application examples), Górecki, et al, (1989) (analysis and synthesis) and Diekman, et al, (1995) (operator theory approach). The recent comprehensive introductions are (Kolmanovskii and Myshkis, 1992) and Hale and Verduyn Lunel (1993).

Time delay systems belong to the class of infinite dimensional systems (Bensoussan, et al, 1993). The modelling approach using time delays is largely used to describe propagation and transport phenomena, which can be met in the applications throughout the fields of mechanical, chemical and electrical engineering. Other typical areas of the application of time delay systems are populations dynamics (Kuang, 1993) and the economics. From the mathematics point of view, there are several approaches to describe such systems, e. g., by differential equations on abstract and functional spaces, or over rings of operators. As regards the differential equations on abstract linear space of infinite dimension (Bensousan, et al, 1993), (Curtain and Zwart, 1995), it is rather general mathematical approach that makes possible to treat the systems described by functional and partial differential equations in the same way, generally called distributed parameter systems. The discussions on such approaches and the system classification can be found in (Curtain, et al., 1989, 1993) and in Delfour (1981). Although the approach is very general, the methods are not always easy to be applied to a specific practical problem. In the second possible approach the functional differential equations are used. The system can be considered either as evolutions in a finite-dimensional space (vector space interpretation), (Kolmanovskii and Nosov, 1986) or in a functional space (Hale and Verduyn Lunel, 1993). In both cases, the initial conditions are always defined by functions. In this approach, the analogy of the functional description, with the finite vector space, with the classical description (finite dimensional) is utilised. Thus, the advantage of this approach is that some tools developed for finite-dimensional systems can be used to analyse the functional systems. However, the specific features of the TDS, i. e., functionality, infinite spectrum, have to be taken into consideration. The idea of using the finiteness of the space vector is also utilised in the third mentioned approach, where the functional differential equations are defined over a ring of operators (Kamen, 1978a) and the theory of systems over rings is used (Kamen, 1978b). The approach is advantageous in case of commensurate delays where the system is considered over the polynomial ring involving the delays. Another possibility is to represent the systems as differential equations over a ring of distributions using convolution operators (Kamen, 1975, 1978a), which allows the initial conditions to be involved in the model. The more comprehensive review of the modelling approaches can be found in Niculescu, (2001). In conclusion, regarding the questions of

generality, conventionality and the applicability for describing real plants, the approach using the functional differential equations seems to be the most suitable one. Therefore, this representation of the system and particularly the approaches presented in the monoghraphs Górecki, et al., (1989) and Hale and Verduyn Lunel, (1993) will be used in this thesis.

1.2.1 Description of TDS, neutral and retarded functional differential equations, the anisochronic approach

A general description of a linear time delay system can be considered in the following functional form, (Górecki, et al., 1989)

$$\begin{aligned}\frac{d\mathbf{x}(t)}{dt} &= \mathbf{A}_0\mathbf{x}(t) + \mathbf{B}_0\mathbf{u}(t) + \sum_{i=1}^{N}\left[\mathbf{H}_i\frac{d\mathbf{x}(t-\eta_i)}{dt} + \mathbf{A}_i\mathbf{x}(t-\eta_i) + \mathbf{B}_i\mathbf{u}(t-\eta_i)\right] + \\ &+ \int_0^T \mathbf{A}(\tau)\mathbf{x}(t-\tau)d\tau + \int_0^T \mathbf{B}(\tau)\mathbf{u}(t-\tau)d\tau \qquad (1.11)\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t)\end{aligned}$$

where $\mathbf{x}\in\mathrm{R}^n$, $\mathbf{u}\in\mathrm{R}^m$, $\mathbf{y}\in\mathrm{R}^p$ and $\mathbf{A}_i$, $\mathbf{A}(\tau)$, $\mathbf{B}_i$, $\mathbf{B}(\tau)$, $\mathbf{H}_i$, $\mathbf{C}$ are matrices of compatible dimensions, $\eta_1 < \eta_2 < \ldots < \eta_N < T$ are the values of the lumped delays and the integrals in (1.11) describe the distributed delays involved in the model. Actually, in Górecki, et. al., (1989) and also in, e.g., Hale and Verduyn Lunel, (1993), the integrals in (1.11) have the form $\int_{-T}^{0}\mathbf{A}(\tau)\mathbf{x}(t+\tau)d\tau$ and $\int_{-T}^{0}\mathbf{B}(\tau)\mathbf{x}(t+\tau)d\tau$. Thus the delay variable τ has the opposite sign than in (1.11) and the functional matrices $\mathbf{A}(\tau)$ and $\mathbf{B}(\tau)$ are defined on the interval [-T, 0] instead of [0, T] as in (1.11). The advantages of the functional matrices defined on the interval [0, T] can be found in Diekman, et al, (1995), section 1.2. The approach has also been used, e.g., in Zítek, (1986). Since model (1.11) involves the system history, the initial conditions can not be only given by the values of the state variables at time t=0, but the initial conditions are predetermined by the functions $\boldsymbol{\varphi}$, $\boldsymbol{\kappa}$ and $\boldsymbol{\zeta}$ (of the particular dimensions, as will be shown further, the state space is Banach space)

$$\left.\begin{aligned}\mathbf{x}(t) &= \boldsymbol{\varphi}(t)\\ \frac{d\mathbf{x}(t)}{dt} &= \boldsymbol{\kappa}(t)\\ \mathbf{u}(t) &= \boldsymbol{\zeta}(t)\end{aligned}\right\} \quad t\in[-T,0] \qquad (1.12)$$

On its right hand side, description (1.11) does not only have the miscellaneously shifted state and input variables, but also shifted derivatives of the state variables. Such a time delay system description is called neutral, and the systems described by (1.11) are referred to as the neutral systems, (Kolmanovskii and Myshkis, 1992), (Kolmanovskii and Nosov, 1986), (Hale and Verduyn Lunel, 1993). Neutral functional differential equations (NFDE) are largely used for describing lossless propagation phenomena, see Niculescu, (2001), which is encountered, e.g., in modelling of distributed networks. In this application of NFDE, the connection of NFDE with partial differential equations of the hyperbolic type (which is the classical tool for describing the propagation phenomena) is used.

Neutral system (1.11) becomes a retarded one if $\mathbf{H}_i$=$\mathbf{0}$, i = 1, 2, ... ,N

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}_0\mathbf{x}(t) + \mathbf{B}_0\mathbf{u}(t) + \sum_{i=1}^{N}\left[\mathbf{A}_i\mathbf{x}(t-\eta_i) + \mathbf{B}_i\mathbf{u}(t-\eta_i\right] + \\ + \int_0^T \mathbf{A}(\tau)\mathbf{x}(t-\tau)d\tau + \int_0^T \mathbf{B}(\tau)\mathbf{u}(t-\tau)d\tau \qquad (1.13)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

The initial conditions of system (1.13) are analogous to (1.12). Describing plants by retarded functional differential equations (RFDE) can be encountered in more applications in modelling and control engineering than the describing the plants by NFDE. The typical areas where the RFDE are applied are heat transfer, chemical and combustion processes, and the physical processes where the transportation phenomena and distributed parameters are encountered. The theory of RFDE is more developed than the theory of NFDE and its results can be found already in publications (Krasovskii, 1963), (Halanay, 1966) and in more recent ones (Kolmanovskii and Myshkis, 1992), (Hale and Verduyn Lunel, 1993).

Using Rieman-Stieltjes integrals (Kolmogorov and Fomin, 1999), the notations of equations (1.11) can be written in the form

$$\frac{d\mathbf{x}(t)}{dt} = \sum_{i=1}^{N}\left[\mathbf{H}_i\frac{d\mathbf{x}(t-\eta_i)}{dt}\right] + \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) + \int_0^T d\mathbf{B}(\tau)\mathbf{u}(t-\tau) \qquad (1.14)$$

and analogously (1.13) can be written in the form

$$\frac{d\mathbf{x}(t)}{dt} = \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) + \int_0^T d\mathbf{B}(\tau)\mathbf{u}(t-\tau) \qquad (1.15)$$

respectively. The use of Rieman-Stieltjes integrals allows both the lumped and the distributed delays to be involved in one convolution. It is possible to do so because of the following feature of the integral. Let us consider the function $f(\tau)$ which is continuous on the interval $[a, b]$, Rieman-Stieltjes integral results in

$$\int_a^b f(\tau)dg(\tau) = \sum_i f(\tau_i)l_i \qquad (1.16)$$

if function $g(\tau)$ is step function with the step sizes l_i in τ_i, and in

$$\int_a^b f(\tau)dg(\tau) = \int_a^b f(\tau)g'(\tau)d\tau \qquad (1.17)$$

if function $g(\tau)$ is continuous. In general, the continuity of $f(\tau)$ is necessary condition for using the Rieman-Stieltjes integral, but the functions $g(\tau)$ and $g'(\tau)$ can be only piecewise continuous with a finite number of finite step discontinuities of $g(\tau)$ on $[a, b]$. The definition and other features of Rieman-Stieltjes integral can be found in, e.g., Kolmogorov and Fomin (1999). Thus regarding the previous definition, the functional matrices in (1.14) and (1.15) can be defined as

$$\mathbf{A}(\tau)=\begin{bmatrix} a_{11}d_{11}(\tau)\,.....\,a_{1n}d_{1n}(\tau) \\ - \quad - \quad - \\ a_{n1}d_{n1}(\tau)\,.....\,a_{nn}d_{nn}(\tau) \end{bmatrix},\ \mathbf{B}(\tau)=\begin{bmatrix} b_{11}d_{u_{11}}(\tau)\,.....\,b_{1m}d_{u_{1m}}(\tau) \\ - \quad - \quad - \\ b_{n1}d_{u_{n1}}(\tau)\,.....\,b_{nm}d_{u_{nm}}(\tau) \end{bmatrix} \tag{1.18}$$

where $d_{ij}(\tau)$ and $d_{u_{ik}}(\tau)$ are the normalised monotonous functions of delay distributions ($d(\tau)=0, \forall\,\tau<0$ and $d(\tau)=1,\ \forall\,\tau>T$), piecewise continuous with a finite number of step discontinuities, see Zítek, (1986).

Rather unusual description of time delay systems introduced in the previous text acquires more convenient form after applying the Laplace transform. Considering the zero initial conditions, the L-transform of neutral system (1.14) and retarded system (1.15) are of the following forms

$$s\mathbf{x}(s)=\left[s\sum_{i=1}^{N}\left[\mathbf{H}_i\exp(-s\eta_i)\right]+\mathbf{A}(s)\right]\mathbf{x}(s)+\mathbf{B}(s)\mathbf{u}(s) \tag{1.19}$$

$$s\mathbf{x}(s)=\mathbf{A}(s)\mathbf{x}(s)+\mathbf{B}(s)\mathbf{u}(s) \tag{1.20}$$

where the transform functional matrices are given by the Laplace-Stieljes integrals

$$\mathbf{A}(s)=\int_0^T\exp(-s\tau)d\mathbf{A}(\tau),\ \mathbf{B}(s)=\int_0^T\exp(-s\tau)d\mathbf{B}(\tau) \tag{1.21}$$

The assumption of zero initial conditions implies restrictions with respect to the analytical solution. However, because of the functionality of the initial conditions, obtaining the analytic solution using the Laplace transform is rather tedious, see (Górecki, et al., 1989). The benefit of the transform is in the possibility to use the algebraic and matrix operations in the system analysis. The L-transform of retarded system (1.20) is formally identical with the transform of classical system (1.2). On one hand, this analogy can be utilised in the system dynamics analysis and control design. However, on the other hand, the functionality of the matrices brings about some indispensable drawbacks, particularly the infinite spectrum of spectra of poles and zeros, the difficult isolation of the important modes of the system dynamics and the causality of the designed control.

The novel modelling approach called anisochronic model formulation, which has been developed by Zítek (1983, 1986), is based on model with Stieljes integrals (1.15). The main idea of the approach consists in involving all the system delays in one convolution. The objective of the approach is to define true distribution of the system dynamics within the state equations of the model. Using the various distributions of delays, the number of integrations (lumped accumulations) in the model is considerably reduced. The aim of the approach is not only to obtain a low order model, but also to obtain the model in which some of the state variables are identical with the system outputs. This identity of the state variables and the system outputs is the essential feature of the anisochronic modelling approach. This feature is advantageous especially in applying the state feedback control, because no observer is needed for estimation of the unmeasured state variables. From the publications dealing with the control problem based on anisochronic model formulation, let us mention Zítek, (1997), Zítek, et al., (1995, 2001), Zítek and Vyhlídal, (1999, 2000), where the problem of state feedback control of time delay systems is worked out. The potentials and limits of anisochronic internal model control are studied in Zítek and Hlava (1998, 2001),

Hlava (1998). In Vyhlídal and Zítek, (2001), the properties of anisochronic first order models are analysed and an identification method of the model parameters is proposed. As has been shown in Zítek, (1998a), Zítek and Garagič, (1997), the observers based on the anisochronic model formulation are suitable to be applied in the fault detection and diagnosis. The separability of the parts of the anisochronic model, usually corresponding to the physical parts of the particular plant, provides the possibility to perform the decomposition of the observer into several local observers.

1.2.2 The notion of state of TDS

Unlike the state of classical system (1.1), which is given by the vector of state variables $\mathbf{x}(t)$, the state of a time delay system (1.11) is given by the function segments of the system state variables and the system inputs $\{\boldsymbol{x}_t, \boldsymbol{u}_t\}$ on the segment of the last system history, see, e.g., Górecki, et. al., (1989), Zítek, (1998) where

$$\boldsymbol{x}_t(\tau) = \mathbf{x}(t+\tau),\ \boldsymbol{u}_t(\tau) = \mathbf{u}(t+\tau) \quad -T \le \tau \le 0 \tag{1.22}$$

and the state space is the Banach space of continuous real functions on the interval of length T

$$\mathcal{C} = C([-T,\ 0], \mathbb{R}^n) \tag{1.23}$$

provided with the supremum norm, i.e., $\|\boldsymbol{\varphi}\| := \max_{-T \le \tau \le 0} |\boldsymbol{\varphi}(\tau)|$, see Hale and Verduyn Lunel, (1993). In many publications dealing with the TDS, the delay in the inputs are not considered or features of autonomous systems are investigated (the system inputs are not involved in the model). The state of such system is obviously given by $\boldsymbol{x}_t$. This notion of state has already been used by Krasovskii, (1963), and the state theory of the class of linear functional differential equations of the retarded type has been further developed by Delfour (1977). From the more recent publications dealing with the concept of state and the state theory of TDS, let us mention (Hale and Verduyn Lunel, 1993) and (Diekman, et. al., 1995). The state given by a function element of the last history coheres with the definition of initial conditions, which are also defined as elements of functions $\boldsymbol{\varphi}$.

1.2.3 Linear autonomous TDS, the shift semigroup

Investigating the class of autonomous systems provides the opportunity to analyse the homogeneous system dynamics, i.e., the dynamics of the systems with inputs. In Hale and Verduyn Lunel, (1993) and Diekman, et. al., (1995) the description of the autonomous systems and their analysis on the basis of the solution operators is investigated. The autonomous system corresponding to RFDE (1.15) is of the form

$$\begin{aligned} &\frac{d\mathbf{x}(t)}{dt} = \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) \\ &\mathbf{x}(t) = \boldsymbol{\varphi}(t) \quad t \in [-T,\ 0] \end{aligned} \tag{1.24}$$

Following the mentioned literature sources, let us introduce an alternative system description

$$\begin{aligned} &\frac{d\mathbf{x}(t)}{dt} = L\boldsymbol{x}_t \qquad t > 0 \\ &\boldsymbol{x}_0 = \boldsymbol{\varphi} \end{aligned} \tag{1.25}$$

where L is a continuous linear mapping from $\mathcal{C}$ to $\mathbb{R}^n$, satisfying

$$L\phi = \int_0^T d\zeta(\theta)\phi(-\theta), \qquad \phi \in \mathcal{C} \tag{1.26}$$

where function ζ is defined on [0, T] with values in $\mathbb{R}^{n\times n}$, whose elements are of bounded variation, normalised so that ζ is continuous from the right on [0, T] and $\zeta(0)=0$.

Remark 1. The bounded variation is defined as follows: Let us consider the function $f(\tau)$ that is said to have bounded variation over the closed interval $\tau \in [a, b]$, if there exists M such that $|f(\tau_1)-a|+|f(\tau_2)-f(\tau_1)|+...+|f(\tau_b)-f(\tau_{n-1})| \le M$ for all $a<\tau_1<\tau_2<...<\tau_{n-1}<b$.

Remark 2. Let function f be of bounded variation with countable number of discontinuities. Function f is continuous from the right on the open interval (a, b) if $f(\tau)=f(\tau+)$, where $f(\tau+)=\lim_{\sigma\downarrow\tau} f(\sigma)$, at every point $\tau \in (a, b)$.

Regarding the features of distribution functions $d_{ij}(\tau)$ of $\mathbf{A}(\tau)$ in (1.18), the functions satisfy the previous definitions, therefore, it is possible to rewrite (1.26) into

$$L\phi = \int_0^T d\mathbf{A}(\theta)\phi(-\theta), \qquad \phi \in \mathcal{C} \tag{1.27}$$

Let us also introduce the concept of the solution operator $\mathcal{T}^\circ(t)$. There exists a family $\mathcal{T}^\circ = \{\mathcal{T}^\circ(t)\}$ of bounded linear operators on a Banach space such that

$$\begin{aligned} &\text{(i)} \quad \mathcal{T}^\circ(0) = \mathbf{I} \quad \text{(the identity matrix)} \\ &\text{(ii)} \quad \mathcal{T}^\circ(t)\mathcal{T}^\circ(p) = \mathcal{T}^\circ(t+p) \quad \text{for } t, p \ge 0 \\ &\text{(iii)} \quad \lim_{t\to 0}\left\|\mathcal{T}^\circ(t)\phi - \phi\right\| = 0, \quad \phi \in \mathcal{C} \end{aligned} \tag{1.28}$$

This family $\mathcal{T}^\circ$ is called the strongly continuous semigroup, in short, a $\mathcal{C}_0$-semigroup, see Pazy, (1983), which is given by the translation along the solutions of (1.24). Consequently, the time evolution of the state $\boldsymbol{x}_t$ is given by an abstract ordinary differential equation in the infinite dimensional state space $\mathcal{C}$. The abstract differential equation can be expressed in the form

$$\frac{d}{dt}(\mathcal{T}^\circ(t)\phi) = \mathcal{A}(\mathcal{T}^\circ(t)\phi) \tag{1.29}$$

where

$$\mathcal{A}\phi = \lim_{t\to 0}\frac{1}{t}\left(\mathcal{T}^\circ(t)\phi - \phi\right) \tag{1.30}$$

with

$$\mathcal{D}(\mathcal{A}) = \left\{\phi, | \frac{1}{t}\lim_{t\to 0}\left(\mathcal{T}^\circ(t)\phi - \phi\right) \text{ exists}\right\} \tag{1.31}$$

i.e., $\mathcal{D}(\mathcal{A})$ consists of all $\phi \in \mathcal{C}$ for which the limit in (1.31) exists. The linear operator $\mathcal{A}$: $\mathcal{D}(\mathcal{A}) \to \mathcal{C}$, which is the derivation of $\mathcal{T}^\circ(t)$ at t=0 and is, in general, unbounded, is called infinitesimal generator of the semigroup. For system (1.24) the generator is defined as follows

$$\mathcal{D}(\mathcal{A}) = \left\{ \boldsymbol{\phi} \,|\, \boldsymbol{\phi}' \in C([-T,\ 0], \mathrm{R}^n), \quad \boldsymbol{\phi}'(0) = \int_0^T d\mathbf{A}(\theta)\boldsymbol{\phi}(-\theta) \right\}; \quad \mathcal{A}\boldsymbol{\phi} = \boldsymbol{\phi}' \tag{1.32}$$

The state $\boldsymbol{x}_t$ of the autonomous system at time t is uniquely determined by the initial condition function $\boldsymbol{\varphi}$, i.e. state at $t = 0$, by the solution operator, thus

$$\mathcal{T}(t)\boldsymbol{\varphi} = \boldsymbol{x}_t, \quad \boldsymbol{\varphi} \in \mathcal{C} \tag{1.33}$$

i.e., operator $\mathcal{T}(t)$, maps the initial state $\boldsymbol{\varphi}$ at time zero to $\boldsymbol{x}_t$. According to definition (1.32), description (1.24) can be rewritten into

$$\begin{aligned} &\frac{d}{dt}\boldsymbol{x}_t = \mathcal{A}\boldsymbol{x}_t, \ t > 0 \\ &\boldsymbol{x}_0 = \boldsymbol{\varphi} \end{aligned} \tag{1.34}$$

As will be shown further, discretizing either the solution operator or the infinitesimal generator $\mathcal{A}$ is one possible way of computing the approximate values of the poles of retarded systems. More detailed definitions, further information and applications based on the semigroup theory can be seen, besides the literature which have been already mentioned, in Curtain and Pritchard, (1978), Verduyn Lunel, (1995, 2001a and 2001b), Hale and Verduyn Lunel, (2001), or Mastinšek, (1994).

The operational description analogous to (1.25) can also be used for description of the neutral systems, see (Hale and Verduyn Lunel, 1993)

$$\begin{aligned} &\frac{d}{dt}\mathcal{D}\boldsymbol{x}_t = \mathcal{L}\boldsymbol{x}_t \qquad t > 0 \\ &\boldsymbol{x}_0 = \boldsymbol{\varphi} \end{aligned} \tag{1.35}$$

where $\mathcal{L}$ and $\mathcal{D}$ are bounded linear mappings from $\mathcal{C}$ to R^n, where $\mathcal{L}$ is defined by (1.26) and

$$\mathcal{D}\boldsymbol{\phi} = \boldsymbol{\phi}(0) - \int_0^T d\upsilon(\theta)\boldsymbol{\phi}(-\theta), \qquad \boldsymbol{\phi} \in \mathcal{C} \tag{1.36}$$

see Hale and Verduyn Lunel, (1993). The difference equation associated with (1.35) and (1.36) is given by $\mathcal{D}\boldsymbol{x}_t = 0$, i.e.

$$\mathbf{x}(t) = \int_0^T d\upsilon(\theta)\mathbf{x}(t-\theta) \tag{1.37}$$

For any $\boldsymbol{\varphi} \in \mathcal{C}$ for which $\mathcal{D}\boldsymbol{\varphi} = 0$, there exists a unique solution $\mathbf{x}(t)$ of (1.37) which is continuous on $[-T, 0]$ and satisfies $\boldsymbol{x}_0 = \boldsymbol{\varphi}$. For the NFDE in the form (1.11), (1.14), equation (1.37) acquires the following form

$$\mathbf{x}(t) = \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i), \qquad t \geq 0 \tag{1.38}$$

see also Hale and Verduyn Lunel, (2002). The state of the NFDE has the same notion as the state of RFDE and the solution operator is again given as the solution map $T(t)$: $\mathcal{C} \to \mathcal{C}$ with the infinitesimal generator defined as

$$\mathcal{D}(\mathcal{A}) = \left\{ \boldsymbol{\phi} \,|\, \boldsymbol{\phi}' \in C([-T,\ 0], \mathrm{R}^n),\ \ \mathcal{D}\boldsymbol{\phi}' = \mathcal{L}\boldsymbol{\phi} \right\};\ \ \mathcal{A}\boldsymbol{\phi} = \boldsymbol{\phi}' \tag{1.39}$$

1.2.4 Poles and zeros of TDS

The definition of poles and zeros of system with time delays is analogous to their definition for classical system (1.1). Considering general TDS of form (1.14), the input output relation arising from the state description is of the form of transfer matrix

$$\mathbf{G}(s) = \frac{\mathbf{y}(s)}{\mathbf{u}(s)} = \mathbf{C}\left[s\left(\mathbf{I} - \sum_{i=1}^{N} \left[\mathbf{H}_i \exp(-s\eta_i)\right]\right) - \mathbf{A}(s) \right]^{-1} \mathbf{B}(s) \tag{1.40}$$

The poles λ_i and the zeros μ_i , $i = 1..\infty$ of retarded system (1.15), ($\mathbf{H}_i = \mathbf{0}$, $i = 1..N$) are the solutions of the following equations

$$M(s) = \det(s\mathbf{I} - \mathbf{A}(s)) = 0 \tag{1.41}$$

for the system poles and

$$N_{kl}(s) = \mathbf{C}_k \operatorname{adj}\left[s\mathbf{I} - \mathbf{A}(s)\right]\mathbf{B}_l(s),\ k = 1..p,\ l = 1..m \tag{1.42}$$

for the zeros of transfer function $G_{k,l}$, where $\mathbf{C}_k$ is the row sub-matrix of $\mathbf{C}$ corresponding to the k^{th} output and $\mathbf{B}_l(s)$ is the column sub-matrix (vector) corresponding to the l^{th} input. In case of neutral system (1.14), the equations acquire the form

$$M(s) = \det(s(\mathbf{I} - \sum_{i=1}^{N} \left[\mathbf{H}_i \exp(-s\eta_i)\right]) - \mathbf{A}(s)) = 0 \tag{1.43}$$

$$N_{kl}(s) = \mathbf{C}_k \operatorname{adj}\left[s(\mathbf{I} - \sum_{i=1}^{N} \left[\mathbf{H}_i \exp(-s\eta_i)\right]) - \mathbf{A}(s) \right]\mathbf{B}_l(s) = 0 \tag{1.44}$$

Also in case of TDS, the location of the system poles determines the stability and the modes of the system dynamics. Therefore, let us turn the attention to the features of characteristic equation and its solutions, the system poles.

The characteristic equations of both retarded and neutral systems are transcendental. Therefore, in contrast to the characteristic equation of classical system (1.5), (1.41) and (1.43) have infinitely many solutions. The transcendental character of the equations arises from functionality of the terms $Q_i(s)$, i=1..n, see (1.45), of the powers of s. Thus, the characteristic function is not polynomial, but it is of the quasipolynomial form

$$M(s) = \sum_{i=0}^{n} s^i Q_i(s) \tag{1.45}$$

If the system is retarded, Q_n is a constant. The following stability condition is identical with the stability condition of classical system (1.1). A system with time delays is stable if and

only if all the solutions of its characteristic equation are located on the left half of the complex plane, see, e.g., Hale and Verduyn Lunel, (1993), Zítek, (1986) i.e.

$$\{\lambda_i \in \mathbb{C}: \ \mathrm{Re}(\lambda_i) \geq 0, M(\lambda_i) = 0\} = 0 \tag{1.46}$$

For the retarded systems the following theorem holds: Let all the poles of system (1.15) be ordered in a sequence λ_1, λ_2, ... , λ_k with respect to their magnitudes, $|\lambda_k| \to \infty$ as $k \to \infty$. The real parts of the poles are uniformly bounded from above and for any real a, the number of poles whose real parts are greater than a is at most finite. All the poles have finite multiplicities. For proofs see Hale (1977) and Myshkis (1972). The feature of the finite number of the poles whose real parts are grater than an arbitrarily chosen a is very important because it allows us to define the set of system poles which are decisive in the system dynamics with respect to the stability of the system and its basic features. There also exist other rules for the pole distribution, but they are mostly valid only for a class of retarded systems, see Bellman and Cooke, (1963), Kolmanovskii and Myshkis, (1992) and also Levin, (1964). One of the features that is also valid for the retarded systems of form (1.15) and (1.24) is the following, see Stépán, (1989). Consider the sequence of λ_k defined above, $\mathrm{Re}(\lambda_k) \to -\infty$, as $k \to \infty$. The poles of a retarded system are mostly distributed as a finite number of (asymptotic) chains of the poles. The origins of these chains are the right most poles, which are close to the real. All the chains depart asymptotically from their origins, which are close to the real axis, to the left with increasing values of $|\lambda_k|$. As will be shown further, the finite number of right most poles is the crucial feature of retarded systems with respect to a control system design based on pole assignment. Also the significance of the right most poles will be taken into consideration and a criterion evaluating their significance will be introduced.

Let us turn the attention to the poles of neutral systems. The poles of the neutral systems display very different properties than the poles of the retarded systems, see (Hale, 1977). Let us demonstrate the difference on the following simplified description of a neutral system

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{H}_1 \frac{d\mathbf{x}(t-\tau)}{dt} + \mathbf{A}_0\mathbf{x}(t) + \mathbf{A}_1\mathbf{x}(t-\tau) \tag{1.47}$$

As can be seen in Stépán, (1989), if all the eigenvalues of $\mathbf{H}_1$ are not zero, the system poles are distributed between two horizontal boundaries, i.e., $\alpha < \mathrm{Re}(\lambda_k) < \beta$, $k = 1,2,....,\infty$. Another very important feature of this class of neutral systems is bound up with the solutions υ_k of the equation

$$D(s) = \det[\mathbf{I} - \mathbf{H}_1 \exp(-s\tau)] = 0 \tag{1.48}$$

If there exists a sequence of poles λ_k of system (1.47) such that $|\lambda_k| \to \infty$ as $k \to \infty$, there also exists a sequence υ_k of the solutions of (1.48) such that $(\lambda_k - \upsilon_k) \to 0$ as $k \to \infty$. Thus the necessary condition for the stability of neutral system (1.47) is the stability of the part corresponding to (1.48). The mentioned features of the distribution of the poles are the typical features of the neutral systems. In general, investigating the spectrum of (1.48) provides the crucial information about the stability and stabilizability of the neutral system. For more general neutral system (1.14), the equation equivalent to (1.48) is of the form

$$D(s) = \det\left[\mathbf{I} - \sum_{k=1}^{N} \mathbf{H}_k \exp(-s\eta_k)\right] = 0 \tag{1.49}$$

Analysis of spectrum of (1.49), its features and its impact to the spectrum of neutral system and thereby to the system stability has been studied, e.g., by Henry (1974), Avelar and Hale (1980) and Michiels, et al, (2001). Most of the results in the mentioned literature have been investigated for the scalar case for which

$$D(s) = 1 - \sum_{k=1}^{N} a_k \exp(-s\eta_k) = 0 \tag{1.50}$$

and then generalised for (1.49). $D(s)$ in (1.50) is called exponential polynomial and its spectrum, as well as the spectrum of (1.49), is called essential spectrum of the neutral system. It has been proved (Avelar and Hale, 1980) that the essential spectrum may be very sensitive even to infinitesimal changes in delays. The arbitrary small changes in the delays may destabilise the system. Let us denote the smallest upper bound of the essential spectrum as

$$c = \sup\{\mathrm{Re}(s) : D(s) = 0\} \tag{1.51}$$

The upper bound generally does not only depend on the values of the delays, but it also depends on their relationship. Moreover the supremum of the spectrum can change discontinuously with respect to the delays whereas the individual delays change continuously, see, e.g., Avelar and Hale (1980). In order to deal with the features of the essential spectrum, the concept of rationally dependent and independent delays has been introduced. The limit case of rationally dependent delays are the commensurate delays, for which equation (1.50) can be rewritten as a polynomial in exp(-sr) (r is the common divisor of the delays). For commensurate delays, the spectrum consists of finite number of solutions that are periodic with respect to $2\pi i/r$, i=1, 2,.. . On the other hand if all the delays are rationally independent, c satisfies the condition

$$1 - \sum_{k=1}^{N} |a_k| \exp(-c\eta_k) = 0 \tag{1.52}$$

for proof see, e.g., Michiels, et al., (2001). In Avelar and Hale (1980) the dependence of the spectrum on the mutual ratio of the delays is demonstrated on the following example. Consider the difference equation

$$x(t) = -0.5x(t - (\eta_1 + \varepsilon)) - 0.5x(t - \eta_2) \tag{1.53}$$

with the exponential polynomial

$$D(s) = 1 + 0.5\exp(-s(\eta_1 + \varepsilon)) + 0.5\exp(-s\eta_2) \tag{1.54}$$

where $\eta_1 = 1$ and $\eta_2 = 2$. If $\varepsilon = 0$, (1.54) is quadratic equation in exp(-s) and the upper bound of the spectrum is given by c = - ln(2)/2, which is identical with the real parts of all the eigenvalues of (1.54). If ε is considered irrational, i.e., the delays are rationally independent, the upper bound can be calculated by means of (1.52) from the equation

$$1 - 0.5\exp(-c(\varepsilon)(1 + \varepsilon)) - 0.5\exp(-2c(\varepsilon)) = 0 \tag{1.55}$$

For $\varepsilon \to 0$, the upper bound is given by $[\lim_{\varepsilon \to 0} c(\varepsilon) = 0] > [c(0) = -\ln(2)/2]$. Hence $c(\varepsilon)$ changes discontinuously with respect to ε and even infinitesimal delay change ε brings the system to the stability boundary. From the robustness point of view, it is necessary to consider the delays as rationally independent in order to obtain a reliable upper bound of the spectrum.

Even if the delays are identified in a real system as rationally dependent, there always exists certain uncertainty in their values that could result in the loss of stability. Therefore, it is necessary to introduce a new concept of stability for the neutral systems. In Hale, (1977), Hale and Verduyn Lunel (2002), the so-called strong stability is defined as follows. The difference equation is strongly stable if it is stable for all variations in the delays. In scalar case (1.50), the system is strongly stable if and only if

$$\sum_{k=1}^{N} |a_k| < 1 \tag{1.56}$$

see Avelar and Hale (1980). Thus difference equation (1.53) is not strongly stable for $\eta_1=1$ and $\eta_2=2$ although all the eigenvalues of (1.54) are located on the left half of the complex plane. The formula analogous to (1.56) exists also for the n-dimensional neutral systems with difference equation corresponding to (1.50). This formula, other features and the theoretical background for analysis of the neutral systems on the basis of the operational calculus will be shown and explained in the next section.

To sum up, the spectrum of poles of the neutral systems has very different features comparing to the spectrum of the retarded systems. The key feature of the neutral systems is that the number of their unstable poles does not have to be finite as it is in case of the retarded systems. In the spectrum of the neutral systems, the poles constitute the chains that do not asymptote to -∞ in the real parts of poles as a rule, but they are more or less parallel (at least a part of the spectrum of the poles) to the imaginary axis (located in a vertical strip of the complex plane). The important information about the stability of the neutral systems provides the analysis of the essential spectrum and especially its behaviour with respect to the changes in the delays. The importance of the essential spectrum is given by the fact that the spectrum of the neutral system asymptotes to the essential spectrum.

As regards the system zeros, their distribution is also important for the final input-output dynamics of TDS. The role of the zeros is identical as in the case of classical system (1.1). Their significance is also considered with respect to their positions towards the dominant poles. The number of system zeros depends on the structure of the system description. If there are no terms corresponding to the L-transform of delays in equation (1.42) and (1.44) the number of zeros is given by the highest power of s operator. Otherwise the spectrum is infinite and equations (1.42) and (1.44) have the analogous features as (1.41) and (1.43). The character of the spectrum depends on the term or coefficient corresponding to the maximum power of $N(s)$. If the maximum power of $N(s)$ is multiplied by a constant coefficient, the spectrum system zeros, i.e., the spectrum of roots of $N(s)$, has analogous features as the spectrum of poles of the retarded system. If the maximum power of $N(s)$ is multiplied by a term (given by Laplace transform of delays), the spectrum of system zeros has analogous features as the spectrum of poles of the neutral system.

1.2.5 Spectrum of infinitesimal generator and solution operator

In section 1.2.3, the concepts of the solution operator $\mathcal{T}^{\circ}(t)$ and the infinitesimal generator of the semigroup $\mathcal{A}$ have been introduced. Both these operators determine the dynamics of the autonomous system. Therefore, investigating their spectra provides important information about the system and about its asymptotic behaviour. First, let us consider retarded system (1.24). The spectrum of infinitesimal generator $\mathcal{A}$ consists of the eigenvalues only, i.e., complex numbers λ satisfying $\mathcal{A}\phi = \lambda\phi$ for some nonzero $\phi \in \mathcal{C}$. The spectrum of eigenvalues is given by

$$\sigma(\mathcal{A}) = \{\lambda \in \mathbb{C},\ \det \Delta(\lambda) = 0\} \tag{1.57}$$

where $\Delta(\lambda)$ is so-called characteristic matrix of (1.24)

$$\mathbf{\Delta}(\lambda) = \lambda \mathbf{I} - \int_0^T \exp(-\lambda\tau) d\mathbf{A}(\tau) \tag{1.58}$$

Obviously, the spectrum of the generator $\mathcal{A}$ is identical with the spectrum of system poles, compare (1.57) and (1.58) with equation (1.41). The spectrum of the solution operator is according to Henry (1974), see also Kaashoek and Verduyn Lunel, (1994),

$$\sigma(\mathcal{T}^{\circ}(t)) = \{\exp(\lambda t),\ \lambda \in \sigma(\mathcal{A})\} \text{ plus possibly } \{0\} \tag{1.59}$$

The asymptotic behaviour of the semigroup $\mathcal{T}^{\circ}(t)$ is determined by the spectral radius of $\mathcal{T}^{\circ}(t)$

$$r_\sigma(\mathcal{T}^{\circ}(1)) = \max\left\{|\gamma| \in \mathbb{C} : \gamma \in \sigma(\mathcal{T}^{\circ}(1))\right\} = \sup\{\exp(\mathrm{Re}(\lambda)),\ \lambda \in \sigma(\mathcal{A})\} \tag{1.60}$$

If $r_\sigma(\mathcal{T}^{\circ}(1))<1$, then the system is exponentially stable. If $r_\sigma(\mathcal{T}^{\circ}(1))>1$, there are exponentially unbounded orbits and $\mathcal{T}^{\circ}(t)$ is unstable. Obviously, the condition $r_\sigma(\mathcal{T}^{\circ}(1))<1$ is equivalent to the stability condition $\max\{\mathrm{Re}(\lambda),\ \lambda \in \sigma(\mathcal{A})\}<0$. As will be shown later, discretising either the semigroup $\mathcal{T}^{\circ}(t)$ or its generator $\mathcal{A}$ corresponding to a retarded system is one possible way of computing the approximation of the right-most system spectrum.

Let us turn the attention to the analysis of neutral system (1.35). The spectrum of infinitesimal generator $\mathcal{A}$ corresponding to the neutral autonomous system

$$\frac{d\mathbf{x}(t)}{dt} = \sum_{i=1}^{N}\left[\mathbf{H}_i \frac{d\mathbf{x}(t-\eta_i)}{dt}\right] + \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) \tag{1.61}$$

is also defined by (1.57) where

$$\mathbf{\Delta}(\lambda) = \lambda\left[\mathbf{I} - \sum_{i=1}^{N}\mathbf{H}_i \exp(-\lambda\eta_i)\right] - \int_0^T \exp(-\lambda\tau) d\mathbf{A}(\tau) \tag{1.62}$$

The spectrum of $\mathcal{T}^{\circ}(t)$ and the spectral radius of $\mathcal{T}^{\circ}(1)$ are defined in the same way as for the retarded system. Let us define the essential spectrum radius $r_e(\mathcal{T}^{\circ}(1))$ of $\mathcal{T}^{\circ}(t)$. The essential spectrum radius is defined as

$$r_e(\mathcal{T}^{\circ}(1)) = \max\left\{|\gamma| \in \mathbb{C} : \gamma \in \sigma_e(\mathcal{T}^{\circ}(1))\right\} = \sup\{\exp(\mathrm{Re}(\lambda)),\ \det \Delta_0(\lambda) = 0)\} \tag{1.63}$$

where

$$\mathbf{\Delta}_0(\lambda) = \mathbf{I} - \sum_{i=1}^{N}\mathbf{H}_i \exp(-\lambda\eta_i) \tag{1.64}$$

The solutions of (1.64), i.e., eigenvalues of the difference equation

$$\mathbf{x}(t) = \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i) \tag{1.65}$$

play the fundamental role in the asymptotic behaviour of the neutral system (1.61). If $r_e(\mathcal{T}^{\circ}(1)) < 1$, there are only finitely many independent unstable solutions. On the other hand $r_e(\mathcal{T}^{\circ}(1)) \geq 1$ implies that there are infinitely many independent unstable solutions. Analogously to the differential equations with delays, a semigroup of bounded linear operators can be associated with the difference equation (1.65)

$$\mathcal{C}_D = \left\{ \boldsymbol{\phi} \in \mathcal{C}, \quad \boldsymbol{\phi}(0) = \sum_{i=1}^{N} \mathbf{H}_i \boldsymbol{\phi}(-\eta_i) \right\} \tag{1.66}$$

where $\mathcal{C}_D$ is the closed subspace of $\mathcal{C}$. The translation along the solution of (1.65)

$$\mathcal{T}_D^{\circ}(t)\boldsymbol{\varphi} = \boldsymbol{x}_t \tag{1.67}$$

defines a strongly continuous semigroup. If the operator is considered $L_0 = \mathbf{0}$, then $r_\sigma(\mathcal{T}^{\circ}(1)) = r_e(\mathcal{T}_D^{\circ}(1))$. If $r_e(\mathcal{T}_D^{\circ}(1)) \geq 1$, the difference equation is not exponentially stable and if $r_e(\mathcal{T}_D^{\circ}(1)) \geq 1$ there are infinitely many unstable solutions. Let us introduce

$$\gamma_0 = \max\{ r_\sigma(\sum_{k=1}^{N} \exp(\mathrm{j}\theta_k) \mathbf{H}_k) \mid \theta_k \in [0, 2\pi],\ k = 1,2,....,N \} \tag{1.68}$$

If the components of $\eta = (\eta_1, \eta_2, ... , \eta_N)$ in the difference operator are rationally independent, then $\mathcal{T}_D^{\circ}(t)$ is exponentially stable if and only if $\gamma_0 < 1$. As can be seen, the condition is independent of η_k. Thus, the condition is equivalent to (1.56), which has been defined for the scalar case for which $\gamma_0 = \sum_{k=1}^{N} |a_k|$, and it defines the strong stability for difference equation (1.65). For further details, definitions and proofs see (Hale and Verduyn Lunel, 1993, Theorem 9.6.1) or (Hale and Verduyn Lunel, 2002). As will be shown later, the concept of the strong stability is particularly important in stabilizing neutral systems by means of the functional feedback from derivations of the state variables, (Hale and Verduyn Lunel, 2002), (Salamon, 1984). The feedback should not only to assess the new spectrum into the stable region, but it should also preserve the stability with respect to the changes in the delays η_i.

1.2.6 Analyzing the spectrum of poles of TDS - analytic methods

As has already been mentioned, the position of the system poles with respect to the imaginary axis is decisive from the system stability point of view. Moreover, the positions of the system poles do not only determine the stability of the system, but also the modes of the system dynamics. It has been shown in section 1.1.4 that it is not trivial to find the solutions of algebraic characteristic equation corresponding to system (1.1). The poles and zeros of (1.1) can be computed analytically only for low order systems. As a matter of course, it is much more complicated to analyse the features (upper real bound, asymptotic behaviour, etc.) of the quasipolynomial spectrum analytically. Such an analysis can only be done for characteristic functions of a specific form. To compute the roots of transcendental equation using elementary analytic operations and functions is not probably possible at all.

As the example of the analysis of TDS spectrum, let us start with the analysis of the first order system with single delay, whose characteristic function is

$$M(s) = s + \exp(-s\tau) \tag{1.69}$$

Regarding $s = \beta + j\omega$, equation (1.69) can be split into real and imaginary parts and the set of equations for computing the system poles is

$$\begin{aligned} \beta + \exp(-\beta\tau)\cos(\omega\tau) &= 0 \\ \omega - \exp(-\beta\tau)\sin(\omega\tau) &= 0 \end{aligned} \tag{1.70}$$

Eliminating the trigonometric terms leads to $|s|^2 = \exp(-2\beta\tau)$ which implies that for sufficiently large |s|, β is negative, i.e., the poles are in the left half-plane. Eliminating the exponential terms leads to $\tan(\omega\tau) = -\omega/\beta$, thus the spacing of the imaginary parts of the poles with the large imaginary parts are approximately $2\pi/\tau$. It is also easy to show that the poles cross the imaginary axis (β=0, cos($\omega\tau$)=0, sin($\omega\tau$)=ω, which yields $\omega = \pm 1$, cos(τ)=0, sin(τ)=1) for the values of $\tau = \pi/2 + 2\pi k$, $k = 1, 2, \ldots$. Differentiating equation (1.69) implies s=ln(τ)/τ. Evaluating equation (1.69) for this value of s result in the value of delay τ = exp(-1) for which the double real pole $s_{1,2}$= exp(1) is present in the spectrum of the system poles. Considering $\tau = 0$, the system has only one pole s = -1. Because the poles move continuously with respect to changes in the parameters (Hale, 1977), the preceding results imply that for $\tau <$ exp(-1), there exist two real poles in the spectrum.

In Marshall, et al., (1992), an algorithm for the stability analysis of the system with the characteristic function

$$M(s,\tau) = \sum_{i=0}^{n} a_i s^i + \sum_{i=0}^{l} b_i s^i \exp(-s\tau) \tag{1.71}$$

has been introduced. The algorithm investigates the system stability with respect to value of τ. The system poles are not directly computed, but the aim is to find the values of τ for which the distribution of the poles is important from the system stability point of view. The procedure consists of three steps. The first step is to examine the stability for τ =0. In the second step, the case of infinitesimally small positive τ is considered, i.e., when the number of poles increases from n to infinity. The objective of the third step is to find the positive values of τ, at which there are system poles located on the imaginary axis. The algorithm and its extension for the case of characteristic function with more than one delay, which are commensurate in τ, can be found in Marshall, et al., (1992). Another algorithm for analysing the character of the pole-spectrum of the system with characteristic function (1.71) can be found in El'sgol'c and Norkin, (1971), presented also in Górecki, et al, (1989). For analogous algorithm providing the analysis of the asymptotic behaviour of the spectrum of poles corresponding to a system with more than one delay see Bellman and Cooke (1963).

As can be seen, to analyse the spectra of poles and zeros of the system with time delays analytically is rather tedious. There exist algorithms that perform the analysis, but the algorithms can usually be applied only to a narrow class of systems with only one or few lumped delays. Therefore, as in the case of higher order systems (1.1), the numerical-based method has to be used for computing the poles of time delay systems.

1.2.7 Computing the spectrum of poles of retarded system - numerical methods

As has been mentioned in section 1.1.4, high order polynomials are likely to be ill conditioned. Therefore, from the numerical point of view, it is rather inconvenient to compute the system poles from the characteristic polynomial, which is extracted from the system matrix **A**. It is more convenient to compute the poles directly as the eigenvalues of the matrix **A**. Analogously, in case of TDS, it is also convenient to avoid extracting the characteristic function, particularly, if the system is of a higher order. (It should be noted that higher order models result rarely if the delays are used in the modelling. Considering the characteristic matrix $\mathbf{\Delta}(\lambda)$, given by (1.58), the solutions of characteristic equation (1.41) correspond to the solutions of the following set of equations

$$\begin{aligned} &\mathbf{\Delta}(\lambda)\mathbf{v} = 0 \\ &\mathbf{c}^{\mathrm{T}}\mathbf{v} - 1 = 0 \end{aligned} \tag{1.72}$$

where $\mathbf{v}$ is the eigenvector of matrix $\mathbf{A}(s)$, normalised so that $\mathbf{c}^{\mathrm{T}}\mathbf{v} = 1$, where $\mathbf{c}$ is a constant vector. Note that vector $\mathbf{c}$ should not be orthogonal to $\mathbf{v}$, i.e., $\mathbf{c} \cdot \mathbf{v} = \sum_{i=1}^{n} c_i v_i \neq 0$, see Engelborghs and Roose (1999) and Moore and Spence, (1980).

Let us briefly explain the basic features of the available methods for computing the approximate positions of the rightmost poles of the retarded systems. A lot of work in this particular field has been done by Engelborghs and Roose and also by Ford and Wulf. In their common research field dealing with the extension of the bifurcation analysis to the class of delay differential equations, they have introduced two different approaches for computing the approximate positions of the right-most system poles (Engelborghs and Roose, 1999, 2002), (Engelborghs, Luzyanina and Roose, 2000), (Ford and Wulf, 1998), (Wulf and Ford, 2000). In the mentioned papers, the system being analysed consists of a set of nonlinear delay differential equations. The system with single delay is considered in the mentioned papers of Ford and Wulf, while in those of Engelborghs and Roose a following system with multiple fixed lumped delays is considered

$$\frac{d\mathbf{x}(t)}{dt} = f(\mathbf{x}(t), \mathbf{x}(t-\tau_1), ..., \mathbf{x}(t-\tau_N), \alpha) \tag{1.73}$$

where $\mathbf{x} \in \mathrm{R}^n$, f: $\mathrm{R}^{n\times(N+1)} \times \mathrm{R}^n$, $\alpha \in \mathrm{R}$, $\tau_i > 0$, i=1..N and $0 < \tau_1 < \tau_2 < ..... < \tau_{N-1} < \tau_N$. Steady state solution $\mathbf{x}^* \in \mathrm{R}^n$ (equilibrium) of (1.73), given as the solution of $f(\mathbf{x}^*, \mathbf{x}^*, ..., \mathbf{x}^*, \alpha) = 0$, does not depend on the delays. In order to analyse the stability of (1.73) around the steady state solutions $\mathbf{x}^*$, the linearization of (1.73) is performed resulting in the so-called variational equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}_0\mathbf{x}(t) + \sum_{j=1}^{N} \mathbf{A}_j \mathbf{x}(t-\tau_j) \tag{1.74}$$

where, using $f(\mathbf{x}^0, \mathbf{x}^1, ..., \mathbf{x}^N, \alpha) = 0$

$$\mathbf{A}_j = \left.\frac{\partial f}{\partial \mathbf{x}^j}\right|_{(\mathbf{x}^*(t), \mathbf{x}^*(t-\tau_1), ..., \mathbf{x}^*(t-\tau_N), \alpha)}, j = 0, ..., N \tag{1.75}$$

The key to the stability analysis of (1.73) is the calculation of the roots of the characteristic equation of (1.74)

$$\det(\boldsymbol{\Delta}(\lambda)) = 0 \tag{1.76}$$

where the characteristic matrix is given by

$$\boldsymbol{\Delta}(\lambda) = \lambda\mathbf{I} - \mathbf{A}_0 - \sum_{j=1}^{N} \mathbf{A}_j \exp(-\lambda_j) \tag{1.77}$$

The steady state solution $\mathbf{x}^*$ of (1.73) is asymptotically stable if all the solutions of (1.77) have strictly negative real parts, i.e., the poles of 1.74 are located on the left half of the complex plane. As can be seen, function f in (1.73) has a parameter α. Therefore a branch of steady state solutions $\mathbf{x}^*(\alpha)$ can be computed as a function of the parameter using a continuation procedure. The aim of the bifurcation analysis is to analyse the stability of (1.73) in the vicinity of its steady state solution $\mathbf{x}^*(\alpha)$ with respect to parameter α, particularly to find the bifurcation points. Bifurcation occur whenever roots of (1.76) move through the imaginary axis as the parameter α acquires a certain value, provided that the other roots of (1.76) have strictly negative real parts. Fold bifurcation occurs when the root is real and a Hopf bifurcation occurs when it is a complex pair. To find out more on bifurcation analysis see, e.g., Marsden and McCracken, (1976), Hassard, et al, (1981), Chow and Hale, (1982), Kuznetsov (1995). Obviously, the behaviour of the rightmost poles is decisive in the bifurcation analysis. Note that the variational equation belongs to the class of retarded systems and has finitely many poles (roots of (1.76)) whose real parts are greater than any real a, see section 1.2.4. Engelborghs and Roose (1999) have developed the algorithm for computing the rightmost roots of (1.76), see also Luzyanina, et al, (1997). The outline of the idea is as follows: Firstly, using a numerical method, the infinite dimensional continuous system is transformed into the finite dimensional discrete system. Secondly, the finitely many poles of the numerical approximation are computed. Finally, using the rightmost poles from the computed set of poles as the starting values for the iterations of Newton's method applied to (1.72), the rightmost roots of set of equation (1.76) are found.

There are two possibilities that can be used to obtain the approximate positions of the roots by means of discretization method. The first approach is based on the discretization of the solution operator $\mathcal{T}^{\circ}(t)$, whose spectrum is given according to (1.59) by $\sigma(\mathcal{T}^{\circ}(t))=\exp(\lambda t)$. If an eigenvalue z of $\mathcal{T}^{\circ}(t)$ is found, the corresponding root of the characteristic equation can be extracted using

$$\mathrm{Re}(\lambda) = \frac{1}{t}\ln(|z|),\ \mathrm{Im}(\lambda) = \arg(z) = \frac{1}{t}\arctan\left(\frac{\mathrm{Im}(z)}{\mathrm{Re}(z)}\right)(\mathrm{mod}\,\frac{\pi}{2t}) \tag{1.78}$$

The unfavourable feature of (1.78) is that because of evaluating the function arctangent, the imaginary parts of the poles result in $\mathrm{mod}(\pi/2t)$. If the four-quadrant inverse tangent is used, (function *atan2* in Matlab), the result is obtained in $\mathrm{mod}(\pi/t)$, which is obviously more favourable result. Thus the imaginary parts of the poles have to be further evaluated to obtain the real values of the poles. The way of discretising $\mathcal{T}^{\circ}(t)$ as well as the algorithm for computing the rightmost roots of the characteristic equation and the features of the algorithm can be seen in Engelborghs and Roose (1999). The second approach, which has been worked out by Ford and Wulf, (1988), see also Wulf and Ford, (2000), is based on discretization of the infinitesimal generator $\mathcal{A}$ whose eigenvalues are identical with the roots of the characteristic equation. The basic ideas and the features of both methods can be found in Engelborghs, et al, (2000). In Engelborghs and Roose (2002), the stability of linear multistep methods (LMS) in computing the approximation of the righmost system poles is investigated.

The approximation by means of LMS methods is also used in DDE-Biftool (Engelborghs 2000), (Engelborghs, et al, 2001b), the Matlab package for bifurcation analysis of delay differential equations.

Let us outline the fundamental ideas of the methods. A general linear k-step LMS method may be written as

$$\sum_{i=0}^{q} \alpha_i \mathbf{x}_{k+i} = h \sum_{i=0}^{q} \beta_i f_{k+i} \tag{1.79}$$

where h is the step of the discretization (sampling period) and α_i, β_i are the coefficients of the method, see, e.g., Hairer, et al, (1987). It is conventional to normalise (1.79) so that $\alpha_q = 1$. The method can be rewritten into the form in which the k+1st sample of $\mathbf{x}$ is computed from the last samples of $\mathbf{x}$ and f

$$\mathbf{x}_{k+1} = -\sum_{i=0}^{r} \alpha_{r-i}\ \mathbf{x}_{k-i} + h \sum_{i=-1}^{r} \beta_{r-i} f_{k-i} \tag{1.80}$$

where $r = q - 1$. The method is explicit if β_q=0, otherwise it is implicit. The right hand side $f_i := f(\mathbf{x}_i, \tilde{\mathbf{x}}(t_i - \tau_1), ..., \tilde{\mathbf{x}}(t_i - \tau_N)$ is computed using approximations $\tilde{\mathbf{x}}(t_i - \tau_l)$ obtained from the past values of $\mathbf{x}_j$ regarding the ratio of τ_j and h (it is due to fact that the delays cannot be expected as integer multiples of the sampling period). Therefore, it is necessary to express the delayed variables by means of an interpolation formula using their available samples. An easy way of approximating a delayed variable is the use of the linear approximation

$$\begin{aligned} &\tilde{x}(t - \tau_j)\big|_{t=kh} \cong (1 - \varsigma_j) x_{k-d_j} + \varsigma_j x_{k-d_j+1} \\ &d_j = \mathrm{int}\left(\frac{\tau}{\mathrm{h}}\right), \varsigma_j = \frac{\tau_j - d_j h}{h} \end{aligned} \tag{1.81}$$

where the function int (the greatest integer function) gives the largest integer less than or equal to τ/h. Another possibility, which has been used in Engelborghs, et al, (2001) is based on the use of more general Nordsieck interpolation, given by

$$\tilde{x}(t_k + \varepsilon h) = \sum_{j=-l}^{v} P_j(\varepsilon) x_{k+j},\ \varepsilon \in [0,1) \tag{1.82}$$

where

$$P_j(\varepsilon) = \prod_{w=-l, w \neq j}^{v} \frac{\varepsilon - w}{j - w} \tag{1.83}$$

thus

$$\begin{aligned} &\tilde{x}(t - \tau_j)\big|_{t=kh} \cong \sum_{w=-l}^{v} P_w (1 - \varsigma_j) x_{k-d_j+1+w} \\ &d_j = \mathrm{int}\left(\frac{\tau}{\mathrm{h}}\right), \varsigma_j = \frac{\tau_j - d_j h}{h} \end{aligned} \tag{1.84}$$

for more details see, e.g., Hong-Jiong and Jiao-Xun (1996), Hairer, et al, (1987).

In the mentioned papers that deals with the system discretization performed in order to obtain the approximation of the rightmost roots, the system considered are not in the uniform form. Therefore, let us accomplish the procedure of discretizing the retarded system step by step, using the ideas of the authors mentioned. In order to make the method more transparent let us use the interpolation (1.81), which usually provides sufficient interpolation if $h < \tau_{\min}$. Applying the interpolation to (1.74), the right-hand side is given by

$$f_i = \mathbf{A}_0 \mathbf{x}_i + \sum_{j=1}^{N} \left((1-\varsigma_j) \mathbf{A}_j \mathbf{x}_{i-d_j} + \varsigma_j \mathbf{A}_j \mathbf{x}_{i-d_j+1} \right) \tag{1.85}$$

Considering the mesh on the interval $[0, h(d_{\max}+1)]$

$$\Gamma = \{0,\ h,\ 2h, \ldots\ldots, d_{\max} h, (d_{\max}+1)h\} \tag{1.86}$$

it is convenient to rewrite (1.81) into the form

$$f_i = \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \mathbf{x}_{i-l} \tag{1.87}$$

where $H=d_{\max}+1$ and the matrices $\tilde{\mathbf{A}}_l$ corresponding to $\mathbf{x}_{i-l}$ are given by the values of matrices $\mathbf{A}_0$ and $\mathbf{A}_j$ from (1.85), as follows

$$\left.\begin{aligned} &\tilde{\mathbf{A}}_l = (1-\varsigma_j)\mathbf{A}_j, \quad l = d_j \\ &\tilde{\mathbf{A}}_l = \varsigma_j \mathbf{A}_j, \quad l = d_j + 1 \\ &\tilde{\mathbf{A}}_l = (1-\varsigma_j)\mathbf{A}_j + \varsigma_{j-1}\mathbf{A}_{j-1}, \quad \tau_j - \tau_{j-1} < h,\ l = d_j \\ &\text{otherwise } \tilde{\mathbf{A}}_l = \mathbf{0}, \end{aligned}\right\} \quad j = 0...N,\ l = 0..H \tag{1.88}$$

If the Nordsieck interpolation of the delays is used, the discrete form of f_i can also be described by the expression (1.87) with the matrices $\tilde{\mathbf{A}}_l$ obtained analogously to (1.88).

1.2.8 Discretization of the solution operator

In order to find the discrete approximation of the solution operator $\mathcal{T}^{\vartheta}(t)$, let us introduce the discrete form of the autonomous system state. Considering the mesh on the last segment of the system history given by (1.86) the discrete approximation of the system state at step k is obviously given by the values of $\mathbf{x}_{k-l}$, $l=0..H^*$, H^* depends on H and on the order of the numerical method used. Putting the samples of $\mathbf{x}$ into a vector

$$\boldsymbol{x}_k = \left[\mathbf{x}_k, \mathbf{x}_{k-1}, \ldots., \mathbf{x}_{k-H^*+1}, \mathbf{x}_{k-H^*} \right]^T \tag{1.89}$$

determining the system state, the state at the step $k+1$ is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi} \boldsymbol{x}_k \tag{1.90}$$

which is the discrete form of

$$\boldsymbol{x}_{t+h} = \mathcal{T}^{\circ}(h)\boldsymbol{x}_t \tag{1.91}$$

and where $\boldsymbol{\Phi} \in \mathbb{R}^{(H^*+1)n \times (H^*+1)n}$ is the discrete approximation of $\mathcal{T}^{\circ}(t)$. Therefore, the approximations of the rightmost system poles are obtained from (1.78) by substituting $t=h$.

As the demonstration of the approach in discretizing the solution operator, let us use the simplest LMS method, i.e., Euler's explicit method, for the discretization given by the formula

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h f_k \tag{1.92}$$

Considering f_i given by (1.87)

$$\mathbf{x}_{k+1} = \mathbf{x}_k + h\sum_{l=0}^{H} \tilde{\mathbf{A}}_l \mathbf{x}_{k-l} \tag{1.93}$$

and the discrete approximation of the solution operator over the step h is

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{I} + h\tilde{\mathbf{A}}_0 & h\tilde{\mathbf{A}}_1 & h\tilde{\mathbf{A}}_2 & \ldots\ldots & h\tilde{\mathbf{A}}_{H-1} & h\tilde{\mathbf{A}}_H \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{I} & \mathbf{0} \end{bmatrix} \tag{1.94}$$

Next, let us consider the general form of the second order linear numerical method

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + h(\beta_1 f_{k+1} + \beta_0 f_k) \\ &= \mathbf{x}_k + h(\beta_1 \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \mathbf{x}_{k-l+1} + \beta_0 \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \mathbf{x}_{k-l}) \end{aligned} \tag{1.95}$$

for which the matrix $\boldsymbol{\Phi}$ acquires the following form

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{PQ}_0 & \mathbf{PQ}_1 & \mathbf{PQ}_2 & \ldots\ldots & \mathbf{PQ}_{H^*-1} & \mathbf{PQ}_{H^*} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{I} & \mathbf{0} \end{bmatrix} \tag{1.96}$$

where

$$\begin{aligned} &\mathbf{P} = (\mathbf{I} - h\beta_1 \tilde{\mathbf{A}}_0)^{-1} \\ &\mathbf{Q}_0 = \mathbf{I} + h(\beta_1 \tilde{\mathbf{A}}_1 + \beta_0 \tilde{\mathbf{A}}_0), \mathbf{Q}_m = h(\beta_1 \tilde{\mathbf{A}}_{m+1} + \beta_0 \tilde{\mathbf{A}}_m),\ m = 1..H, \\ &\tilde{\mathbf{A}}_{m+1} = \mathbf{0},\ H^* = H \end{aligned} \tag{1.97}$$

For the most frequently used first and second order numerical methods, the coefficients β_0 and β_1 are of the following values: Euler explicit - $\beta_1 = 0$, $\beta_0 = 1$, Euler implicit - $\beta_1 = 1$, $\beta_0 = 0$,

Trapezoidal standard $\beta_1 = 0.5$, $\beta_0 = 0.5$ and Trapezoidal modified $\beta_1 = 1-\kappa$, $\beta_0 = \kappa$. Analogously, for the general form of LMS (1.80)

$$\mathbf{x}_{k+1} = -\sum_{i=0}^{r} \alpha_{r-i}\, \mathbf{x}_{k-i} + h \sum_{i=-1}^{r} \beta_{r-i} \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \mathbf{x}_{k-i-l} \tag{1.98}$$

The matrix $\boldsymbol{\Phi}$ is of form (1.96) where

$$\begin{aligned} &\mathbf{P} = (\mathbf{I} - h\beta_{r+1}\tilde{\mathbf{A}}_0)^{-1} \\ &\mathbf{Q}_m = -\alpha_{r-m}\mathbf{I} + h\sum_{i=0}^{r+1} \beta_i \tilde{\mathbf{A}}_{m-r+i},\ m = 0..H^*, \\ &\{\tilde{\mathbf{A}}_g = \mathbf{0}, (g<0 \text{ or } g>H)\}, \{\alpha_h = 0, h<0\},\ H^* = H + r \end{aligned} \tag{1.99}$$

The necessary condition for the successful implementation of the numerical method is the choice of the suitable value of step h. The value of h determines the stability region for which the numerical method is stable, see Engleborghs and Roose, (2002). Usually, the smaller the value of h is, the numerical method is more likely to be stable. On the other hand, it should be noted that the step h should not be too small. Inadequately small h enhances the role of truncation errors, which influences the accuracy of the result in a negative way. Discretizing a time delay system, the step length of h also determines the order of the resultant discretized system. Therefore, h should also be chosen with respect to the values of the maximum delay. To sum up, there are two contradictory requirements for the length of h. Smaller h results in more precise approximation of the rightmost system poles, however, on the other hand, higher order of the resultant discrete system is obtain which can cause difficulties in computing the eigenvalues of the matrix $\boldsymbol{\Phi}$. It is well known, that the stability of the implicit methods is not so sensitive with respect to the discretization step h. Therefore, in case of TDS, it is better to use an implicit numerical method, which allows the higher values of h to be chosen and thereby to obtain reasonably high order of the resultant discrete system. The problem of the choice of h is also discussed in Engelborghs and Roose (2002), where the related heuristic criteria can be found.

Tab. 1.1 R-K method Radau II, matrix $\mathcal{A}$ from the Butcher tableau with respect to the order of the method p

	$p=1$, $(s=1)$	$p=3$, $(s=2)$	$p=5$, $(s=3)$
$\mathcal{A}$	1	$\begin{bmatrix} \frac{5}{12} & -\frac{1}{12} \\ \frac{3}{4} & \frac{1}{4} \end{bmatrix}$	$\begin{bmatrix} \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{bmatrix}$

An alternative approach in discretizing the solution operator and the infinitesimal generator based on the application of Runge-Kutta (R-K) methods can be found in Breda, et al., (2001). In the mentioned paper, the method is applied to the scalar system with a single delay. Using the ideas from the paper, let us extend the method to TDS of form (1.75). Lets us

apply s-stage R-K method, with the Butcher tableau given by $\mathcal{A}$, $\mathcal{b}$ and $\mathcal{c}$ (Butcher, 1987), (Hairer, et al, 1987, 1996, 1989), see Tab. 1.1, to system (1.74) with the right-hand side approximated by (1.87) acquires the following form

$$\begin{aligned}\mathbf{Y}^{(k+1)} &= \mathbf{1}_s \otimes \mathbf{x}_k + h\sum_{l=0}^{H}\left(\mathcal{A}\otimes\tilde{\mathbf{A}}_l\right)\mathbf{Y}^{k+1-l}\\ \mathbf{x}_{k+1} &= \mathbf{Y}_s^{(k+1)}\end{aligned} \tag{1.100}$$

where $\mathbf{Y}^{(k+1)} = [\mathbf{Y}_1^{(k+1)},.....,\mathbf{Y}_s^{(k+1)}]^T$, $\mathbf{Y}^{(k+1)} \in \mathbb{R}^{ns}$ is the stage vector at the step (k+1) and $\mathbf{1}_s = [1,1,.....,1]^T \in \mathbb{R}^s$ and the symbol $\otimes$ denotes the Kronecker product, called also matrix direct product, (Schafer, 1996). Combining equations (1.100) results in

$$\mathbf{Y}^{(k+1)} = (\mathbf{I}_s - h\mathcal{A}\otimes\tilde{\mathbf{A}}_0)^{-1}\left((\mathbf{E}_s\otimes\mathbf{I})\mathbf{Y}^k + h\sum_{l=1}^{H}\left(\mathcal{A}\otimes\tilde{\mathbf{A}}_l\right)\mathbf{Y}^{k+1-l}\right) \tag{1.101}$$

where $\mathbf{I}_s \in \mathbb{R}^{ns}$ and $\mathbf{I} \in \mathbb{R}^n$ are identity matrices and $\mathbf{E}_s = [\mathbf{0}_s,....,\mathbf{0}_s,\mathbf{1}_s]$, $\mathbf{0}_s = [0,0,.....,0]^T \in \mathbb{R}^s$ $\mathbf{E}_s \in \mathbb{R}^{s\times s}$. In order to obtain the matrix $\mathbf{\Phi}$, i.e., the approximation of the solution operator $\mathcal{T}(h)$, let us rewrite (1.101) into the formula analogous to (1.90)

$$\boldsymbol{X}_{k+1} = \mathbf{\Phi}\boldsymbol{X}_k \tag{1.102}$$

with the vector

$$\boldsymbol{X}_k = \begin{bmatrix}\mathbf{Y}^{(k)}\\ \vdots \\ \mathbf{Y}^{(k+1-H)}\end{bmatrix} \in \mathbb{R}^{nsH} \tag{1.103}$$

representing the state of the numerical approximation of TDS, where the matrix $\mathbf{\Phi} \in \mathbb{R}^{nsH\times nsH}$ is formally of the same form as (1.96), i.e.

$$\mathbf{\Phi} = \begin{bmatrix}\mathbf{PQ}_0 & \mathbf{PQ}_1 & \mathbf{PQ}_2 & ..... & \mathbf{PQ}_{H^*-1} & \mathbf{PQ}_{H^*}\\ \mathbf{I}_s & \mathbf{0} & \mathbf{0} & ..... & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}_s & \mathbf{0} & ..... & \mathbf{0} & \mathbf{0}\\ ..... & ..... & ..... & ..... & ..... & .....\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & ..... & \mathbf{I}_s & \mathbf{0}\end{bmatrix} \tag{1.104}$$

where

$$\begin{aligned}&\mathbf{P} = (\mathbf{I}_s - h\mathcal{A}\otimes\tilde{\mathbf{A}}_0)^{-1}\\ &\mathbf{Q}_0 = \mathbf{E}_s\otimes\mathbf{I} + h\left(\mathcal{A}\otimes\tilde{\mathbf{A}}_1\right),\ \mathbf{Q}_m = h\left(\mathcal{A}\otimes\tilde{\mathbf{A}}_l\right),\ m = 2..H,\\ &H^* = nsH\end{aligned} \tag{1.105}$$

As can be seen, the size of $\mathbf{\Phi}$ is not only given by the order of the original system and the number of delays plus possibly few steps corresponding to the numerical method. The size of the discrete approximation of the solution operator is given by the stage s of the numerical R-K method which is applied. In Breda, et al, (2001), the method Radau IIA, which belongs to the class of stiffly accurate R-K methods (Hairer, et al, 1996) is used. The implementation of R-K methods and particularly of the progressive method Radau IIA to a class of delay differential equations has been studied by Zenaro, (1986), Weiner and Strehmel, (1988), Hout, (1992) or Guglielmi, et al, (2001). Depending on the order of the method p, the matrix is given according to Table 1.1. Let us note that the method is equivalent to the implicit Euler method for p=1. As in the application of a LMS method, computing the right-most eigenvalues of the numerical approximation of the solution operator $\mathbf{\Phi}$, the rightmost eigenvalues of original continuous time delay system (1.74) are obtained from (1.78) by substituting $t=h$.

1.2.9 Discretization of the infinitesimal generator of the semigroup

The second method for computing the approximations of the rightmost poles of a time delay system is based on the discretization of the infinitesimal generator of the semigroup $\mathcal{A}$, defined by (1.32). Considering equation (1.34), multiplying the generator $\mathcal{A}$ by the actual system state $\boldsymbol{x}_t$, the derivation of the system state $\boldsymbol{x}_t$ is obtained. Analogously to the discrete form of the solution operator, also the discrete form of the generator $\mathcal{A}_h$ can be found. Considering the discrete approximation of the system state (1.89), the discrete form of equation (1.34) acquires the following form

$$\begin{aligned} &\frac{d}{dt}\boldsymbol{x}_k = \mathcal{A}_h \boldsymbol{x}_k \,,\; k>0, \\ &\boldsymbol{x}_0 = \boldsymbol{\varphi}_h \,,\; \boldsymbol{\varphi}_h = \left[\boldsymbol{\varphi}(0)\, \boldsymbol{\varphi}(-h), ..., \boldsymbol{\varphi}((H^*-1)h), \boldsymbol{\varphi}(H^* h)\right]^T \end{aligned} \tag{1.106}$$

In Ford and Wulf, (1988), see also Wulf and Ford, (2000), two possible methods for discretizing the generator $\mathcal{A}$ of a system with a single delay has been introduced. Let us extend the approach to system (1.74) whose right hand side is considered in discrete form (1.87). With respect to the definition of generator (1.32), using the Euler explicit approximation scheme

$$\boldsymbol{\phi}(kh) = \boldsymbol{\phi}((k-1)h) + h\boldsymbol{\phi}'((k-1)h) \tag{1.107}$$

the following equations hold

$$\begin{aligned} &\boldsymbol{\phi}'(0) = \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \boldsymbol{\phi}(-lh) \\ &\boldsymbol{\phi}'(-lh) = \frac{1}{h}\left[\boldsymbol{\phi}(-(l-1)h) - \boldsymbol{\phi}(-lh)\right],\; l = 1..H \end{aligned} \tag{1.108}$$

which determine the form of the discrete approximation of $\mathcal{A}$

$$\mathcal{A}_h = \frac{1}{h}\begin{bmatrix} h\tilde{\mathbf{A}}_0 & h\tilde{\mathbf{A}}_1 & h\tilde{\mathbf{A}}_2 & \ldots\ldots & h\tilde{\mathbf{A}}_{H-1} & h\tilde{\mathbf{A}}_H \\ \mathbf{I} & -\mathbf{I} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & -\mathbf{I} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{I} & -\mathbf{I} \end{bmatrix} \tag{1.109}$$

The rightmost eigenvalues of $\mathcal{A}_h$ are directly the approximations of the rightmost system poles. Comparing the discrete form of the infinitesimal generator $\mathcal{A}_h$ (1.109) with the discrete approximation of the solution operator $\mathbf{\Phi}$ (1.94), it can be seen, that the following condition holds

$$\mathcal{A}_h = \frac{1}{h}\left(\mathbf{\Phi} - \mathbf{I}_H\right) \tag{1.110}$$

where $\mathbf{I}_H \in \mathbb{R}^{(H+1)\times(H+1)}$. Obviously, it is in accordance with (1.30). If the Trapezoidal rule is applied

$$\boldsymbol{\phi}(kh) = \boldsymbol{\phi}((k-1)h) + h[0.5\boldsymbol{\phi}'(kh) + 0.5\boldsymbol{\phi}'((k-1)h)] \tag{1.111}$$

the equations analogous to (1.109) are of the form

$$\begin{aligned} \boldsymbol{\phi}'(0) &= \sum_{l=0}^{H} \tilde{\mathbf{A}}_l \boldsymbol{\phi}(-lh) \\ \boldsymbol{\phi}'(-lh) &= \frac{2}{h}\left[\boldsymbol{\phi}(-(l-1)h) - \boldsymbol{\phi}(-lh)\right] - \boldsymbol{\phi}'(-(l-1)h),\ l = 1..H \end{aligned} \tag{1.112}$$

To obtain the numerical approximation of the generator $\mathcal{A}$, let us introduce the matrix

$$\mathbf{\Theta} = \frac{2}{h}\begin{bmatrix} 0.5h\tilde{\mathbf{A}}_0 & 0.5h\tilde{\mathbf{A}}_1 & 0.5h\tilde{\mathbf{A}}_2 & \ldots\ldots & 0.5h\tilde{\mathbf{A}}_{H-1} & 0.5h\tilde{\mathbf{A}}_H \\ \mathbf{I} & -\mathbf{I} & \mathbf{0} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & -\mathbf{I} & \ldots\ldots & \mathbf{0} & \mathbf{0} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots\ldots & \mathbf{I} & -\mathbf{I} \end{bmatrix} \tag{1.113}$$

The rows of $\mathcal{A}_h$ are then given by the following recursive rule

$$\begin{aligned} \mathcal{A}_{h0} &= \mathbf{\Theta}_0 \\ \mathcal{A}_{hl} &= \mathbf{\Theta}_l - \mathcal{A}_{hl-1}, l = 1..H \end{aligned} \tag{1.114}$$

where $\mathcal{A}_{hl}$ and $\mathbf{\Theta}_l$, are the l^{th} rows of the matrices $\mathcal{A}_h$ and $\mathbf{\Theta}$, respectively. In Ford and Wulf, (1988), the approach is directly based on the first and second order approximations of the derivation operator. If the first order approximation is used, the result is equivalent to (1.109). Using a quadratic approximation of the derivative, e.g. second order approximation, the discrete form of the infinitesimal generator acquires the form

$$\mathcal{A}_h = \frac{1}{2h}\begin{bmatrix} 2h\tilde{\mathbf{A}}_0 & 2h\tilde{\mathbf{A}}_1 & 2h\tilde{\mathbf{A}}_2 & 2h\tilde{\mathbf{A}}_3 & ..... & 2h\tilde{\mathbf{A}}_{H-2} & 2h\tilde{\mathbf{A}}_{H-1} & 2h\tilde{\mathbf{A}}_H \\ \mathbf{I} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & ..... & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} & ..... & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ ..... & ..... & ..... & ..... & ..... & ..... & ..... & ..... \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & ..... & \mathbf{I} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & ..... & -\mathbf{I} & 4\mathbf{I} & -3\mathbf{I} \end{bmatrix} \tag{1.115}$$

The similar approaches in discretising the generator $\mathcal{A}$ can also be found in Bellen and Maset, (2000), where instead of the approximation $\phi'(kh) = [\phi((k+1)h) - \phi((k-1)h)]/(2h)$ the approximation $\phi'(kh) = [-\phi((k+2)h) + 4\phi((k+1)h) - 3\phi(k)h)]/(2h)$ is used.

In Breda, et al, (2001), the results of the discretization of the generator $\boldsymbol{A}$ of the semigroup associated with the scalar system with a single delay by means of the numerical method Runge-Kutta Radau IIA can be found. Analogously to the first and the second order approximation, considering the scheme of a R-K method Radau IIA, let the approximation of the derivation function segment $\boldsymbol{\phi}$ of the initial conditions be defined in the following form

$$\begin{aligned} \boldsymbol{\phi}'(0) &= \sum_{l=0}^{H} (\tilde{\mathbf{A}}_l \boldsymbol{\phi}(-lh)) \\ \boldsymbol{F}'(-lh) &= \frac{1}{h} \mathcal{A}^{-1} \otimes \left(\mathbf{1}_s \otimes \boldsymbol{\phi}(-(l-1)h) - \boldsymbol{F}(-lh)\right),\ l = 1...H \end{aligned} \tag{1.116}$$

where $\mathcal{A}$ is the matrix determining the R-K method Radau IIA, see Table 1.1, and $\boldsymbol{F}(t) = [\boldsymbol{F}_1(t),....,\boldsymbol{F}_s(t)]^T$, $\boldsymbol{F}_s(t) = \boldsymbol{\phi}(t)$. The approximation of the functions $\boldsymbol{F}_i(t)$, $i = 1..s-1$, can be obtained by means of applying the particular R-K scheme to the known function segment $\boldsymbol{\phi}$. Considering equations (1.116), the discrete approximation of the generator yields the following form

$$\mathcal{A}_h = \begin{bmatrix} \tilde{\mathbf{A}}_0^s\ \tilde{\mathbf{A}}_1^s .....\tilde{\mathbf{A}}_{H-1}^s\ \tilde{\mathbf{A}}_H^s \\ \mathcal{B}_h \end{bmatrix} \tag{1.117}$$

where $\mathcal{A}_h \in \mathbb{R}^{(Hs+1)\times(Hs+1)}$, $\tilde{\mathbf{A}}_l^s = [\tilde{\mathbf{A}}_l, \mathbf{Z}]$, $\tilde{\mathbf{A}}_l^s \in \mathbb{R}^{n\times ns}$, $l = 0...H-1$, $\mathbf{Z} = [\mathbf{0}_n, \mathbf{0}_n, ..., \mathbf{0}_n]$, $\mathbf{Z} \in \mathbb{R}^{n\times n(s-1)}$, $\mathbf{0}_n = [0,0,.....,0]^T$, $\mathbf{0}_n \in \mathbb{R}^s$, $\tilde{\mathbf{A}}_H^s = \tilde{\mathbf{A}}_H$, n is the order of the system, s is the order of the stage of R-K method and

$$\mathcal{B}_h = \frac{1}{h}\begin{bmatrix} \mathbf{w} & \mathbf{W} & \mathbf{0} & ..... & \mathbf{0} \\ \mathbf{0} & \mathbf{w} & \mathbf{W} & ..... & \mathbf{0} \\ ..... & & \ddots & \ddots & ..... \\ \mathbf{0} & \mathbf{0} & ..... & \mathbf{w} & \mathbf{W} \end{bmatrix}, \qquad \begin{aligned} \mathbf{w} &= (\mathcal{A}^{-1}\mathbf{1}_s) \otimes \mathbf{I} \\ \mathbf{W} &= \mathcal{A}^{-1} \otimes \mathbf{I} \end{aligned} \tag{1.118}$$

$\mathcal{B}_h \in \mathbb{R}^{Hs\times(Hs+1)}$, $\mathbf{w} \in \mathbb{R}^{ns\times n}$, $\mathbf{W} \in \mathbb{R}^{ns\times ns}$ and the matrix $\mathbf{w}$ is aligned with the s^{th} part of the matrix $\mathbf{W}$, i.e. with its columns within $((n\text{-}1)s\text{+}1)..ns$. Using this method, particularly if s=3 and $h << \tau_{\max}$, considerably large matrix $\mathcal{A}_h$ results. On the other hand, the number of the rightmost eigenvalues of $\mathcal{A}_h$ which approximate the rightmost poles of the system with a high accuracy is much larger than in case of using the first or the second order LMS method. Let us

recall again that according to the definition of the spectrum of the infinitesimal generator $\mathcal{A}$, see section 1.2.5, the right-most eigenvalues of $\mathcal{A}_h$ are directly the approximations of the rightmost poles of the system.

1.2.10 Discrete approximation of TDS using Delta transform

An alternative approach to the TDS discretization approaches described above has been worked out by Petrová, (2001) and Zítek and Petrová, (2002) based on the delta transform, see, Middleton and Goodwin, (1990). Zítek and Petrová have extended the discretization approach that provides a suitable state-space forms for higher order discrite-time models (Comeau and Hori, (1998)) to the class of TDS. The basic idea of the approach consists in replacing the integration operator of equation (1.74) by the l^{th} order discrete-time integrator $I(\delta)$ given by

$$I(\delta) = \frac{b_l \delta^l + \ldots + b_1 \delta + b_0}{\delta^l + a_{l-1}\delta^{l-1} + \ldots + a_1 \delta + a_0} \tag{1.119}$$

and the right-hand side of (1.74) is substituted by (1.87) using an interpolation method. Thus, the δ - model acquires the form

$$\mathbf{x}(\delta) = I(\delta) f(\delta) \tag{1.120}$$

where $f(\delta)$ arises from applying the δ transform to (1.87). Computing the eigenvalues of finite order approximation (1.120), given as the solutions of the characteristic equation

$$M(\delta) = \det[\delta \mathbf{I} - I(\delta) f(\delta)] = 0 \tag{1.121}$$

we can obtain the approximation of the rightmost poles of system (1.74) using the formula analogous to (1.78) given by

$$\mathrm{Re}(\lambda) = \frac{1}{h}\ln(|1 + h\delta|),\ \mathrm{Im}(\lambda) = \arg(1 + h\delta) = \frac{1}{h}\arctan\left(\frac{\mathrm{Im}(1 + h\delta)}{\mathrm{Re}(1 + h\delta)}\right)(\mathrm{mod}\,\frac{\pi}{2h}) \tag{1.122}$$

considering the mutual relationship of the Laplace operator *s* and the operator of δ transform δ in the form $\exp(sh) = 1 + h\delta$. Note that if four-quadrant inverse tangent is used in (1.122) (function *atan2* in Matlab), the result is obtained in $\mathrm{mod}(\pi/t)$. As can be seen, this method used for approximating the rightmost poles of TDS is quite similar to the method based on discretization of the solution operator described above. In the implementations, it is convenient to use the discrete-time integrator $I(\delta)$ corresponding to a discretization numerical method, e.g., LMS or Runge-Kutta method and so on, see Zítek and Petrová, (2002). The main merit of using the described method of computing the approximation of the system rightmost poles is given by the convergence of $\delta \to s$ as $h \to 0$, because

$$\lim_{h \to 0} \delta = \lim_{h \to 0} \frac{\exp(sh) - 1}{h} = s \tag{1.123}$$

which provides a bi-directional bridge between the continuous and discretized model of TDS. Thus, if only the rough approximation of the rightmost poles is sufficient, the roots of (1.121)

may be considered directly as the approximations of the rightmost poles of TDS (provided that h is sufficiently small)

1.2.11 Numerical computation of the eigenvalues of large sparse matrices

The aim of both methods for computing the approximate positions of the rightmost poles of the retarded systems is to obtain as much precise approximations of the poles as possible. To achieve such an enhanced accuracy of the computed poles, discretization step h should be chosen sufficiently small. As regards the mutual relationship between the system delays and the step h, the following conditions should be satisfied $h<<\tau_{\max}$, and at least $h<\tau_{\min}$. Consequently, the choice of a small step h results in the considerably large matrices $\mathbf{\Phi}$ or $\mathcal{A}_h$. This fact of dealing with the matrices of possibly very large sizes should be taken into consideration and the appropriate method for computing the eigenvalues of the matrices should be chosen. In Engelborghs, et al, (2000) the numerical methods like Subspace iteration and Arnoldi's method, if necessary combined with the shift-invert and Caley transformation, are suggested to use in order to compute the rightmost eigenvalues of the large sparse matrices, see also Saad, (1992), Meerbergen and Roose, (1996). For a comprehensive practical guide for the methods of computing the system eigenvalues, where the theory overview as well as the final code-written algorithms can be found, see Bai, et al, (2000). According to the decision tree for choosing the most suitable method for the solution of the particular eigenvalue problem, which is available in the mentioned book, the problem being solved can be characterised as the standart non-Hermitian eigenvalue problem of a sparse matrix of a modest order (order<1000), where only a set of right-most poles is of the main interest. For this type of problem, the decision tree does not exclude using QR method, which belongs to the class of direct methods and is commonly used in practice. It is well known, that for any non-Hermitian matrix $\mathbf{A}$, there exists a unitary matrix $\mathbf{U}$, $\mathbf{U}^T\mathbf{U}=\mathbf{I}$, that transforms the matrix to upper triangular form $\mathbf{T}=\mathbf{U}^T\mathbf{A}\mathbf{U}$. The matrix $\mathbf{T}$ is called the Schur form of $\mathbf{A}$ and the eigenvalues of $\mathbf{A}$ appear along the diagonal of $\mathbf{T}$. The Schur matrix $\mathbf{T}$ can be obtained by using the QR iteration method. However, if QR method is applied directly, the convergence of the iterations is extremely slow. Therefore, before starting the iteration it is convenient to reduce the matrix $\mathbf{A}$ to the upper Hessenberg form, which can be done in a finite number of steps, and then to apply the QR iteration. For more details see, e.g., Wilkinson, (1965) or Demmel, (1997). The possibility to apply this method is especially convenient from the practical point of view because the QR algorithm is available in the package LAPACK, see Anderson, et al, (1999), whose subroutines are accessible from the commonly used software Matlab via the command *eig*. On the other hand, it should be noted that the computational costs increase rapidly with the size of the matrix, (in the range proportional to the order cubed). Also the memory requirements increase substantially with the increasing size of the matrix in the range proportional to the order squared. To sum up, from the practical point of view, the method can be used for the matrices of the size up to few hundreds. For the matrices of the size close to 1000 (practically the limit is rather 500) or larger, an advanced method should be used to compute the eigenvalues. It should be noted that not only size of the matrices is decisive in the process of choosing the method. It should also be taken into account, that the matrices that result from the approximations of the operators are quite sparse (George and Liu, 1981). Moreover, the matrices, as well as the polynomials, see the section 1.1.4, can also be ill conditioned. It means that a small perturbation in the matrix coefficients, caused, e.g., by rounding errors, can considerably change the eigenvalue spectrum. The eigenvalues can be perturbed by much more than the perturbation of the matrix. For example, this ill-conditioning tends to occur when two or more eigenvalues are very close together.

Especially, it can happen if the matrix is very far from the Hermitian (the matrix $\mathbf{A}$ is Hermitian if $\overline{\mathbf{A}}^T = \mathbf{A}$, where $\overline{\mathbf{A}}$ denotes the matrix whose elements are complex conjugate to $\mathbf{A}$. An important feature of the Hermitian matrices is that their spectrum only consists of the real eigenvalues).

Besides the direct methods for solving the eigenvalue problem, there exists a class of so-called iterative methods. These methods, which are completely iterative, are usually suitable only for the computation of a handful of interesting eigenvalues or eigenpairs. The advanced modern iteration methods are based on so-called subspace iteration. The fundamentals of the subspace iteration methods are in the attempt to build a subspace that is rich in the eigenvectors that correspond to the eigenvalues that are of our interest. Thus, dealing with the given number of eigenvectors, the matrices are usually reduced to a more manageable form. The Arnoldi methods, which have been already mentioned, belong to the class of subspace iteration methods. From the other methods, let us mention the Lanczos and Jacobi-Davidson methods. The Arnoldi and Lanczos methods are particularly effective if applied with the shift-and-invert transformation $\mathbf{T}_{\mathrm{SI}} = (\mathbf{A} - \hat{\lambda}\mathbf{I})^{-1}$, where $\hat{\lambda}$ is the shift or pole (user-defined close to the desired eigenvalue), e.g., the methods work formally with $\mathbf{T}_{\mathrm{SI}}$ instead of $\mathbf{A}$. Another possibility consists in the use of the Cayley transform $\mathbf{T}_{\mathrm{SI}} = (\mathbf{A} - \hat{\lambda}\mathbf{I})^{-1}(\mathbf{A} - \hat{\mu}\mathbf{I})$, where $\hat{\lambda}$ is the pole and $\hat{\mu}$ is the zero. Usually both transforms can be performed directly, but if it is not possible, so-called inexact iterative methods has to be used. The mentioned transformations belong to the preconditioned techniques, the aim of which is to accelerate the convergence of the particular iterative method. Since the matrices resulted from the discretization of the solution operator and infinitesimal generator are relatively sparse, the efficiency of the particular iterative method can also be improved by the use of the efficient storage scheme, see Eijkhout, (1992).

To conclude, let us mention some of the software tools or subroutines available to solve the eigenvalue problem of large sparse matrices. Let start with the routines EB12 (multivector iteration, (Duff and Reid, 1993)) and EB13 (Arnoldi algorithm, (Scott, 1995)) for subspace, which are parts of the Harwell Subroutine Library (Fortran 77). Given a real unsymmetric matrix, this routine computes the set of eigenvalues that have the largest positive real part, the largest (in magnitudes) negative real part, or the largest magnitudes. The another collection of the Fortran subroutines for the implicitly restarted Arnoldi method is available in the package ARPACK, (Lehoucq, 1998). The algorithm based on Jacobi-Davidson method is available as the Matlab function JDQR, which can be used to comput a few selected eigenvalues with some desirable property, besides others, also the rightmost eigenvalues, see Fokkema, et al, (1998) and also Sleijpen and Vorst, (1996). The algorithm is effective especially if the matrix is sparse and of large size. From the algorithms providing preconditioning technique let us mention LOBPCG (Matlab, (Knyazev, 2001)).

1.3 State feedback control of TDS

1.3.1 Overview of methods used in control of TDS

In case of TDS, the control design has the specific features and requirements which have to be taken into consideration. For example, if the classical PID feedback control is applied to a single input and single output (SISO) system with a large input delay (dead time), the controller gains should be set considerably low, according to the length of the controller action time lag. Rather slow control system responses result from such a mild setting of the controller parameters. The idea of avoiding the influence of the time lag in the controller action has been introduced by Smith, (1959). The idea is based on a special control loop arrangement known as Smith predictor or Smith dead time compensator that provides the compensation of the input time delay. However, the necessary condition for the successful implementation of Smith predictor is to have a reliable model that truly describes the controlled system. The delay compensation principle in Smith predictor is very sensitive to system uncertainties, which are, however, always encountered in the practical applications. The chances for the implementation of Smith predictor in the real plant control applications are rather reduced. There exist many modifications of Smith predictor that are designed especially to enhance the robustness of the control scheme, see the overview in Palmor, (1996). A frequency domain analysis of Smith predictor can be found in Åström, (1977). Extension of Smith predictor to multivariable systems can be found in Ogunnaike and Ray, (1979).

It has been mentioned that using the classical PID control loop implies rather slow control system dynamics. Nevertheless, from the practical point of view, the use of PID controllers for controlling time delay system can not be ignored. A lot of the industrial applications of control are finally based on miscellaneously assembled loops of PID controllers. An approach of tuning the parameters of PID controllers based on the knowledge of the model of system has been proposed by Morari and Zafiriou, (1989) and is referred to as internal model control, see also Seborg, et al., (1989). The functional extension of the internal model control strategy applied to TDS can be found in Zítek, (1998), Hlava, (1998), Zítek and Hlava, (2001). Above all, the aim of this chapter is to provide a survey of methods performing the control design in time delay systems based on pole placement. Therefore, the authors who contributed significantly to the development of control theory of TDS will be mentioned only briefly. Let us start with the contribution of Oguztoreli (1966) where the extension of optimal control design to TDS can be found. The control techniques and controllers' design are the prime topics in Marshal, (1979), Górecki, et al., (1989), Kolmanovskii and Shaikhet, (1996). Control analysis based on optimization of performance criteria has been studied in Malek-Zaverei and Jamshidi, (1987), Marshal, et al., (1992). In Chukwu, (1989), stability and time optimal control has been investigated. The control of neutral systems has been studied by Salamon, (1984), Hale and Verduyn Lunel, (2002).

1.3.2 Finite spectrum assignment

More advanced method for eliminating the delays from the characteristic function of TDS with the control feedback is the famous method of Finite Spectrum Assignment (FSA) (Manitius and Olbrot, 1979). In the mentioned paper, which is considered as the keystone of huge theory of FSA, the problem of FSA is solved for two classes of systems, systems with delays in control and systems with commensurate delays in state variables. The systems with delays in control are considered in the following form

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \int_0^T d\mathbf{B}(\tau)\mathbf{u}(t-\tau) \tag{1.124}$$

for which the feedback acquires the form

$$\mathbf{u}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{F}\int_0^T\int_0^\theta \exp(\mathbf{A}(\tau - \theta))d\mathbf{B}(\theta)\mathbf{u}(t-\tau)d\tau \tag{1.125}$$

in the mentioned paper of Manitius and Olbrot, (1979), it has been proved that the spectrum of the feedback system given by (1.124) and (1.125) coincides with the spectrum of the matrix $\mathbf{A} + \mathbf{B}(\mathbf{A})\mathbf{F}$ where

$$\mathbf{B}(\mathbf{A}) = \int_0^T \exp(-\tau\mathbf{A})d\mathbf{B}(\tau) \tag{1.126}$$

Assuming the controllability (stabilizability) of the pair (**A**, **B**(**A**)), the finite spectrum of the feedback system, which consist of *n* self-conjugate points in the complex plain, can be placed arbitrarily by a suitable choice of the matrix **F**.

The other problem solved in Manitius and Olbrot, (1979), i.e., FSA applied to a system with commensurate delays in state variables, is more complicated from both the controllability and the spectrum assignment point of view. The aim of this section is not to provide the comprehensive theoretical background of FSA, but it is only to outline the basic features of FSA, the method will not be explained. The method can be found in the paper of Manitius and Olbrot, (1979) and also in the papers of Watanabe et al, (1983, 1984, 1986) where the method was further developed and extended to the commensurate delays in both system state and input variables. There have also been made attempts to extend the FSA to more general time delay systems (already in paper of Manitius and Olbrot, (1979)), however the results obtained are by no means straightforward. The problem of FSA solved in frequency domain can be found in Ichikawa (1985) and Wang, et al., (1993).

Even though FSA method has been proved to be a tool that can perform the whole spectrum assignment of a class of time delay systems, the method had retained only in the theoretical stage for a long time and even simulation verification of the results had not been done. Probably Wang and his co-workers were the first who brought forward the FSA method into applications. In Wang et al (1995) a modification of FSA is presented to achieve asymptotic tracking and regulation. Also the practical issues of FSA implementation are discussed. As the application example, the modified FSA method is applied to a system described by first order model with input time delay, which is often used for rough, but often sufficient, description of the industrial processes. Also an identification method based on relay feedback test is presented in the paper. Further development of the mentioned modified FSA and the relay feedback identification can be found in monograph of Wang et al. (1999).

Even if the difficulties of the FSA method implementation are not considered, there is another questionable issue connected with the method. A specific problem of FSA method is the existence of feedback loops closed just from the control input itself, which may result in neutral character of the obtained feedback system. Due to this feature, this control strategy sometimes does not ensure enough system robustness. In Engelborghs, et al., (2001) a problem of equivalence of the feedback system with system of neutral type was discussed and important consequences have been derived. As has been mentioned in section 1.2.4, the necessary condition for the stability of a neutral system is the stability of the difference part of

the system, which provides the essential spectrum. It is shown in the paper of Engelborghs that the feedback system is unstable with respect to the arbitrarily small perturbations caused by a practical implementation of the integral term (e.g., using a numerical rule) whenever its difference part is unstable. This fact further reduce the number of systems for which the FSA method results in more or less robust feedback system. To sum up the FSA is a nice theory that provides a broad research field and which can be further studied with the results likely to be accepted for publication in prestigious journals. However, except the contributions of Wang, et al, and maybe few others, from the practical point view the method has not been proved to provide valuable benefits to the field of applied control engineering.

1.3.3 Ackerman formula and its extension to TDS, concept of spectral controllability

There exist several methods to accomplish the pole assignment for delay free system (1) by means of proportional feedback

$$u = -\mathbf{K}\mathbf{x}(t) \tag{1.127}$$

where $\mathbf{K} \in \mathrm{R}^{1\times n}$. Most of the methods are based more or less on the system transformation into a canonical form, for which the task is trivial, provided that the system is controllable. Obviously feedback (1.127) allows the eigenvalues of the closed loop system, i.e., the eigenvalues of the matrix **A**-**BK**, to be assigned. From the first contributions, where the problem of pole assignment was solved, let us mention Wonham, (1967) and Davison, (1970). Nevertheless, the method that is commonly used for the pole placement and which can be found in most of the monographs dealing with the subject of modern control engineering is the method introduced by Ackermann, (1972). The method known as Ackermann formula is based on Cayley-Hamilton theorem, which states that any square matrix satisfies its own characteristic equation. Suppose the desired characteristic function of the feedback system is prescribed by the choice of n system poles λ_i

$$M_d(s) = \det(s\mathbf{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) = s + \sum_{i=0}^{n-1} a_i s^i = \prod_{i=1}^{n} (s - \lambda_i) \tag{1.128}$$

The gain coefficients **K** performing such a pole assignment are given by the Ackermann formula

$$\mathbf{K} = [0,0,...1]\mathbf{R}^{-1} M_d(\mathbf{A}) \tag{1.129}$$

where

$$\mathbf{R} = \left[\mathbf{B},\ \mathbf{A}\mathbf{B},\ \ldots\ \mathbf{A}^{n-1}\mathbf{B}\right] \tag{1.130}$$

is the controllability matrix. The necessary condition is that the system has to be controllable, i.e., the condition $\mathrm{rank}(\mathbf{R}) = n$ is satisfied, see, e.g. Ogata, (1997), Sontag, (1998). The system is controllable if the control input u influences all the state variables and using the control input only, an arbitrary state of the system can be achieved. If the controllability condition is satisfied, the whole spectrum of the feedback system can be assigned by means of the setting of the feedback gains.

The concept of controllability does not change in case of TDS. The fully consistent concept of controllability of TDS to the delay free case requires considering the functional character of the system state. The relevant conditions determining the system controllability

are then rather complicated, see, e.g., Górecki, et al., (1989). However, such an exact attainability of a functional state in a prescribed time interval is rather unsubstantial from the practical realisation point of view. On the other hand, the condition that must be satisfied to change the system dynamics in a desired way via a feedback is the same as in the case of delay free system, i.e., the control input u influences all the state variables and the positions of all the system poles. This controllability notion is known as spectral controllability and is defined as follows. System (1.15) is spectral controllable by means of the control input u if there exists such a feedback

$$u(t) = -\int_0^T d\mathbf{K}(\tau)\mathbf{x}(t-\tau) \tag{1.131}$$

that assigns n prescribed eigenvalues to the spectrum of the feedback system

$$\frac{d\mathbf{x}(t)}{dt} = \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) - \int_0^T d\mathbf{B}(\tau)\int_0^T d\mathbf{K}(\theta)\mathbf{x}(t-\tau-\theta) \tag{1.132}$$

The condition of the spectral controllability of TDS is analogous to the controllability of delay free system, see Zítek, (1998), i.e.

$$\text{rank}(\mathbf{R}(s)) = n\,, \quad \mathbf{R}(s) = \left[\mathbf{B}(s),\ \ \mathbf{A}(s)\mathbf{B}(s),\ \ \ldots\ \ \mathbf{A}(s)^{n-1}\mathbf{B}(s)\right] \tag{1.133}$$

for any s where $\mathbf{R}(s)$ is the controllability matrix of TDS. The functional feedback can be designed by means of the extension of the Ackermann formula to TDS suggested in Zítek, (1998) and in Zítek and Vyhlídal, (2001). The aim of the method is to accomplish cancellation of all the delays in the characteristic function of (1.132) in order to obtain a delay free feedback system with the dynamics determined by the prescribed n system poles λ_i, i.e.,

$$M_c(s) = \det(s\mathbf{I} - \mathbf{A}(s) + \mathbf{B}(s)\mathbf{K}(s)) = s + \sum_{i=0}^{n-1} a_i s^i = \prod_{i=1}^{n}(s-\lambda_i) \tag{1.134}$$

The problem of implementation of the modified Ackermann formula

$$\mathbf{K}(s) = [0,0,\ldots 1]\mathbf{R}(s)^{-1} M_d(\mathbf{A}(s)) \tag{1.135}$$

is given by obtaining a non-causal feedback as a rule. The non-causality is caused by the fact that performing the inversion of the controllability matrix R(s), there may appear anticipative factors $\exp(\tilde{\tau}_i)$, i = 1..n, in K(s) if the delays of TDS are lumped, or more complex anticipative terms if some of the delays are distributed. Let us consider the system with lumped delays only. Performing the factorisation

$$\mathbf{K}(s) = [\tilde{K}_1(s),\ \tilde{K}_2(s),\ldots,\tilde{K}_n(s)]\,\text{diag}[\exp(\tilde{\tau}_1),\exp(\tilde{\tau}_2),\ldots,\exp(\tilde{\tau}_n)] = \tilde{\mathbf{K}}(s)\mathbf{E}(s) \tag{1.136}$$

a trimmed matrix $\tilde{\mathbf{K}}(s)$ is obtained free of the predictive terms. The predictive terms in $\mathbf{E}(s)$ can be cancelled out in the feedback system using a special loop arrangement analogous to Smith compensator for MIMO systems. It has been shown in Zítek and Vyhlídal, (2001), where such a modification of Smith compensator has been introduced, that the system description can be transformed into the form

$$\mathbf{x}(s) = [s\mathbf{I} - \mathbf{A}(s)]^{-1}\mathbf{B}(s)u(s) = [s\mathbf{I} - \tilde{\mathbf{A}}(s)]^{-1}\tilde{\mathbf{B}}(s)[\mathbf{E}(s)]^{-1}u(s) \tag{1.137}$$

Performing time shifting given by the diagonal elements in $\mathbf{E}(s)$ across the system equation, the system matrices $\mathbf{A}(s)$ and $\mathbf{B}(s)$ are transformed into matrices $\tilde{\mathbf{A}}(s)$ and $\tilde{\mathbf{B}}(s)$. The functional feedback that involves the Smith compensator separately for each equation of the system description is of the following form

$$u(s) = -\tilde{\mathbf{K}}(s)\,\mathbf{x}(s) + \tilde{\mathbf{K}}(s)\left[[\mathbf{E}(s)]^{-1} - \mathbf{I}\right]\left[s\mathbf{I} - \tilde{\mathbf{A}}(s)\right]^{-1}\tilde{\mathbf{B}}(s)\,u(s) \tag{1.138}$$

where $[\mathbf{E}(s)]^{-1} = \mathrm{diag}[\exp(-sT_1),\ldots,\exp(-sT_n)]$ is a diagonal matrix of delays and $\mathbf{x}$ are the measured or estimated system state variables (for the estimation technique for TDS using a functional model based observer see Zítek, (1998)). The second part of the right-hand side of (1.138) represents the role of the parallel model of the plant. Obviously, if the model and the real plant behaviour are in a perfect accordance the terms $-\tilde{\mathbf{K}}(s)\mathbf{x}(s)$ and $\tilde{\mathbf{K}}(s)[\mathbf{E}(s)]^{-1}\left[s\mathbf{I} - \tilde{\mathbf{A}}(s)\right]^{-1}\tilde{\mathbf{B}}(s)u(s)$ cancel each other and only the predicting term remain as follows

$$u(s) = -\tilde{\mathbf{K}}(s)\left[s\mathbf{I} - \tilde{\mathbf{A}}(s)\right]^{-1}\tilde{\mathbf{B}}(s)u(s) = -\tilde{\mathbf{K}}(s)\mathbf{E}(s)\mathbf{x}(s) \tag{1.139}$$

However, due to the unavoidable plant-model differences such a complete compensation for delays is not achieved in fact. The main drawback of the extension of the Ackermann formula to TDS is the necessity of using Smith compensator, which is known to reduce the robustness of the feedback system. As a matter of fact, also the delays in the state variables are compensated by feedback (1.139) and presumably the robustness of the feedback system is also rather sensitive to the changes of the state variable delays. The robustness issue of the feedback designed has not been systematically studied yet. The consequences of the parameter changes, especially the changes in delays, should be investigated in order to evaluate the applicability of the method to improve the dynamics of the real plants.

1.3.4 Pole placement based control of TDS

From the control implementation point of view, the structure of the control algorithm plays an important role in the control design. The aim of the design is to obtain not too complicated control law. According to that point, the mentioned algorithms, i.e. FSA and the modification of the Ackerman formula, are not too convenient because both the method provides the functional feedback. In contrast to the extensive theory of automatic control, the simple control algorithms usually based on well-known PID control law can be found in most of the industrial applications. Obviously, it is given by the fact that these algorithms are easy to understand and implement. Supposedly, only the controllers that are based on transparent and easily applicable control lows (comparable with the PID algorithm) have the real chance to substitute the classical PID-based controllers in the industrial applications. Regarding the structure of the control law, the classical state feedback control (1.127) is very convenient from the implementation point of view. It should be noted (1.127) is used to stabilize or improve the dynamics of a system. Nevertheless, the control, i.e., the set-point tracking and the disturbance rejection, can also be performed using formula (1.127) provided that an additional state variable x_I is introduced to accomplish the integration of the control error

$$\frac{dx_{\mathrm{I}}(t)}{dt} = w(t) - y_c(t) \tag{1.140}$$

where y_c is the controlled output of the system and w is its desired value (Zítek and Vyhlídal, 2000). Feedback (1.127) applied to this extended system performs the control with the dynamics given by assigned poles. However, in spite of the simplicity of the control law, the algorithm has not been successfully implemented in many industrial applications. Unavailability of the parameter setting rules based on experiments may be one reason why the state feedback controllers are not broadly used. The feedback parameter design is based on the model representation of the plant. Therefore an identification of the model parameters has to be accomplished before designing the feedback gains. Another drawback of the state feedback control is the fact that if classical system (delay free) description (1.1) is used to describe a real plant, the state variables are not often available as the measured system outputs. These unmeasured or even unmeasurable state variables have to be estimated by means of state observers, see, e.g. Ogata, et al, (1997). The whole control low that consists of the state feedback and the observer acquires quite complicated form as a rule.

As has already been mentioned, the functional description of system (1.15) provides the possibility to select the state variables as the available system outputs. Thus, usually, it is not necessary to use the observers because the states are measured on the plant. Even though some of the state variables are not available and have to be estimated, the observer can usually be divided into independent local observers of a very simple structure. Consequently, the application of the feedback from the state variables (1.127) becomes much easier. On the other hand applying the coefficient feedback to TDS described by a functional model does not allow the whole system spectrum (which is infinite) to be assigned. However, if the condition of spectral controllability (1.133) is satisfied, n selected system poles can be assigned. The features of the extension of classical state feedback design by the gain coefficient feedback to the class of TDS has been studied by Zítek, (1997, 1998), Zítek and Vyhlídal, (2000, 2002b). The idea of applying such a pole assignment consists in selecting system poles $\sigma_i, i = 1..n$ to be assigned in order to stabilize or improve the dynamics of TDS. The gain coefficients K_1, K_2, ... , K_n are then computed from the set of equations

$$M(\sigma_i, K_1, K_2, ..., K_n) = 0, \; i = 1..n \tag{1.141}$$

where $M(s, \mathbf{K}) = \det(s\mathbf{I} - \mathbf{A}(s) + \mathbf{B}(s)\mathbf{K})$. The poles assigned can be both real and complex conjugate. In case of prescribing the complex conjugate pair $\sigma_{h,h+1} = \beta_h + \mathrm{j}\omega_h$, the corresponding equation has to appear in (1.141) in the following form $\mathrm{Re}[M(\beta_h, \omega_h, \mathbf{K})] = 0$ and $\mathrm{Im}[M(\beta_h, \omega_h, \mathbf{K})] = 0$. The key problem of the method is that only n poles (from the infinitely many poles) can be assigned. In case of improper choice of the poles to be assigned, some poles from the infinite set of unassigned poles can acquire the positions in the s-plane that will bring on the unfavourable modes into the dynamics of the feedback system. In order to obtain a satisfactory result of the assignment, the assigned poles have to be the dominant poles of the dynamics. It is obvious that a proper analysis of the original system dynamics has to precede the pole assignment. For example, the ultimate frequencies of the feedback system loops provide helpful information in the process of choosing the poles to be assigned. The frequencies, which can be obtained from experiments, are the limit frequencies of the feedback system that should not be exceeded by the pole assignment. Nevertheless, much more valuable information in the process of choosing the poles to be assigned is the knowledge of the original system spectrum. Very important issue to be studied is the

evaluation of the significance of the poles in the system dynamics. Some of the results of solving the problem of pole assignment can be found in Zítek and Vyhlídal, (2002a,b).

1.3.5 Continuous pole placement for TDS

An alternative approach for assessing the gain coefficients of feedback (1.127) has been proposed by Michiels, et al, (2002) known as continuous pole placement for delay equations. The method has been designed to stabilize the unstable systems with delays. The stabilization procedure consists in step-by-step shifting of the rightmost system poles to the left with the simultaneous monitoring of the other poles. The algorithm of the stabilization procedure is as follows:

Algorithm 1.1 Stabilization procedure of continuous pole placement

A. Initialize l=1
B. Compute the rightmost system poles
C. Compute the sensitivity of the l rightmost poles with respect to the changes in the feedback gain $\mathbf{K}$
D. Move l rightmost poles in direction to the left half plane by applying a small changes to the feedback gain $\mathbf{K}$, using the computed sensitivities.
E. Monitor the rightmost uncontrolled poles. If necessary, increase the number of controlled poles l. The algorithm is stopped when stability is reached or when the available degree of freedom in the controller does not allow sup(Re(λ)) to be further reduced. In the other case, go to step B.

Each step of Algorithm 1.1 is described in more detail in Michiels, et al, (2002). In the paper, the features of the method are investigated and applied to the system with single input delay. In the end of the paper, it is shown that the method can also be used to stabilize systems with several lumped delays in both state and input variables. There is also an example of applying the algorithm to a system with distributed delays in the paper. Let us briefly explain the fundamentals of the method considering a linear TDS with the characteristic function $M_c(s,\mathbf{K}) = \det[s\mathbf{I} - \mathbf{A}(s) + \mathbf{B}(s)\mathbf{K}]$. Let us assume that the position of $l<n$ poles $\lambda_1, \lambda_2, ..., \lambda_l$ is to be assigned in the actual step of the stabilization procedure and the sensitivity matrix $\mathbf{S}_l$ is defined by

$$\mathbf{S}_l = [S_{i,j}] \in \mathrm{R}^{l \times n}, \; S_{i,j} = \frac{\partial \lambda_i}{\partial K_j} \tag{1.142}$$

A coefficient $S_{i,j}$ of the matrix $\mathbf{S}_l$ can be computed from the following formula

$$\frac{\partial \lambda_i}{\partial K_j} = \frac{d\, M_c(\lambda_i, \mathbf{K})}{dK_j} \left(\frac{d\, M_c(s, \mathbf{K})}{ds} \right)^{-1} \Bigg|_{s=\lambda_i} \tag{1.143}$$

In the paper of Michiels, et al, (2002), an alternative method for computing the elements of the sensitivity matrix $\mathbf{S}_l$ has been proposed. The method is based on the system of equations (1.72) and provides numerically more robust computation. However, if TDS is of a low order, which is the usual result of using delays in modelling, the use of formula (1.143) is quite safe from the numerical point of view. If desired small displacements of the poles are given by

$\Delta\Lambda_l = [\Delta\lambda_1, \Delta\lambda_2, ..., \Delta\lambda_l]^T$ the change of the gain coefficients can be computed from the linearized difference formula

$$\mathbf{S}_h \Delta\mathbf{K} = \Delta\Lambda_l \tag{1.144}$$

If $l<n$ equation (1.144) has infinitely many solutions. It is possible to choose n-l values of the coefficients of $\Delta\mathbf{K}$ in order to obtain unique solution of l coefficients of the feedback increment. An alternative way of computing $\Delta\mathbf{K}$ consist in using the generalized Moore-Penrose pseudo-inverse $\mathbf{S}_l^+$ of $\mathbf{S}_l$, (Ben-Israel and Greville, 1977), (Campbell and Meyer, 1991), minimizing the square errors in satisfying (1.144)

$$\Delta\mathbf{K} = \mathbf{S}_l^+ \Delta\Lambda_l \tag{1.145}$$

If the displacements of the poles given by $\Delta\Lambda_l$ are sufficiently small, new feedback gain matrix $\mathbf{K}_{i+1} = \mathbf{K}_i + \Delta\mathbf{K}$, causes the shift of the feedback system spectrum towards the prescribed $\Delta\Lambda_l$, where i denotes the last step of the procedure. Nevertheless, the computed changes in the feedback coefficients do not precisely result in the prescribed displacements in $\Delta\Lambda_l$ since equation (1.144) is based on linearization. On the other hand, if the displacements in $\Delta\Lambda_l$ are sufficiently small, the errors in the prescribed and the resultant part of the spectrum are almost negligible. Therefore, the new spectrum of feedback system can be obtained within few iterations of Newton's numerical method (provided that $\hat{\Lambda}_{l\,i+1} = \Lambda_{l\,i} + \Delta\Lambda_i$ are the starting values for the iteration) to compute the current values of $\Lambda_l = [\lambda_1, \lambda_2, ..., \lambda_l]^T$. Together with displacements of the poles being controlled, also the behaviour of the other rightmost poles has to be monitored in order to avoid loosing the control of the stability determining poles. In Michiels, et al., (2002), the monitoring is performed by DDE-BIFTOOL (Engelborghs, 2000) using an discretization based algorithm for computing the approximation of the rightmost poles described in sections 1.2.7-9.

If the aim of the shifting the poles is to control both real and imaginary parts of the poles, the method described above is rather cumbersome. The equivalent results can be achieved by using the algorithm mentioned in section 1.3.4, using direct prescription of the poles. However, the method becomes advantageous if only the real parts of the poles are to be assigned. Such a requirement to control only the real parts of the poles is particularly useful if the primary task of the control design is to assure the robust stability of the system, i.e., to place the poles as much to the left as possible. Considering that

$$\frac{\partial[\mathrm{Re}(\lambda_i)]}{\partial K_j} = \mathrm{Re}\left(\frac{\partial\lambda_i}{\partial K_j}\right) \tag{1.146}$$

equation (1.145) can be rewritten into the form

$$\Delta\mathbf{K} = (\mathrm{Re}(\mathbf{S}_l))^+ \Delta\Lambda_l^{\mathrm{R}} \tag{1.147}$$

where $\Delta\Lambda_h^{\mathrm{R}}$ the desired displacement of the real parts of the controlled poles. This approach is particularly advantageous because it provides the possibility to control real parts of the number of poles varying from n to $2n$ depending on the character of the poles. Obviously,

prescribing a shift in the real part to a complex pole results in the same shift of its complex conjugate pole.

The procedure of stabilizing the retarded system using the continuous pole placement method can be summarised as follows. The first step of the procedure is to compute the rightmost poles of the system. Then, the rightmost pole is shifted to the left applying the feedback coefficients with the increments computed from formula (1.147). Naturally, also the positions of the uncontrolled poles are changed. Therefore, the rightmost system spectrum has to be computed before starting the next step of the procedure. Such a stepwise shifting of the rightmost pole is performed until the real part of another pole (or couple) appears close to the real part of the rightmost pole. Then, instead of one pole (or couple) two poles (or couples) are being shifted to the left. Such an procedure of increasing the number of the real parts of the poles (couples) being controlled is performed until the group of n poles (or couples) gets close to another pole. The further shifting of the poles to the right is not possible, because only n real parts of poles (or couples) can be controlled. It should be noted, that from the numerical point of view, it is necessary to keep a certain distance between the neighbouring poles that are controlled otherwise the procedure fails, see Michiels, et al., (2002).

Unlike in case of classical system description, there exist certain limits in stabilising the system with delays. Obviously, if the number of unstable poles is higher than the maximum number of poles that we can control, the stabilisation by the gain feedback cannot be performed as a rule. Such a problem can also appear if the unstable pole(s) are too far to the right from the stability boundary. Anyway, it is very difficult to prove in advance that an unstable system can be stabilized by the gain coefficient feedback. Stabilisability of TDS with single input delay has been studied in Michiels and Roose, (2002b).

1.3.6 Strong stabilization of neutral systems

It has been mentioned in section 1.2.4 that the essential spectrum of (1.14), i.e., the spectrum corresponding to the difference equation

$$\mathbf{x}(t) = \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i) \tag{1.148}$$

associated with (1.14), determines the number of unstable poles of (1.14). Let c is defined by (1.51). If $c \geq 0$, there are infinitely many unstable poles, since the spectrum of (1.14) converges to the spectrum of (1.148) with the increasing magnitude of the poles. On the contrary, $c < 0$ indicates that the number of unstable system poles is finite. Therefore the stabilization task of a neutral system differs with respect to the value of c. If $c \geq 0$, the neutral system has to be stabilised in two steps. In the first step difference equation (1.148) has to be stabilised and in the second step the stabilization of the whole system is to be performed. The problem of stabilization of the neutral systems has been studied by Salamon, (1984), Pandolfi, (1976) and Hale and Verduyn Lunel (2002). Let the neutral system is considered in the form

$$\frac{d}{dt}\left[\mathbf{x}(t) - \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i)\right] = \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t - \tau) + \mathbf{B}\mathbf{u}(t) \tag{1.149}$$

In the first step it is necessary to achieve the stability of the difference equation

$$\mathbf{x}(t) = \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i) + \mathbf{B}\mathbf{u}(t) \tag{1.150}$$

Considering the feedback

$$u(t) = \sum_{i=1}^{N} \mathbf{F}_i \mathbf{x}(t - \eta_i) \tag{1.151}$$

yields

$$\mathbf{x}(t) = \sum_{i=1}^{N} \mathbf{H}_i \mathbf{x}(t - \eta_i) + \mathbf{B} \sum_{i=1}^{N} \mathbf{F}_i \mathbf{x}(t - \eta_i) \tag{1.152}$$

The characteristic matrix of (1.152) is given by

$$\mathbf{\Delta}_{0,\mathrm{cl}}(s) = \mathbf{\Delta}_0(s) - \mathbf{B} \sum_{i=1}^{N} \mathbf{F}_i \exp(-s\eta_i) \tag{1.153}$$

where

$$\mathbf{\Delta}_0 = \mathbf{I} - \sum_{i=1}^{N} \mathbf{H}_i \exp(-s\eta_i) \tag{1.154}$$

is the characteristic matrix of the difference equation associated with original system (1.149). Difference equation (1.152) is exponentially stable if and only if the roots of the equation $D(s) = \det[\mathbf{\Delta}_{0,\mathrm{cl}}(s)] = 0$ are located in the left half of the complex plane. In particular, a necessary condition for the stabilizability of (1.149) is given by

$$\mathrm{rank}[\mathbf{\Delta}_0(s), \mathbf{B}] = n, \ \forall s \in \mathbb{C}, \text{with } \mathrm{Re}(s) \geq -\varepsilon \tag{1.155}$$

for some $\varepsilon > 0$. The necessity of this condition has been studied in Salamon, (1984), for general difference equation and in O'Connor and Tarn, (1983a, 1983b) for systems with a single point delay. The condition is analogous to the controllability condition for retarded systems $\mathrm{rank}[s\mathbf{I} - \mathbf{A}(s), \mathbf{B}(s)] = n, \ \forall s \in \mathbb{C}, \text{with } \mathrm{Re}(s) \geq -\varepsilon$, from which condition (1.133) results. Considering the concept of strong stability of the neutral systems, condition (1.155) is only necessary condition but not the sufficient due to the sensitivity of the stability properties of difference equations with variations in the delays. Condition (1.155) is also sufficient in case that the delays are commensurable (Salamon, 1984) and by an appropriate choice of the control matrices $\mathbf{F}_1$, $\mathbf{F}_2$, ..., $\mathbf{F}_N$, the decay rate of system (1.152) can be made arbitrarily fast. If the delays η_i are rationally independent, the problem is more complicated. The aim of applying feedback (1.151) is to make the difference equation strongly stable, i.e., to make γ_0<1, where γ_0 is defined by (1.68). According to Hale and Verduyn Lunel (2002), difference equation (1.152) is strongly stable, i.e., independently on the variations in the delays, if and only if for any choice of the delays $\eta_1, \eta_2, .., \eta_N$ the roots of the determinant

$$\det[\mathbf{\Delta}_{0,\mathrm{cl}}(s)] = \det\left[\mathbf{I} - \sum_{i=1}^{N} (\mathbf{M}_i + \mathbf{B}\mathbf{F}_i) \exp(-s\eta_i)\right] \tag{1.156}$$

are in the open half-plane $\{\lambda_i \in \mathbb{C}, \mathrm{Re}(\lambda_i) \leq -\varepsilon < 0\}$ for some positive ε depending on the values of $\eta_1, \eta_2, .., \eta_N$. Thus a necessary condition for strong stabilizability of (1.156) is that, for any choice of the delays $\eta_1, \eta_2, .., \eta_N$

$$\operatorname{rank}\left[\mathbf{I}-\sum_{i=1}^{N}\mathbf{M}_i\exp(-s\eta_i),\mathbf{B}\right]=n,\ \forall s\in\mathbb{C},\text{with Re}(s)\geq -\varepsilon \tag{1.157}$$

for some positive $\varepsilon=\varepsilon(\eta_1,\eta_2,..,\eta_N)$. Condition (1.157) is not only sufficient but also necessary for strong stabilizability of (1.150), for proof see Hale and Verduyn Lunel (2002).

The second step in the stabilization of (1.149) is to assure that any of the poles will not remain in the right half of the complex plane. Performing the first step of the stabilization, i.e., stabilizing the difference equation associated to (1.149), there is finite number of poles located to the right from the stability boundary. Involving the velocity feedback from the state variables that is designed to stabilize (1.150), the feedback acquires the following form

$$u(t)=\sum_{i=1}^{N}\mathbf{F}_i\mathbf{x}(t-\eta_i)-\int_0^T d\mathbf{K}(\tau)\mathbf{x}(t-\tau) \tag{1.158}$$

The feedback system is then given by

$$\frac{d}{dt}\left[\mathbf{x}(t)-\sum_{i=1}^{N}\left(\mathbf{H}_i+\mathbf{B}\mathbf{F}_i\right)\mathbf{x}(t-\eta_i)\right]=\int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau)-\mathbf{B}\int_0^T d\mathbf{K}(\tau)\mathbf{x}(t-\tau) \tag{1.159}$$

with the characteristic matrix

$$\mathbf{\Delta}_{\text{cl}}(s)=s\mathbf{\Delta}_{0,\text{cl}}(s)-\int_0^T\exp(-s\tau)d[\mathbf{A}(\tau)-\mathbf{B}\mathbf{K}(\tau)] \tag{1.160}$$

System (1.149) is exponentially stable if and only if the roots of the characteristic equation $M_{\text{cl}}(s)=\det[\mathbf{\Delta}_{\text{cl}}(s)]=0$ are located in the left half of the complex plane. In particular, a necessary condition for stabilizability is given by (1.157) and

$$\operatorname{rank}[\mathbf{\Delta}(s),\mathbf{B}]=n,\ \forall s\in\mathbb{C},\text{with Re}(s)\geq -\varepsilon \tag{1.161}$$

for some $\varepsilon>0$. where

$$\mathbf{\Delta}(s)=s\left[\mathbf{I}-\sum_{i=1}^{N}\mathbf{M}_i\exp(-s\eta_i)\right]-\int_0^T\exp(-s\tau)d\mathbf{A}(\tau) \tag{1.162}$$

The necessity of these conditions and the fact that these conditions (1.157) and (1.161) are independent has been claimed in Salamon, (1984).

Let us suppose that the matrices $\mathbf{F}_1$, $\mathbf{F}_2$, ..., $\mathbf{F}_N$ are chosen such that difference equation (1.152) is exponentially stable and independent of the delays. This implies that the feedback system has a finite number of unstable poles that should be shifted to stabilize the feedback system. To sum up, condition (1.157) is necessary and sufficient for strong stabilization of the difference equation of closed loop neutral equation (1.159) and the second condition is necessary and sufficient to stabilize the remaining finitely many unstable modes. Since the retarded part of (1.159) is robust with variations in the delays and the associate difference equation of (1.159) is strongly stable, it follows that the stability properties of the closed-loop system (1.159) are robust with small variations in the delays.

2. OBJECTIVES OF THE THESIS

The main topics of this thesis are the frequency domain based analysis and control of time delay systems. Particularly, the stress will be laid on the methods of synthesis and control design based on the knowledge of the dominant part of the spectrum of the system poles and zeros. As has been mentioned in chapter 1, the applicability of the available algorithms is often restricted on a narrow class of systems. For example, the method for computing the approximation of the rightmost poles using either the discretization of the infinitesimal generator of the semigroup or the discretization of the solution operator (described in sections 1.2.7-1.2.9) can be applied only to the retarded systems, preferably only with the lumped delays. The distribution of the poles in the complex plane determines the stability and the modes of the dynamics. However, the distribution as such is not decisive in determining the significance of the modes of the dynamics. The significance of the modes is determined by the distribution of the system zeros. Thus, not only the distribution of the system poles but also the distribution of the system zeros should be known to evaluate completely the system dynamics.

- **Objective 1**. The primary objective of this thesis is to develop an algorithm for computing both the system poles and zeros of TDS located in a chosen region of the complex plane. The algorithm should provide the possibility to solve the task for large class of systems, both retarded and neutral. Considering that both the system poles and zeros are the solutions of the quasipolynomial equations, the algorithm is to be based on computing the roots of quasipolynomials. Since the quasipolynomials (as well as polynomials) tend to be ill conditioned as its degree increases, the robustness of the algorithm is to be investigated. The result of the investigation should be the definition of the class of TDS for which the algorithm may be successfully used. This objective of the thesis will be the most important one. The other objectives are chosen in order to verify the features of the algorithm that will be designed.

- **Objective 2**. The modeling approach involving the delays allows the real plants to be described by considerably lower order models (if the order is considered as the number of differential equations) than if a modeling approach based on the delay free models is used. Thus, the second objective of this thesis is the investigation of the features of the low order models as the basic element units in building the plant model. The investigation should result in the mapping of the distribution of the poles and zeros of the low order TDS. The other result of this part of the thesis should be the choice of the suitable structure of the low order model able to fit the dynamics of a wide class of the SISO systems.

- **Objective 3**. The third objective is motivated by the fact that the evaluation of the significance of the poles in the infinite spectra based on evaluating the distances of the poles from the stability boundary is sufficient only if the stability of the system is evaluated. If the character of the system input-output dynamics is to be evaluated, the criterion is insufficient. Also the other criterion used to evaluate the significance of the system poles based on the magnitudes of the poles does not provide satisfactory evaluation of the pole significance. Thus, the third objective of this thesis is to define a group of system poles (from the infinite spectrum) that determine the system input-output dynamics. Therefore, I am going to try and find a criterion that will truly evaluate the significance of the system poles.

- **Objective 4**. In section 1.3 I have explained some of the methods for control design of TDS. As the last topic of the thesis, the methods based on the pole placement using the proportional feedback from the state variables are to be investigated. The stress should to be laid on comparing the method based on the direct prescription of the poles with the method known as continuous pole placement. Both the control feedback design methods are also to be extended to the class of neutral systems. As the result of this part of the thesis, the suitable strategy for pole placement applied to TDS is to be proposed.

In order to summarize the contributions that will be achieved in this thesis and to demonstrate the applicability of the methods that will be developed, the methods are to be applied for analyzing the dynamics of a real (laboratory) plant. The control methods based on the continuous and the direct pole placement are to be applied to control the plant as well.

3. ALGORITHMS FOR COMPUTING QUASIPOLYNOMIAL ROOTS

3.1 Introduction

According to the first objective of this thesis stated in chapter 2, in this chapter, I am going to introduce and explain two algorithms for computing the roots of quasipolynomials that I have developed. The first algorithm is based on the modification of Weyl's algorithm (Pan, 1997), which is one of the basic algorithms for locating the roots of polynomials. The modification of the algorithm consist in the use of the argument principle based test instead of the proximity test, which is used in the original algorithm. This modification allows Weyl's algorithm to be used also for computing the roots of quasipolynomials. The second algorithm for computing the quasipolynomial roots that I am going to introduce in this chapter, which is original in fact and more powerful than Weyl's construction based algorithm, is based on mapping the quasipolynomial function in the complex plane. Also the features of the algorithms will be studied and several application examples will be included.

3.2 Argument principle based algorithm

Let us consider a quasipolynomial characteristic function of TDS in the following form

$$M(s) = \sum_{i=0}^{n} s^i Q_i(s) \tag{3.1}$$

where $Q_i(s)$, i=1..n are the functions involving the terms corresponding to the distribution of the system delays. It has been mentioned that it is not an easy task to compute the roots of the polynomials, especially if its order is higher than four. Naturally, the issue is even more difficult if the roots of a quasipolynomial are to be computed.

Some of the powerful algorithms for computing the roots of polynomials use the iteration methods based on iterative computing the eigenvalues of a companion matrix of the polynomial. The task of computing the polynomial roots is changed into the task of computing the eigenvalues of the accompanying matrix, see, e.g., Fortune, (2001). Since the companion matrix of quasipolynomial (3.1) is a functional matrix, such an approach is not suitable for computing the roots of quasipolynomials. Weyl's geometric construction based algorithm is the alternative way of computing the polynomial roots, see, e.g., Wilf, (1978) or Pan, (1997) and the references therein. (In fact, it is one of the first advanced numerical algorithms used for computing the roots of polynomials). Weyl's construction, also known as quadtree construction, is a two-dimensional version of the bisection algorithm of a line interval, see Fig. 3.1. The basic idea of the construction is the following. On the complex plane, the search for the roots starts with an initial suspect region $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$, containing all the polynomial roots. Then the region is partitioned into four congruent subregions. At the centre of each of them, the proximity test is performed (Henrici, 1974), i.e., a distance of the closest root from the centre is estimated. If this distance exceeds the length of the diagonal of the subregion then the subregion does not contain any roots and is discarded. If the result is opposite, the subregion is called suspect and undergoes the same recursive process of partitioning into four subregions. Then the proximity tests are performed at the centres of those smaller subregions. The polynomial roots lying in each suspect region are approximated by its centre with errors bounded by the half-length of its diagonal. Thus, in k iteration steps, the approximation errors cannot exceed $0.5\mathrm{diag}(\mathcal{D})/2^k$, where $\mathcal{D}$ represents the actual suspect region.

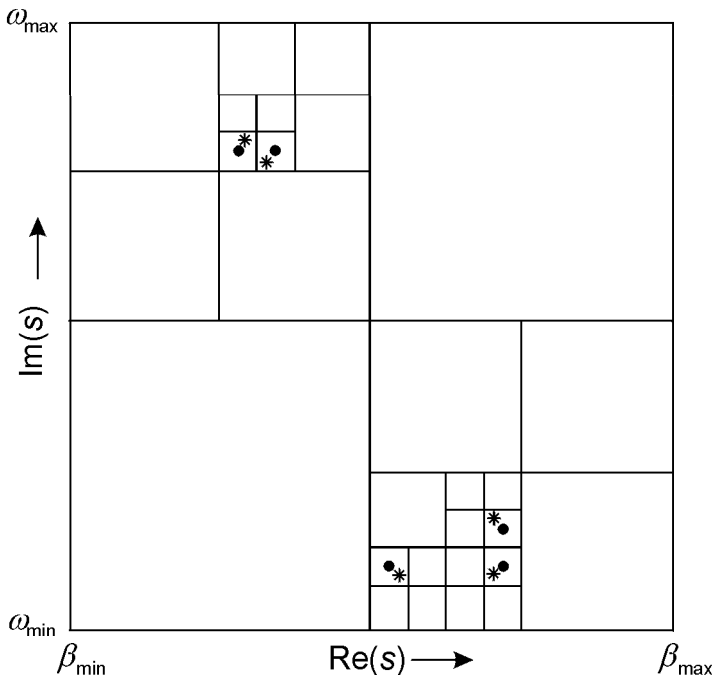

Fig. 3.1 Locating the roots by Weyl's (quadtree) algorithm
asterisks - polynomial roots, dots - approximation of the polynomial roots

Let us modify the algorithm in order to compute the roots of a quasipolynomial $M(s)$ located inside the suspect region $\mathcal{D}$. Instead of the proximity test let us use the argument principle, which holds for any analytic function including quasipolynomials, (El'sgol'ts and Norkin, 1973), and by means of which it is possible to compute the number of roots in a region. Let $\mathcal{D}$ is a domain in the complex plane whose boundary φ is a closed, Jordan curve, i.e., it is of the finite length and has no multiple points. Let $G(s)\in\mathbb{C}$ is a meromorphic function, i.e., a single valued analytic function that has no singular points beside poles. Then

$$\frac{1}{2\pi j}\int_{\varphi}\frac{G'(s)}{G(s)}ds = N_{\mathcal{D}} - P_{\mathcal{D}} \tag{3.2}$$

and

$$\frac{1}{2\pi}\Delta_{\varphi}\arg G(s) = N_{\mathcal{D}} - P_{\mathcal{D}} \tag{3.3}$$

where $N_{\mathcal{D}}$ is the total number of zeros of the function G in $\mathcal{D}$, $P_{\mathcal{D}}$ is the total number of poles of the function G in $\mathcal{D}$, both counting their multiplicities and $G'(s) = dG(s)/ds$.

The principle of argument is based on the relation between the increments of the logarithm and the argument of a function G along a function $\varphi:[a,b]\to\mathbb{C}$, i.e.,

$$\Delta_{\varphi}\arg G(s) = \mathrm{Im}[\Delta_{\varphi}\ln G(s)] \tag{3.4}$$

where

$$\Delta_{\varphi}\ln G(s) = \ln\left|\frac{G(\varphi(b))}{G(\varphi(a))}\right| + j\Delta_{\varphi}\arg G(s) = \int_{\varphi}\frac{G'(s)}{G(s)}ds \tag{3.5}$$

If the function φ is closed, then

$$\Delta_\varphi \arg G(s) = k_1 2\pi \quad \text{and} \quad \Delta_\varphi \ln G(s) = k_2 2\pi\mathrm{j} \tag{3.6}$$

where k_1 and k_2 are integers and since $\varphi(a) = \varphi(b)$

$$\Delta_\varphi \ln G(s) = \mathrm{j}\Delta_\varphi \arg G(s) \tag{3.7}$$

Obviously, if formulas (3.2) or (3.3) are applied to a quasipolynomial $M(s)$, the result is the number of $M(s)$ zeros in $\mathcal{D}$. Although the number of $M(s)$ zeros in $\mathcal{D}$ can be computed from (3.2), such an approach is only suitable for low order quasipolynomials with not too complicated exponential terms Q_i in (3.1). From the numerical point of view, it should not be a problem to perform the integration in (3.2) using a method based on a trapezoidal or quadrature summation rule. However, such a numerical evaluation of the integral of M'/M is rather computationally and memory demanding. It may result in an unacceptably long computational time if the term M'/M is too complicated. The advantage of such a numerical computation of (3.2) is that the result is directly the number of the roots and no algorithm for graphical evaluation of the argument is to be performed as in case of using formula (3.3).

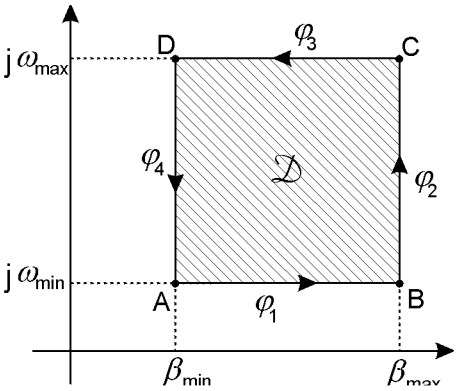

Fig. 3.2 Region $\mathcal{D}$ in s-plane

Consider the region $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$. The region is enclosed by four line-segments, see Fig. 3.2

$$\left.\begin{aligned} \varphi_1(\beta) &= \beta + \mathrm{j}\omega_{\min} \\ \varphi_2(\omega) &= \beta_{\max} + \mathrm{j}\omega \\ \varphi_3(\beta) &= -\beta + \beta_{\min} + \beta_{\max} + \mathrm{j}\omega_{\max}, \\ \varphi_4(\omega) &= \beta_{\max} + \mathrm{j}(-\omega + \omega_{\max} + \omega_{\min}) \end{aligned}\right\} \begin{aligned} &\beta \in [\beta_{\min}, \beta_{\max}] \\ &\omega \in [\omega_{\min}, \omega_{\max}] \end{aligned} \tag{3.8}$$

The number of $M(s)$ roots inside the region $\mathcal{D}$ is given by the formula

$$\begin{aligned} N_{\mathcal{D}} = \frac{1}{2\pi\mathrm{j}} \Bigg[&\int_{\beta_{\min}}^{\beta_{\max}} \frac{dM(\varphi_1(\beta))}{d\beta} \frac{1}{M(\varphi_1(\beta))} d\beta + \int_{\omega_{\min}}^{\omega_{\max}} \frac{dM(\varphi_2(\omega))}{d\omega} \frac{1}{M(\varphi_2(\omega))} d\omega + \dots \\ \dots + &\int_{\beta_{\min}}^{\beta_{\max}} \frac{dM(\varphi_3(\beta))}{d\beta} \frac{1}{M(\varphi_3(\beta))} d\beta + \int_{\omega_{\min}}^{\omega_{\max}} \frac{dM(\varphi_4(\omega))}{d\omega} \frac{1}{M(\varphi_4(\omega))} d\omega \Bigg] \end{aligned} \tag{3.9}$$

The alternative way of computing the number of $M(s)$ roots in $\mathcal{D}$ is based on a graphical evaluation of formula (3.3) which allows $N_{\mathcal{D}}$ to be evaluated from the variation of the argument $\Phi(s) = \arg(M(s))$

$$\tan(\Phi(s)) = \frac{\mathrm{Im}(M(s))}{\mathrm{Re}(M(s))} \tag{3.10}$$

as s moves once around the boundary of $\mathcal{D}$ in the counter-clockwise sense. Considering the features of the trigonometric function arctangent, the argument Φ results in the form of

$\mathrm{mod}(\pi/2)$, i.e., within -π/2 and π/2. The number of roots $N_{\mathcal{D}}$ results from the number of times $\Phi(s)$ crosses the value π/2 or -π/2 as s varies around the contour φ, see Fig 3.3, considering the course of the function. The algorithm of evaluating the change of $\Phi(s)$ as s moves around the boundary of the region $\mathcal{D}$ is quite simple and easy to implement. Starting from a certain arbitrarily chosen point on φ, e.g., A, see Fig. 3.2, and varying s continuously in the counter-clockwise sense, the argument changes piecewise continuously (piecewise continuity is due to φ, which is also piecewise continuous, see Fig. 3.2) until it reaches π/2 (or -π/2). Then the argument changes discontinuously to -π/2 (or π/2) and it changes piecewise continuously again until it reaches one of the boundary values ±π/2. In this way the argument is evaluated around φ, passing the points B, C and D and coming back to the starting point A, see Fig. 3.2. A set of curves starting and ending at ±π/2 is obtained as the result of the argument evaluation providing the argument change π, -π or 0, see Fig. 3.3. In order to obtain the final $\Delta_{\varphi}\Phi(s)$ the changes in arguments on these curves, i.e., π, -π or 0, have to be summed. If a curve starts at -π/2 and ends at π/2, the contribution to $\Delta_{\varphi}\Phi(s)$ is π. Conversely, if a curve starts at π/2 and ends at -π/2, the contribution to $\Delta_{\varphi}\Phi(s)$ is -π. If the curve starts and ends at π/2 (-π/2), the contribution to $\Delta_{\varphi}\Phi(s)$ is 0. Obviously, the final change of the argument, whose variation along the contour φ is shown in Fig. 3.3, is $\Delta_{\varphi}\Phi(s) = 2\pi$. It indicates that there is one root of $M(s)$ in the region $\mathcal{D}$., $N_{\mathcal{D}}$=1.

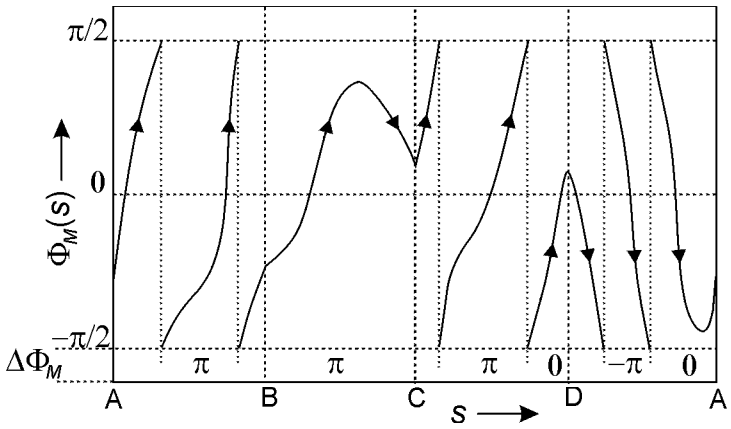

Fig. 3.3 Variation of $\Phi(s)$ along the closed contour φ, see Fig. 3.2

The presented algorithm for computing $N_{\mathcal{D}}$ based on (3.3), which has also been used in Luzyanina and Roose, (1996) to detect bifurcation points for delay differential equations, is more suitable for computer implementation than the algorithm based on (3.2) due to its distinctly lower number of numerical operations to be performed.

Applying the argument principle first to the region $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$, see Fig. 3.1, the number of roots of $M(s)$ in this prime suspect region is obtained. Than dividing the region into four subregions $\mathcal{D}_{\mathrm{TL}}$, $\mathcal{D}_{\mathrm{TR}}$, $\mathcal{D}_{\mathrm{BL}}$ and $\mathcal{D}_{\mathrm{BR}}$ (TL= top-left, TR= top-right, BL= bottom-left, BR= bottom-right), the number of $M(s)$ roots is computed in each of the subregions. If a subregion contains some roots, it is called suspect and it is divided into four subregions again to which the argument based test is applied. The whole algorithm of dividing the suspect subregions is performed recursively until the required accuracy of the locations of the roots is achieved. The approximate position of a root is considered as the position of the centre of the smallest suspect region, in which the root has been located.

The error of the root approximation is bounded by the half-length of the diagonal of the smallest suspect subregion. If no root is located in a subregion, the recursive algorithm is stopped on this part of the complex plain. It is advisable to check whether or not the number of roots located in a region agrees with the sum of the numbers of roots located in its subregions.
Non-satisfying this condition indicates a failure of the algorithm and may result in omitting some roots in $\mathcal{D}$.

Although the algorithm of recursive splitting the suspect regions is quite simple, its computer implementation requires an elaborate approach performing the recursive operations. Obviously, in the program implementation, there should be subroutines that perform the algorithm for dividing the suspect regions and the algorithm for locating the roots. The more difficult problem to solve is to decide in which way the regions are to be checked and how to assure that no piece of the prime suspect region will remain unchecked. The possible solution of the problem is to store the information of checked subregions (i.e., whether or not the particular subregion contains any roots and also their number) in a data structure corresponding to a general tree (nonlinear data structure used in database systems, see, e.g., Trembley and Bunt, (1989), see Fig. 3.4. The root of the tree is the prime region $\mathcal{D}$. The root has four branches, each of them corresponding to one of the subregions $\mathcal{D}_{TL}$, $\mathcal{D}_{TR}$, $\mathcal{D}_{BL}$ or $\mathcal{D}_{BR}$. If the particular subregion does not contain any roots, the corresponding branch is marked with the stop leaf, which denotes that the partitioning of the region is not to be performed. If there are any roots in the region, their number is written in the corresponding branch node from which four branches grow corresponding to four parts of the partitioned region. In this way, the algorithm is performed recursively until the regions with diagonals less than required maximum error of the approximation result. The branches corresponding to these smallest regions are marked either with the stop leaves or with root leaves depending on whether or not they contain any roots. Also the number of roots in the smallest regions (denoting the multiplicity of the final root approximations) are written in the root-leaves.

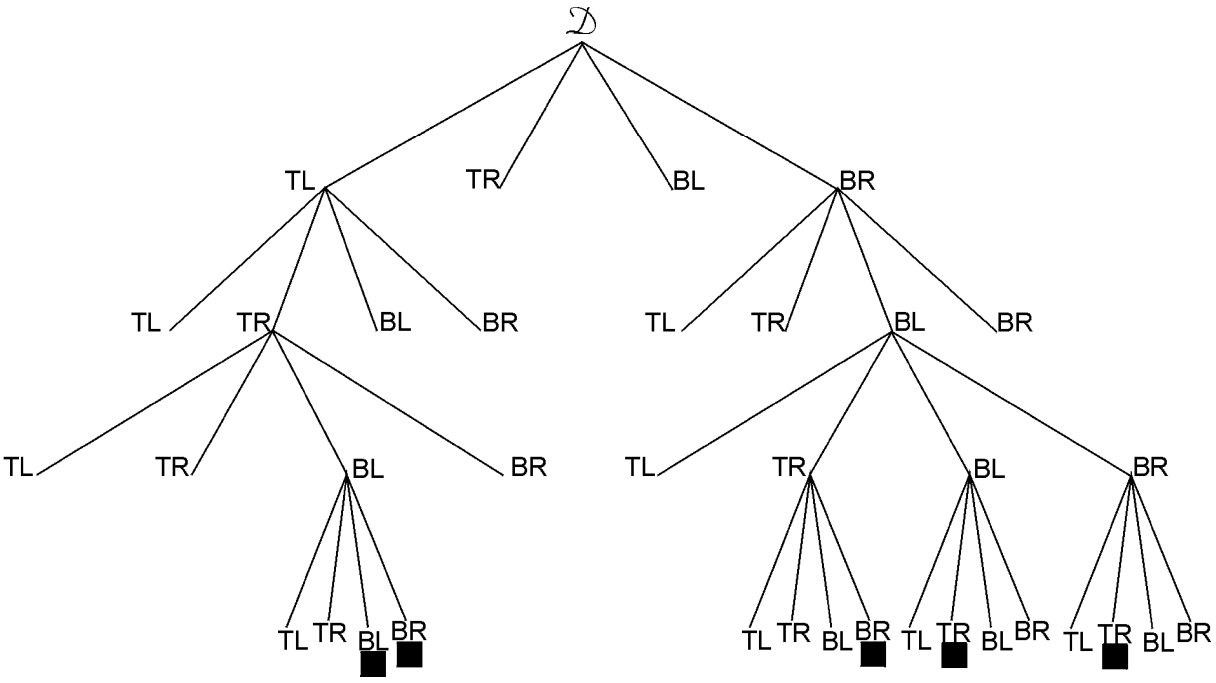

Fig. 3.4 Tree representing recursive process of Weyl's construction applied to region seen in Fig. 3.1

The algorithm is performed until all the branches of the tree are marked either with the stop leaves or with root leaves. Such a tree representing the recursive process in locating the roots in region $\mathcal{D}$ from Fig. 3.1 is shown in Fig. 3.4 where the root leaves are marked by black squares. Each part of the prime region can be identified using, e.g., the dot notation. For example the root leaves of the tree in Fig. 3.4, are identified as follows (in the left-right direction): $\mathcal{D}_{\text{TL.TR.BL.BL}}$, $\mathcal{D}_{\text{TL.TR.BL.BR}}$, $\mathcal{D}_{\text{BR.BL.TR.BR}}$, $\mathcal{D}_{\text{BR.BL.BL.TR}}$ and $\mathcal{D}_{\text{BR.BL.BR.TR}}$. The application of the described rootfinding algorithm based on Weyl's construction and the argument principle will be shown in the following Example 3.1.

Example 3.1

Let us find the roots of the quasipolynomial

$$M(s) = 0.5s^3 + 2\exp(-s)\,s^2 + 2.5\exp(-2s)\,s + \exp(-3s) \tag{3.11}$$

located in the region $\mathcal{D} = [-1, 1]\times[0, 2]$. Applying Weyl's construction combined with argument based test, the following results are obtained, see Fig.3.5 - Fig. 3.12.

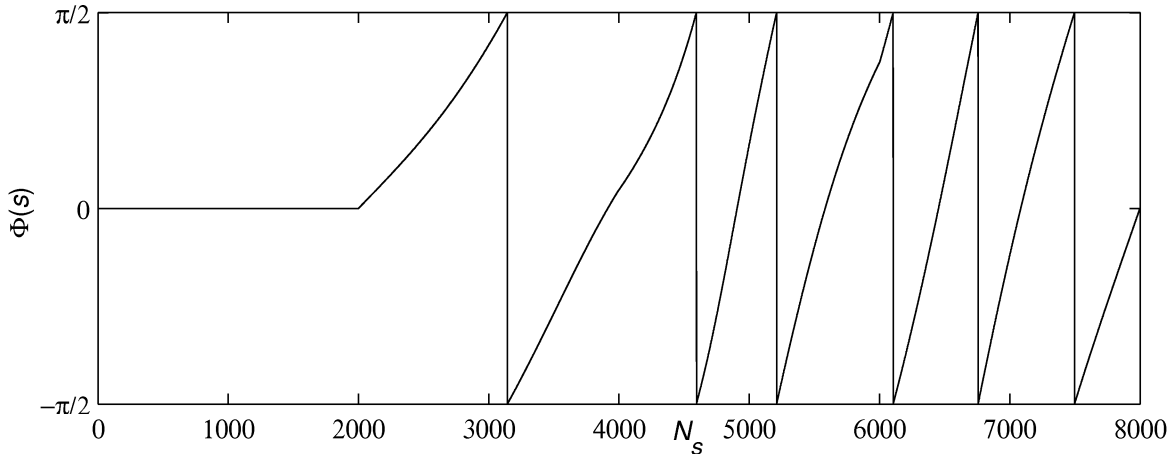

Fig. 3.5 Variation of $\Phi(s)$ along the boundary of $\mathcal{D} = [-1, 1]\times[0, 2]$, $\Delta\Phi(s) = 6\pi$, $N_{\mathcal{D}} = 3$

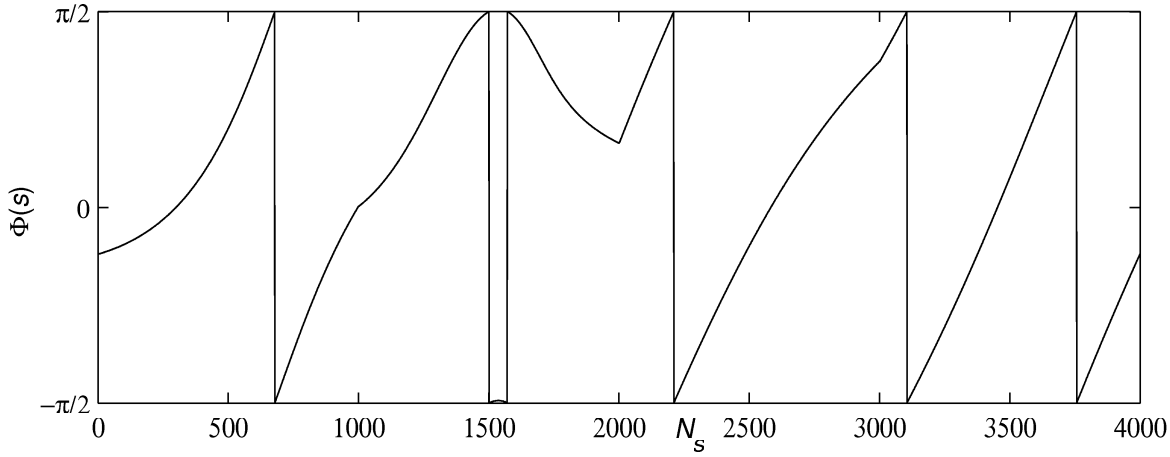

Fig. 3.6 Variation of $\Phi(s)$ along the boundary of $\mathcal{D}_{\text{TL}} = [-1, 0]\times[1, 2]$, $\Delta\Phi(s) = 4\pi$, $N_{\mathcal{D}_{\text{TL}}} = 2$

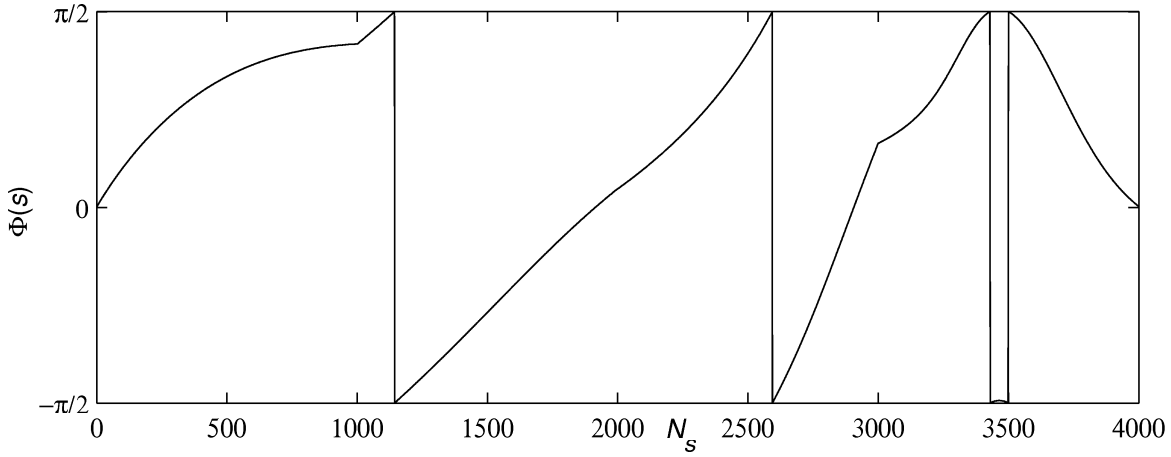

Fig. 3.7 Variation of $\Phi(s)$ along the boundary of $\mathfrak{D}_{\mathrm{TR}} = [0, 1]\times[1, 2]$, $\Delta\Phi(s) = 2\pi$, $N_{\mathfrak{D}_{\mathrm{TR}}} = 1$

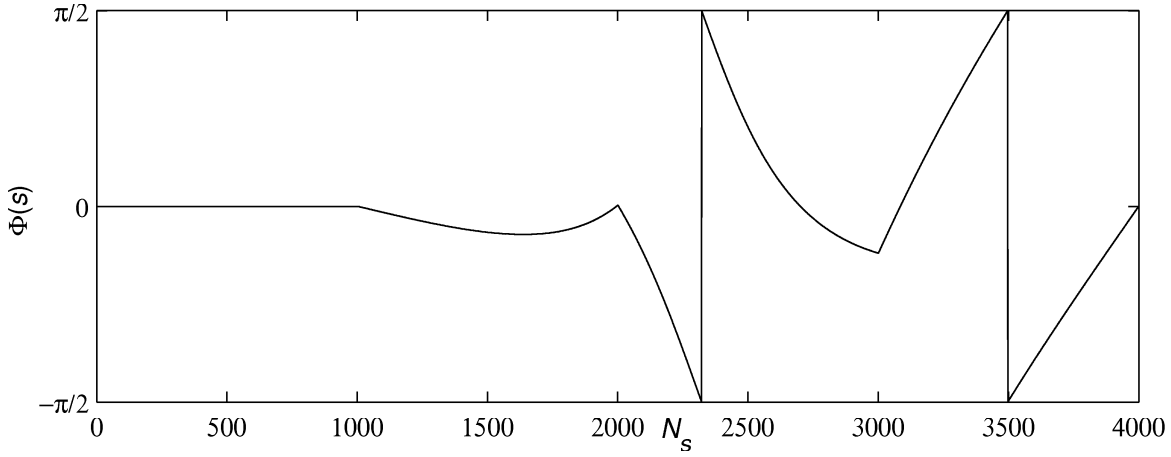

Fig. 3.8 Variation of $\Phi(s)$ along the boundary of $\mathfrak{D}_{\mathrm{BL}} = [-1, 0]\times[0, 1]$, $\Delta\Phi(s) = 0$, $N_{\mathfrak{D}_{\mathrm{BL}}} = 0$

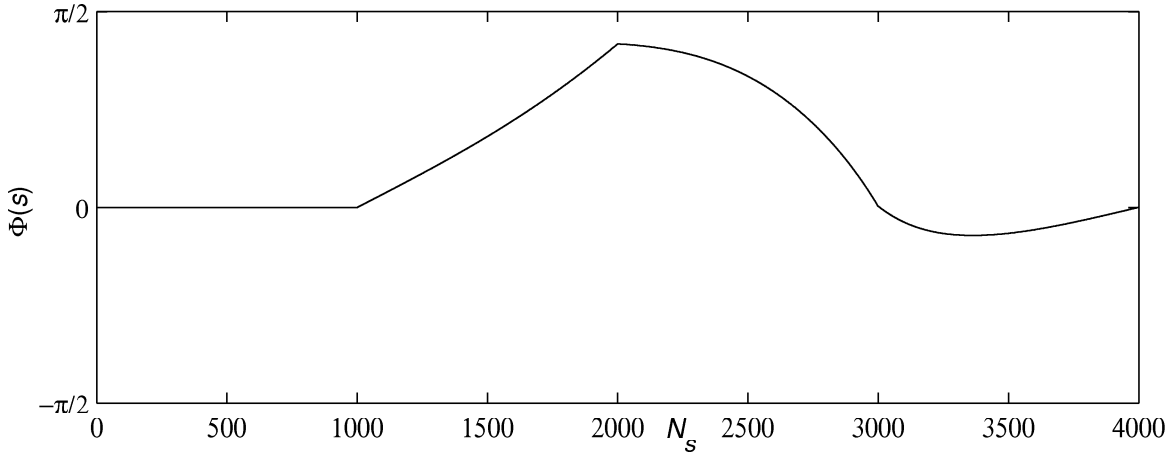

Fig. 3.9 Variation of $\Phi(s)$ along the boundary of $\mathfrak{D}_{\mathrm{BR}} = [0, 1]\times[0, 1]$, $\Delta\Phi(s) = 0$, $N_{\mathfrak{D}_{\mathrm{BR}}} = 0$

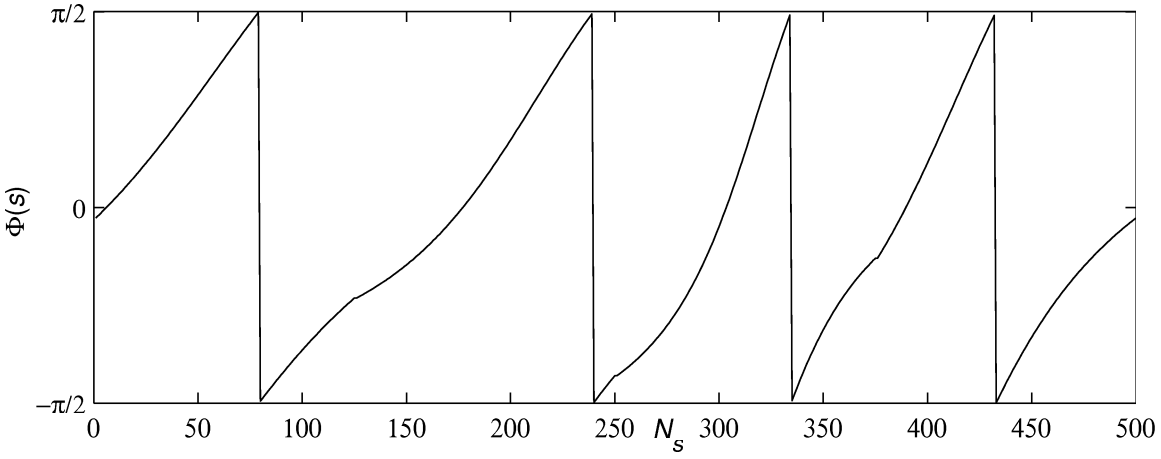

Fig. 3.10 Variation of $\Phi(s)$ along the boundary of $\mathcal{D}_{\text{TL.BR.TL.BR}} = [-0.375, -0.25]\times[1.25, 1.375]$, $\Delta\Phi(s) = 4\pi$, $N_{\mathcal{D}_{\text{TL.BR.TL.BR}}} = 2$

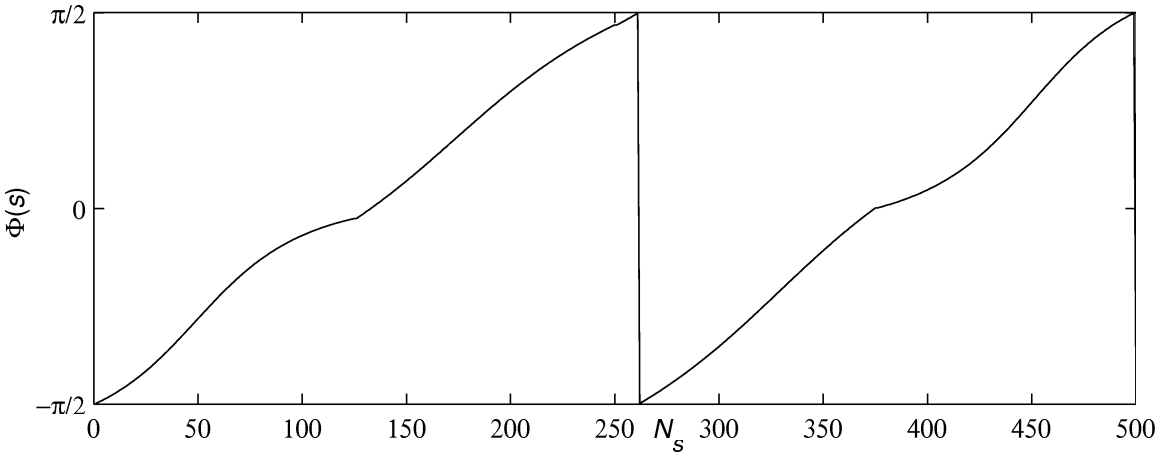

Fig. 3.11 Variation of $\Phi(s)$ along the boundary of $\mathcal{D}_{\text{TR.TL.BL.TL}} = [0.125, 0.25]\times[1.625, 1.75]$, $\Delta\Phi(s) = 2\pi$, $N_{\mathcal{D}_{\text{TR.TL.BL.TL}}} = 1$

In order to evaluate the argument along the boundary of the region $\mathcal{D}$, i.e., along φ given by (3.8), consider a mesh of points on φ with the increment Δs. Thus, the argument is to be evaluated for the points $N_{s,i}$, $i = 1..2(\text{int}((\beta_{\max} - \beta_{\min})/\Delta_s) + \text{int}((\omega_{\max} - \omega_{\min})/\Delta_s) + 2)$ of φ. Let us chose $\Delta s = 0.001$ to evaluate the argument increment of (3.11) along the boundary of prime region $\mathcal{D} = [-1, 1]\times[0, 2]$, and its subregions. As can be seen in Fig. 3.5, the argument increment along the boundary φ of $\mathcal{D}$ is $\Delta_\varphi \Phi(s) = 6\pi$, i.e., $N_{\mathcal{D}} = 3$. It means, according to (3.3) (quasipolynomial (3.11) does not have any poles, $P_{\mathcal{D}} = 0$), that there are three zeros (roots) of (3.11), in $\mathcal{D}$. Partitioning the region into $\mathcal{D}_{\text{TL}}$, $\mathcal{D}_{\text{TR}}$, $\mathcal{D}_{\text{BL}}$ and $\mathcal{D}_{\text{BR}}$, it can be seen in Fig. 3.6 -Fig. 3.9, that there are 2 roots in $\mathcal{D}_{\text{TL}}$ since $\Delta_{\varphi_{TL}} \Phi(s) = 4\pi$, 1 root in $\mathcal{D}_{\text{TR}}$ since $\Delta_{\varphi_{TR}} \Phi(s) = 2\pi$, and no roots in $\mathcal{D}_{\text{BL}}$ and $\mathcal{D}_{\text{BR}}$, since $\Delta_{\varphi_{BL}} \Phi(s) = \Delta_{\varphi_{BR}} \Phi(s) = 0$. Consequently, in the second step, the regions $\mathcal{D}_{\text{TL}}$, $\mathcal{D}_{\text{TR}}$ are partitioned into four subregions and in each of them the argument based test is performed.

Applying this modified recursive Weyl's construction, finally, two roots are located in

$$\mathcal{D}_{\text{TL.BR.TL.BR}} = [-0.375, -0.25] \times [1.25, 1.375]$$

see Fig. 3.10, and one root in

$$\mathcal{D}_{\text{TR.TL.BL.TL}} = [0.125, 0.25] \times [1.625, 1.75]$$

Provided that such approximation of the roots is sufficient, the roots of quasipolynomial (3.11) are given by

$$s_{1,3} = (-0.3125 + \text{j}1.3125) \pm 0.0625(1 \pm \text{j})$$
$$s_5 = (0.1875 + \text{j}1.6875) \pm 0.0625(1 \pm \text{j})$$

i.e., the root approximations are located in the centres of $\mathcal{D}_{\text{TL.BR.TL.BR}}$ and $\mathcal{D}_{\text{TR.TL.BL.TL}}$, and the maximum approximation error is given by the half of their diagonals. The partitioning process of the prime region with the final position of the approximations of the roots can be seen in Fig. 3.12.

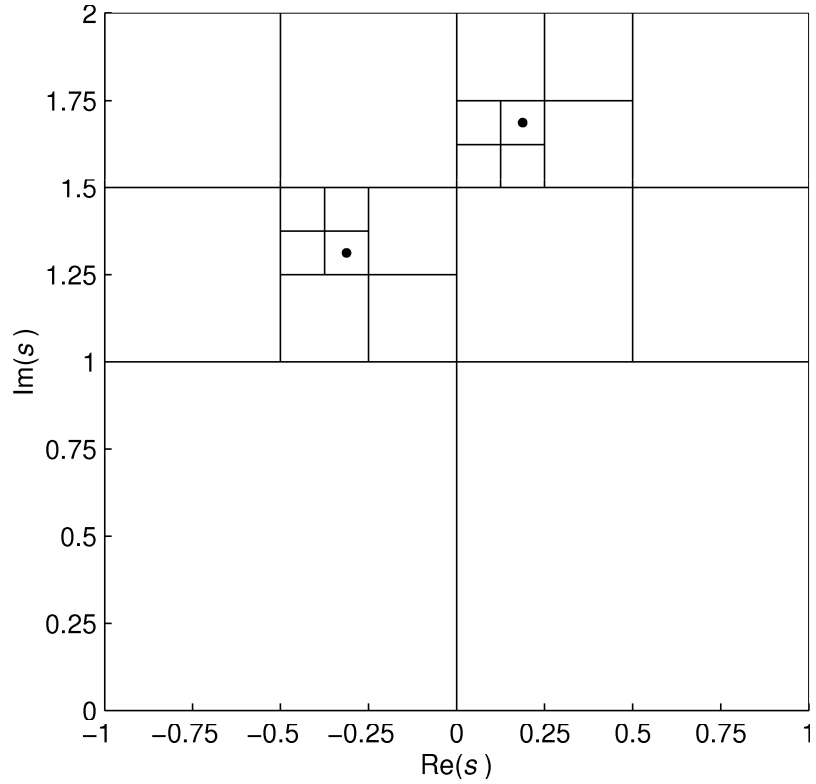

Fig. 3.12 Weyl's construction applied to (3.11)

As can be seen, there are two roots located in the region $\mathcal{D}_{\text{TL.BR.TL.BR}}$. However, the character of this couple remains unknown. The roots can be either distinct or in the form of double root. Further carrying on Weyl's algorithm could be used to solve the problem. However, only distinctness of the roots can be confirmed after some iterations. The multiplicity of a root cannot be proved in this way, because the result is always given with an approximation error. On the other hand, from the practical point of view, it does not really matter if two or several roots are multiple or distinct but located within small mutual distances (considering the physical meaning of the roots, i.e., they represent the system poles or zeros).

3.3 Computing the root approximations using Newton's method

To sum up, the designed modification of Weyl's construction seems to be a suitable tool to locate the approximate positions of the roots located in a part of the complex plane. On the other hand, its convergence speed, given by the linear convergence rate, is relatively low. It is possible to accelerate the iteration process, for example by using Newton's method. This numerical method guarantees the quadratic convergence from a starting point of the iteration s_0 to a root of $M(s)$. Let $M(s)$ be an analytic function on a region $\mathcal{D}$, then

$$s_{i,k+1} = s_{i,k} - \frac{M(s_{i,k})}{M'(s_{i,k})} \tag{3.12}$$

where s_i is i^{th} root of $M(s)$, $M'(s) = dM(s)/ds$, and k=0,1,..., is the step of Newton's iteration. The accuracy of the i^{th} root approximation can be indirectly prescribed by the value ε_N. The Newton's iteration stops if

$$\varepsilon_N \geq \varepsilon_{i,k}, \quad \varepsilon_{i,k} = \left| s_{i,k} - s_{i,k-1} \right| \tag{3.13}$$

Since Newton's method iterates in a quadratic rate, it can be assumed that the approximation error is close to ε_N but probably not less than ε_N.

The problem in applying Newton's method consists in the requirement that the starting point of the iteration is to be located close to the root being approximated. The procedure of Newton's iteration can be unstable near a horizontal tangent or a local minimum of $M(s)$. However, with a good choice of the initial root's position, the algorithm is likely to converge, see, e.g., Rektorys, (1994). Using Newton's algorithm is also risky if not only one but several poles are close to the starting point of the iterations. In such a case, depending on the character of $M(s)$, the method may incorrectly result in the multiple roots whereas some of the roots are not located. To avoid such a failure, Weyl's construction should be carried out until the sufficient distinctions of the root approximations are achieved, particularly in the subregions with high density of the roots. If any of the roots are multiple or close to multiple roots, it is advisable to carry out Weyl's construction in the subregion where the roots are located until the required accuracy of the root approximations are achieved. The convergence features of both the methods will be shown and compared in Example 3.2

Example 3.2

Let us compare the convergence features of Newton's method and Weyl's algorithm. Let us use the algorithms for approximating root s_5 of $M(s)$ given by (3.11) starting from $s_{5,0}$=0.5+j1.5. As can be seen in Fig. 3.13 and Fig 3.14, respectively, the convergence speed of both the algorithms is more or less equivalent at the steps 1-4 of the iterations, in which the approximations get very close to the roots being approximated. However, as can be seen in Tab. 3.1, the convergence of Newton's method is much faster in the further steps than the convergence of Weyl's construction. Obviously, employing Newton's method is reasonable if the approximation having resulted from Weyl's algorithm gets close to the root of $M(s)$. Weyl's construction is much more robust in locating the prime distribution of the roots in $\mathcal{D}$ and can not be substituted by Newton's method in this point.

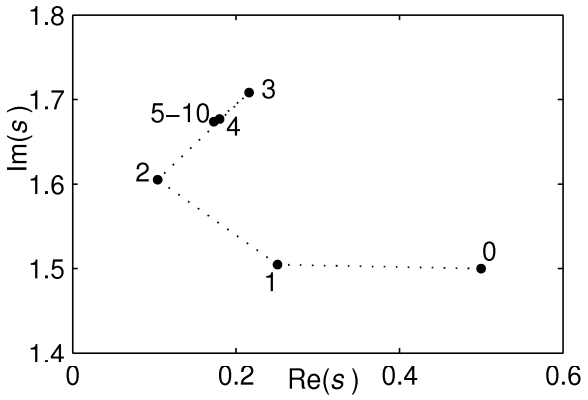

Fig. 3.13 Iterations of Newton's method

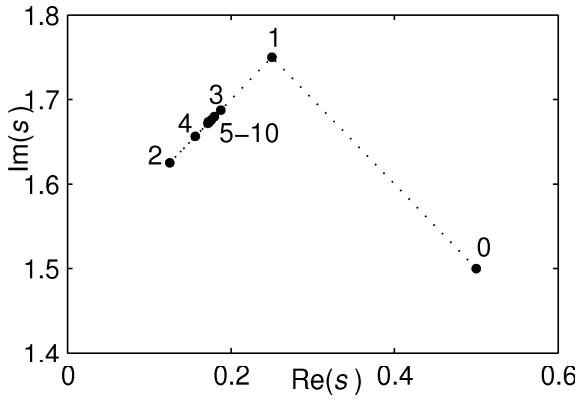

Fig. 3.14 Iterations of Weyl's construction

Tab. 3.1 Comparing Newton's method and Weyl's algorithm

k	$\varepsilon_k = \lvert s_{5,k} - s_{5,k-1} \rvert$	
	Newton's method	Weyl's algorithm
1	0.24924597344752	0.35355339059327
2	0.17764319986182	0.17677669529664
3	0.15220071700968	0.08838834764832
4	0.04797851095093	0.04419417382416
5	0.00760192169348	0.02209708691208
6	0.00018530326764	0.01104854345604
7	0.00000010929595	0.00552427172802
8	0.00000000000004	0.00276213586401
9	0.00000000000000	0.00138106793200
10	0.00000000000000	0.00069053396600

3.4 Rootfinding algorithm based on mapping the quasipolynomial function

It has been shown in section 3.3 that Newton's method is an effective tool for computing the roots of quasipolynomial $M(s)$ if the approximate positions of the roots are known. These prime approximations can be provided by modified Weyl's construction using argument principle based test. On one hand the construction is quite reliable, however, on the other hand the method is rather cumbersome, especially if there are many roots in the selected region $\mathcal{D}$. In this section I am going to introduce an original method for locating the approximate positions of the quasipolynomial roots based on mapping the function $M(s)$ in s-plane.

Suppose a quasipolynomial $M(s)$ is given by (3.1) and we want to find the approximate positions of its real roots located on the interval $\mathcal{B} = [\beta_{\min}, \beta_{\max}]$. Real roots are the solutions of the equation $M(s)=0$, $s \in \mathrm{R}$. Thus, an intuitive way of obtaining the approximate positions of the real roots consists in locating the intersection points of the curve described by $M(s)$ with the real axis s, see Fig. 3.15. In order to locate the

approximations $s_{i,0}$, let us consider a mesh on $\mathcal{B} = [\beta_{\min}, \beta_{\max}]$ given by $\beta_k = \beta_{\min} + k\Delta_s$, $k = 0,1,\ldots,\text{int}|\beta_{\max} - \beta_{\min}|/\Delta_s + 1$, where Δ_s is the chosen increment on the real axis. Real roots of $M(s)$ are located between the values β_{r_i} and β_{r_i+1} for which

$$M(\beta_{r_i})M(\beta_{r_i+1}) \le 0 \tag{3.14}$$

The approximation of the roots are given by

$$s_{i,0} = \beta_{r_i} + \left|\frac{M(\beta_{r_i})}{M(\beta_{r_i}) - M(\beta_{r_i+1})}\right|\Delta_s \tag{3.15}$$

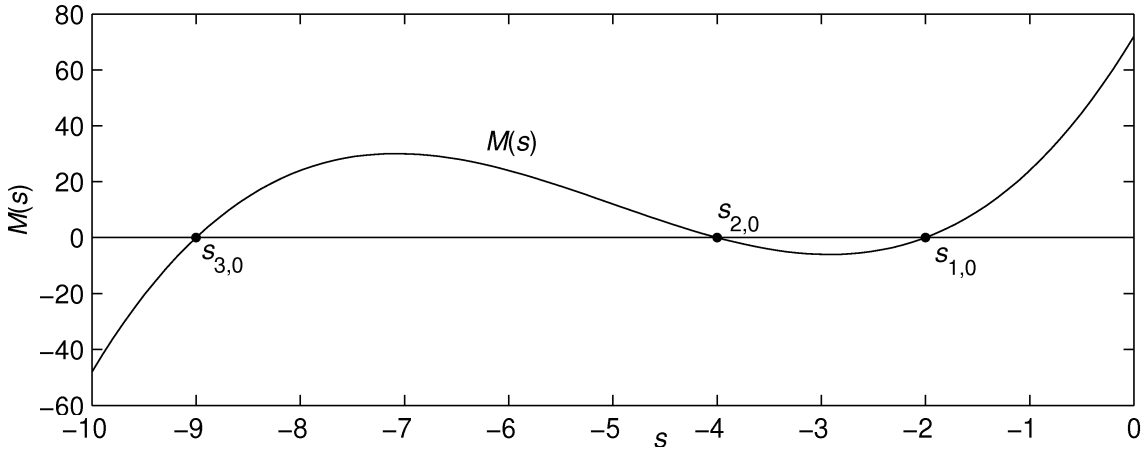

Fig. 3.15 Locating approximate positions of the real roots of $M(s)$

As has been shown, it is relatively easy to obtain the approximate positions of the real roots of $M(s)$. The task becomes much more complicated if the complex roots are to be found. Quasipolynomial $M(s)$ as a function of the complex variable $s = \beta + \mathrm{j}\omega$ can be split into real and imaginary parts

$$M(\beta + \mathrm{j}\omega) = R(\beta,\omega) + \mathrm{j}I(\beta,\omega) \tag{3.16}$$

where $R(\beta,\omega) = \mathrm{Re}\{M(\beta+\mathrm{j}\omega)\}$ and $I(\beta,\omega) = \mathrm{Im}\{M(\beta+\mathrm{j}\omega)\}$. Consequently, characteristic equation $M(s)=0$ can be split into

$$\begin{aligned} R(\beta,\omega) &= 0 \\ I(\beta,\omega) &= 0 \end{aligned} \tag{3.17}$$

Analogously to the task to locate the positions of the real roots, let us look at the problem from the geometric point of view. The complex roots of $M(s)$ are the intersection points of the curves described by the implicit functions $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$. Mapping the surfaces $R(\beta,\omega)$ and $I(\beta,\omega)$ on the chosen (suspect) region $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$ the zero-level contours are given by the intersections of the surfaces with the s-plane, see Fig. 3.16. Since the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ can be found analytically only for most trivial quasipolynomials, see Example 3.3., a numerical method has to be used to solve the problem.

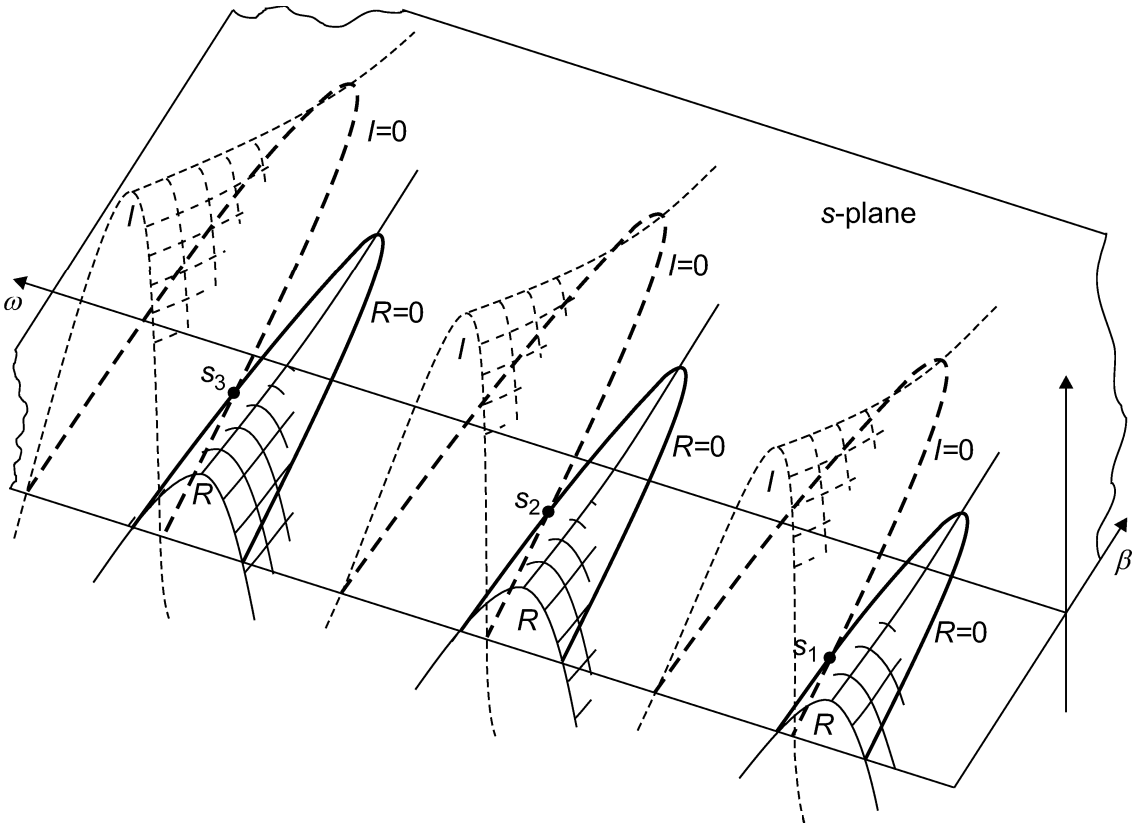

Fig. 3.16 The principle of locating $M(s)$ roots

Example 3.3

Let us find the contours $R(\beta,\omega)=0$ and $R(\beta,\omega)=0$ corresponding to the quasipolynomial

$$M(s)=s+\exp(-s) \tag{3.18}$$

Splitting the function $M(\beta+\mathrm{j}\omega)$ into real and imaginary parts

$$R(\beta,\omega)=\beta+\exp(-\beta)\cos\omega \tag{3.19}$$

$$I(\beta,\omega)=\omega-\exp(-\beta)\sin\omega \tag{3.20}$$

Both equations (3.19) and (3.20) allow the explicit forms of $R(\beta,\omega)=0$ and $I(\beta,\omega)=0$ to be found given by

$$\omega=(2k+1)\pi\pm\arccos(\beta\exp(\beta)),\ \beta\in(-\infty,\beta_u],\ k=0,1,2,... \tag{3.21}$$

$$\beta=-\ln\frac{\omega}{\sin\omega},\ \ \omega\in[2k\pi,(2k+1)\pi],\ k=0,1,2,... \tag{3.22}$$

Let us analyse the functions in order to obtain the zero-level contours analytically. Since the function arccosine provides real values only for its argument taken from the interval [-1,1], the function $\beta\exp(\beta)$ is to be limited by these values. Evaluating first and second derivation of this function, it is easy to find its absolute minimum that is reached for $\beta=-1$, thus $\min(\beta\exp(\beta))=-0.368>-1$. Evaluating the limit $\lim_{\beta\to-\infty}\beta\exp(\beta)=0$ and taking into

consideration the fact that the function is decreasing on the interval $[-\infty,-1)$ and increasing on the interval $[-1\infty)$ there only exists the limitation given by $\beta \le \beta_u$. Numerical evaluation of the equation $\beta\exp(\beta)=1$ results in $\beta_u = 0.567144$. The boundaries of ω in (3.22) are given by the fact that the logarithm function provides the real results only for its argument greater than 0. Since function sine gives positive result for $\omega \in [2k\pi,(2k+1)\pi]$, $k=0,1,2,\ldots$, this is the condition for ω in (3.22), provided that $\omega \ge 0$. The complex plane is symmetric therefore it is not necessary to investigate the features of (3.21) and (3.22) for $\omega < 0$. Both the sets of contours described by (3.21) and (3.22) are shown in Fig. 3.17. Their intersection points correspond to the positions of the roots of $M(s)$.

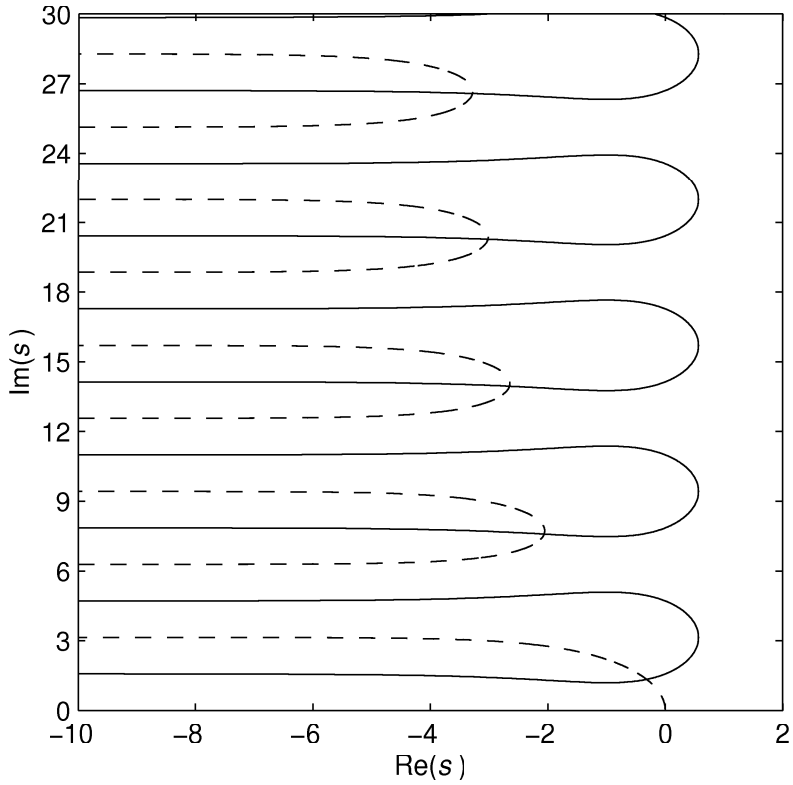

Fig. 3.17 Contours given by (3.21) (solid, corresponding to $R(\beta,\omega)=0$) and (3.22) (dashed, corresponding to $I(\beta,\omega)=0$)

3.4.1 Algorithm for contour plotting

The numerical technique to obtain the level-contours is known as contour plotting. Most the math software programs provide an algorithm for contour plotting. Such algorithms are implemented, e.g., in Mathematica[2], Maple[3] and Mathcad[4] as *ContourPlot*, in Matlab[5] as *Contour*. Let us also mention the data analysis and visualisation software PV-Wave[6], where the algorithm *Contour* is available for gridded data and *Contour2* for scattered data and NCAR Graphics[7] with the advanced contouring package of Fortran subroutines *Conpact*.

[2] http://www.wolfram.com/
[3] http://www.mapleapps.com/
[4] http://www.mathsoft.com/
[5] http://www.mathworks.com/
[6] http://www.vni.com/
[7] http://ngwww.ucar.edu/

Easily applicable method, that is used in some of the mentioned software tools, e.g., in Matlab, is known as level curve tracing algorithm, see, e.g., Cottafava and Le Moli, (1969), Snyder, (1977), (see also the thesis of Aramini, 1981). The method seems like the obvious way to perform contour plotting if the function values are available only at the vertices of a regular grid. The fundamentals of the algorithm can be summarised as follows. All the edges in the grid that are intersected by the level curve are marked. Subsequently, for all the marked edges, the interpolation is performed in order to obtain a better approximation of the intersection points of the curve and the edges, see Fig. 3.18. Finally, starting from an intersection point, the method searches for the neighbouring point to draw a line segment of the contour. Such a procedure is carrying on until the boundary of the region is reached or the contour is closed. There exist many modifications of the method that differ particularly in accomplishing the curve tracing. However, most of them are based on the approximation of intersection points of the curve and the edges. It should be noted that the grid cells do not have to be necessarily rectangular. Triangular grid cell for contouring process has been used, e.g., in Preusser, (1984). A different approach has to be used if the data are scattered. The algorithm based on approximation of the level curves by piecewise polynomials, which is used in software PV-Wave as *Contour2*, has been proposed in Preusser, (1986).

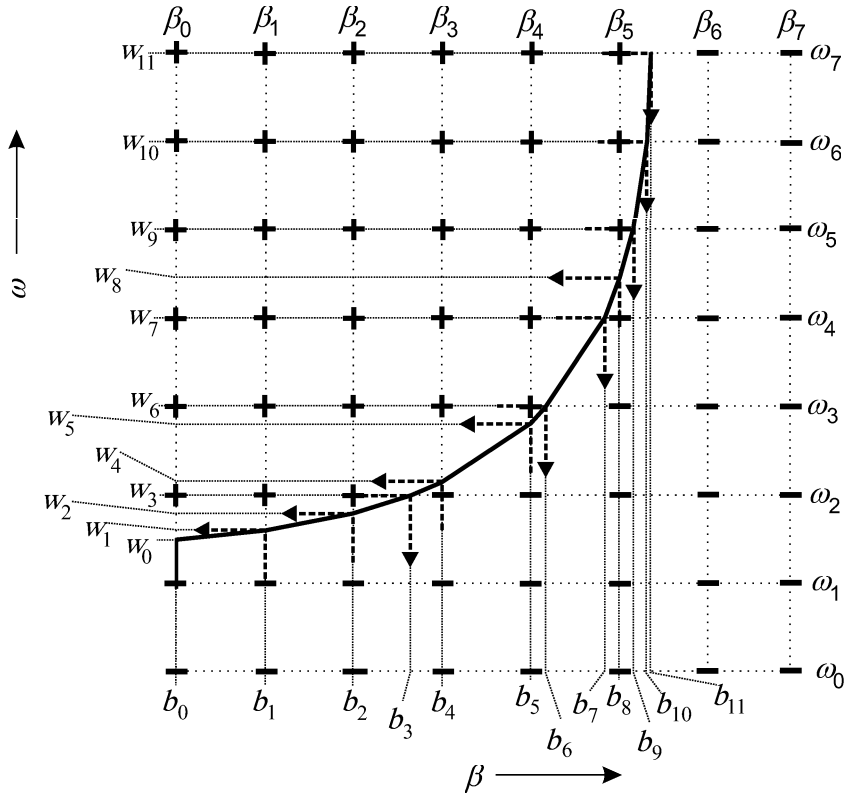

Fig. 3.18 The idea of constructing the zero-level contour

Since software Matlab is commonly used in the field of control engineering, I have decided to use its function *contour* for accomplishing the contour plotting. Let us explain the implementation of the level curve tracing algorithm for obtaining, e.g., the curve $R(\beta,\omega) = 0$ ($\mathbf{R}^0$). The first step consists in evaluating the function $R(\beta,\omega)$ for all the nodes of the grid on $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$ given by

$$\begin{aligned}\beta_k &= \beta_{\min} + k\Delta_s,\ k = 0,1,....,k_{\max},\ k_{\max} = \text{int}\left|\beta_{\max} - \beta_{\min}\right|/\Delta_s + 1\\ \omega_l &= \omega_{\min} + l\Delta_s,\ l = 0,1,....,l_{\max},\ l_{\max} = \text{int}\left|\omega_{\max} - \omega_{\min}\right|/\Delta_s + 1\end{aligned} \tag{3.23}$$

where Δ_s is a chosen increment (step) of the grid. Thus the values of $R(\beta,\omega)$ can be stored in the following matrix

$$\mathbf{R}^{\mathrm{d}} = \begin{bmatrix} R(\beta_0,\omega_{l_{\max}}) & R(\beta_1,\omega_{l_{\max}}) & \dots & R(\beta_{k_{\max}},\omega_{l_{\max}}) \\ & & \vdots & \\ R(\beta_0,\omega_1) & R(\beta_1,\omega_1) & \dots & R(\beta_{k_{\max}},\omega_1) \\ R(\beta_0,\omega_0) & R(\beta_1,\omega_0) & \dots & R(\beta_{k_{\max}},\omega_0) \end{bmatrix} \tag{3.24}$$

The way of approximating the zero-level contour based on matrix (3.24) is sketched in Fig. 3.18 and is accomplished according to the method described as the level curve tracing algorithm. Thus the approximations of the intersection points of the edges and the zero-level contour are obtained using the linear interpolation method. The level curve approximation is stored in the matrix

$$\mathbf{R}^0 = \begin{bmatrix} b_0 & b_1 & b_2 & \hat{b}_3 & b_4 & b_5 & \hat{b}_6 & \hat{b}_7 & b_8 & \hat{b}_9 & \hat{b}_{10} & \hat{b}_{11}\cdots\cdots \\ \hat{w}_0 & \hat{w}_1 & \hat{w}_2 & w_3 & \hat{w}_4 & \hat{w}_5 & w_6 & w_7 & \hat{w}_8 & w_9 & w_{10} & w_{11}\cdots\cdots \end{bmatrix} \tag{3.25}$$

in which each column corresponds to one point of the contour in the *s*-plane. As can be seen, one value in each column of $\mathbf{R}^0$ is marked by the chevron. Non-marked values correspond to the values on the grid while the chevron marked values are the results of the interpolations performed on the edges, see Fig. 3.18. $\mathbf{R}^0$ given by (3.25) corresponds to the contour in Fig. 3.18. For example

$$b_2 = \beta_2,\quad \hat{w}_2 = \omega_1 + \left|\frac{R(\beta_2,\omega_1)}{R(\beta_2,\omega_1) - R(\beta_2,\omega_2)}\right|\Delta_s$$

or (3.26)

$$\hat{b}_9 = \omega_1 + \left|\frac{R(\beta_5,\omega_5)}{R(\beta_5,\omega_5) - R(\beta_6,\omega_5)}\right|\Delta_s,\quad w_9 = \omega_5$$

In this way, the both the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ are approximated. The idea of the algorithm will be explained in the Example 3.4.

Example 3.4

Let us find the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ corresponding to quasipolynomial (3.18) from Example 3.4, (where the contours have been found analytically), by means of described contour algorithm. First step of this task consist in mapping the surfaces $R(\beta,\omega)$ and $I(\beta,\omega)$ on the grid region $\mathcal{D} = [-10,2]\times[0,30]$ with chosen $\Delta_s = 0.3$, see Fig. 3.19 and Fig. 3.20, respectively. Consequently the zero-level contours result from applying the contour plotting algorithm (*contour*, Matlab). As can be seen, putting the contours from Fig. 3.19 and Fig.3.20 into one figure, the result would obviously correspond to Fig. 3.17, where the analytically found contours may be seen.

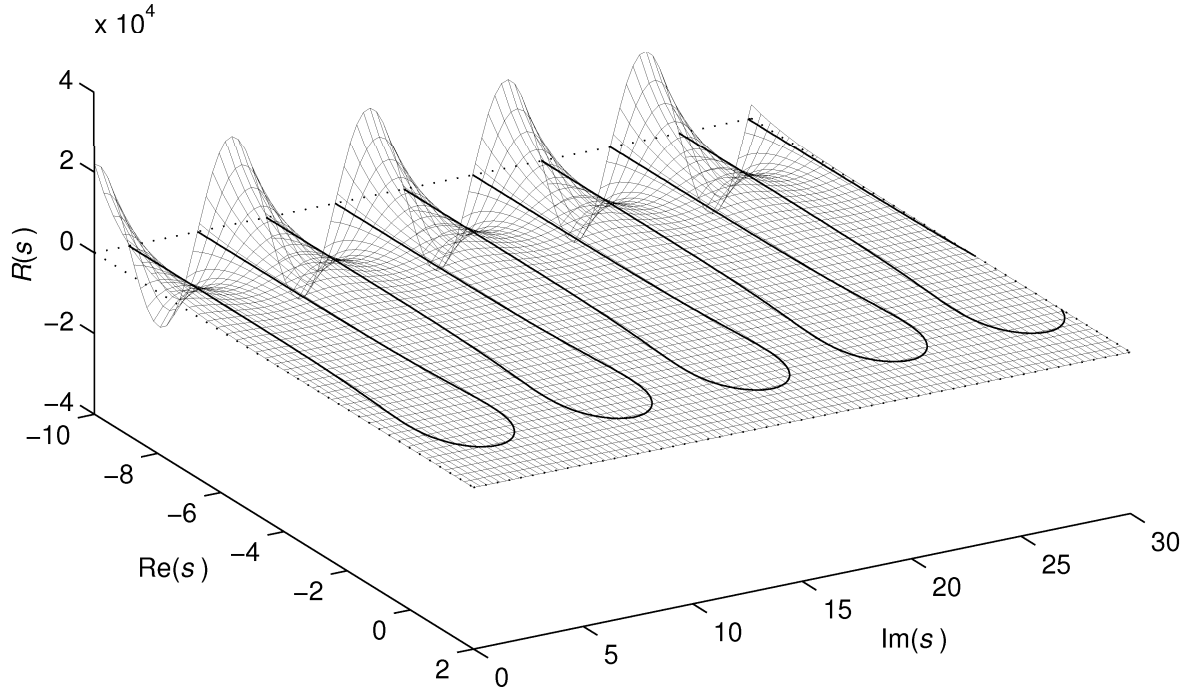

Fig 3.19 Surface $R(s)$ ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.18) and the contours $R(\beta, \omega) = 0$ (intersections of the surface with the s-plane)

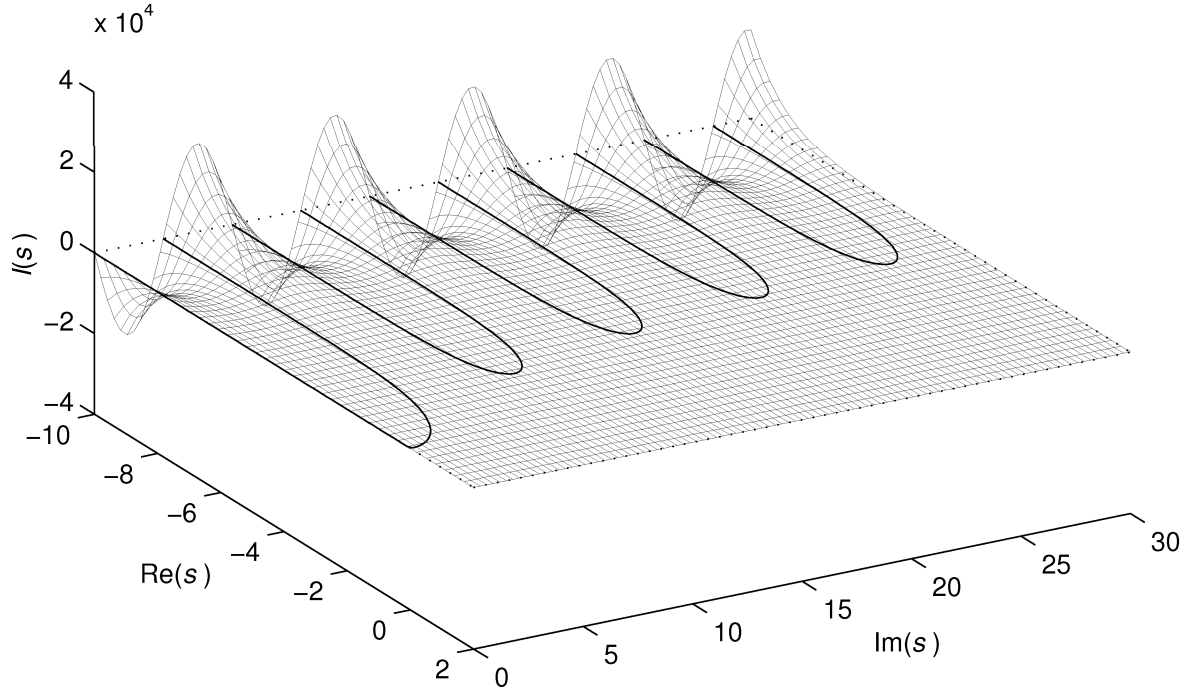

Fig. 3.20 Surface $I(s)$, ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.18) and the contours $I(\beta, \omega) = 0$ (intersections of the surface with the s-plane)

Example 3.5

As in Example 3.4 let us find the zero-level contours $R(\beta,\omega) = 0$, $I(\beta,\omega) = 0$ corresponding to the following quasipolynomial

$$M(s) = \det(s\mathbf{I} - \mathbf{A}(s)) = s^3 + Q_2(s)s^2 + Q_1(s)s + Q_0(s) \tag{3.27}$$

$$\mathbf{A}(s) = \begin{bmatrix} -\exp(-8.4s) & \exp(-4.1s) & \exp(-6.6s) \\ \dfrac{\exp(-5.2s) - \exp(-12.5s)}{7.3s} & -\exp(-4.3s) & \exp(-3.7s) \\ \exp(-7.8s) & \dfrac{\exp(-6.5s) - \exp(-18.9s)}{12.4s} & \exp(-5.2s) \end{bmatrix}$$

Evaluating the determinant, we obtain

$$Q_0(s) = -\exp(-17.9s) - \frac{\exp(-5.2s) - \exp(-12.5s)}{7.3s}\frac{\exp(-6.5s) - \exp(-18.9s)}{12.4s}\exp(-6.6s) -$$
$$- \frac{\exp(-5.2s) - \exp(-12.5s)}{7.3s}\exp(-12.1s) + \frac{\exp(-5.2s) - \exp(-12.5s)}{7.3s}\exp(-9.3s) -$$
$$- \exp(-15.6s) - \exp(-18.7s)$$

$$Q_1(s) = -\exp(-9.5s) - \frac{\exp(-6.5s) - \exp(-18.9s)}{12.4s}\exp(-3.7s) - \exp(-13.6s) + \exp(-12.7s) -$$
$$- \frac{\exp(-5.2s) - \exp(-12.5s)}{7.3s}\exp(-4.1s) - \exp(-14.4s)$$

$$Q_2(s) = \exp(-8.4s) + \exp(-4.3s) - \exp(-5.2s)$$

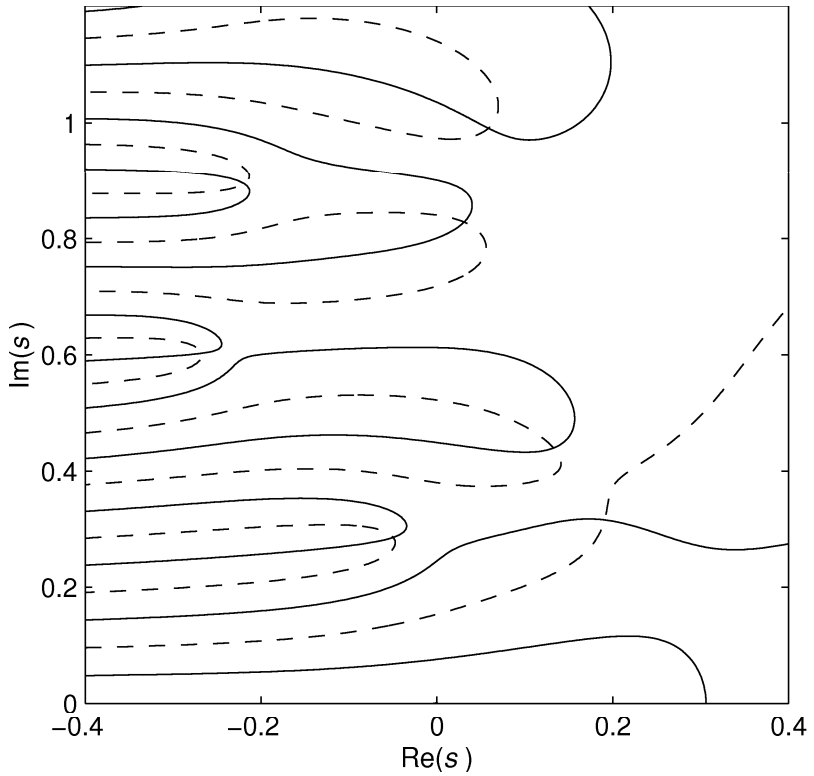

Fig. 3.21 Contours $R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed) corresponding to (3.27)

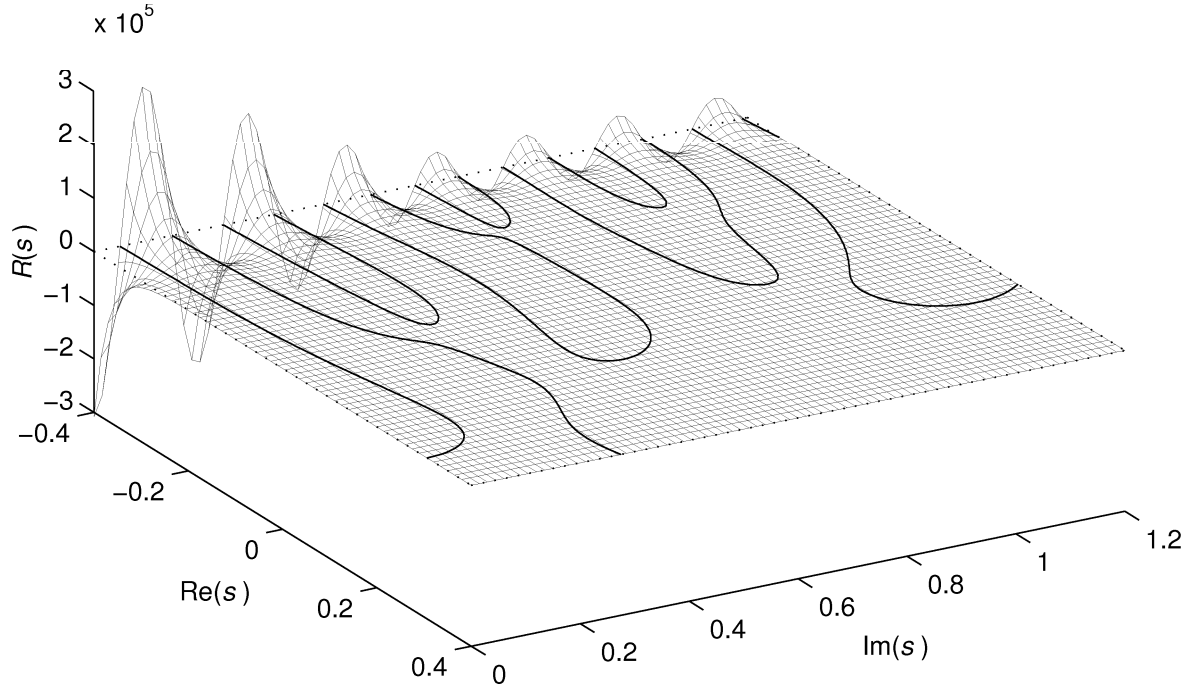

Fig. 3.22 Surface $R(s)$ ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.27) and the contours $R(\beta,\omega) = 0$ (intersections of the surface with the s-plane)

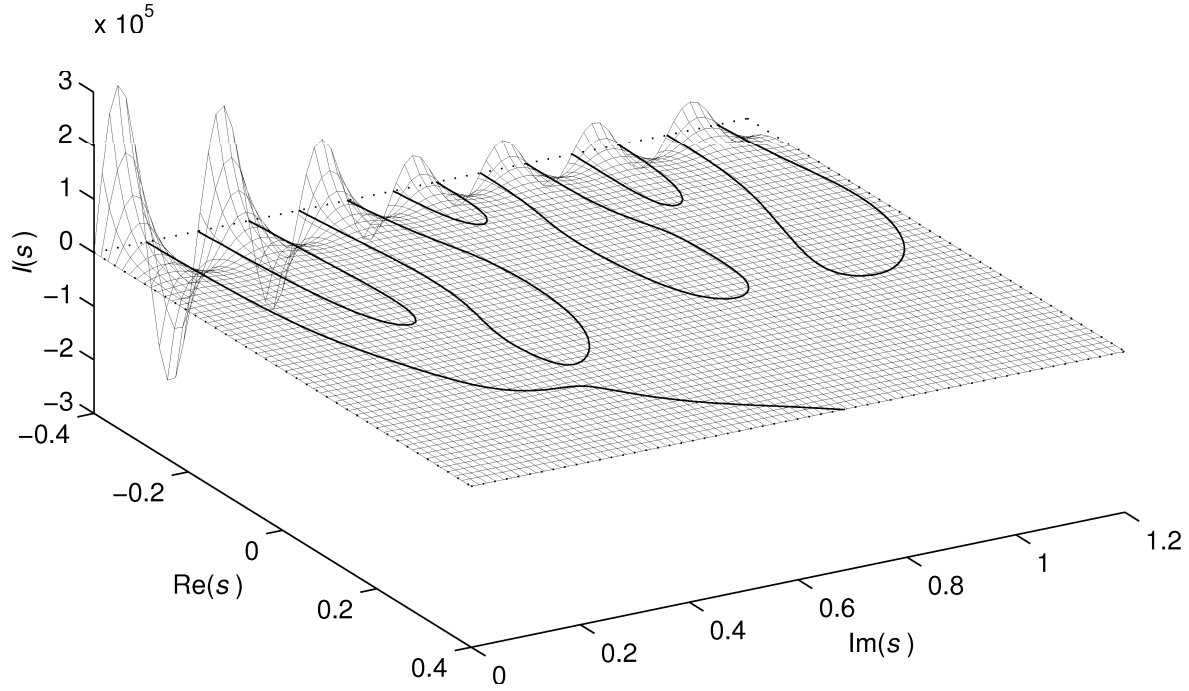

Fig. 3.23 Surface $I(s)$, ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.27) and the contours $I(\beta,\omega) = 0$ (intersections of the surface with the s-plane)

The contours have been found using the same procedure as in Example 3.4. The surfaces of $R(\beta,\omega)$ and $I(\beta,\omega)$ are mapped on the region $\mathcal{D} = [-0.4,0.4]\times[0,1.2]$ with the increment $\Delta_s = 0.015$, see Fig. 3.22 and Fig. 3.23. The contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ can be seen in Fig. 3.21. Obviously, there are seven complex roots (intersection points of $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$) and one real root (intersection point of $R(\beta,\omega) = 0$ and axis Re(s)) located in the region $\mathcal{D}$. As can be seen, the terms Q_i, i=0,1,2 of $M(s)$ given by (3.27) are quite complicated functions. Even though the quasipolynomial is only of the third order, the functions $R(\beta,\omega)$ and $I(\beta,\omega)$ are not trivial. Due to the terms representing the linear distribution of two delays in (3.28), computing the eigenvalues of the matrix by means of a method based on discretization (seen in sections 1.2.7-1.2.10) would be rather complicated task. On the other hand the method described above indicates the positions of the eigenvalues of $\mathbf{A}(s)$ directly and without extensive effort, using only the tools available in software Matlab.

3.4.2 Locating the intersection points

The positions of the quasipolynomial roots are given by the intersection points of the zero-level contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$. These intersections can be easily identified visually, see e.g. Fig. 3.17 or Fig. 3.21. The estimates of the intersection points can be used as the starting points for Newton's numerical method. In this secion, I am going to introduce a method for automatic locating the positions of the intersections of $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$. In case of computing the real roots of $M(s)$, see Fig. 3.15, the roots are the intersection points of the function $M(s)$ with the s-axis. The problem of locating the complex roots of $M(s)$ can be solved analogously to that case. Let us consider that we have found the points of one of the zero-level contours, e.g., $R(\beta,\omega) = 0$. The points are stored in the matrix

$$\mathbf{R}^0 = [\beta_k \ \omega_k]^T \quad k=0,1,.. \tag{3.28}$$

see (3.25), ordered to approximate the contour by connecting the points by the line segments. Let us introduce the function

$$I_R(k) = I(\beta_k, \omega_k), \quad k = 0,1,... \tag{3.29}$$

Let the values $\beta_k + \mathrm{j}\omega_k$ that are closest to the roots be marked by the indexes r_i, the following inequality holds

$$I_R(r_i)\, I_R(r_i+1) \le 0 \tag{3.30}$$

Thus, selecting $\beta_{r_i} + \mathrm{j}\omega_{r_i}$ using (3.30) the approximations of the roots are given by

$$\begin{aligned} \beta_{i,0} &= \beta_{r_i} + \left|\frac{I_R(r_i)}{I_R(r_i) - I_R(r_i+1)}\right| \left|\beta_{r_i+1} - \beta_{r_i}\right| \\ \omega_{i,0} &= \omega_{r_i} + \left|\frac{I_R(r_i)}{I_R(r_i) - I_R(r_i+1)}\right| \left|\omega_{r_i+1} - \omega_{r_i}\right| \end{aligned} \tag{3.31}$$

These approximate positions may be used as the starting points for Newton's method if the enhanced accuracy of the root approximations is required. If the step of the grid is not too large, $s_{i,0} = \beta_{r_i} + \mathrm{j}\omega_{r_i}$ can be directly considered as the starting points for Newton's method.

The described approach for locating the roots of $M(s)$ is visualised in Fig. 3.24 where the contours $R(\beta,\omega) = 0$ corresponding to $M(s)$ given by (3.27) from Example 3.5 are shown. The region is chosen $\mathcal{D} = [-0.2,0.4]\times[-0.15,65]$ in order to show the way of locating a real root as well. On these contours, the normalised function

$$\bar{I}_R(k) = \frac{I_R(k)}{|I_R(k)|}, \quad k = 0,1,... \tag{3.32}$$

has been evaluated and it is visualised in Fig. 3.24 by the thicker lines. Due to the normalisation, the function $\bar{I}_R(k)$ acquires only two values, either 1 or -1 (if $I_R(p) = 0$, (3.32) has to be solved as the limit with $I_R(p) \to 0_+$ and $I_R(p) \to 0_-$).

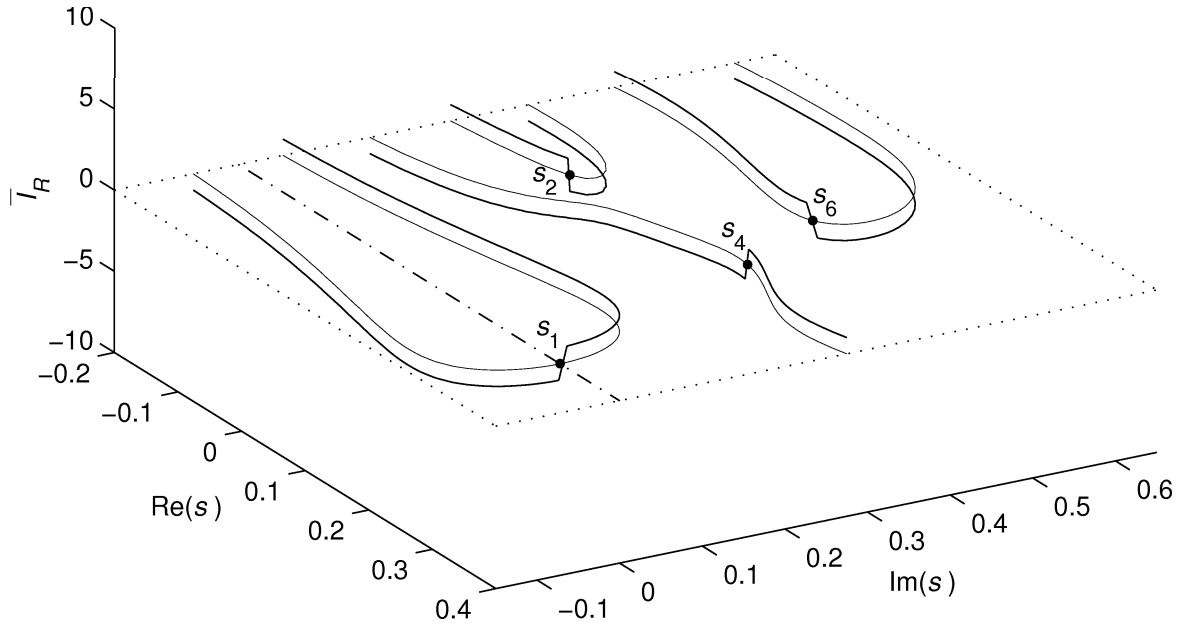

Fig. 3.24 Zero-level contours $R(\beta,\omega) = 0$ with the curves $\bar{I}_R(k)$ evaluated on the contours

The positions of the roots are indicated by the intersection points of $\bar{I}_R(k)$ and $R(\beta,\omega)$ (given by (3.28)). Obviously, these are the points where $\bar{I}_R(k)$ changes its values from 1 to -1 or vice versa. As can be seen in Fig. 3.24, there are four roots in the region $\mathcal{D}$. One root is real and the other roots are complex. The complex roots are marked with even indexes, because they have complex conjugates with negative imaginary parts. Let us compare Fig. 3.24 with the corresponding part of Fig. 3.21. The complex roots are in the positions of the intersection points of the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$. However, there is no intersection of the contours in the real root. It is given by the nature of the real roots where the necessary and sufficient condition is $R(\beta) = 0$. As can be seen in Fig. 3.24, $\bar{I}_R(k)$ intersects the contour $R(\beta,\omega) = 0$ in the root s_1 anyway. It is due to the symmetry of the s-plane (The symmetry is not simply mirror like, but the mirror reflex is inverted, i.e., the vertical axis is inverted).

To conclude, using the described approach for automatic locating the positions of the roots, only one of the surfaces is necessary to be mapped. The roots are located as the intersection points of the zero-level contours and the curves that correspond to the values of

the other surface (not mapped one) which are evaluated on the contours. However, I recommend to map both the surfaces anyway. Mapping both the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ provides the possibility to check the resultant approximations of the root positions visually. Therefore, I will keep on displaying both the surfaces and sets of contours.

3.4.3 Locating multiple roots

As will be shown below, the rootfinding method based on quasipolynomial function mapping does not solve the problem of multiple roots in a different way. The multiplicity is simply given by the number of intersecting contours in the particular point. This very convenient feature of the algorithm will be demonstrated in the following example.

Example 3.6

Let us find the roots of the quasipolynomial

$$M(s) = (s + \exp(-s))^n \tag{3.33}$$

for $n = 3$ located in the region $\mathcal{D} = [-10,2]\times[0,30]$. In order to achieve smooth contours, let us choose $\Delta_s = 0.1$.

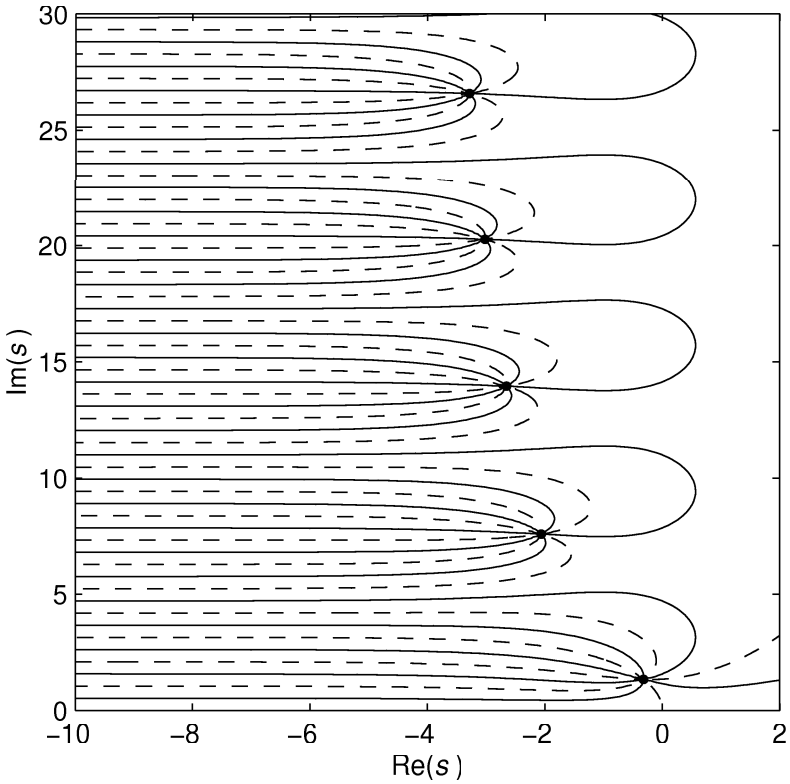

Fig. 3.25 Mapping $M(s)$ given by (3.33), $n = 3$
$R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.1$

The resultant maps are shown in Fig. 3.25. As can be seen, at each of the points corresponding to the root approximations, three couples of contours are intersected. Thus, the points correspond to the triple roots. However, each of the roots is located independently. The values of the starting points of the Newton's method $s_{i,0}$ resulted from the contour mapping and the more precise approximations s_i obtained by using Newton's method are shown in

Tab. 3.2. There are three sets of results of Newton's method in Tab. 3.2 differing in ε_N that indirectly prescribes the accuracy of Newton's method, see (3.13). The values $s_{i,0}$ are not computed from (3.31) but they are taken from (3.28), given by $s_{i,0} = \beta_{r_i} + \mathrm{j}\omega_{r_i}$ where r_i is determined by (3.30) (one of the values $\beta_{r_i}, \omega_{r_i}$ is a grid value). Obviously, with decreasing value of ε_N the root triples tend to converge to their common target that is the value of the corresponding triple root.

Tab. 3.2 Approximation of the roots with respect to ε_N

i	$s_{i,0},\ \Delta_s = 0.1$	$s_i,\ \varepsilon_N = 0.01$	$s_i,\ \varepsilon_N = 0.001$	$s_i,\ \varepsilon_N = 0.0001$
1	-3.2000 +26.5742j	-3.2713 +26.5793j	-3.2863 +26.5804j	-3.2876 +26.5805j
3	-3.2305 +26.7000j	-3.2793 +26.5951j	-3.2870 +26.5817j	-3.2877 +26.5806j
5	-3.4000 +26.5831j	-3.3037 +26.5808j	-3.2892 +26.5805j	-3.2880 +26.5805j
7	-2.9000 +20.2633j	-3.0056 +20.2714j	-3.0183 +20.2723j	-3.0201 +20.2724j
9	-3.0000 +20.3001j	-3.0112 +20.2846j	-3.0190 +20.2740j	-3.0201 +20.2726j
11	-3.0169 +20.2000j	-3.0189 +20.2582j	-3.0201 +20.2706j	-3.0202 +20.2723j
13	-2.7000 +13.9241j	-2.6673 +13.9414j	-2.6544 +13.9485j	-2.6534 +13.9491j
15	-2.6000 +14.0078j	-2.6426 +13.9601j	-2.6518 +13.9506j	-2.6531 +13.9493j
17	-2.7000 +13.8977j	-2.6623 +13.9384j	-2.6544 +13.9478j	-2.6533 +13.9491j
19	-2.0000 + 7.5815j	-2.0445 + 7.5866j	-2.0607 + 7.5885j	-2.0621 + 7.5886j
21	-2.1000 + 7.6048j	-2.0793 + 7.5960j	-2.0638 + 7.5893j	-2.0624 + 7.5887j
23	-2.1048 + 7.6000j	-2.0815 + 7.5938j	-2.0640 + 7.5891j	-2.0624 + 7.5887j
25	-0.2564 + 1.4000j	-0.3054 + 1.3490j	-0.3170 + 1.3383j	-0.3180 + 1.3373j
27	-0.4000 + 1.3051j	-0.3344 + 1.3300j	-0.3196 + 1.3366j	-0.3183 + 1.3372j
29	-0.4000 + 1.2996j	-0.3343 + 1.3289j	-0.3196 + 1.3365j	-0.3183 + 1.3372j

Seemingly, locating the triple roots of (3.33) by means of the method based on $M(s)$ mapping has not brought about any difficulties. The problem has been solved in the same way as if the roots were single. In order to highlight the potentials and to show the limits of the method, let us try to solve the same task for $M(s)$ given by (3.33) with $n = 10$. The maps of such quasipolynomial are shown in Fig. 3.26. Since the figure is not sufficiently transparent, a detailed view of one set of the intersection points of the contours is shown in Fig. 3.27 ($\Delta_s = 0.01$). As can be seen, the result is in accordance with the assumption that there are 10 couples of intersecting contours.

In contrast to the visual perception that the intersecting contours are quite smooth, in fact, the contour plotting algorithm used provides worse results. As can be seen in Fig. 3.28 (for $n = 3$) and Fig. 3.29 (for $n = 10$), where the contours are shown in the vicinity of one of the multiple roots, the contours are formed in a different way. In case of $n = 3$, this incorrect fitting of the contours did not influence the result. However in case of $n = 10$, there are located 15 intersection points, which does not correspond to the multiplicity of the root. The contour mismatch is given by the plotting algorithm used and would not be probably eliminated by choosing a smaller Δ_s.

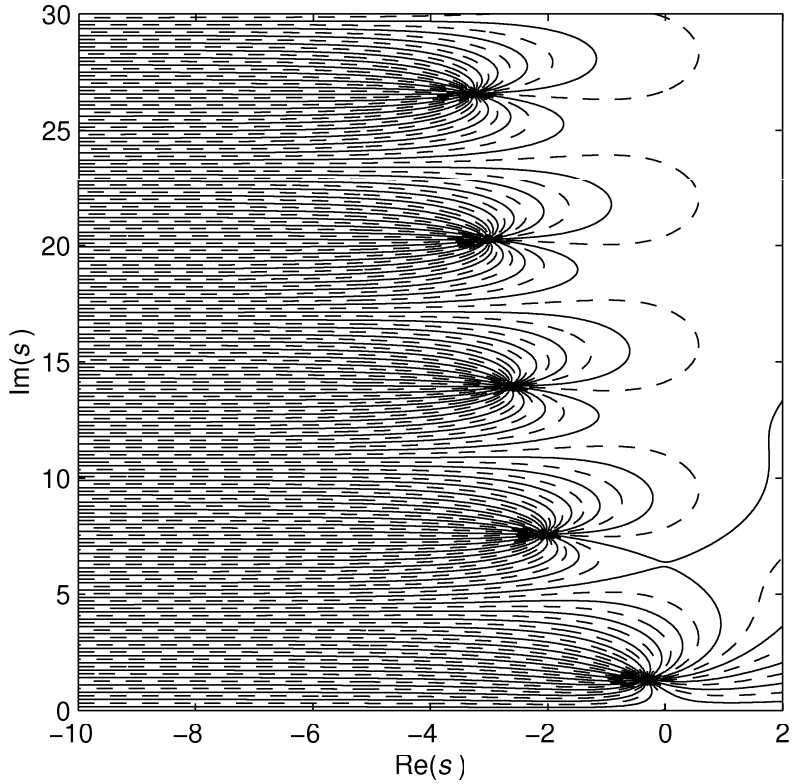

Fig. 3.26 Mapping $M(s)$ given by (3.33), $n = 10$
$R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.1$

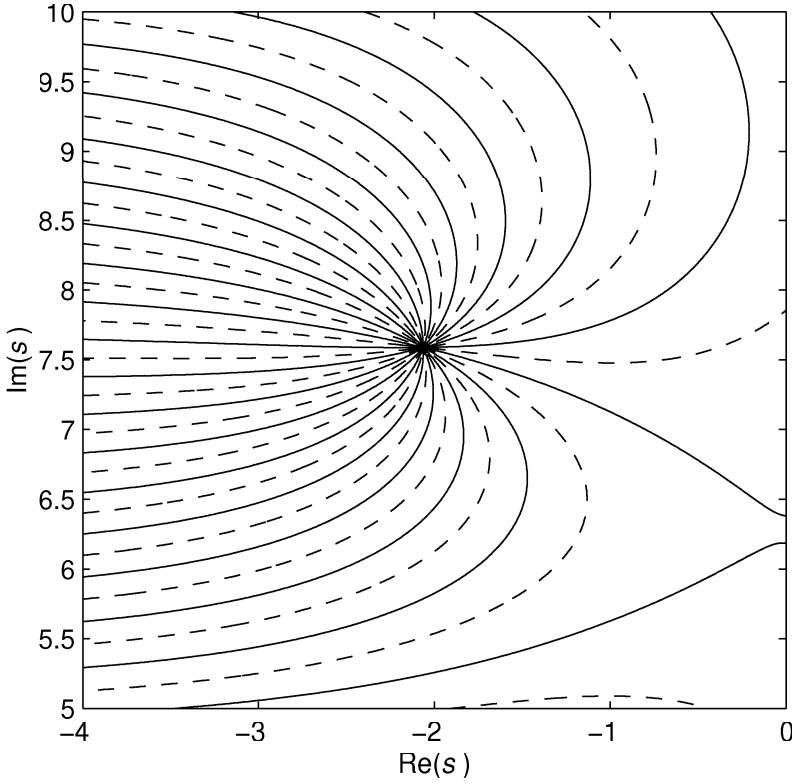

Fig. 3.27 Detailed view of mapping $M(s)$ given by (3.33), $n = 10$
$R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.01$

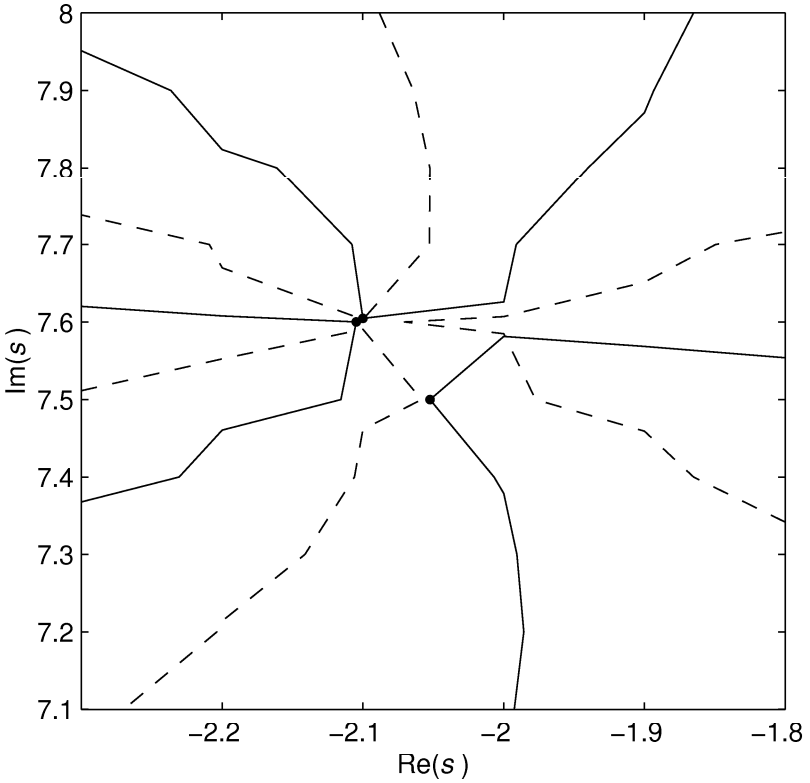

Fig. 3.28 Detailed view of mapping $M(s)$ given by (3.33), $n = 3$ in the vicinity of the location of a multiple root, $R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.1$

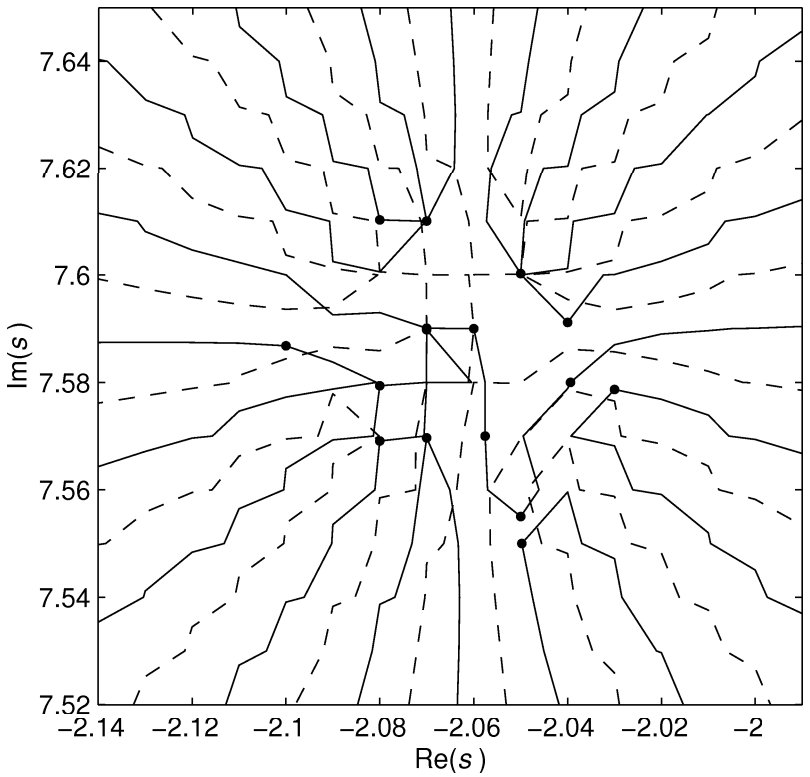

Fig. 3.29 Detailed view of mapping $M(s)$ given by (3.33), $n = 10$ in the vicinity of the location of a multiple root, $R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.01$

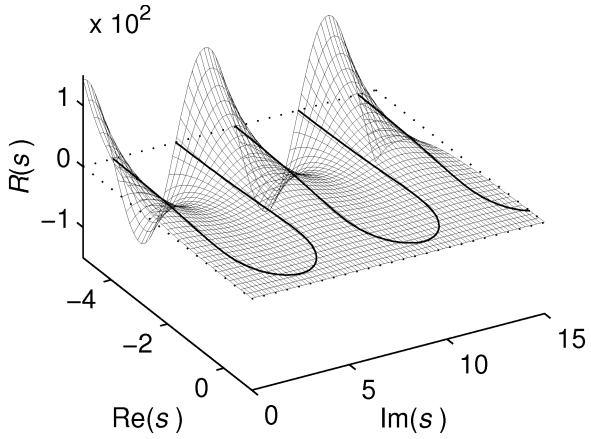

Fig. 3.30 Mapping real part of $M(s)$ given by (3.33), n=1

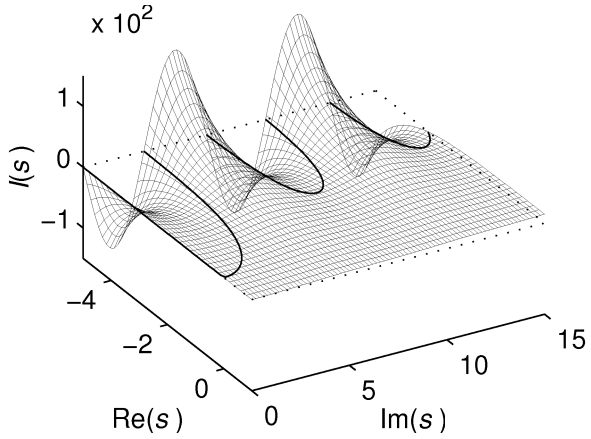

Fig. 3. 31 Mapping imag. part of $M(s)$ given by (3.33), n=1

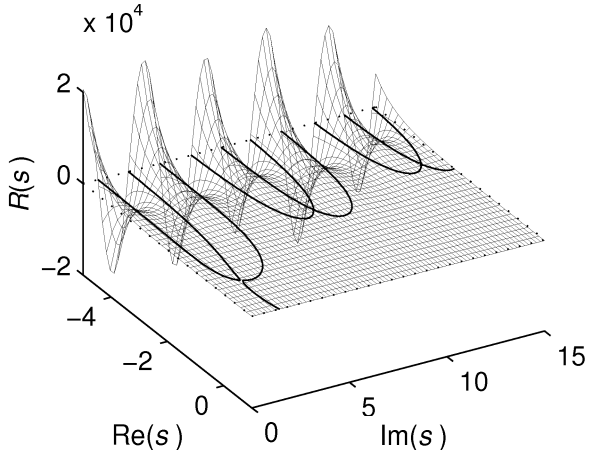

Fig. 3.32 Mapping real part of M(s) given by (3.33), n=2

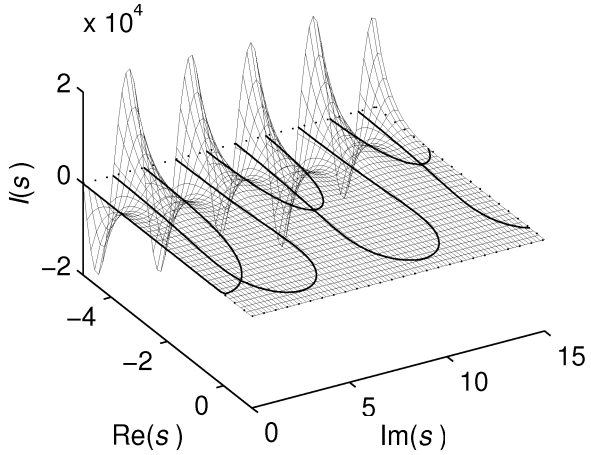

Fig. 3.33 Mapping imag. part of $M(s)$ given by (3.33), n=2

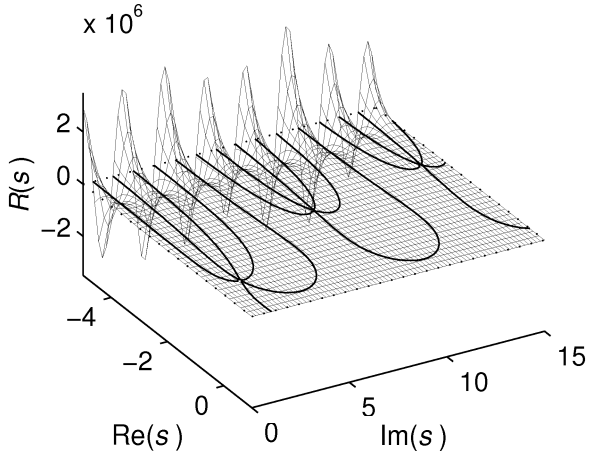

Fig. 3.34 Mapping real part of $M(s)$ given by (3.33), n=3

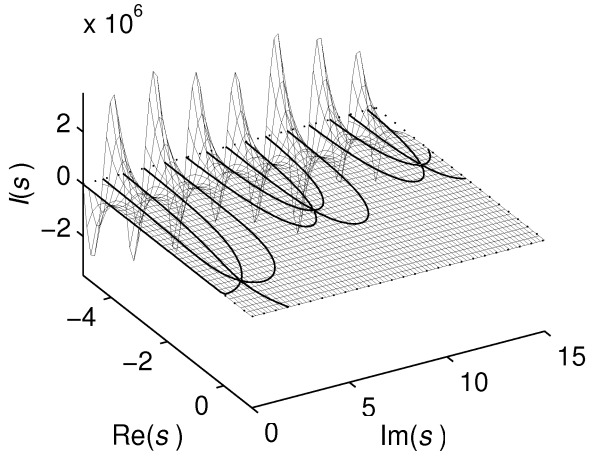

Fig. 3.35 Mapping imag. part of $M(s)$ given by (3.33), n=3

To sum up, the mapping based method for computing the roots may be used for locating the approximate positions of the multiple roots without any modifications. However, the danger of the contour mismatching in the vicinity of the multiple roots has to be taken into consideration. It is advisable to check the contours visually by inspecting the view of the vicinity of the multiple roots. In Fig. 3.30 - Fig. 3.35, the surfaces corresponding to quasipolynomial (3.33) with $n = 1..3$ are shown with the zero-level contours. Obviously, considering the character of the contours with increasing n, the contours are supposed to be smooth even in the vicinities of the multiple roots. The unsmoothness seen in Fig. 3.28 and Fig. 3.29 is obviously caused by the contour plotting algorithm used in which the requirement of contour smoothness is not involved.

Example 3.7

Let us return back to the task solved in Example 3.1, i.e., locating the root positions of $M(s)$ given by (3.11) in the region $\mathcal{D} = [-1, 1] \times [0, 2]$, by means of Weyl's construction and the argument principle rule. Applying the mapping based algorithm the maps shown in Fig. 3.36 indicates that the roots $s_{1,3}$ are the double complex roots. Using $\Delta_s = 0.01$ and $\varepsilon_N = 0.0001$ the root approximations seen in Tab. 3.3 have been obtained.

Tab. 3.3 Approximation three roots of $M(s)$ given by (3.11)

i	$s_{i,0}$, $\Delta_s = 0.01$	s_i, $\varepsilon_N = 0.0001$
1	-0.3200 + 1.3407j	-0.3182 + 1.3373j
3	-0.3300 + 1.3336j	-0.3182 + 1.3372j
5	0.1700 + 1.6707j	0.1728 + 1.6737j

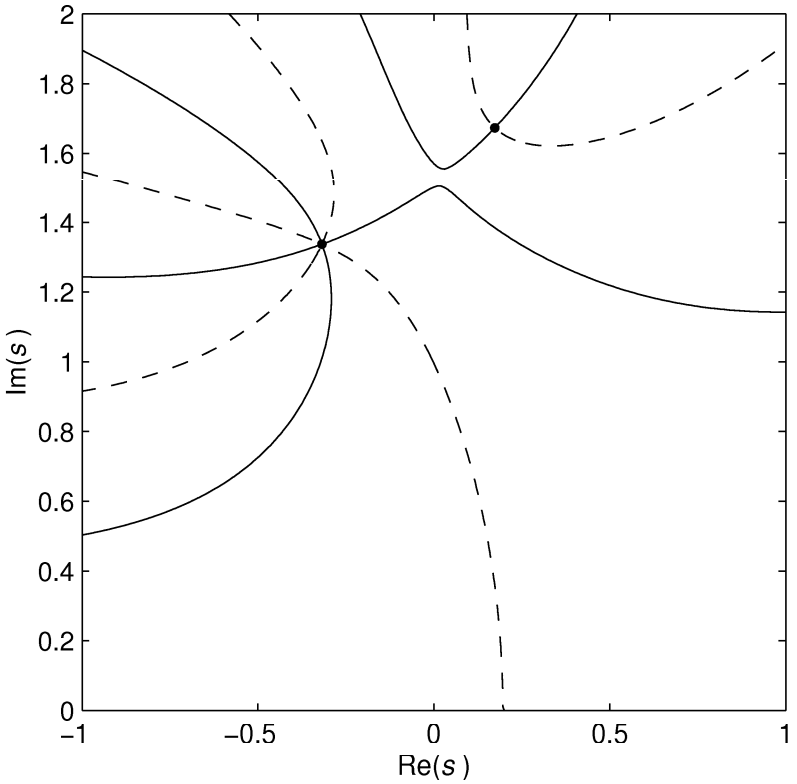

Fig. 3.36 Mapping $M(s)$ given by (3.11),
$R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed), $\Delta_s = 0.01$

3.4.4 Ill-conditioned (quasi)polynomials

The major limitation of the mapping based algorithm is the incapability to deal with the ill-conditioned (quasi)polynomial functions. Such a problem is often encountered in solving the task of computing the roots of higher order polynomials, see Wilkinson, (1963, 1984), Peters and Wilkinson, (1971). A root is called ill-conditioned if it is sensitive to small changes in the coefficients of the polynomial caused, e.g., by truncation errors. Conversely a root is called well-conditioned if it is comparatively insensitive to such perturbations. Roughly speaking a root which is well separated from the other roots is likely to be well conditioned, while roots that are close together are likely to be ill conditioned. The closeness is not meant as the absolute distance between neighbouring roots, but the ratio of their magnitudes. If the ratio is close to one, the zeros are ill conditioned. Therefore, obviously, the multiple roots are ill conditioned. A multiple root is often split into a cluster of roots because of the perturbations in the (quasi)polynomial or computational inaccuracies. Dealing with ill-conditioned polynomials requires using iterative algorithms with multi-precision arithmetic, see Bini, (1996), Bini and Fiorentino, (1999, 2000), Fortune, (2001).

Well-known ill-conditioned polynomial is Wilkinson polynomial

$$M(s) = \prod_{i=1}^{n} (s - i) \tag{3.34}$$

which is ill-conditioned already for $n = 20$. Although the mapping based rootfinding method has been designed for computing the roots of quasipolynomials, obviously, it can also be used for locating the roots of polynomials. Thus, let us try to locate the roots of (3.34) by means of mapping based method, see the result of the mapping in Fig. 3.37. As can be seen, the located roots truly correspond to the roots of (3.34), i.e., $s_i = i,\ \ i = 1..20$.

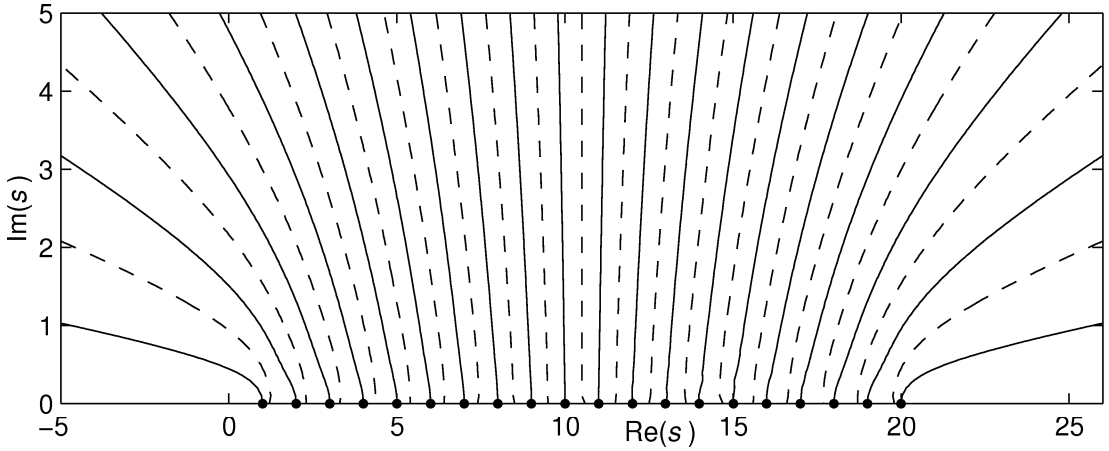

Fig. 3.37 Mapping $M(s)$ given by (3.34), n=20,
$R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed), $\Delta_s = 0.1$

Let us show the effect of ill-conditioned eigenvalues in practise. For example if the coefficient that corresponds to the 19th power of s in (3.34) is multiplied by a factor of 1.0000000005, the distribution of the roots is considerably changed, see Fig. 3.38. Obviously the roots $s_{10}..s_{20}$ (indexing the roots in the left-right direction) are extremely ill-conditioned roots because their displacements are quite large even though the coefficient is multiplied by such a small constant. On the other hand the other roots are not apparently so sensitive.

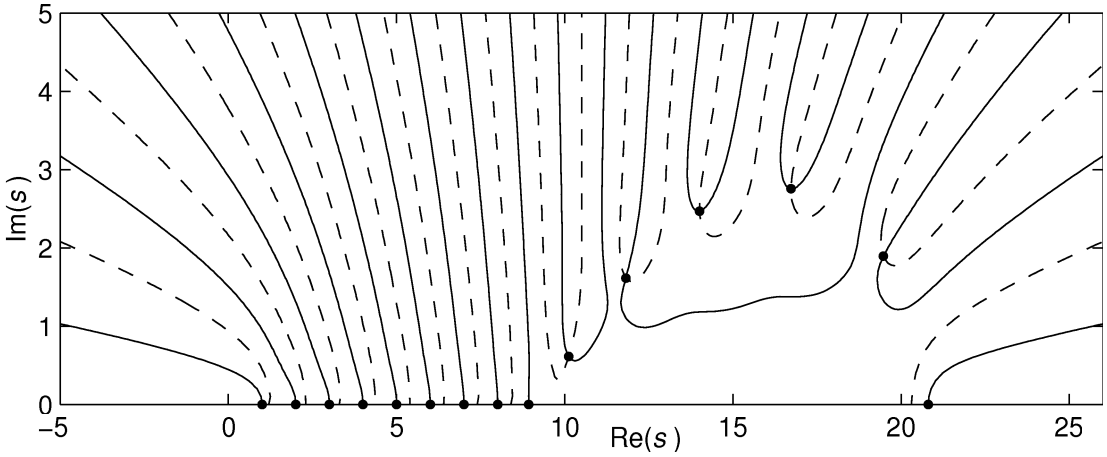

Fig. 3.38 Mapping $M(s)$ given by (3.34), n=20, with coefficient corresponding to 19th power of s multiplied by 1.0000000005, $R(\beta,\omega) = 0$ - solid, $I(\beta,\omega) = 0$ - dashed, $\Delta_s = 0.1$

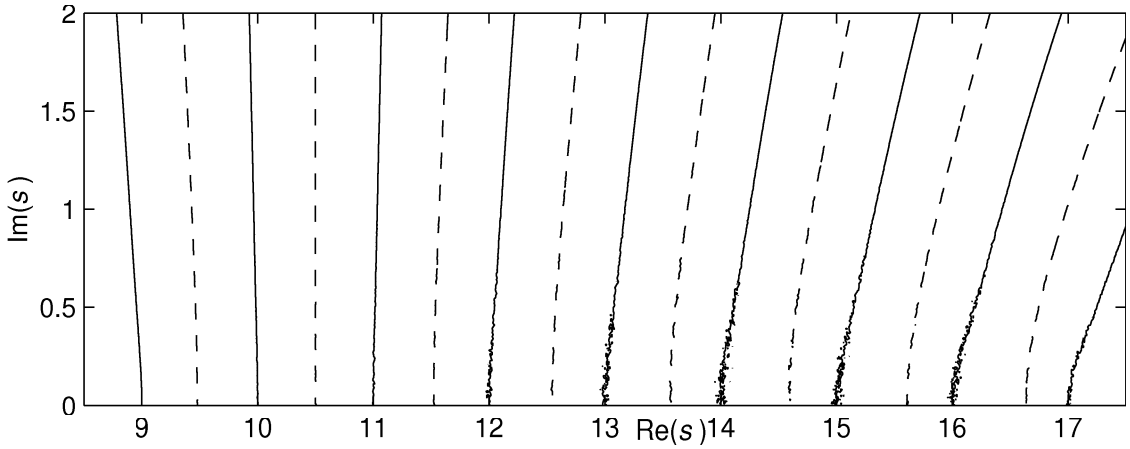

Fig. 3.39 Mapping $M(s)$ given by (3.34), n=20, detailed view
$R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed), $\Delta_s = 0.01$

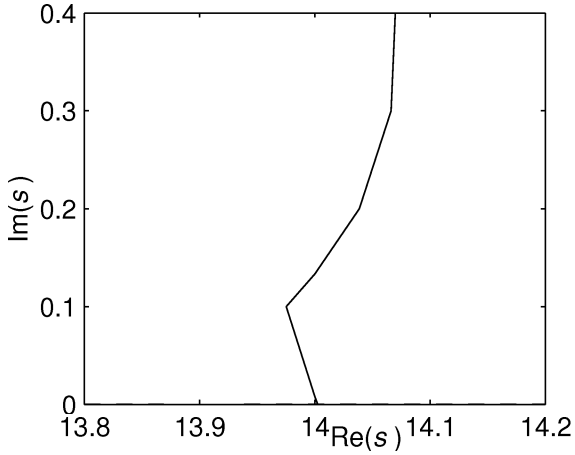

Fig. 3.40 Detail of $R(\beta,\omega) = 0$ of (3.34) from Fig. 3.37, $\Delta_s = 0.1$

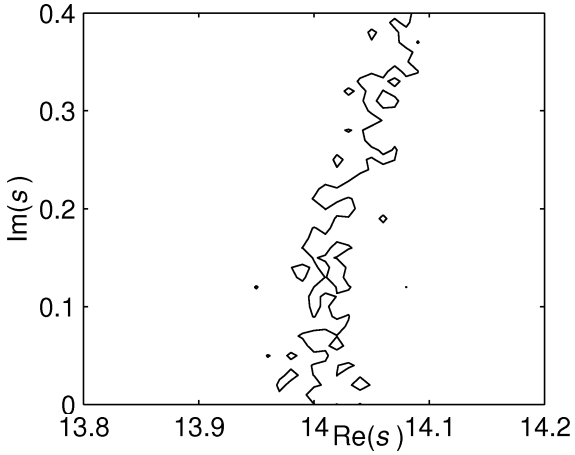

Fig. 3.41 Detail of $R(\beta,\omega) = 0$ of (3.34) from Fig. 3.39, $\Delta_s = 0.01$

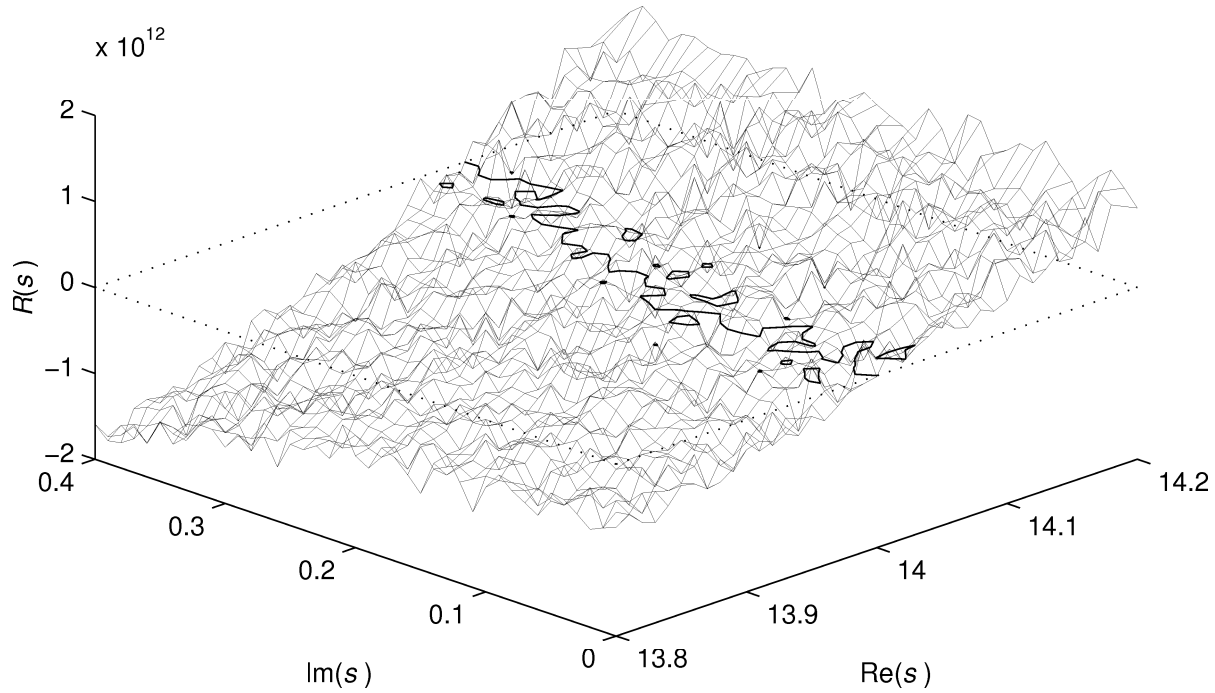

Fig 3.42 Surface $R(s)$ ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.34) and the contours $R(\beta, \omega) = 0$

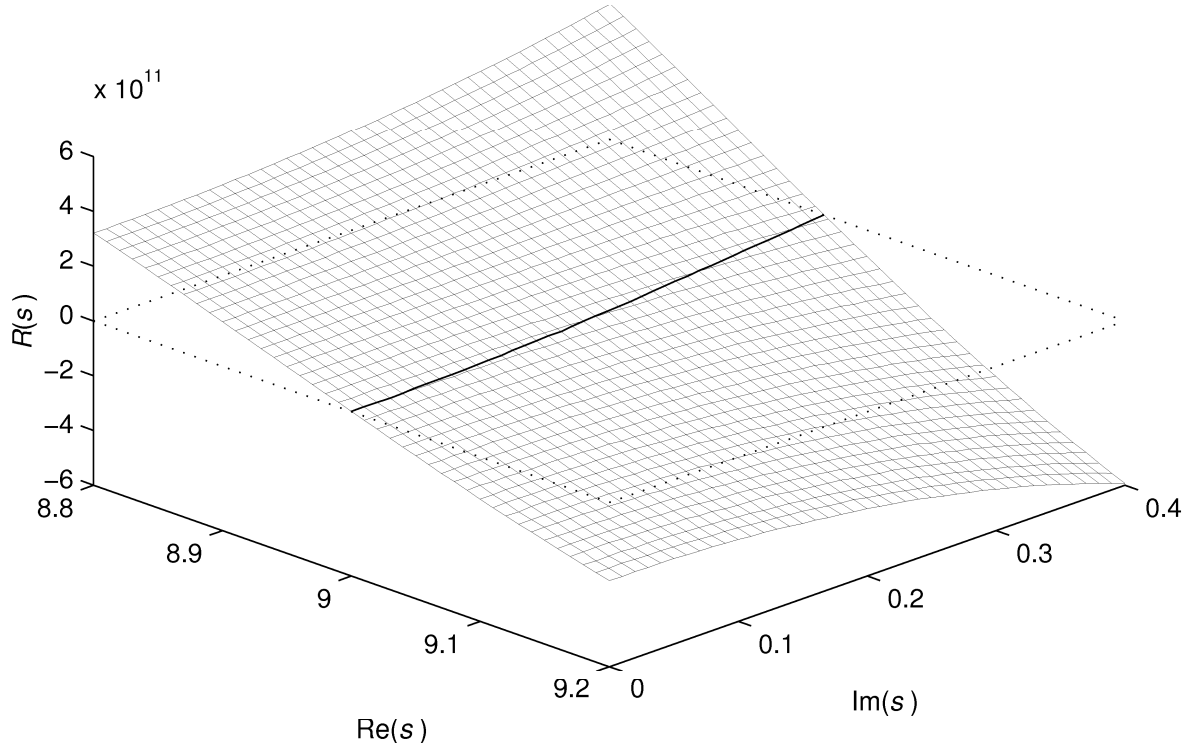

Fig 3.43 Surface $R(s)$ ($s = \beta + \mathrm{j}\omega$) corresponding to $M(s)$ given by (3.34) and the contours $R(\beta, \omega) = 0$

On the basis of the application of the mapping based algorithm performed, the results of which are seen in Fig. 3.37 and Fig. 3.38, one could conclude that the task of computing the roots of (3.34) for n=20 has been managed without any difficulties. However, as will be shown, the task has been solved successfully only because of suitable choice of Δ_s. Let us look at locating some of the most ill-conditioned roots in more detail using $\Delta_s = 0.01$, i.e., ten times less Δ_s than in the previous case. The result of such experiment can be seen in Fig. 3.39. Obviously, something unexpected occurs as the curves approach the real axis. From the certain points, the curves are not smooth and compact. This incompactness of one of the curves is shown in Fig. 3.41 in more detail. As can be seen in Fig. 3.42, the incompactness of the curve is caused by unsmoothness of the surface $R(\beta,\omega)$ in the domain of ill-conditioned roots. The surface is apparently unsmooth only in the domain of ill-conditioned roots since the surface is smooth in the vicinity of the well conditioned root s_9, see Fig. 3.43. The correct result of locating the roots of (3.34) with $\Delta_s = 0.1$ was given by overcoming the noise-like unsmoothness of $R(\beta,\omega)$ by choosing rather rougher grid, compare Fig. 3.40 and Fig. 3.41. However, this trick does not work for (3.34) with higher n. In Fig. 3.44 the mapping of (3.34) with $n = 25$ is shown using $\Delta_s = 0.1$. As can be seen, the algorithm has broken down completely in the domain of ill-conditioned roots because of enhanced unsmoothness of $R(\beta,\omega)$.

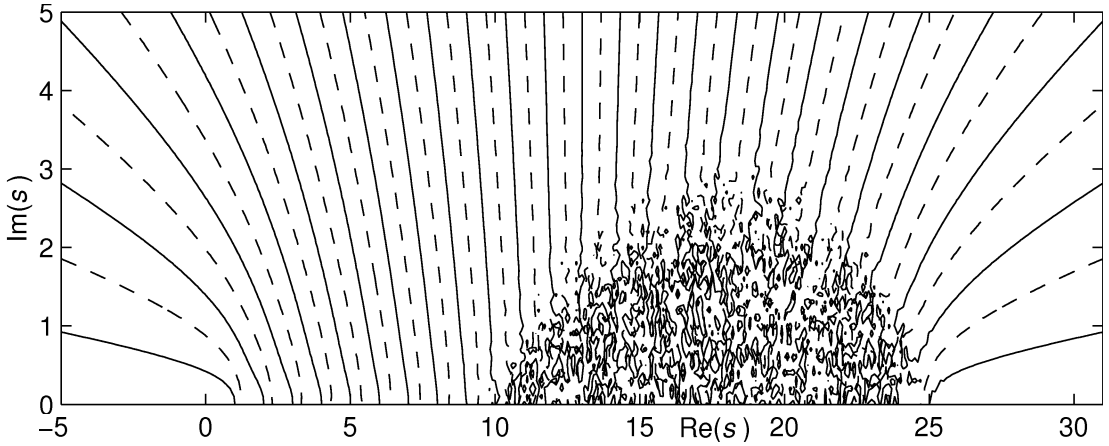

Fig 3.44 Mapping $M(s)$ given by (3.34), n=25
$R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed), $\Delta_s = 0.1$

The results of performed experiments imply that the mapping based algorithm is not reliable if used for locating the roots of ill-conditioned (quasi)polynomials. However, since the method is not iterative, the converse result could have been hardly expected. In case of polynomials, the application capability of the algorithm is comparable with the non-iterative methods that consist in computing the eigenvalues of companion matrices of polynomials (available, e.g., in Matlab as function *roots*). Nevertheless, the mapping based rootfinding algorithm has been primarily designed for analysing the dynamic features of anisochronic (TDS) system. As has been already emphasised, the modelling approach using the delays as the dynamic elements results in low order plant models as a rule. Consequently, the characteristic equations are given by the low degree quasipolynomials for which the probability of occurrence of ill-conditioned roots is relatively low.

3.4.5 Choosing the step of the grid

A very important issue that has not been discussed yet is the task of choosing the step (increment) of the grid Δ_s. Suitable choice of Δ_s is crucial in the successful application of the mapping based rootfinder. The step should be chosen small enough so that the contours were smooth and mapped well. Too large Δ_s may even cause omitting some of the roots. On the other hand, too small Δ_s results in unacceptably large duration of the root-locating process. Solving the task of choosing Δ_s requires balancing these two contradictory requirements. Some aspects of the task will be discussed in the following example.

Example 3.8

Let us compare the duration of the rootfinding process using the mapping based method for locating the approximate positions of the roots and Newton's method to enhance the accuracy of the approximations of the poles. Let us consider the quasipolynomial from Example 3.5, i.e., given by (3.26) and $\mathcal{D} = [-0.4, 0.4] \times [0, 1.2]$. Intuitively, Δ_s should be chosen according to the range of the shorter side of $\mathcal{D}$. The step of the grid that assures visually smooth contours is approximately given by 0.01 of the shorter side. Thus, let us denote this step as $\Delta_{s,0.01}$, duration of the corresponding mapping based root locating process as $t_{L0.01}$, and the duration of the process of the final root assessment by means of Newton's method with $\varepsilon_N = 0.0001$ as $t_{N0.01}$. Considering the boundary values $\mathcal{D} = [-0.4, 0.4] \times [0, 1.2]$, $\Delta_{s,0.01} = 0.008$, $t_{L0.01} = 5.4\text{s}$, $t_{N0.01} = 1.3\text{s}$ (PC: 450 MHz, 128 MB RAM).

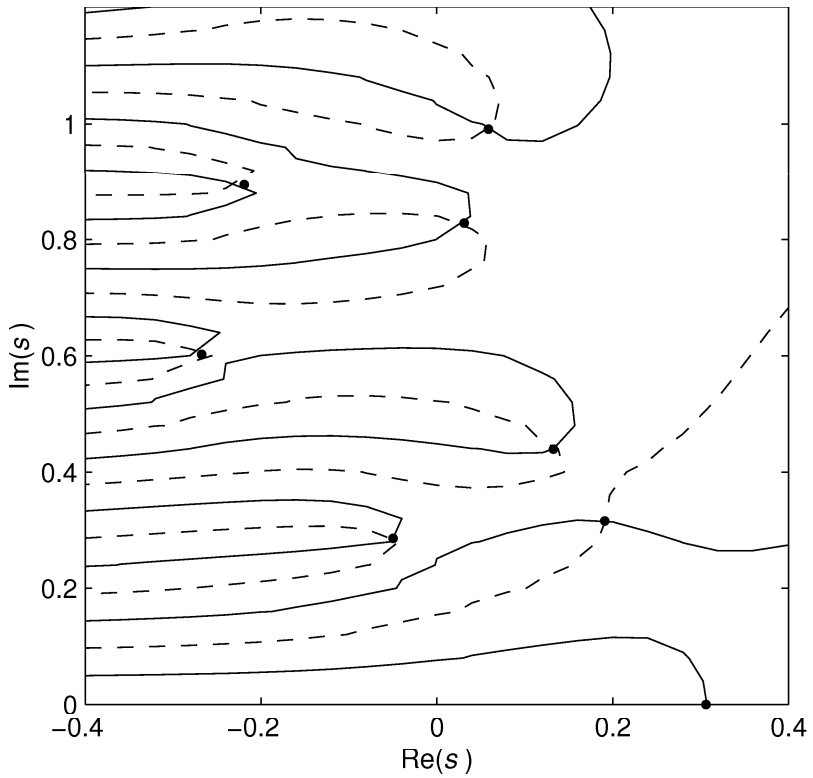

Fig 3.45 Mapping $M(s)$ given by (3.27),
$R(\beta,\omega) = 0$ (solid), $I(\beta,\omega) = 0$ (dashed), $\Delta_s = 0.032$

Tab. 3.4 Time of the periods of rootfinding process depending on Δ_s in time units $t_{L0.01}$ and $t_{N0.01}$

	$\Delta_{s,0.01}/4$	$\Delta_{s,0.01}/2$	$\Delta_{s,0.01}$	$2\Delta_{s,0.01}$	$4\Delta_{s,0.01}$	$8\Delta_{s,0.01}$
t_L	14.3	3.5	1	0.39	0.22	0.19
t_N	0.85	0.98	1	1.15	1.85	3.08
$t_L + t_N$	11.7	3.0	1	0.53	0.54	0.75

In Tab. 3.4 we can see the duration of the periods of the rootfinding process with respect to Δ_s. Time periods are related to $t_{L0.01}$ and $t_{N0.01}$. According to Tab. 3.4, the optimum step of the grid seems to be $\Delta_s = 2\Delta_{s,0.01}$. Further increasing the step does not result in shorter duration of the computation. The time period of locating the root is shorter but on the other hand, due to the rather rough grid the located roots $s_{i,0}$ are far from s_i. Therefore more iterations of Newton's method have to be done which results in longer t_N. On the other hand if $\Delta_s < \Delta_{s,0.01}$, t_L increases considerably while the change of t_N is very small. $M(s)$ mapping obtained using $\Delta_s = 4\Delta_{s,0.01} = 0.032$ is shown in Fig. 3.45. As can be seen, the curves are not smooth which implies the step should be chosen shorter.

With respect to the results obtained in Example 3.8 and with respect to my experience with choosing Δ_s, the step of the grid should be chosen according to the following formula

$$\Delta_s = (0.01..0.03)\min\{(\beta_{\max} - \beta_{\min}),(\omega_{\max} - \omega_{\min})\} \tag{3.35}$$

Such a choice of Δ_s should guarantee both, visual smoothness of the curves and acceptable duration of the root-finding process. If Δ_s chosen according to (3.35) is still too large, i.e., the curves are not formed well, instead of reducing Δ_s, the region $\mathcal{D}$ should be partitioned into several regions in which the task should be solved with smaller Δ_s. The prime guess of the size of $\mathcal{D}$ can be done with respect to the values of the system ultimate frequency considered as the primary scale factor of the s-plane.

3.4.6 Practical aspects of the algorithm implementation

The algorithm based on Weyl's construction with the argument principle based test is suitable to be applied for locating the roots of $M(s)$ if there are not many roots in the region $\mathcal{D}$. Although the algorithm of recursive dividing the suspect regions is quite simple, its computer implementation requires using an elaborate searching approach that would guarantee that none piece of the region remains unsearched. On the other hand, the second rootfinding algorithm based on $M(s)$ mapping can be easily implemented, e.g., by means of Matlab functions. Therefore, I have chosen this mapping based rootfinding algorithm for the final computer implementation. Math environment Matlab has been chosen for the implementation because its environment allows user-friendly combining of the predefined and user-written functions in the program. Another reason of choosing Matlab is given by the fact that the program is quite familiar in the field of control engineering.

To sum up the algorithm for locating the roots of (quasi)polynomials can be summarised as follows:

Algorithm 3.1 Mapping based rootfinder

1 The region $\mathcal{D} = [\beta_{\min}, \beta_{\max}] \times [\omega_{\min}, \omega_{\max}]$ in which the roots of $M(s)$ are to be found and suitable step of the grid Δ_s are defined.

2 The region $\mathcal{D}$ is covered by the grid of nodes given by (3.23)

3 For each of the nodes $\beta_k + \mathrm{j}\omega_l$, the function $M(\beta_k, \omega_l)$ is evaluated and the values of $R(\beta_k, \omega_l)$ and $I(\beta_k, \omega_l)$ are found by splitting $M(\beta_k, \omega_l)$ into real and imaginary part. Thus the values are stored in matrix $\mathbf{R}^{\mathrm{d}}$ in the form of (3.24) and in the analogous matrix $\mathbf{I}^{\mathrm{d}}$.

4 Using a contour plotting method, the zero-level contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ are mapped over the values in matrices $\mathbf{R}^{\mathrm{d}}$ and $\mathbf{I}^{\mathrm{d}}$. The points of the contours given by the couples β and ω are stored in matrix $\mathbf{R}^0$ in the form of (3.28) and in the analogous matrix $\mathbf{I}^0$

5 3D curves I_R are evaluated over the contours $R(\beta,\omega) = 0$. The values of $I_R(h)$ given by (3.29) are obtained by evaluating $I(\beta,\omega)$ for the values stored in $\mathbf{R}^0$.

6 The intersection points of the contours $R(\beta,\omega) = 0$ and $I(\beta,\omega) = 0$ are located by means of searching for the values $\beta_{r_i} + \mathrm{j}\omega_{r_i}$ and $\beta_{r_i+1} + \mathrm{j}\omega_{r_i+1}$ from $\mathbf{R}^0$ for which I_R change the sign, see (3.30). Each of the roots of $M(s)$ located in $\mathcal{D}$ is approximated by $s_{i,0} = \beta_{r_i} + \mathrm{j}\omega_{r_i}$.

7 The accuracy of the approximations of the roots is enhanced by means of Newton's iteration method (3.12) with the starting values $s_{i,0}$, i=1,2,....

Unlike polynomials that are represented by the row matrix of the coefficients corresponding to the powers of the operator s, quasipolynomial are not so easily treatable. If a quasipolynomial represents the characteristic functions of TDS with the various distributions of the delays, its terms corresponding to the powers of s acquire quite complicated forms, see, e.g, Example 3.5. Thus the best representation of quasipolynomials seems to be the form $M(s)$ involving the operator s. Such a requirement can be performed by means of Symbolic Math Toolbox of Matlab that allows symbolic variables to be defined. Symbolic Math Toolbox involve symbolic computation into Matlab's numeric environment using the kernel of Maple as the computational engine. Defining operator s as the symbolic variable, the quasipolynomial can be written directly in this operator and none special form of quasipolynomial is required.

Using the calculus of Symbolic Math Toolbox and functions of Matlab I have programmed the function *aroots* performing the Algorithm 3.1. The command line function *aroots* is of the following syntax

P=aroots(M, $\mathcal{D}$, Δ_s, ε_N)

see Apendix 1, where M is the quasipolynomial in the predefined symbolic variable s, $\mathcal{D}$ is the suspect region, Δ_s is the step of the grid, ε_N is the absolute value of the maximum difference of the root approximations resulting from two successive steps of Newton's

iteration and P is the vector of computed roots. The implementation of the algorithm will be shown in the following example.

Example 3.9

Let us compute the root approximations of the function $M(s)$ given by (3.27) from Example 3.5 in the region $\mathcal{D} = [-0.4, 0.4] \times [0, 1.2]$ with $\Delta_s = 0.01$ and $\varepsilon_N = 0.0001$. The following sequence of the commands written in the command line of Matlab will perform the task

```
>>syms s

>>A=[-exp(-8.4*s)     exp(-4.1*s)     exp(-6.6*s);
      (exp(-5.2*s)-exp(-12.5*s))/7.3/s     -exp(-4.3*s)     exp(-3.7*s);
      exp(-7.8*s)     (exp(-6.5*s)-exp(-18.9*s))/12.4/s     exp(-5.2*s)]

>>P=aroots(det(s*eye(3)-A), [-0.4   0.4   0   1.2], 0.01, 0.0001)
```

In the first line, the symbolic variable s is defined using the function *syms*. The matrix $\mathbf{A}(s)$ is then written using this symbolic variable. Finally the function *aroots* is written in the command line according to its syntax with the characteristic function evaluated using the command *det* (evaluating the determinant). The resultant root approximations are stored in the vector P

```
>> P =
   0.0587 + 0.9910j
   0.3061
   0.0310 + 0.8285j
  -0.2190 + 0.8946j
  -0.2676 + 0.6025j
   0.1326 + 0.4395j
  -0.0497 + 0.2857j
   0.1908 + 0.3158j
```

As can be seen, using the function *aroots* for locating the roots is quite easy. The only drawback of using the symbolic calculus is rather large memory consumption in case of dealing with the quasipolynomials resulting from higher order TDS with complicated distributions of the delays, i.e., if the quasipolynomials acquire too complicated forms.

To conclude, the mapping based rootfinder given by algorithm 3.1 may be applied to compute both the poles and zeros of both retarded and neutral system with lumped and distributed delays (according to the rules for their distribution, see (1.18)). Naturally, the algorithm can also be used to compute the roots of (low degree) polynomials and the roots of exponential polynomials (which determine the essential spectrum of the neutral equations). In fact, the class of functions whose roots can be computed using the rootfinder is broader. Since the subject solved in this thesis deals with the analysis of TDS, the applicability of the mapping based rootfinder to further functions is not investigated. The applicability of the mapping based rootfinder is limited to the well-conditioned functions, which rather bounds the maximum degree of the (quasi)polynomials which can be analysed using the rootfinder. On the other hand, the anisochronic approach provides low order models of real plants as a rule. Thus the mapping based rootfinder is very valuable tool in the analysis of the dynamics of TDS.

4. APPLICATION OF MAPPING BASED ROOTFINDER IN ANALYSIS AND SYNTHESIS OF TIME DELAY SYSTEMS

4.1 Introduction

In this part of the thesis I am going to show some results of the analysis and the synthesis of TDS achieved by using the developed rootfinding algorithm based on characteristic function mapping. Features of both retarded and neutral systems will be investigated on the basis of computed dominant parts of their spectra of poles and zeros.

In section 4.2, according to the second objective of this thesis stated in Chapter 2, the features of the first order anisochronic model will be investigated (It should be noted, that the model is infinite dimensional in fact. By the order, we consider here the number of integration operators used in the model). The model will be further considered as the basic element in the anisochronic modelling approach. Even though the model is described only by a single functional differential equation, thanks to the delays involved, the model can be used to approximate the features of the systems with higher order dynamics. First, the identification method based on the relay feedback test will be extended to the first order anisochronic model with two delays. This method has proved to be a suitable method for identifying the parameters of real plants. Since the identification of the TDS is not the task to be solved in this thesis, only the results of the extension of the mentioned identification method will be presented. Further in section 4.2, the features of the first order model with more than one delay will be investigated. Particularly, the potentials of assessing the dominant poles and zeros will be studied.

In section 4.3, beyond the framework of the objectives of this thesis, some of the features of the internal model control (IMC) design based on the first order anisochronic model will be investigated. Particularly, it will be shown, that the IMC design may result in the neutral character of the closed loop system. This feature will be shown in the several examples and the character of the spectra of the closed loop system will be studied. As the prime result of the section, the robust IMC design will be introduced.

The basic feature of TDS is the infinite spectrum of the poles. Even though TDS have infinitely many poles, the system dynamics are determined by a group of few poles as a rule. In section 4.4, according to the third objective of this thesis stated in Chapter 2, I will introduce the method for selecting the group of the significant poles of TDS. The method is based on the generalised Heaviside expansion of the particular input-output transfer function. Using the expansion, the transfer functions of the system modes are obtained. The evaluation of the dynamics is based on evaluating the weighting functions corresponding to the transfer functions of the modes.

In sections 4.5 and 4.6, according to the fourth objective of this thesis stated in Chapter 2, the potentials of the pole placement methods based on the coefficient feedback applied to TDS will be investigated. First, the gradient based pole placement method will be introduced. The method arises from the linearity of the characteristic function of the closed loop system with respect to the feedback parameters. The method may be used for direct pole placement. In section 4.6, the modification of the continuous pole placement method using the gradient based pole placement design and the quasipolynomial mapping based rootfinder will be introduced. As will be shown in section 4.7, the application of this rootfinding algorithm allows the method of the continuous pole placement to be used also to stabilize a class of the neutral systems. In section 4.8, the features of the continuous and direct pole placement methods will be discussed and the strategy suitable to obtain favourable pole placement result will be suggested.

4.2 First order TDS, features and identification

4.2.1 First order anisochronic model with one delay

In many industrial applications, the following SISO first order model with the input delay can be encountered

$$G(s) = \frac{y(s)}{u(s)} = \frac{K \exp(-s\tau)}{Ts+1} \tag{4.1}$$

where K is static gain coefficient, T is time constant and τ is input time delay, see, e.g., Goodwin, et al, (2001). In fact, according to Hang et al., (1993), the model is sufficient for adequate approximation of most of the industrial processes. An identification method of model (4.1) parameters that proved to be suitable for practical implementation is the method based on relay feedback test. For large classes of processes, relay feedback gives an oscillation with a period close to the process ultimate period t_u ($t_u = 2\pi/\omega_u$, where ω_u is the ultimate frequency). The process ultimate gain k_u is approximately given by Åström and Hägglund (1984,1988a,b) as

$$k_u = \frac{4u_m}{\pi y_m} \tag{4.2}$$

Given k_u and ω_u as the results of relay feedback test and K resulting, e.g., from the process step response, the remaining two parameters of model (4.1) can be obtained from

$$\tau = \frac{\sqrt{k_u^2 K^2 - 1}}{\omega_u} \tag{4.3}$$

$$T = \frac{\pi - \arctan(\omega_u \tau)}{\omega_u} \tag{4.4}$$

see Hang, et al., (1993). In Wang, et al., (1999) a biased relay ($|u_{min}| \neq |u_{max}|$) is suggested to be used instead of the symmetric relay. Using such a modification, all the parameters of model (4.1) can be identified from the relay feedback test. Using this biased relay, the oscillations are not symmetric and the parameter K can be computed from

$$K = \frac{\int_0^t y(h)dh}{\int_0^t u(h)dh}, \quad t >> t_u \tag{4.5}$$

In my experience, the biased relay based identification is not suitable to be applied to real plants because the values of integrals in (4.5) are very sensitive to the deflections of the system from its operational point (caused, e.g., due to the process noise) during the identification process. Moreover, the non-linearity of the plant may cause difficulties in computing K from (4.5). An alternative way of identifying the parameters only from the relay feedback test consists in the estimation of the input delay τ from the interval between an extremum of $y(t)$ and the preceding relay switch. This method is more robust than the method based on the biased relay. However, if the plant dynamics are higher order ones, it may be difficult to identify the extremum of $y(t)$. If the delay is identified in this way, the remaining

parameters of (4.1), i.e., T and K can be computed from the critical parameters k_u and ω_u by formulas analogous to (4.3) and (4.4).

It should also be noted, that in real plant applications, the relay with hysteresis should be used in order to make the identification less sensitive to the measurement noise. Using the relay with hysteresis implies that the critical parameters k_u and ω_u resulting from the oscillations do not correspond to the critical parameters of the system. The parameters correspond to the point of the system frequency response which is of slightly lower frequency than the critical one. Nevertheless, if the hysteresis is small, the values k_u and ω_u can be considered as the critical parameters.

Although first order model (4.1) is referred to as the suitable model for describing real plants, the only one mode of the model given by the pole $\lambda_1 = -1/T$ provides rather restricted potentials in fitting the true system dynamics. Therefore, model (4.1) is suitable only for rather rough description of non-oscillatory plants. Moreover, this model usually describes truly the system dynamics in a rather narrow frequency range.

4.2.2 First order anisochronic model with two delays

In order to extend the applicability of the first order model, let us introduce another delay η into the first order anisochronic model. Besides the delayed system input, let us also use delayed state variable in the model. The anisochronic model then acquires the following form

$$G(s) = \frac{y(s)}{u(s)} = \frac{K \exp(-s\tau)}{Ts + \exp(-s\eta)} \tag{4.6}$$

see Vyhlídal and Zítek, (2001). Analogous but second order model has been used in Zítek, (1998). As has been shown in the mentioned references, thanks to the delay in the denominator, model (4.6) can be used for describing the plants that are usually described by higher order models. As has been shown in Vyhlídal and Zítek, (2001), if the parameters K and τ are known, the remaining two parameters of model (4.5) can be obtained from

$$\eta = \frac{\pi - \arccos\{Kk_u \cos(\omega_u \tau)\}}{\omega_u} \tag{4.7}$$

$$T = \frac{\sin(\omega_u \eta) - \tan(\omega_u \tau)\cos(\omega_u \eta)}{\omega_u} \tag{4.8}$$

where k_u (given by (4.2)), and ω_u result from the relay feedback test. The parameters K and τ can be obtained, e.g., from the system step response, where $K = \Delta y / \Delta u$ and τ is determined by drawing a tangent line at the inflexion point of the step response and locating the intersection point with the time axis, see Fig. 4.2. The identification procedure will be shown in the following example.

Example 4.1

Let us approximate the dynamics of the tenth order classical model

$$G_p(s) = \frac{1}{(2s+1)^{10}} \tag{4.9}$$

by first order anisochronic models (4.1) and (4.6) using the identification method based on the relay feedback experiment and the step response. The ultimate frequency $\omega_u = 0.163$ ($t_u = 38.5$) and gain $k_u = 1.654$ ($y_m = 0.77$, $u_m = 1$) result from the relay feedback test applied to system (4.9), see Fig. 4.1. The input delay of model (4.6) $\tau_2 = 11.5$ is obtained from the step response of (4.9), see Fig. 4.2. Obviously considering $K_1 = K_2 = 1$, the remaining parameters of model (4.1) are obtained from evaluating (4.3) and (4.4), which results in $\tau_1 = 13.6$, $T_1 = 8.1$, and the remaining parameters of model (4.6) are obtained from evaluating (4.7) and (4.8), which results in $T_2 = 15$, $\eta_2 = 6.4$.

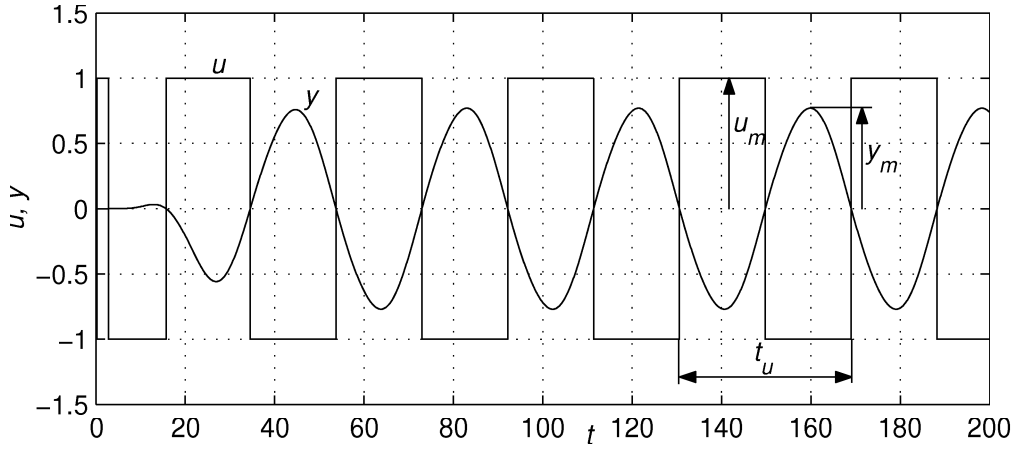

Fig. 4.1 Relay feedback experiment applied to model (4.9)

As can be seen from the step and frequency responses, see Fig. 4.2 and Fig. 4.3, from the practical point of view, first order model with only one delay (4.1) approximates model (4.9) quite well. However, comparing the characteristics of model (4.1) with the characteristics of model with two delays (4.6), the approximation obtained using (4.1) is considerably worse. The step responses of (4.9) and (4.6) get very close to each other after the response of (4.6) attains the response of (4.9). The discrepancies of the responses close to $t = \tau$ are given by the fact that the first order model does not filter the step change of the input as it is done in case of using higher order models. Therefore, the response of a first order model can not be smooth in the vicinity of $t = \tau$. Analogously, the frequency responses of (4.9) and (4.6) are very close to each other up to ultimate frequency. They are quite close to each other even in the third quadrant of the complex plane. Such a good approximation of the higher order model (4.9) by first order anisochronic model (4.6) is achieved thanks to the denominator delay η. In fact, the higher order dynamics is substituted by the anisochronic relation.

Due to practical point of view, it is reasonable to avoid using the step response in the identification process. Performing the relay feedback test, the approximation of the critical point of system response is obtained allowing two parameters of (4.6) to be computed. Thus, if we find another point of the frequency response, we will be able to compute all the parameters of (4.6) (using a numerical method). If an integrator element is put in front of the relay element in the closed loop, a lower frequency of input excitation signal is obtained corresponding to the point of the system frequency response on the negative imaginary axis (with argument $\Phi = -\pi/2$). Using a delay instead of the integrator, other points of the frequency response can be obtained (depending on the delay length). The parameters can be computed from the overdetermined set of equations, e.g., by least square method. However, it should be noted that the relay feedback test provide only the approximate values of the frequency

response points and the accuracy of the approximation falls down with the decreasing frequency.

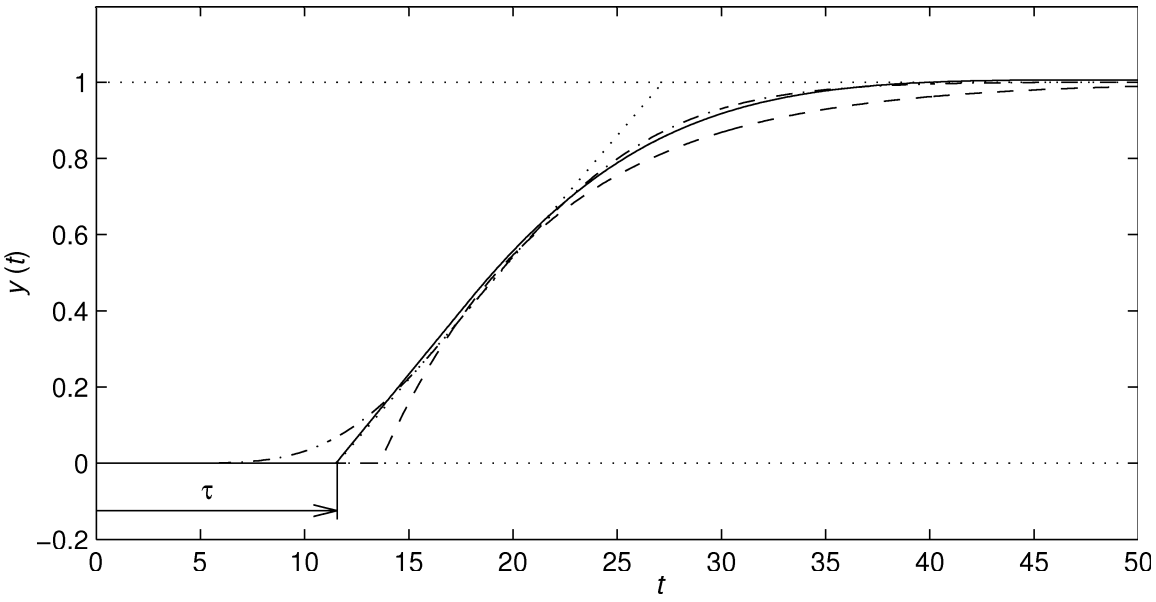

Fig. 4.2 Step responses: model (4.9) (dash-dotted) with tangent line in the inflexion point (dotted), first order approximation (4.6) (solid), first order approximation (4.1) (dashed)

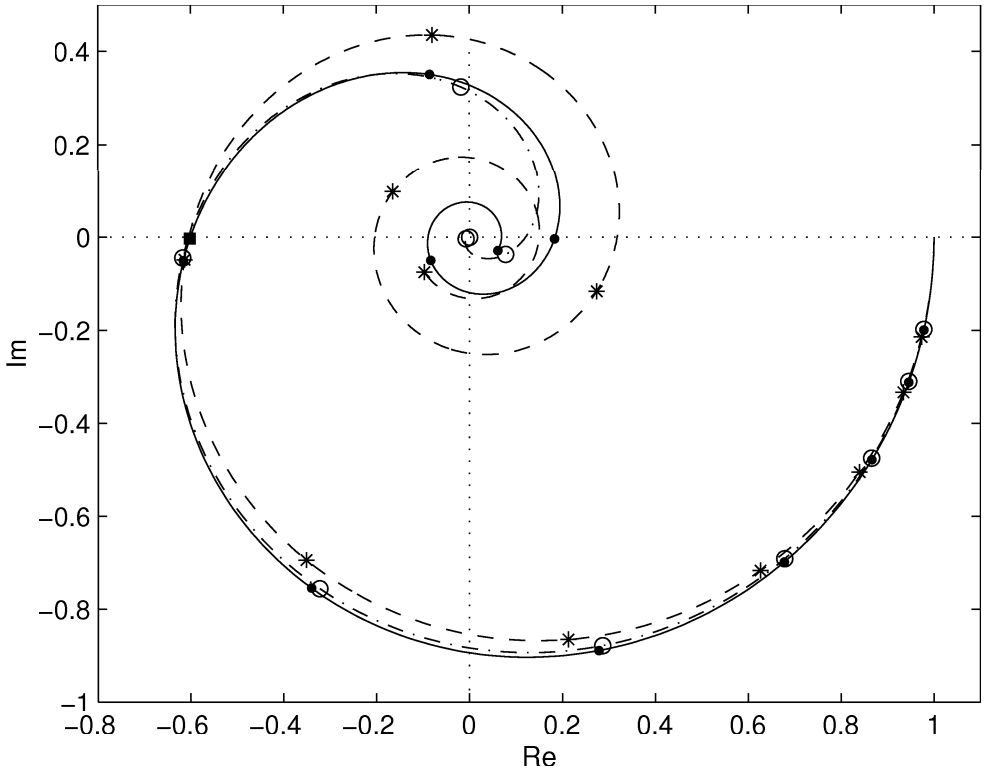

Fig. 4.3 Frequency responses: model (4.9) (dash-dotted+empty circles), first order approximation (4.6) (solid and black circles), first order approximation (4.1) (dashed and asterisks). Marked frequencies: logarithmically spaced vector [0.01 1] into 10 segments. Prescribed ultimate frequency marked by black square

4.2.3 Dynamics of first order model with delay in denominator

Using the mapping based algorithm for computing the quasipolynomial roots, let us investigate the distribution of the roots of the quasipolynomial

$$M(s) = s + \exp(-s\eta) \tag{4.10}$$

that corresponds to the characteristic function of (4.6) if the time constant T is considered as the time unit, i.e., $T/T \to 1$, $\eta/T \to \eta$. In Fig. 4.4 and Fig. 4.5 we can see the trajectories of the roots of (4.10) with respect to the value of the delay η. Let us continuously increase the value of η and let us observe the trajectories of the roots of (4.10) closest to the s-plane origin. Obviously if $\eta = 0$, the equation (4.10) has only one real root $s_1 = -1$. This real root moves to the left as the value of η increases (note that since $\eta \neq 0$ equation (4.10) is transcendental, with infinitely many solutions). The other real root emerges from the minus infinity and moves towards the origin of the complex plane as η increases, see Fig. 4.5. Both the real roots have the same position $s_{1,2} = -\exp(1)$ for $\eta = \exp(-1)$. As can be seen, there are also two complex roots in the region shown in Fig. 4.5 for $\eta = \exp(-1)$ (market with black squares). As η further increases, the former real roots become complex conjugate pair following the trajectories seen in Fig. 4.4. The trajectories of this couple of roots intersect the imaginary axis at the values $\pm$ j for $\eta = \pi/2$. Note that this intersection point with the imaginary axis is common for all the root trajectories and one of the root couple occupy this position as $\eta = \pi/2 + 2\pi k$, k=1,2,.., see Fig. 4.4. As can be estimated from Fig. 4.4, the common destination of all the roots is the origin of the s-plane.

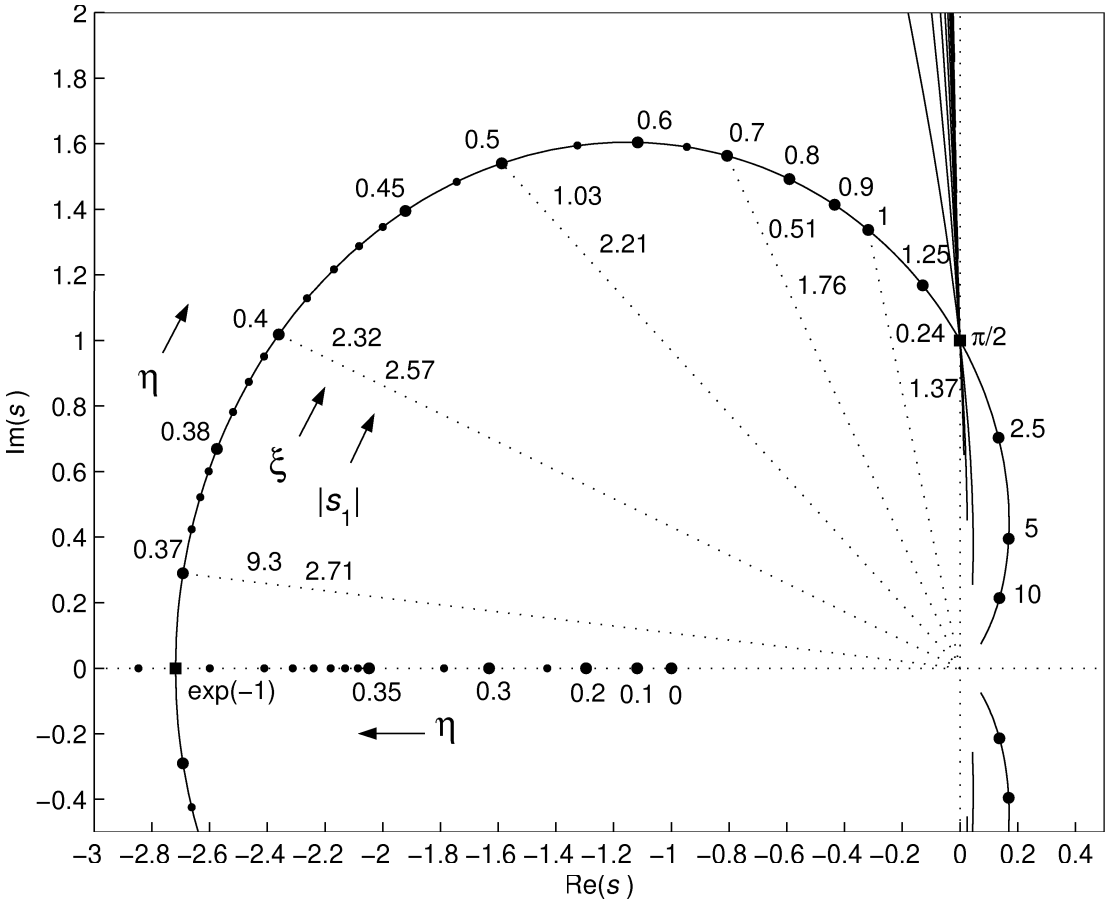

Fig. 4.4 The trajectories of the dominant pair of roots of (4.10) with respect to the value of $\eta = \eta/T$, detailed view of Fig. 4.5

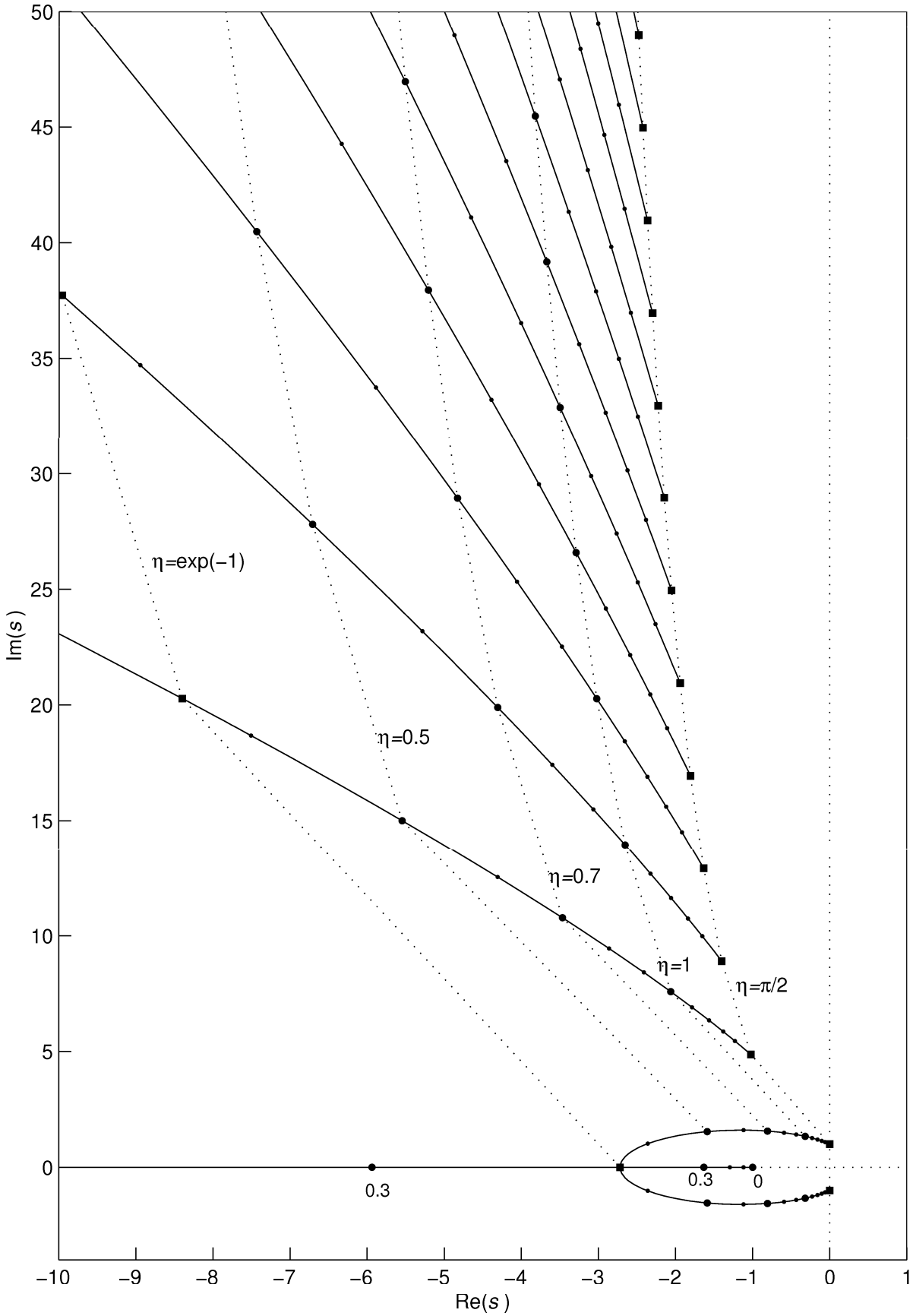

Fig. 4.5 The trajectories of the roots of (4.10) (that are closest to the *s*-plane origin) with respect to the value of $\eta = \eta/T$

On the basis of the obtained root trajectories let us evaluate the dynamics of the first order anisochronic model (4.6). The dynamics of the system is primarily non-oscillatory, at first determined only by one real root. As this root is close to value $s_1 = 0.3$ also the other real root starts influencing the dynamics considerably. Obviously, the dynamics of the system are determined by this pair of roots for all the values of η (for the stable dynamics, i.e., $\eta < \pi/2$). As this pair approaches the stability boundary, the ratio of the magnitude of this pair and the magnitude of the second closest pair to the *s*-plane origin is still close to 5. The dynamics of model (4.6) with respect to η can be read from Fig. 4.4 where the values of magnitude of the dominant couple of roots and their relative damping $\xi = |\beta|/\omega$ are shown. To conclude, although model (4.6) is given only by a single first order functional differential equation, its dynamics is rather analogous to the second or higher order delay free models (if the model dead time is approximated by the input delay τ).

4.2.4 Low order anisochronic model with zeros

The dynamics of the introduced first order anisochronic models are determined by the system poles only. Involving more delays in the numerator, which introduce the zeros into the model dynamics, further extends the applicability of the first order anisochronic model. One possible way of involving a zero-effect to the system dynamics consists in using the model

$$G(s) = \frac{K(Ls + \exp(-s\chi))\exp(-s\tau)}{Ts + \exp(-s\eta)} \tag{4.11}$$

where the ratio χ/L determines the distribution of the roots of $Ls + \exp(-s\chi) = 0$ (the positions of the zeros of (4.11)) in the same way as the ratio η/T determines the distribution of the poles, see Vyhlídal and Zítek, (2001). However, using (4.11) is effective only in case of the requirement to involve the zeros that are located on the left half of the complex plane. If the system is non-minimum phase system, model (4.11) cannot be used because the equation $Ls + \exp(-s\chi) = 0$ does not have positive real roots for any χ/L. If the zero is positive real and single given by $\mu = 1/L$, it can be added to the model simply by using $-Ls + 1$ instead of $Ls + \exp(-s\chi)$ (using $-Ls + \exp(-s\chi)$ does not bring about considerable merits because its dominant root is positive real for any $\chi > 0$). More difficult task is to involve the dominant complex zeros with the positive real parts. Theoretically, it is possible to use the term $Ls + \exp(-s\chi)$ but as $|\mathrm{Re}(\mu)|/\mathrm{Im}(\mu) > 0.5$ the ratio χ/L becomes very large, which is not convenient from the numerical point of view. Another problem arising from the use of model (4.11) is that the degree of numerator is equal to the degree of denominator, i.e., there is a direct input-output link in the model. To avoid such an inconvenient model structure, instead of first order anisochronic model (4.11), the following second order model may be used

$$G(s) = \frac{K(Ls + \exp(-s\chi))\exp(-s\tau)}{(T_1 + 1)(Ts + \exp(-s\eta))} \tag{4.12}$$

with an additional mode with time constant T_1.

The alternative way of involving the zeros into the first order model that do not have the drawback of the equal degrees of numerator and denominator consists in using the following model with the numerator of the form of exponential polynomial

$$G(s) = \frac{K(1 - a\exp(-s\chi))\exp(-s\tau)}{Ts + \exp(-s\eta)} \tag{4.13}$$

The zeros of system (4.13) are the roots of the following equation

$$N(s) = 1 - a\exp(-s\chi) = 0 \tag{4.14}$$

If a>0, the real solution of the equation is given by

$$s = -\frac{1}{\chi}\ln\left(\frac{1}{a}\right) \tag{4.15}$$

In fact, variable s is the complex variable, i.e., $s = \beta + \mathrm{j}\omega$, thus the roots are the solutions of the equations

$$\mathrm{Re}(N(s)) = 1 - a\exp(-\chi\beta)\cos(\chi\omega) = 0 \tag{4.16}$$

$$\mathrm{Im}(N(s)) = a\exp(-\chi\beta)\sin(\chi\omega) = 0 \tag{4.17}$$

which result from splitting equation (4.14) into real and imaginary parts. Separating the exponential term from (4.16)

$$\exp(-\beta\chi) = \frac{1}{a\cos(\omega\chi)} \tag{4.18}$$

and substituting (4.18) into (4.17), the following expression results

$$\tan(\chi\omega) = 0 \tag{4.19}$$

Since (4.19) is satisfied for $\omega = k\pi/\chi$ and the right-hand side of (4.18) has to be positive to obtain real β, the roots $s = \beta + \mathrm{j}\omega$ of (4.14) are given by

$$\beta = -\frac{1}{\chi}\ln\left|\frac{1}{a}\right| \tag{4.20}$$

$$\begin{matrix}\omega = \pm 2k\pi/\chi, & \text{if } a > 0 \\ \omega = \pm(2k+1)\pi/\chi, & \text{if } a < 0\end{matrix}, \quad k = 0,1,2,... \tag{4.21}$$

Thus, by means of a and χ, we can assign the horizontal chain of the roots arbitrarily in the complex plane. Prescribing the real parts of the roots β yields (from (4.20))

$$|a| = \exp(\beta\chi) \tag{4.22}$$

and the parameter χ results from

$$\chi = \frac{2\pi}{\omega_\mathrm{p}} \tag{4.23}$$

where ω_p prescribe the spacing of the imaginary parts of the roots. If a is chosen positive, equation (4.14) has one real root. The closest complex root (of the horizontal chain) to the real one has imaginary part equal to ω_p. If a is chosen negative, equation (4.14) does not have a real root. In this case, the roots of the chain closest to the real axis have the imaginary parts

equal to $\pm\omega_p/2$. To sum up if a >0, the roots are given as $s_1=\beta$ and $s_{2k,2k+1}=\beta\pm j(k\omega_p)$, k=1,2,... and if a<0, the roots are given as $s_{2k+1,2(k+1)}=\beta\pm j((2k+1)\omega_p/2)$, $k=0,1,...$ Thus, by means of the difference equation, we can assign either one real dominant zero or the pair of complex conjugate zeros. The procedure of assigning the zeros of a system will be shown in the following example.

Example 4.2

Let us assign zeros $\mu_{1,2}$=1+j to a system using the numerator $N(s)=1-a\exp(-s\chi)$. First of all, the delay χ is to be computed from (4.23). Since the root of (4.14) is complex conjugate, $\omega_p=2$ thus $\chi=\pi$ and according to (4.22) $a=-\exp(\pi)$. As can be seen in Fig. 4.6, the prescribed root has been obtained. The effect of changing the sign of a is shown in Fig. 4.7. As can be seen, in agreement with (4.21), $N(s)$ has one real root. Obviously, in case of choosing one real root to be assigned, the spacing of the complex roots can be chosen by the parameter χ. Thus, assigning a real zero, the impact of the complex zeros of the chain on the final dynamics can be reduced using smaller values of χ.

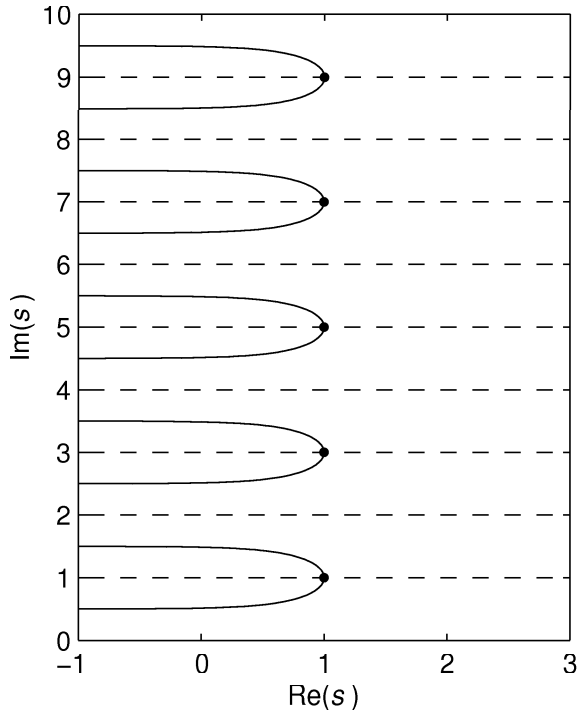

Fig. 4.6 Roots of (4.14) for $\chi=\pi$ and $a=-\exp(\pi)$, Re($N(s)$)=0-solid, Im($N(s)$)=0-dashed

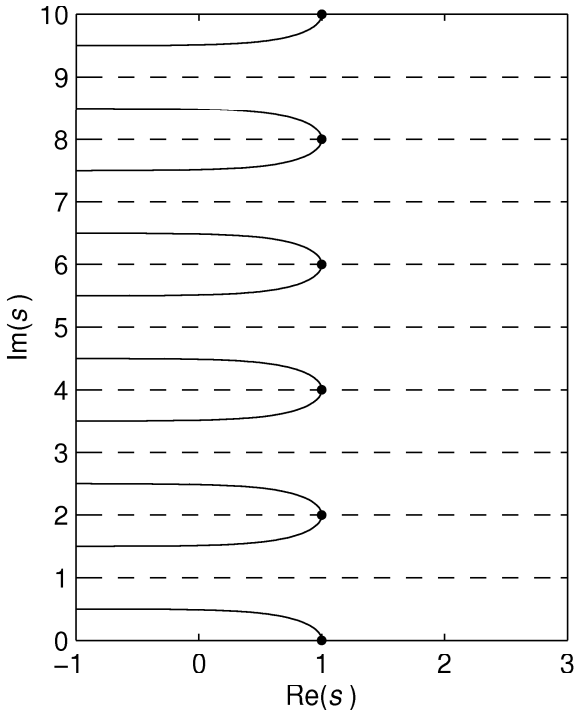

Fig. 4.7 Roots of (4.14) for $\chi=\pi$ and $a=\exp(\pi)$, Re($N(s)$)=0-solid, Im($N(s)$)=0-dashed

Example 4.3

In Goodwin, et al., (2001), (Example 4.6, p. 80) the role of a real zero in the system dynamics is demonstrated on the step response of the system

$$H(s) = \frac{-s + \mu}{\mu(s+1)(0.5s+1)} \tag{4.24}$$

Let us compare the step responses of (4.24) with the step responses of the following system

$$G(s) = \frac{1 - a\exp(-s\chi)}{(1-a)(s+1)(0.5s+1)} \tag{4.25}$$

which has the same values of the dominant zero as the values of the zero of (4.24)

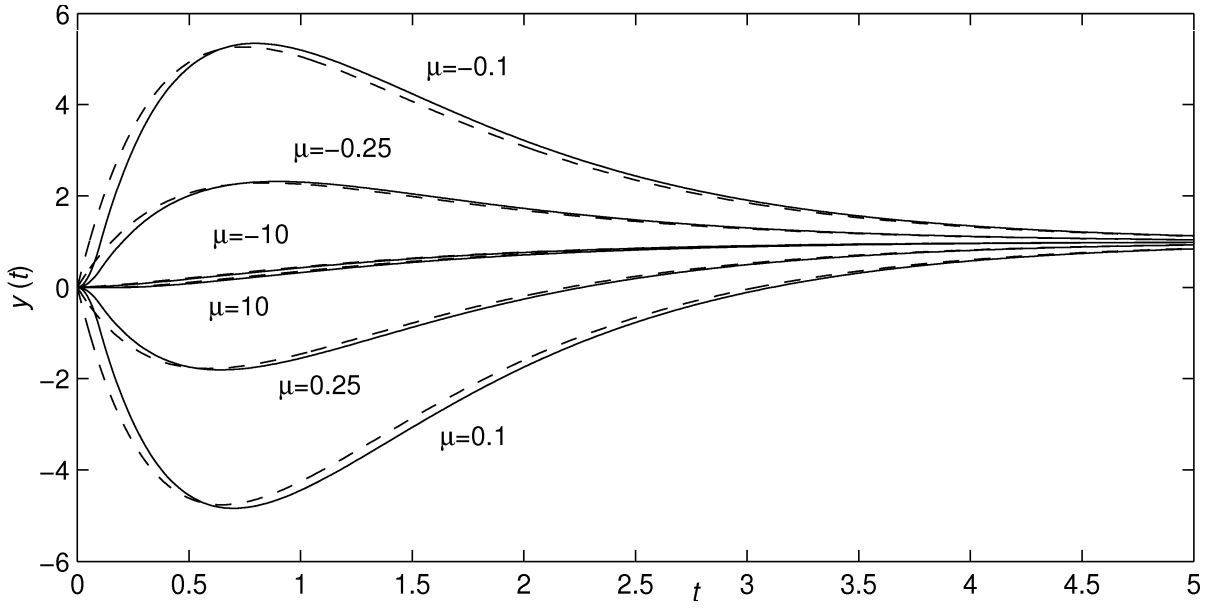

Fig. 4.8 Effect of different zero locations to the step responses of systems (4.24) (dashed) and (4.25) (solid)

The result of the comparison of the step responses of (4.24) and (4.25) is seen in Fig. 4.8. The delay in numerator of (4.25) is chosen $\chi = 0.1$ and the parameter $a = \exp(\mu\chi)$. As can be seen, step responses of model (4.25) fit the responses of (4.24) quite well. The error of the responses depends on the value of the parameter χ. The smaller the delay parameter χ is, the responses are closer to each other (the more distant are the complex zeros of the chain).

To sum up, model (4.13) can be used as the approximation model of fairly broad class of SISO systems. The motivation to find the universal model as simple as possible is due to application of the model in control. Using, e.g., methods of inversion dynamics or internal model control, the structure of the final controller is primarily given by the structure of the plant model. Some aspects of using model (4.13) in internal model control will be discussed in the following section.

4.3 Notes on anisochronic IMC design

4.3.1 Neutral character of closed loop system caused by IMC

It has already been mentioned that the method FSA may result in neutral character of the closed loop system dynamics. In this chapter, I am going to show that also the model based control design may give the same result if applied to TDS. In Zítek, (1998), Hlava, (1988), Zítek and Hlava, (1988), Vyhlídal and Zítek, (2001), the features of internal model control (IMC) design based on low order models with time delays are studied. The scheme of IMC is shown in Fig. 4.9, where $R^*(s)$ is the controller, $P(s)$ denotes the dynamics of the plant which is controlled and $G(s)$ is the model of the plant. Using universal first order anisochronic model (4.13) and provided that the dynamics of the system being approximated does not involve positive zero effect, (the problem of systems with positive zeros will be solved in section 4.3.2), the transfer function of the controller is given by

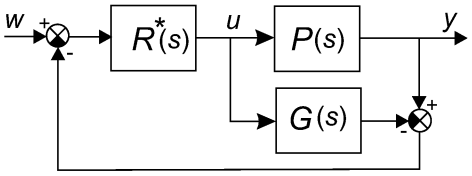

Fig. 4.9 IMC scheme

$$R^*(s) = \frac{1}{G_i(s)} F(s) = \frac{Ts + \exp(-s\eta)}{K(1 - a\exp(-s\chi))(T_f s + \exp(-s\eta_f))} \tag{4.26}$$

where $G_i(s)$ is the invertible part of model (4.13) and $F(s)$ is the first order anisochronic filter with F(0)=1. The transfer function of the inner control loop (which is the controller transfer function if the classical control loop is considered) is given by

$$R(s) = R^*(s) = \frac{R^*(s)}{1 - R^*(s)G(s)} = \frac{Ts + \exp(-s\eta)}{K(1 - a\exp(-s\chi))(T_f s + \exp(-s\eta_f) - \exp(-s\tau))} \tag{4.27}$$

see its block diagram in Fig. 4.10

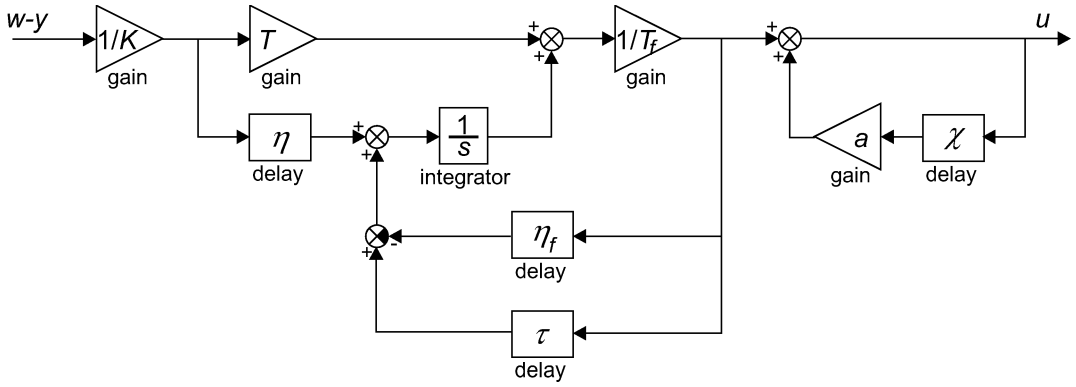

Fig. 4.10 Scheme of the controller (4.27)

If $G(s)=P(s)$, the closed loop dynamics are given by the first order model

$$G_{wy}(s) = \frac{\exp(-s\tau)}{T_f s + \exp(-s\eta_f)} \tag{4.28}$$

However, the model approximates only a part of the dynamics as a rule. Therefore, let us study the closed loop dynamics for the case $G(s) \neq P(s)$. Let the filter be $F(s) = 1/F_f(s)$, ($F_f(s) = T_f s + \exp(-s\eta_f)$), the approximative model $G(s) = KN(s)/M(s)\exp(-s\tau)$, ($N(s) = 1 - a\exp(-s\chi)$, $M(s) = (Ts + \exp(-\eta s))$) and the true plant model $P(s) = Q(s)/S(s)$, then the controller transfer function is given by

$$R(s) = \frac{M(s)}{KN(s)(F_f(s) - \exp(-s\tau))} \tag{4.29}$$

and the transfer function of the closed loop is the following

$$G_{wy}(s) = \frac{M(s)Q(s)}{S(s)KN(s)(F_f(s) - \exp(-s\tau)) + M(s)Q(s)} \tag{4.30}$$

Thanks to $N(s)$, obviously, there are delayed terms of the highest derivative of $y(t)$ in the model of the closed loop. Thus the closed loop is of the neutral system dynamics. The features of the IMC design and the application of the mapping based rootfinder in analysis of the system spectra will be shown in the following example.

Example 4.4

Consider the plant described by the model

$$P(s) = \frac{20s + 1}{(2s + 1)^{10}} \tag{4.31}$$

with the multiple pole $\lambda_{1..10} = -0.5$ and the single zero $\mu = -0.05$. Let us find the parameters of model (4.13) that will approximate model (4.31) in the low frequency range. Assessing $\tau = 8$ and $\chi = 10$, which implies $a = 0.607$ (according to (4.20), β= -0.05) and K=1/(1-a)=2.54, the system dead time and the rising part of the response are approximated quite well, see Fig. 4.11.

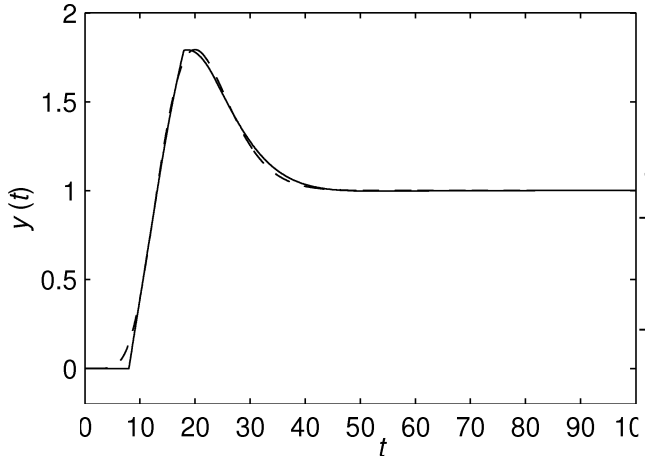

Fig. 4.11 Step responses of system (4.31) (dashed) and of its approximation (4.13) (solid)

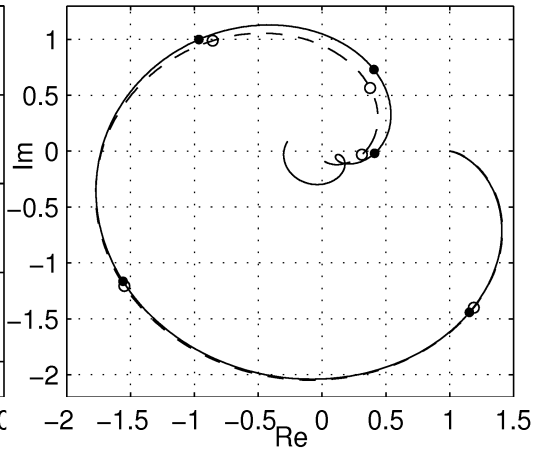

Fig. 4.12 Frequency responses of system (4.31) (dashed) and of its approximation (4.13) (solid)

The remaining two parameters of model (4.13), i.e., $T = 13.1$ and $\eta = 5.5$, have been assessed using least square method to approximate two points of the frequency response of (4.31). Since model (4.13) is supposed to approximate model (4.31) in the low frequency range, the points of the frequency response being approximated have been chosen those with $\Phi(\omega_1) = \arg(P(\mathrm{j}\omega_1)) = -\pi/2$ and $\Phi(\omega_2) = \arg(P(\mathrm{j}\omega_2)) = -\pi$. As can be seen in Fig. 4.12, the frequency response of model (4.13) approximates frequency response of (4.31) very well (even in the fourth quadrant of the frequency response). Also the approximation of the system step response is very good considering the anisochronic model (4.13) is of the first order.

Let us use the given parameters of model (4.13) in controller (4.27) and let us investigate the dynamics of the closed loop. Closed loop system is of the 11^{th} order with transfer function (4.30). Choosing $T_\mathrm{f} = 10$ and $\eta_\mathrm{f} = 7$ the closed loop dynamics are supposed to be given by the dominant couple of poles $\tilde{\lambda}_{1,2} = -0.081 \pm 0.156\mathrm{j}$ (with relative damping $\xi = 0.51$, see Fig. 4.4). Using the quasipolynomial mapping based rootfinder, the poles of the closed loop system can be computed as the roots of the denominator of the closed loop with the transfer function given by (4.30), see Fig. 4.13.

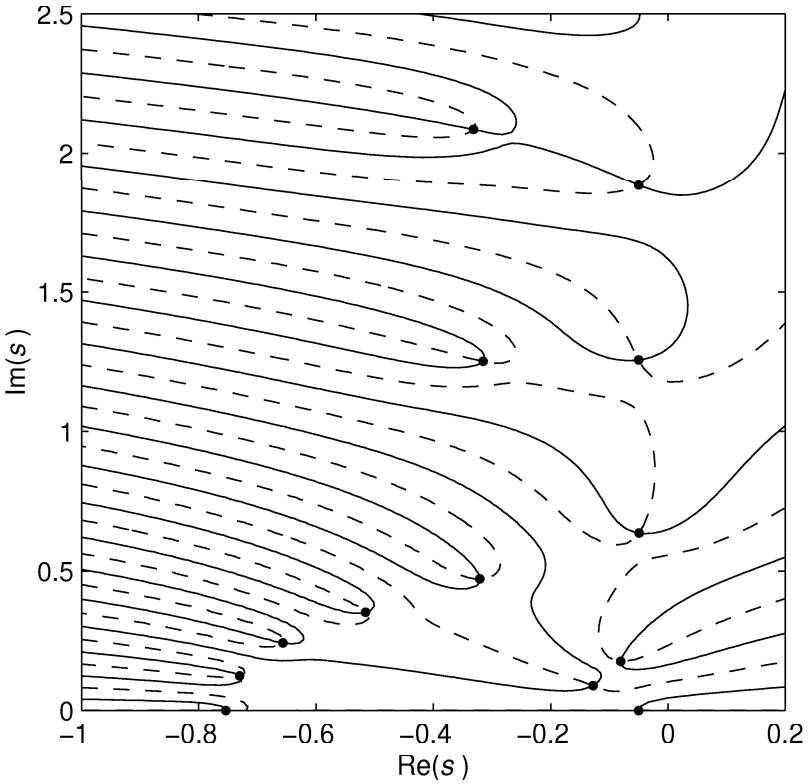

Fig. 4.13 Poles of the closed loop system with plant model (4.31) and controller (4.27), Re($M(s)$)=0-solid, Im($M(s)$)=0-dashed, $M(s)$-characteristic function of the closed loop system

As can be seen in Fig. 4.13, the following poles $\lambda_1 = -0.05$, $\lambda_{2,3} = -0.081 + 0.176\mathrm{j}$ and $\lambda_{4,5} = -0.126 + 0.089\mathrm{j}$ are likely to be dominant (considering the value of the magnitude of the poles as the evaluating criterion). The poles of the couple $\lambda_{2,3}$ are quite close to the prescribed poles $\tilde{\lambda}_{1,2}$. However, from the distribution of the other dominant poles, it is not obvious that the poles $\lambda_{2,3}$ determine the dynamics of the closed loop (λ_1 is even closer to

the origin of the *s*-plane). As has been mentioned, the dynamics of the system are not only determined by the system poles, but also by the system zeros. Thus, let us find the zeros of the closed loop system. Since the closed loop is neutral system we will analyse the spectrum of difference equation as well. The essential spectrum of the system is obviously given by the solutions of equation (4.14). The spectra of poles (black circles), zeros (empty circles) and roots of the difference equation (asterisks) corresponding to the closed loop system with chosen $T_f = 10$ and $\eta_f = 7$ are shown in Fig. 4.16. As can be seen the pole λ_1 is likely to be compensated by the real zero. Also the couple of poles $\lambda_{4,5}$ are quite close to a couple of zeros and is also partly compensated. Consequently, the dominant mode of the set point response is really given by the couple of poles $\lambda_{2,3}$ (see Fig. 4.16, the poles of the prescribed anisochronic dynamics are marked by squares). The dominant role of the couple $\lambda_{2,3}$ in the set-point response dynamics of the closed loop is confirmed by the responses of the real closed loop system (4.30) and ideal closed loop system (4.28) seen in Fig. 4.17. As can be seen, the real set point response (solid) is very close to the ideal one (dashed). The characteristic feature of the class of neutral systems, i.e., some of the poles converge to the roots of the difference equation, can be seen in Fig. 4.16 and in the enlarged region in Fig. 4.20.

Thus, using the difference equation of form (4.14) to approximate the system dominant zero implies that the closed loop system has infinitely many poles with real parts close to the value given by (4.20). Therefore, controller (4.27) cannot be used to control systems with zeros in the right half of the complex plane. On the other hand, if the dominant zero is negative and not too close to the imaginary axis, the neutral character of the closed loop system does not bring about any risky features to its dynamics. Provided that closed loop system (4.30) (with negative dominant zero) does not have any unstable poles close to the *s*-plane origin, it does not have any unstable poles at all. It is given by the fact that the chain of the poles converging to the spectrum of difference equation has the tendency of getting closer to the eigenvalues of the essential spectrum as the magnitudes of the poles in the chain increase.

In order to demonstrate that the dynamics of the closed loop are really determined by the parameters T_f and η_f let us choose different settings of these parameters. Firstly, let $T_f = 10$ and $\eta_f = 0$ corresponding to the dominant pole $\tilde{\lambda}_1 = -0.1$. The poles of real closed loop system that are closest to this prescribed dominant pole $\tilde{\lambda}_1$ are $\lambda_{2,3} = -0.108 + 0.026\mathrm{j}$, see Fig. 4.14. As can be seen from the set point responses in Fig. 4.15, the dominant modes of the dynamics are again determined by the couple $\lambda_{2,3}$. However, even though the set point responses in Fig. 4.15 are very similar, it can be seen that there is a weakly damped parasite mode in the real response. Obviously, this mode corresponds to the couple $\lambda_{6,7} = -0.051 \pm 0.639\mathrm{j}$,(compare the position of λ_6 with the closest eigenvalue of the essential spectrum.

As the third setting of the parameters determining the closed loop dynamics, let us choose $T_f = 10$ and $\eta_f = 10\pi/2$ which implies $\tilde{\lambda}_{1,2} = \pm 0.1\mathrm{j}$. The corresponding spectra can be seen in Fig. 4.18 and the set-point responses in Fig. 4.19. As can be seen, some of the poles of real closed loop (4.30) (not only the dominant ones) are even closer to the poles of the ideal closed loop (4.28) than in the previous cases. As can be seen, the real closed system is not unstable because the dominant couple of poles $\lambda_{2,3} = -0.002 \pm 0.1\mathrm{j}$ are located slightly to the left from the imaginary axis.

Caption 4.1 Spectra and set point responses of closed loop system (4.30) consisting of for different settings of prescribed dynamics to the ideal closed loop system (4.28). Spectra: black circles - poles of (4.30), empty circles - zeros of (4.30), asterisks - roots of $N(s)$, squares - poles of system (4.28) (ideal closed loop). Set point responses: solid - system (4.30), dashed - system (4.28)

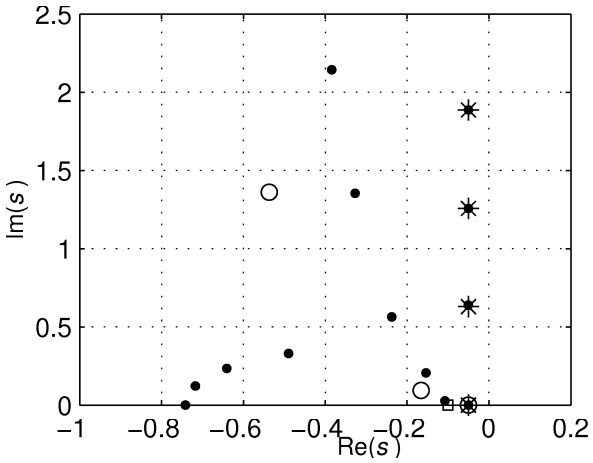

Fig. 4.14 $T_f = 10$ and $\eta_f = 0$

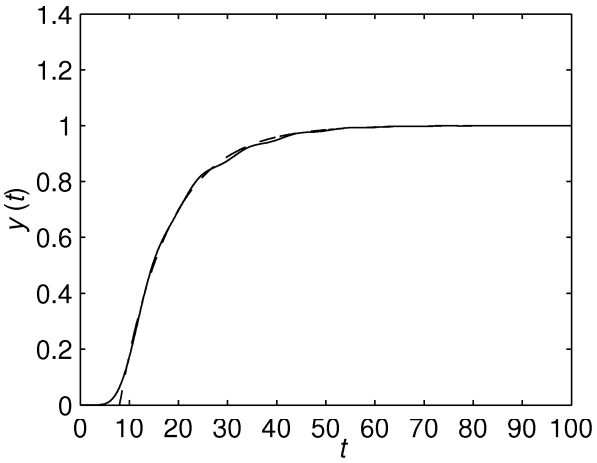

Fig. 4.15 $T_f = 10$ and $\eta_f = 0$

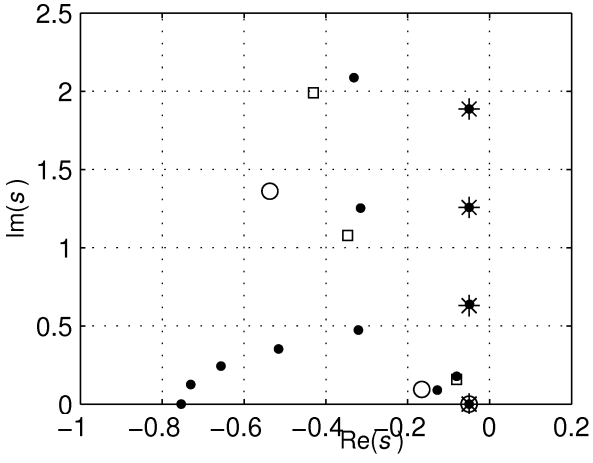

Fig. 4.16 $T_f = 10$ and $\eta_f = 7$

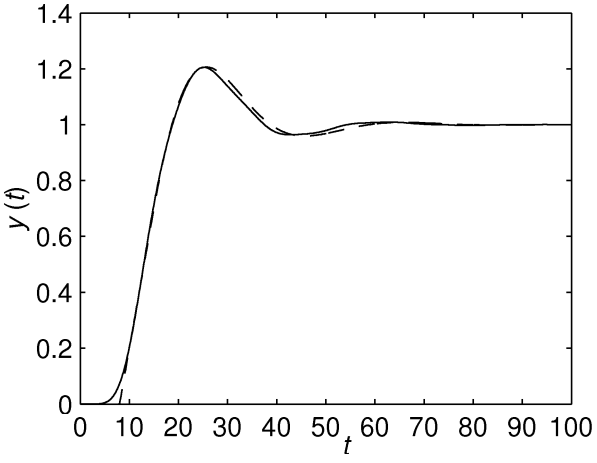

Fig. 4.17 $T_f = 10$ and $\eta_f = 7$

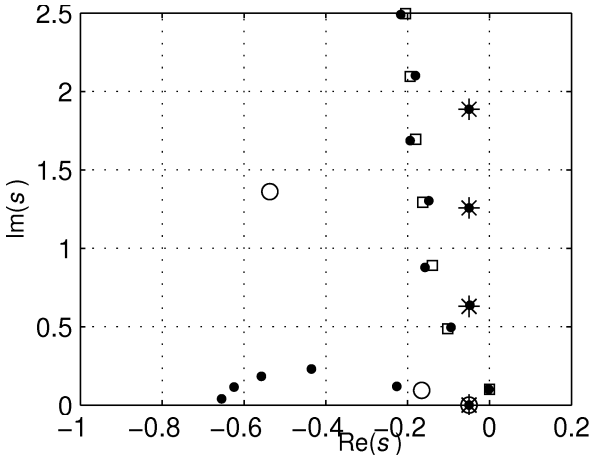

Fig. 4.18 $T_f = 10$ and $\eta_f = 10\pi/2$

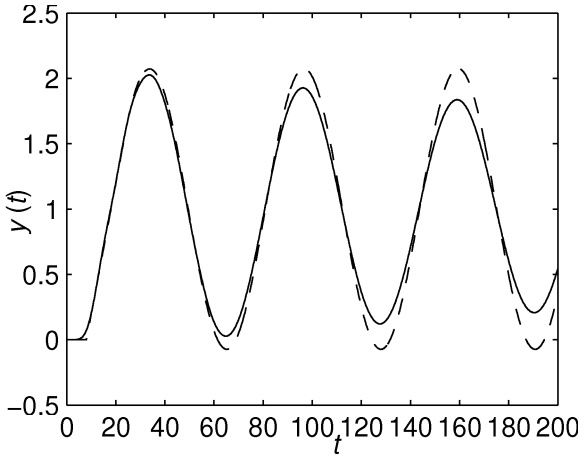

Fig. 4.19 $T_f = 10$ and $\eta_f = 10\pi/2$

On the basis of good results achieved by using the first order anisochronic model with difference part in the numerator to approximate the higher order models with single real zero, one could suppose that using the difference equation

$$N(s)=\prod_{i=1}^{m}\left(1-a_i\exp(-s\chi_i)\right)=1-\sum_{i=1}^{m}\tilde{a}_i\exp(-s\tilde{\chi}_i) \tag{4.32}$$

allows more zeros of the system to be approximated. However, using this way for approximating more than one zero, the features of spectra of exponential polynomials have to be taken into consideration (Avellar and Hale, 1980). Even if all the roots of (4.32) are located in the left half of the complex plane, the positions of the roots may be very sensitive even to small variations in the delays (if the delays are rationally dependent). However, if the following condition is satisfied

$$\sum_{i=1}^{m}\left|\tilde{a}_i\right|<1 \tag{4.33}$$

the equation $N(s)=0$ does not have any solutions in the right half of the complex plane, see, e.g., Avellar and Hale, (1980) and there is no danger of loosing the stability if the delays in (4.32) change slightly. The risk of using (4.32) will be demonstrated in the following example.

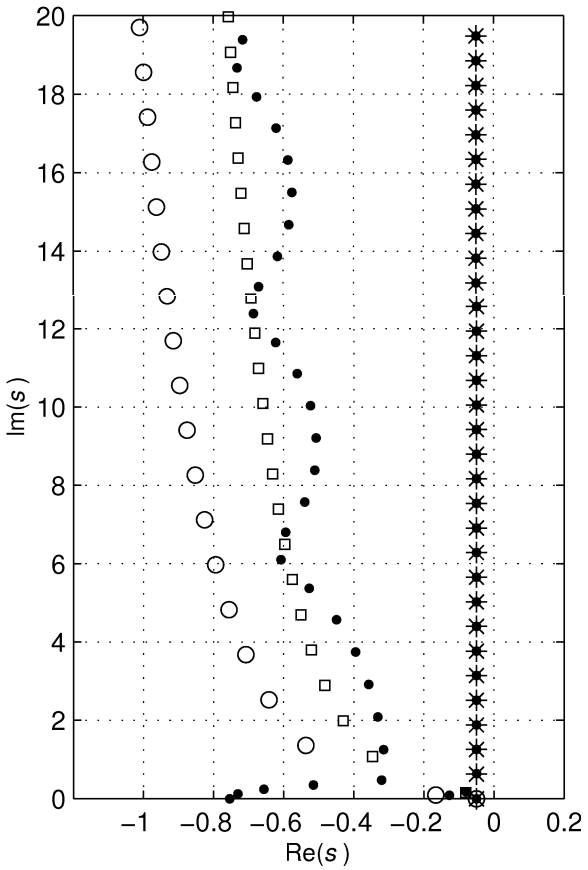

Fig. 4.20 Spectra of closed loop system for $T_f=10$ and $\eta_f=7$, see Caption 4.1

Example 4.5

Consider the plant described by the model

$$P(s)=\frac{(20s+1)(10s+1)}{(2s+1)^{10}} \tag{4.34}$$

with two single zeros $\mu_1=-0.05$ and $\mu_2=-0.1$. The parameters of the first order model with $N(s)=1-\tilde{a}_1\exp(-\tilde{\chi}_1 s)-\tilde{a}_2\exp(-\tilde{\chi}_2 s)$ and $M(s)=Ts+\exp(-\eta s)$, $\chi=7.5$, $a_1=0.687$, $a_2=0.472$, ($\tilde{a}_1=a_1+a_2=1.159$, $\tilde{a}_2=-a_1a_2=-0.324$, $\tilde{\chi}_1=\chi=7.5$, $\tilde{\chi}_2=2\chi=15$), $\tau=5.5$, K=6.06, $T=13.3$ and $\eta=6.5$ have been obtained in the same way as in Example 4.4. As can be seen in Fig. 4.21 and 4.22, the first order functional model approximates the characteristics of model (4.34) quite well. Using the parameters of the first order filter for IMC controller (4.29) $T_f=10$ and $\eta_f=7$ ($\tilde{\lambda}_{1,2}=$ -0.081$\pm$0.156j) the results shown in Fig 4.23 and Fig. 4.24 are obtained.

As can be seen in Fig. 4.23, two of the dominant poles are likely to be compensated by the zeros. Thus the dominant part of the dynamics are given by the couple of poles

$\lambda_{2,3} = -0.078 \pm 0.102\mathrm{j}$ that are quite close to the prescribed couple $\tilde{\lambda}_{1,2}$. Regarding the lower imaginary part of $\lambda_{2,3}$ (comparing to $\tilde{\lambda}_{1,2}$), it may be assumed that the responses of the real closed loop system are likely to be more damped than the responses of the ideal closed loop. This feature is seen in Fig. 4.24, where the set-point responses of the real and ideal closed loops can be seen. Besides the set-point response is more damped, a parasite oscillatory mode is involved in the closed loop dynamics given by the couple of poles $\lambda_{6,7} = -0.048 \pm 0.858\mathrm{j}$ (corresponding to a couple of the essential spectrum).

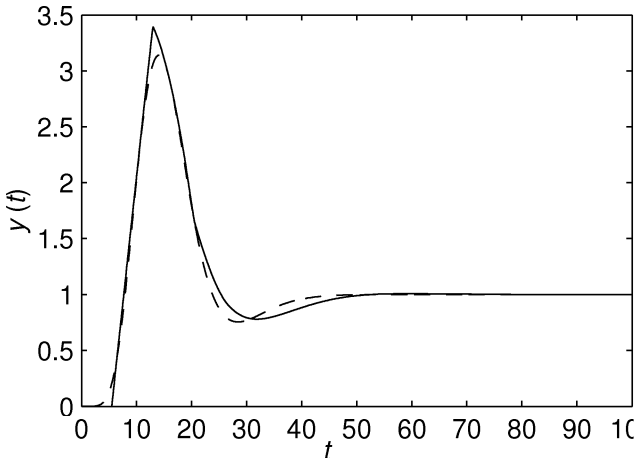

Fig. 4.21 Step responses of system (4.34) (dashed) and of its anisochronic first order approximation (solid)

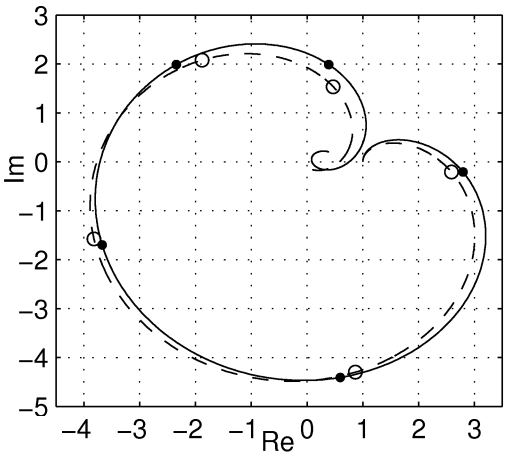

Fig. 4.22 Frequency responses of system (4.34) (dashed) and of its anisochronic first order approximation (solid)

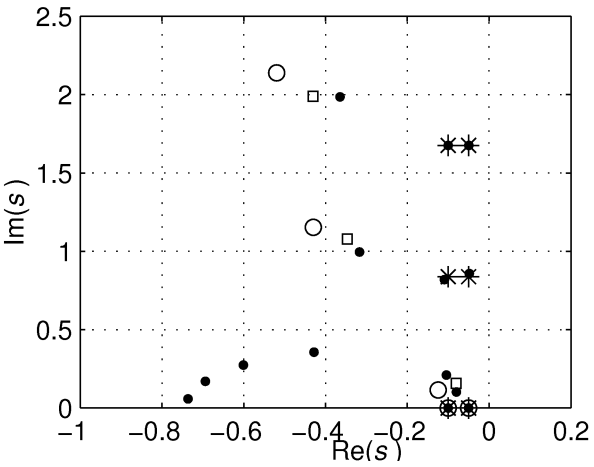

Fig. 4.23 Spectra of closed loop system for $T_\mathrm{f} = 10$ and $\eta_\mathrm{f} = 7$, see Caption 4.1

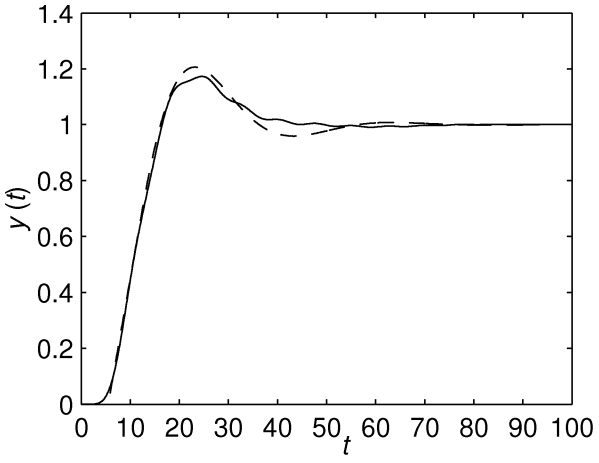

Fig. 4.24 Set-point responses of real (dashed) and ideal (solid) closed loop systems

In Fig. 4.25, the spectra of the closed loop system are shown in the enlarged region. As in the case of the system with one zero, the chains of the roots of $N(s)$ are also parallel with the imaginary axis. However, if one of the delays in (4.32) changes so that the delays are not rationally dependent the character of the essential spectrum considerably changes. Since (4.33) is not satisfied, there are infinitely many poles crossing the stability boundary if one of the delays changes. The deflection of the essential spectrum caused by the change $\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$ are shown in Fig. 4.26. As can be seen, such a change of one of the delays

deflects the essential spectrum and some of the roots of (4.32) (in fact infinitely many) enter the right half of the complex plane. The character of the spectra can be seen in Fig. 4.29. Obviously, the instability of the closed loop is given by the part of the controller $1/N(s)$, see (4.30). Therefore let us show the character of the instability caused by such a small change in

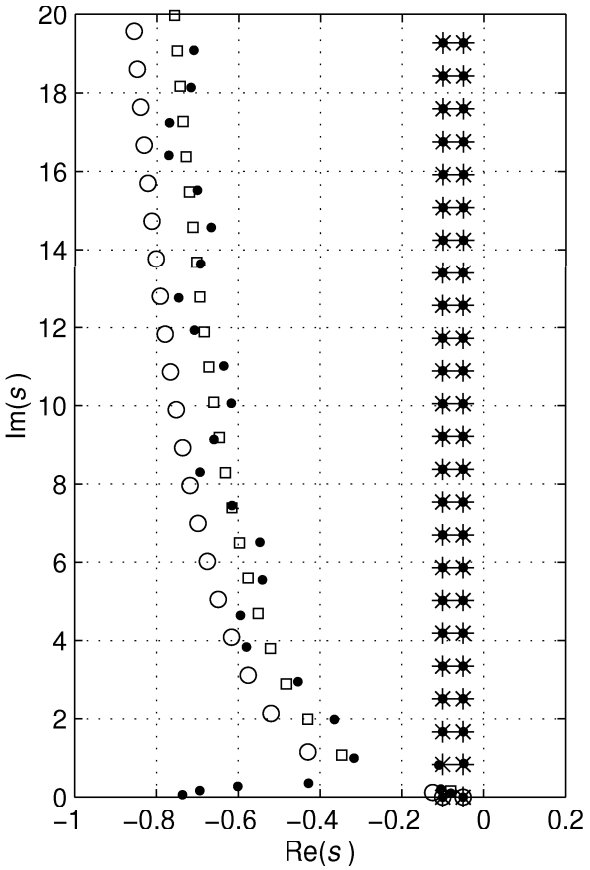

Fig. 4.25 Spectra of closed loop system, rationally dependent delays of difference equation, ($\tilde{\chi}_2 = 2\tilde{\chi}_1$), see Caption 4.1

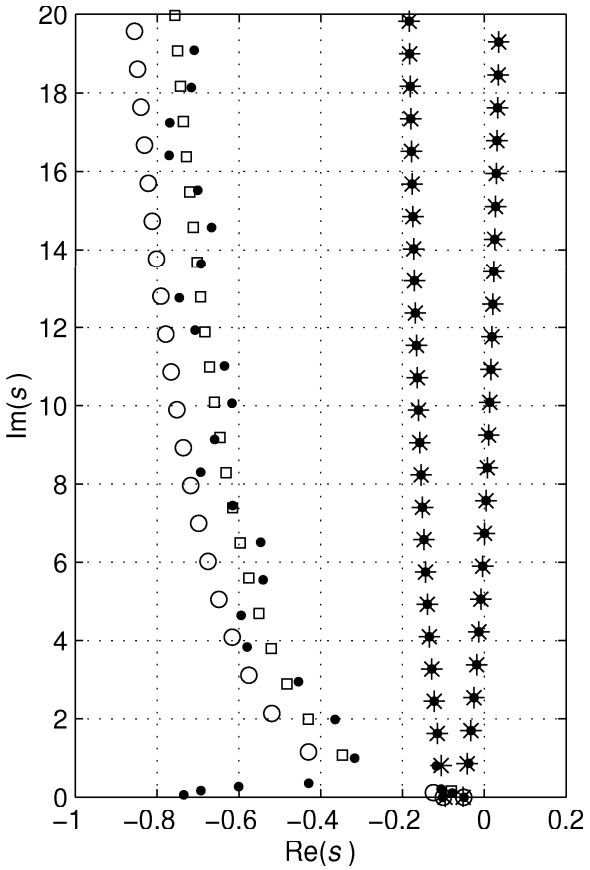

Fig. 4.26 Spectra of closed loop system, rationally independent delays of difference equation, ($\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$), see Caption 4.1

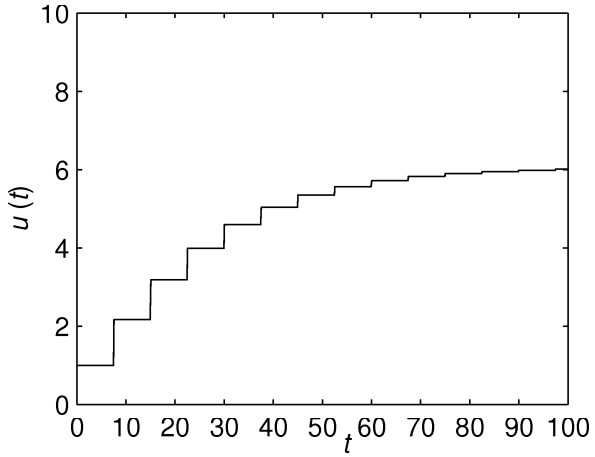

Fig. 4.27 Step response of $1/N(s)$, rationally dependent delays, $\tilde{\chi}_2 = 2\tilde{\chi}_1$, see (4.32)

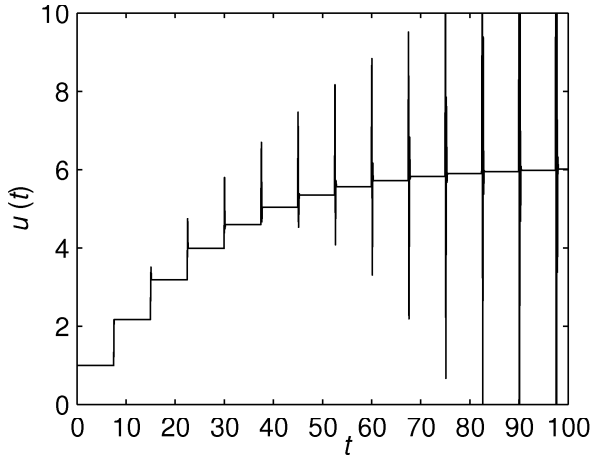

Fig. 4.28 Step response of $1/N(s)$, rationally independent delays, $\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$, see (4.32)

the delays. In Fig. 4.27 and Fig 4.28 we can see the step responses of the transfer function given by $1/N(s)$. In Fig. 4.27, the delays are rationally dependent, $\tilde{\chi}_2 = 2\tilde{\chi}_1$ and in Fig. 4.28, the delays are not rationally dependent, $\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$. The character of the responses is quite similar. The response of $1/N(s)$ with $\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$ differs from the former one only in the beginning of each of the steps. However these impulse-like discrepancies have increasing amplitude responsible for loosing the stability of the closed loop.

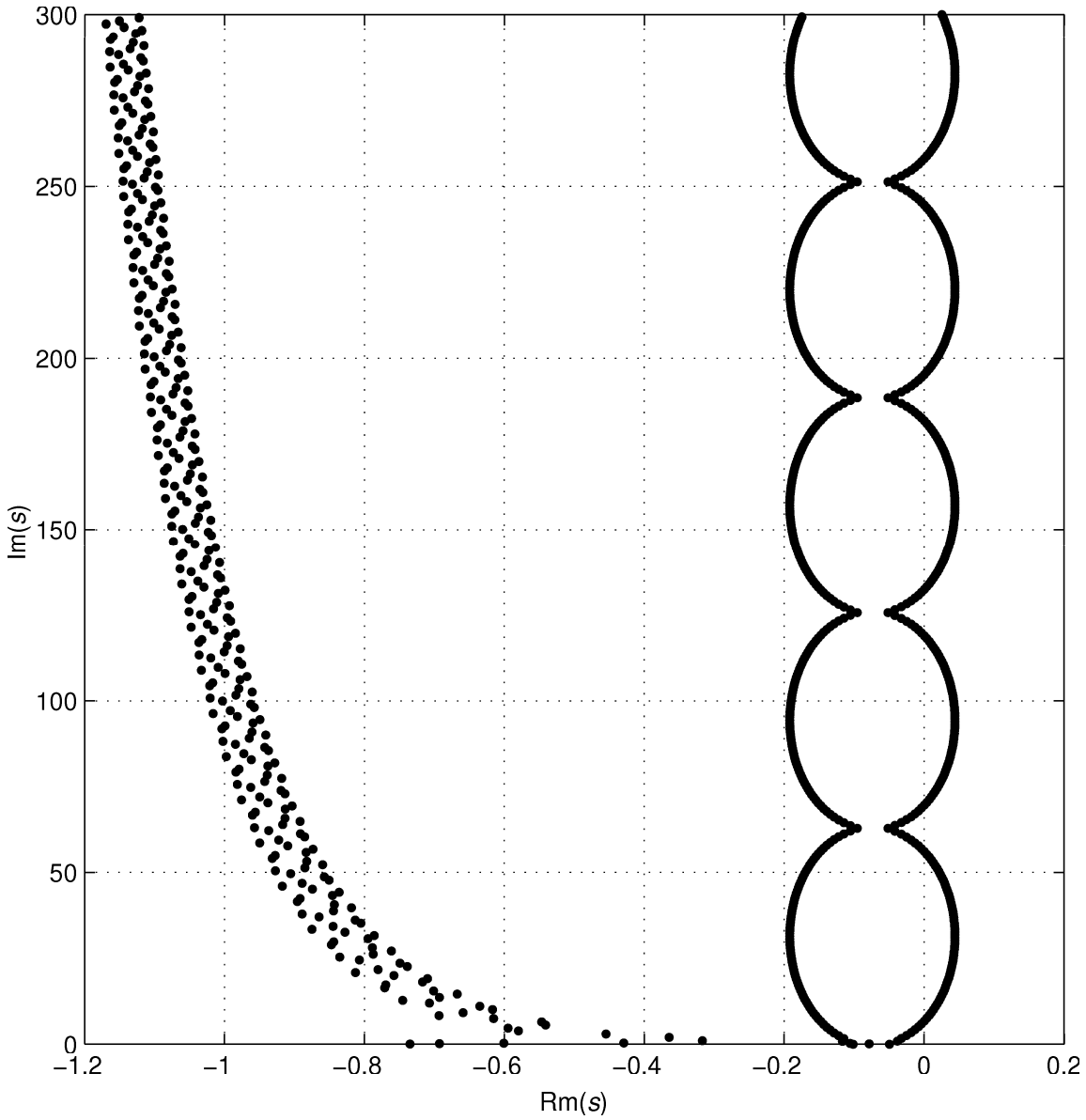

Fig. 4.29 Spectrum of the poles of closed loop system with rationally independent delays in $N(s)$, ($\tilde{\chi}_2 = 2\tilde{\chi}_1 + 0.1$), see (4.32)

4.3.2 Robust anisochronic IMC control design based on first order anisochronic model

According to the results obtained in the previous section, IMC anisochronic controller (4.29) is suitable only if the coefficients of $N(s)$ satisfy condition (4.33). We could also observe that the essential spectrum given by the solutions of $N(s)$=0 may bring the undesirable oscillatory modes to the closed loop dynamics if the spectrum is too close to the stability boundary. In the classical IMC design, see Morari and Zafiriou, (1989), Skogestad, et. al., (1996), the uninvertible part of the delay free system (which is in not involved in the

controller) is given by the non-minimal phase factor as a rule. Thus, analogously to that case, let us factorise $N(s)$ into $N_{\mathrm{d}}(s)N_{\mathrm{n}}(s)$, from which only $N_{\mathrm{d}}(s)$ will be involved in the inverted part of the transfer function. The spectrum of $N_{\mathrm{d}}(s)=\prod_{i=1}^{d}\left(1-a_i\exp(-s\chi_i)\right)=1-\sum_{i=1}^{d}\tilde{a}_i\exp(-s\tilde{\chi}_i)$ has to be free of the roots located in the right half of the complex plane, i.e., $|a_i|<1$ and also the following condition has to be satisfied $\sum_{i=1}^{d}|\tilde{a}_i|<1$, which guaranties the strong stability of the spectrum. Thus the controller of the loop seen in Fig. 4.9 is of the form

$$R^{*}(s)=\frac{M(s)}{K\overline{N}_{\mathrm{n}}N_{\mathrm{d}}(s)F_{\mathrm{f}}(s)} \tag{4.35}$$

where $\overline{N}_{\mathrm{n}}=N_{\mathrm{n}}(s),s\to 0$ and the controller of the classical control loop acquires the form

$$R(s)=\frac{M(s)}{KN_{\mathrm{d}}(\overline{N}_{\mathrm{n}}F_{\mathrm{f}}(s)-N_{\mathrm{n}}(s)\exp(-s\tau))} \tag{4.36}$$

The closed loop transfer function is given by

$$G_{wy}(s)=\frac{N_{\mathrm{n}}(s)\exp(-s\tau)}{\overline{N}_{\mathrm{n}}F_{\mathrm{f}}(s)} \tag{4.37}$$

in case that $G(s)=P(s)$. If $G(s)\neq P(s)$, the closed loop is given by

$$G_{wy}(s)=\frac{M(s)Q(s)}{S(s)KN_{\mathrm{d}}(s)(\overline{N}_{\mathrm{n}}F_{\mathrm{f}}(s)-N_{\mathrm{n}}(s)\exp(-s\tau))+M(s)Q(s)} \tag{4.38}$$

As can be seen in (4.37), the zeros of $N_{\mathrm{n}}(s)$ become the zeros of the closed loop dynamics. The application of the method will be shown in the following example

Example 4.6

Let us use IMC controller (4.36) to control the system

$$P(s)=\frac{(20s+1)(10s+1)(-5s+1)}{(2s+1)^{10}} \tag{4.39}$$

with three zeros $\mu_1=-0.05$, $\mu_2=-0.1$ and $\mu_3=0.2$. Let us consider the parameters of the first order anisochronic model

$$G(s)=\frac{\prod_{i=1}^{3}(1-a_i\exp(-s\chi_i)K\exp(-s\tau)}{Ts+\exp(-s\eta)} \tag{4.40}$$

as follows $T=11.94$, $\eta=4.67$, $\tau=3$, $\chi_1=\chi_2=7.5$, $\chi_3=5.5$ $a_1=0.687$, $a_2=0.472$, $a_3=3.004$ and $K=-3.02$. The comparison of the step and frequency responses of models (4.39) and (4.40) can be seen in Fig. 4.30 and Fig. 4.31, respectively. As can be seen in Fig. 4.30, the step response of first order model (4.40) approximate the response of high order model (4.39) quite well. Also the approximation of the frequency response in the low frequency range is very good.

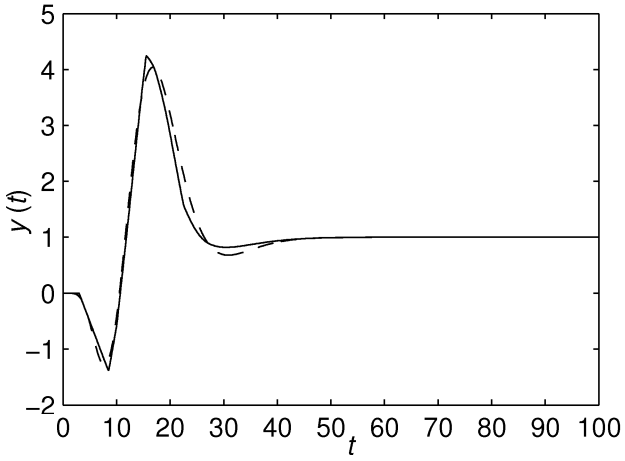

Fig. 4.30 Step responses of system (4.39) (dashed) and of its anisochronic first order approximation (4.40) (solid)

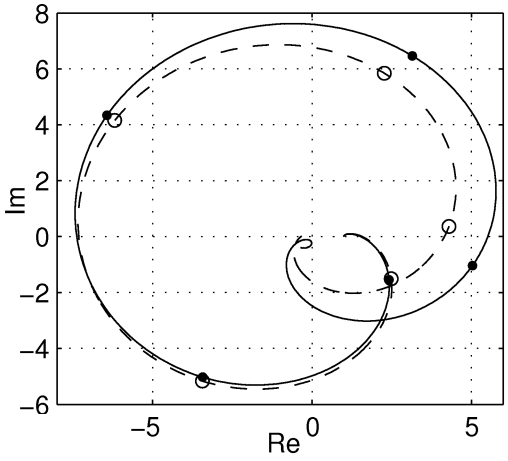

Fig. 4.31 Frequency responses of system (4.39) (dashed) and of its anisochronic first order approximation (4.40) (solid)

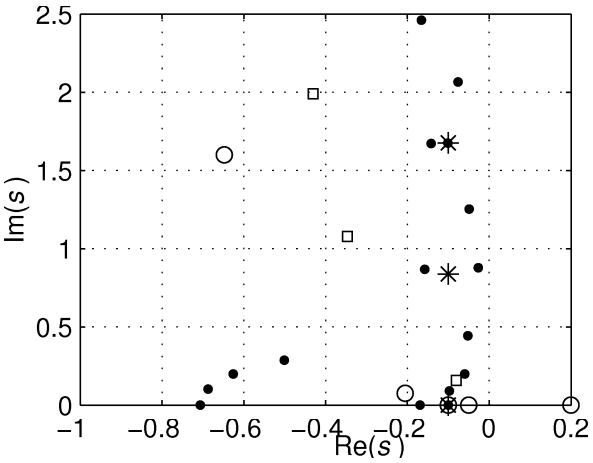

Fig. 4.32 Spectra of closed loop system for $T_f = 10$ and $\eta_f = 7$, see Caption 4.1

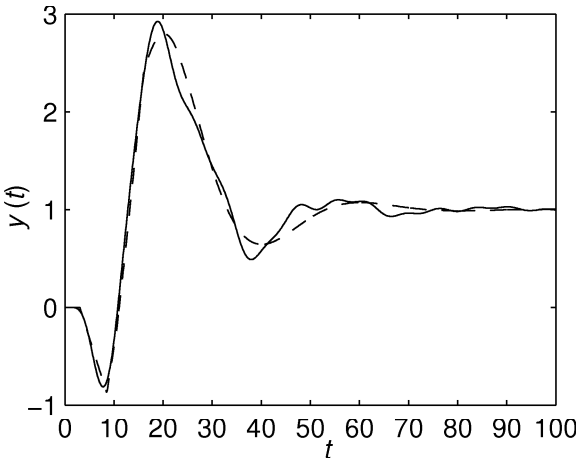

Fig. 4.33 Set-point responses of real (solid) and ideal (dashed) closed loop systems

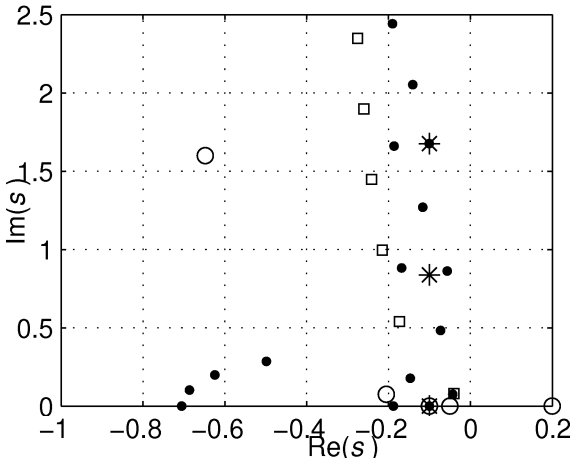

Fig. 4.34 Spectra of closed loop system for $T_f = 20$ and $\eta_f = 14$, see Caption 4.1

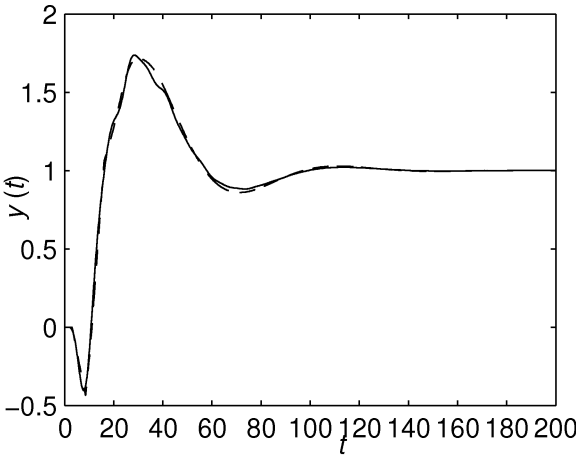

Fig. 4.35 Set-point responses of real (solid) and ideal (dashed) closed loop systems

First step of the robust IMC design consists in the factorisation of $N(s)$. Since $a_3 > 1$ and since $|\tilde{a}_1| + |\tilde{a}_2| = 1.48$ ($\tilde{a}_1 = a_1 + a_2 = 1.159$, $\tilde{a}_2 = -a_1 a_2 = -0.324$) condition (4.33) is not satisfied for $\prod_{i=1}^{2}(1 - a_i \exp(-s\chi_i))$, the function $N(s)$ may be factorised into $N_d(s) = 1 - a_2 \exp(-s\chi_1)$ (μ_2 is more distant from the stability boundary than μ_1, thus the control loop is likely to be more robust) and $N_n(s) = (1 - a_2 \exp(-s\chi_2))(1 - a_3 \exp(-s\chi_3))$. Using controller (4.36) with the optional parameters $T_f = 10$ and $\eta_f = 7$ the spectra of the closed loop dynamics and the set-point responses can be seen in Fig. 4.32 and Fig. 4.33, respectively. As can be seen in Fig. 4.32, the couple of dominant poles $\tilde{\lambda}_{1,2} = -0.081 \pm 0.156\text{j}$ are not so far from the prescribed dominant couple $\lambda_{2,3} = -0.06 \pm 0.198\text{j}$, which implies that the set-point response of the closed loop is quite close to the ideal response. However, the step response involves a parasite oscillatory mode corresponding to the pole $\lambda_{10,11} = -0.027 \pm 0.877\text{j}$. Assigning slower closed loop dynamics by choosing $T_f = 20$ and $\eta_f = 14$ ($\tilde{\lambda}_{1,2} = -0.040 \pm 0.0782\text{j}$), the pole $\lambda_{10,11}$ gets more to the left which enhance the robustness of the closed loop, see Fig. 4.34. The set-point response is also much closer to the ideal response. As can be seen in the spectra in Fig. 4.32 and Fig. 4.34 the zero of $N_d(s)$ is compensated by the pole λ_1. However, other two dominant real zeros corresponding to the solutions of $N_n(s) = 0$ are not compensated and their contributions to the closed loop dynamics are seen in Fig. 4.33 and Fig. 4.35 (compare with Fig. 4.30).

4.3.3 A note on IMC design for higher order TDS

Let us consider SISO retarded system given by the matrices

$$\mathbf{A}(s) = \begin{bmatrix} a_{1,1}(s) & a_{1,2}(s) & \cdots & a_{1,n}(s) \\ a_{2,1}(s) & a_{2,2}(s) & \cdots & a_{2,n}(s) \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}(s) & a_{n,2}(s) & \cdots & a_{n,n}(s) \end{bmatrix}, \quad \mathbf{B}(s) = \begin{bmatrix} b_1 \exp(-s\tau_1) \\ b_2 \exp(-s\tau_2) \\ \vdots \\ b_n \exp(-s\tau_n) \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{4.41}$$

Transforming the system into input-output transfer function, the term of the highest derivative of the quasipolynomial in the numerator is given by

$$Q_n(s) = \sum_{k=1}^{n} b_k c_k \exp(-s\tau_k) \tag{4.42}$$

(Since this result is not the substantial according to the objectives of this thesis, the derivation of (4.42) is omitted). Thus, performing the dynamics inversion of the system with matrices (4.41) we obtain the neutral character of the closed loop if at least one of the couples satisfy $b_k c_k \neq 0$, and $\tau_k > 0$, k=1,..,n. Therefore, in IMC design applied to TDS, it is necessary to analyze the spectrum of the system difference equation and to remove the risky parts from the part of the transfer function being inverted.

4.4 Evaluation of the significance of the poles of TDS

4.4.1 Heaviside series based expansion of dynamics of TDS

In Examples 4.4, 4.5 and 4.6, the significance of the poles has been evaluated with respect to their distances from the *s*-plane origin. Besides such a pole significance evaluation, e.g., Goodwin, (2001), considered as dominant poles those that are closer to the stability boundary than the rest of the poles. Such an evaluation is reasonable if the problem to be solved is the stability issue. Anyway, it is difficult to claim whether the distance from the origin or the distance from the imaginary axis is more important in determining the pole significance. In this section, according to the third objective of the thesis, I am going to introduce an original method for evaluating pole significance based on the analysis of the modes of TDS.

Due to the generalized Heaviside series expansion, see, e.g., Angot, (1952), the function $g(t)$ with the Laplace transform

$$G(s) = \frac{N(s)}{M(s)} \tag{4.43}$$

where the functions $N(s)$ and $M(s)$ are the analytic functions and $N(s)/M(s)$ has only single poles the number of which may be both finite and infinite. Provided that the ratio does not have any singularities besides the poles, $g(t)$ can be expanded into the series

$$g(t) = \sum_{i=0}^{\infty} \frac{N(\lambda_i)}{M'(\lambda_i)} \exp(\lambda_i t) = \sum_{i=0}^{\infty} R(\lambda_i) \exp(\lambda_i t) \tag{4.44}$$

which implies

$$G(s) = \sum_{i=1}^{\infty} \frac{N(\lambda_i)}{M'(\lambda_i)} \frac{1}{s-\lambda_i} = \sum_{i=1}^{\infty} \frac{R(\lambda_i)}{s-\lambda_i} = \sum_{i=1}^{\infty} H_i(s) \tag{4.45}$$

where $\lambda_i, i = 1..\infty$ are the poles of the function $N(s)/M(s)$ and $R(s) = N(s)/M'(s)$, $M'(s) = dM(s)/ds$.

4.4.2 *Evaluation of the significance of the poles in the infinite spectrum*

In Zítek and Vyhlídal, (2002a,b), we have suggested evaluating the significance of the poles on the basis of the absolute value of the residues $|R(\lambda_i)|$. Due to (4.44) the values of the residues determine weight factors of the contribution of the modes to the system dynamics. Obviously, applying such an evaluation is reasonable only in case of real poles. However, the reason of using $|R(\lambda_i)|$ for evaluation of the complex λ_i is not obvious. The oscillatory modes are not given by the single complex poles λ_i but by the complex conjugate pairs $\lambda_{i,i+1} = \beta_i \pm \mathrm{j}\omega_i$. The residues corresponding to $\lambda_{i,i+1}$ are also complex conjugate pairs, given by $R(\lambda_{i,i+1}) = \beta_{\mathrm{R}i} \pm \mathrm{j}\omega_{\mathrm{R}i}$ and their roles in the modes are not so clear. In order to prove or disprove using $|R(\lambda_i)|$ to evaluate the significance of λ_i, let us investigate the features of the system modes. Considering that $G(s)$ is a time delay system (input lumped delay is not involved, it is considered as the separated part of the dynamics not influencing the pole significance, see section 5.2) thus the functions $N(s)$ and $M(s)$ are quasipolynomials of form (1.45). According to (4.45), such an infinite dimensional system $G(s)$ may be expanded into the sum of infinite number of the first order transfer functions given by

$$H_i(s) = \frac{R(\lambda_i)}{s - \lambda_i} \tag{4.46}$$

However, function (4.46) fully describes only the modes corresponding to the real poles. If λ_i is complex, the mode is oscillatory and its transfer function is of the second order given by

$$H_{i,i+1}(s) = \frac{R(\lambda_i)(s - \lambda_{i+1}) + R(\lambda_{i+1})(s - \lambda_i)}{(s - \lambda_i)(s - \lambda_{i+1})} = \frac{2(\beta_{\mathrm{R}i}s - \beta_{\mathrm{R}i}\beta_i - \omega_{\mathrm{R}i}\omega_i)}{s^2 - 2\beta_i s + \beta_i^2 + \omega_i^2} \tag{4.47}$$

where λ_{i+1} denotes here the complex conjugate pole to λ_i. The static gain coefficient of transfer function (4.47) is given by $H_{i,i+1}(0) = -2(\beta_{\mathrm{R}i}\beta_i + \omega_{\mathrm{R}i}\omega_i)/(\beta_i{}^2 + \omega_i{}^2)$. The dynamics of the transfer function (4.47) is not only determined by the couple of poles $\lambda_{i,i+1}$, but also by the single zero $\mu_{\mathrm{H}i} = \beta_i + (\omega_{\mathrm{R}i} / \beta_{\mathrm{R}i})\omega_i$. Performing the inverse Laplace transform to transfer functions (4.46) and (4.47), respectively, we obtain the weighting functions of the modes

$$h_i(t) = R(\lambda_i)\exp(\lambda_i t) \tag{4.48}$$

if λ_i are real poles and

$$h_{i,i+1}(t) = 2(\beta_{\mathrm{R},i}\exp(\beta_i t)\cos(\omega_i t) - \omega_{\mathrm{R},i}\exp(\beta_i t)\sin(\omega_i t)) \tag{4.49}$$

if $\lambda_{i,i+1}$ are complex conjugate pairs of poles, respectively. If the poles λ_i are real, the residue values $R(\lambda_i)$ are the global extrema of the weighting functions $h_i(t)$ for $t \geq 0$. Since function (4.48) is monotonous and $h_i(\infty) = 0$ are the other global extrema of $h_i(t)$ for $t \geq 0$, the differences between the maxima and minima of the weighting functions $h_{\mathrm{e}i} = \left|h_{i\max} - h_{i\min}\right|$ are equal to the absolute values of the residues, i.e., $h_{\mathrm{e}i} = \left|R(\lambda_i)\right|$. This fact support using $\left|R(\lambda_i)\right|$ to evaluate the significance of the real poles. Since, presumably, the higher the value of $h_{\mathrm{e}i} = \left|R(\lambda_{i,i+1})\right|$ is, the more significant the contribution of a particular weighting function $h_i(t)$ to the weighting function of the whole system $g(t)$ is. On the other hand, such an evaluation of the significance of the complex poles using $\left|R(\lambda_{i,i+1})\right|$ is not justified by an analogous role of $\left|R(\lambda_{i,i+1})\right|$ in the weighting functions $h_{i,i+1}(t)$. However, the difference between the maximum and minimum of the weighting function $h_{\mathrm{e}i} = \left|h_{i\max} - h_{i\min}\right|$, $t \geq 0$, is not difficult to obtain even for (4.49). Evaluating the first derivation of (4.49) equal to zero, the extrema of the oscillatory weighting function (4.49) occur at time t given by the solutions of the following equation

$$\tan(\omega_i t) = \frac{\beta_{\mathrm{R}i}\beta_i - \omega_{\mathrm{R}i}\omega_i}{\beta_{\mathrm{R}i}\omega_i + \omega_{\mathrm{R}i}\beta_i} \tag{4.50}$$

i.e., $t = t_{\mathrm{e}i} + \pi/\omega$, k=0,1, 2, ..., ($t_{\mathrm{e}0} > 0$ is the solution of (4.50) that is the closest one to t = 0). Considering the oscillatory mode being analysed is stable, the extrema have decreasing tendency as t increases. Thus, the significance evaluating criterion $h_{\mathrm{e}i}$ is defined as

$$
\begin{aligned}
h_{\mathrm{e}i} &= \left| h_{i\max} - h_{i\min} \right| \\
h_{\mathrm{e}i} &= \left| R(\lambda_i) \right| \text{ if } \lambda_i \text{ is real} \\
h_{\mathrm{e}i} &= \max\left\{ \left| h_i(0) - h_i(t_{\mathrm{e}0}) \right|, \left| h_i(t_{\mathrm{e}0}) - h_i(t_{\mathrm{e}1}) \right| \right\} \text{ if } \lambda_i \text{ is complex}
\end{aligned}
\tag{4.51}
$$

In the following example, the comparison of the evaluation criteria will be performed.

Example 4.7

Consider the closed loop system from Example 4.4 given by (4.30) with the setting $T_{\mathrm{f}} = 10$ and $\eta_{\mathrm{f}} = 7$, i.e.,

$$
G(s) = \frac{(Ts + \exp(-\eta s))Q(s)}{S(s)K(1 - a\exp(-s\chi))(T_{\mathrm{f}} s + \exp(-s\eta_{\mathrm{f}}) - \exp(-s\tau)) + Ts + \exp(-\eta s)Q(s)}
\tag{4.52}
$$

Let us expand the transfer function of the system into the generalized Heaviside series and let us investigate the roles of the modes in the system dynamics. In Tab. 4.1, we can see the values of the poles λ_i of system (4.52) ordered with respect to the values of $\left|R(\lambda_i)\right|$. The values of the residues $R(\lambda_i)$, $\left|R(\lambda_i)\right|$, the zeros μ_{Hi} of transfer functions (4.47) and the values of $h_{\mathrm{e}i}$ are also seen in Tab. 4.1 (the dynamics of some modes, e.g., those of $H_{1,2}(s)$ or $H_{18}(s)$ are considerably influenced by their zeros since $\left|\mu_{Hi}\right| / \left|\lambda_i\right| < 1$, which indicates the dominance of the zeros.) Evaluating the values of both $\left|R(\lambda_i)\right|$ and $h_{\mathrm{e}i}$ shows that the break point values of both the criteria are obviously the values for i = 15. Both the criteria decrease significantly as i changes from 15 to 16. According to that point, presumably, the dynamics of system (4.52) are determined by the poles λ_i, i = 1..15. In order to confirm the assumption, let us investigate the character of the modes. In Fig. 4.37 - Fig. 4.44, we can see the step and impulse responses of the transfer functions $H_i(s)$ (weighting functions) for i = 1..15 and in Fig 4.45 the responses of the systems corresponding to expansion (4.45) for the maximum i = 2..15. Obviously, the most significant modes are those corresponding to the poles $\lambda_{1,2}$ and $\lambda_{3,4}$. As can be seen in Fig. 4.45, the step response of the system $H_{1,2}(s) + H_{3,4}(s)$ approximates the basic features of the original system, i.e., the dead time and the oscillatory character. Increasing i gradually, i.e., involving i transfer functions $H_i(s)$ in the system approximation, the step responses of the approximation get closer to the step response of the system $G(s)$. Finally, as i = 15, the approximation is so good that its approximation error is not seen in Fig. 4.45 and its step response seems to be identical with the response of (4.52). The approximation error can be seen in the very detailed view in Fig. 4.46. According to the step responses shown in Fig. 4.45 and Fig.4.46, all the substantial modes of the TDS are approximated by means of the finite order approximation (4.45) with maximum i = 15, (considering the ordering in Tab. 4.1). As can be seen in Tab. 4.1, there are only small differences in the evaluation of the pole significance using the criteria $\left|R(\lambda_i)\right|$ and $h_{\mathrm{e}i}$. On the basis of such a result, it might be concluded that using the criterion $\left|R(\lambda_i)\right|$ is sufficient and the evaluation of the weighting functions is not necessary. However, as will be shown later, evaluation using $\left|R(\lambda_i)\right|$ fails if there are poles considerably close to each other in the system pole spectrum.

Tab. 4.1 The poles of system (4.52) and the values of their significance evaluating criteria

i	λ_i	μ_{Hi}	$\frac{\lvert\mu_{Hi}\rvert}{\lvert\lambda_i\rvert}$	$R(\lambda_i)$	$\lvert R(\lambda_i)\rvert$	h_{ei}
1	-0.081 + 0.176j	-0.043	0.219	(-1.379 - 0.299j) 10^{-1}	1.411 10^{-1}	3.575 10^{-1}
3	-0.128 + 0.089j	-0.171	1.099	(6.661 - 3.241j) 10^{-2}	7.408 10^{-2}	1.351 10^{-1}
5	-0.516 + 0.352j	-1.111	1.780	(1.749 - 2.960j) 10^{-2}	3.438 10^{-2}	3.559 10^{-2}
7	-0.321 + 0.472j	-2.284	4.003	(0.715 - 2.976j) 10^{-2}	3.061 10^{-2}	3.430 10^{-2}
9	-0.656 + 0.242j	-0.876	1.252	(1.812 - 1.642j) 10^{-2}	2.445 10^{-2}	3.625 10^{-2}
11	-0.730 + 0.124j	-0.779	1.051	(1.943 - 0.755j) 10^{-2}	2.084 10^{-1}	3.885 10^{-2}
13	-0.754			1.986 10^{-2}	1.986 10^{-2}	1.986 10^{-2}
14	-0.049 + 0.636j	6.058	9.490	(-0.785 - 7.528j) 10^{-3}	7.569 10^{-3}	2.379 10^{-2}
16	-0.050 + 1.257j	-5.408	4.300	(-1.776 + 7.574j) 10^{-5}	7.779 10^{-5}	2.779 10^{-4}
18	-0.315 + 1.252j	-0.593	0.459	(-9.372 + 2.079j) 10^{-6}	9.599 10^{-6}	2.725 10^{-5}
20	-0.050 + 1.885j	2.456	1.303	(1.584 + 2.106j) 10^{-6}	2.635 10^{-6}	9.544 10^{-6}
22	-0.050			-1.081 10^{-6}	1.081 10^{-6}	1.081 10^{-6}
23	-0.331 + 2.086j	8.837	4.184	(-0.269 - 1.179j) 10^{-7}	1.209 10^{-7}	2.960 10^{-7}

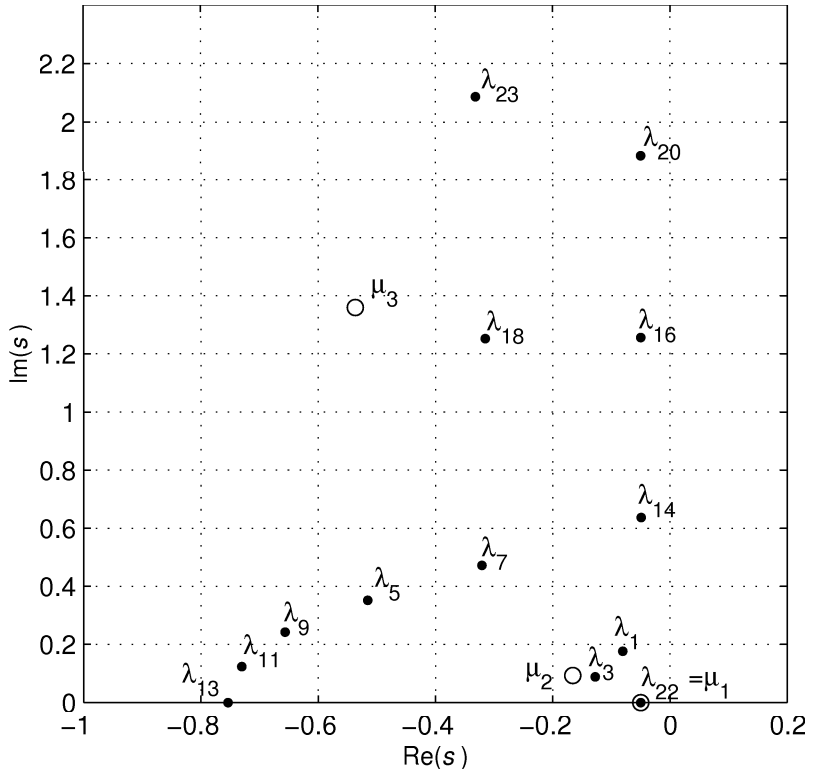

Fig. 4.36 Poles (black circles) and zeros (empty circles) of the system given by (4.52). The poles are ordered with respect to $\lvert R(\lambda_i)\rvert$, see Tab. 4.1

Caption 4.2 The step responses $y_i(t)$ - solid, and the weighting functions $h_i(t)$ - dashed, of the transfer functions $H_i(s)$ given by (4.46) and (4.47) respectively. The impulse responses have different scales in order to show both the step and the impulse responses in one figure. The scale factor (s.f.) is seen in the captions of the figures.

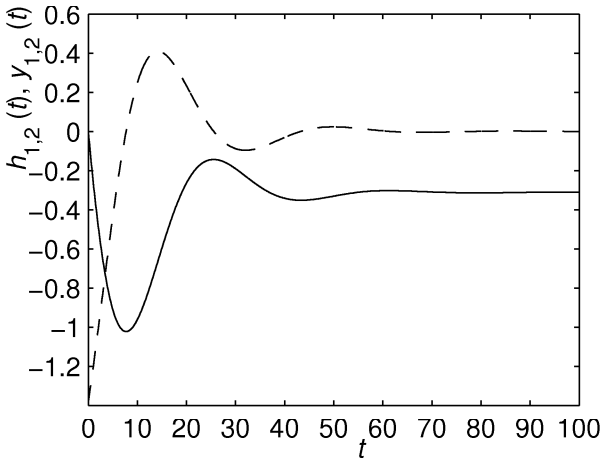

Fig. 4.37 Responses (s.f.=5) of $H_{1,2}(s)$, $\lambda_{1,2} = -0.081 \pm 0.176\,\mathrm{j}$, see Caption 4.2

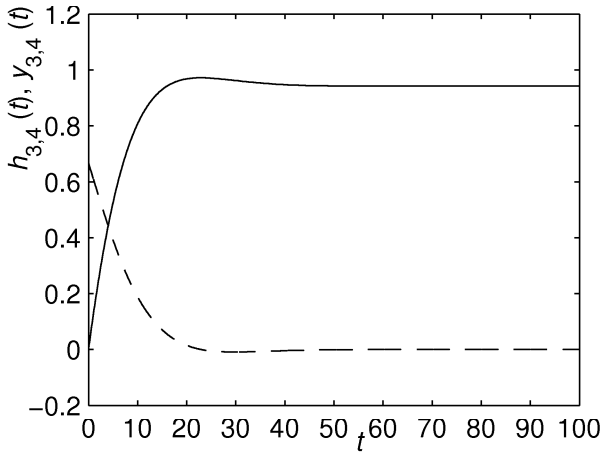

Fig. 4.38 Responses (s.f.=5) of $H_{3,4}(s)$, $\lambda_{3,4} = -0.128 \pm 0.089\,\mathrm{j}$, see Caption 4.2

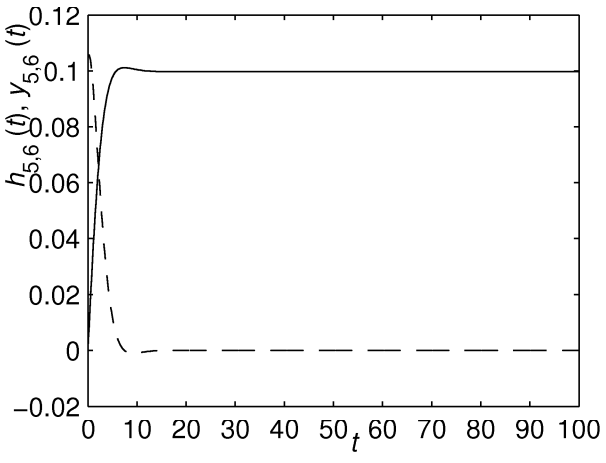

Fig. 4.39 Responses (s.f.=3) of $H_{5,6}(s)$, $\lambda_{5,6} = -0.516 \pm 0.352\,\mathrm{j}$, see Caption 4.2

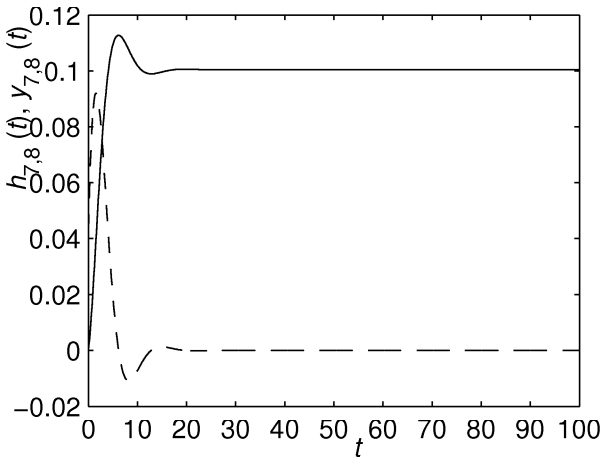

Fig. 4.40 Responses (s.f.=3) of $H_{7,8}(s)$, $\lambda_{7,8} = -0.321 \pm 0.472\,\mathrm{j}$, see Caption 4.2

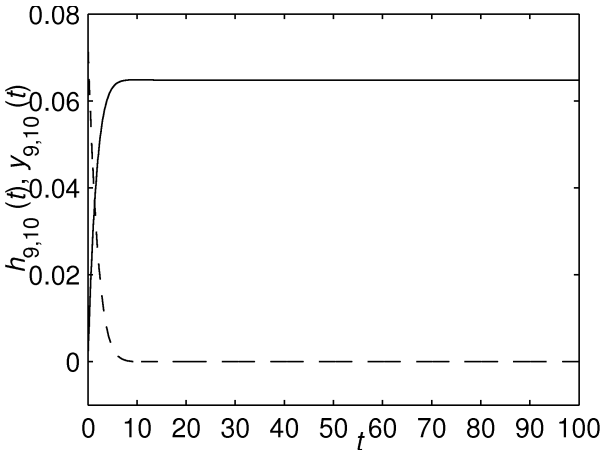

Fig. 4.41 Responses (s.f.=2) of $H_{9,10}(s)$, $\lambda_{9,10} = -0.081 \pm 0.176\,\mathrm{j}$, see Caption 4.2

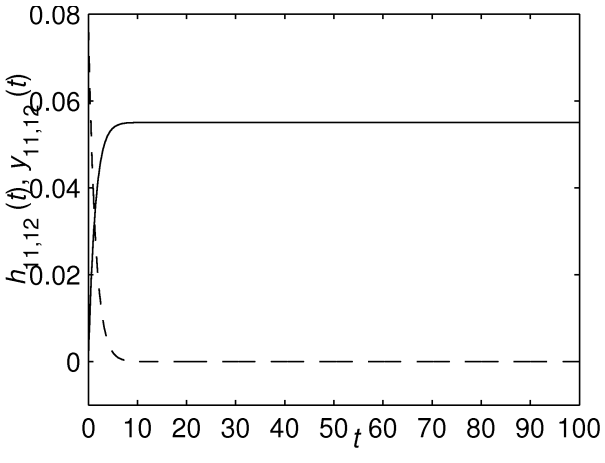

Fig. 4.42 Responses (s.f.=2) of $H_{11,12}(s)$, $\lambda_{11,12} = -0.730 \pm 0.124\,\mathrm{j}$, see Caption 4.2

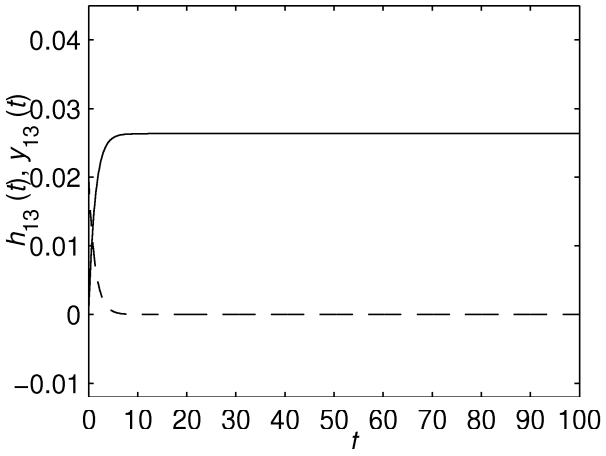

Fig. 4.43 Responses (s.f.=1) of $H_{13}(s)$, $\lambda_{13} = -0.754$, see Caption 4.2

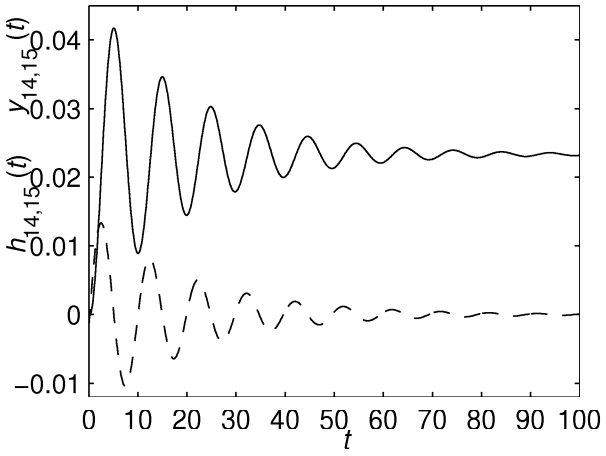

Fig. 4.44 Responses (s.f.=1) of $H_{14,15}(s)$, $\lambda_{14,15} = -0.049 \pm 0.636\mathrm{j}$, see Caption 4.2

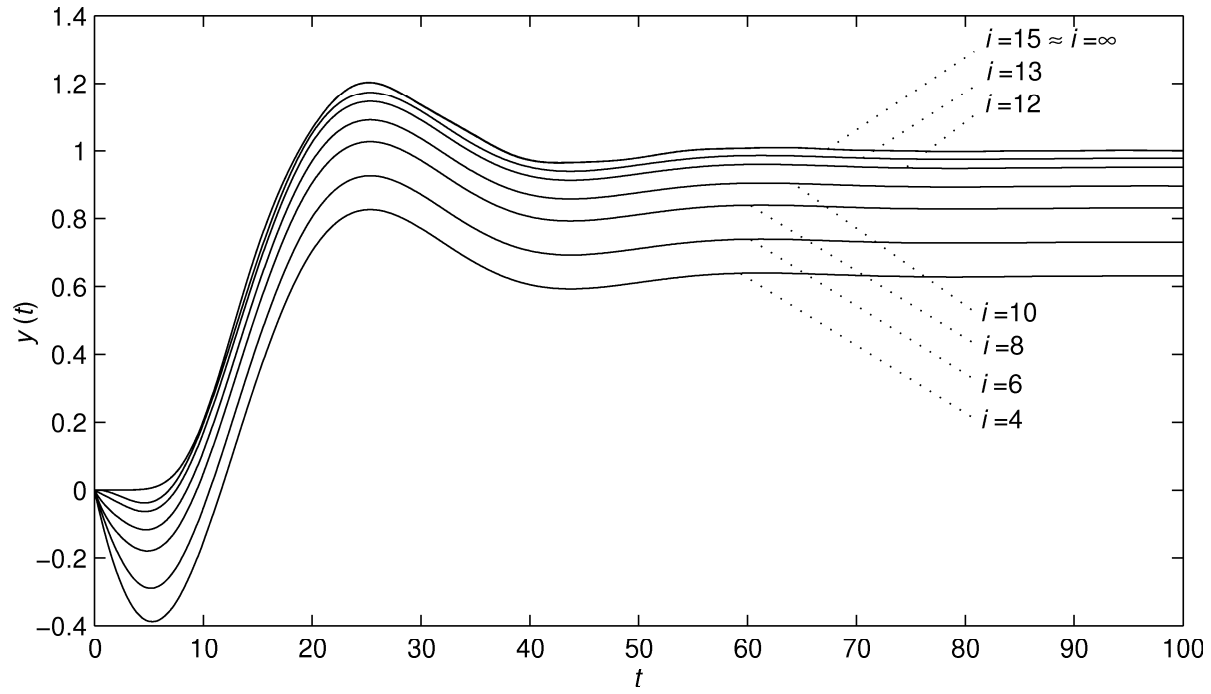

Fig. 4.45 Comparison of the step response of $G(s)$ given by (4.52) (TDS) with the step responses of the systems given by the sums of the first i transfer functions $H_i(s)$ of expansion (4.45)

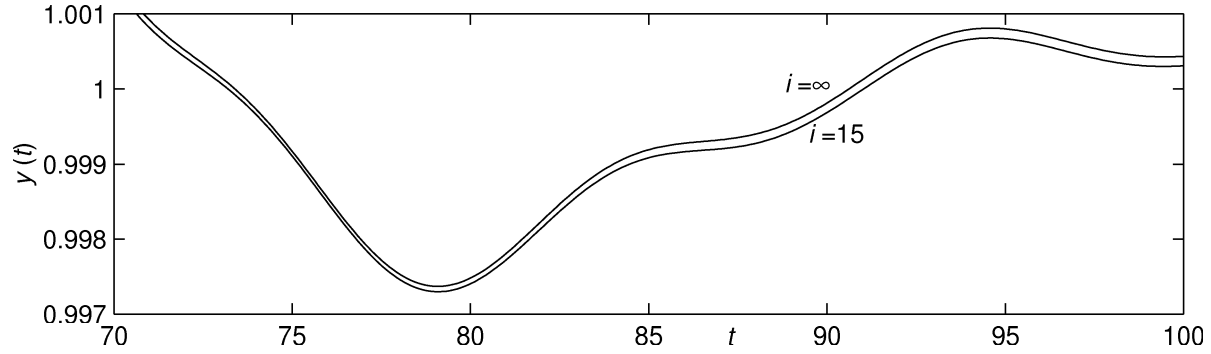

Fig. 4.46 Detailed view of a part of Fig. 4.45

4.4.3 Poles close to each other

In order to demonstrate the problem of evaluating the significance of the poles that are close to each other using the criterion $|R(\lambda_i)|$, let us solve the problem for the simplest TDS

$$G(s) = \frac{1}{s + \exp(-\eta s)} \tag{4.53}$$

As can be seen in Fig. 4.4 and Fig. 4.5, the system has two real distinct poles for $\eta < \exp(-1)$. Starting from $\eta = 0$, as η increases, the mutual distance of the poles decreases. For $\eta = \exp(-1)$ the poles have the same position $\lambda_{1,2} = -\exp(1)$. If the delay η further increases, the poles become complex conjugate pair. In Fig. 4.47, the criteria $|R(\lambda_i)|$ and h_{ei} evaluated for both real poles of system (4.53) for $\eta \in [0.01, \pi/2]$ are seen. If the poles are real, $h_{e1} = |R(\lambda_1)| > h_{e2} = |R(\lambda_2)|$, ($|\lambda_2| > |\lambda_1|$), which is in agreement with the method of evaluating the pole significance on the basis of the distances of the poles from the imaginary axis (the closer the pole, the more significant). However, as can be seen in Fig. 4.47 if the poles are getting close to each other, the values of $|R(\lambda_i)|$, i=1,2, increase rapidly, both asymptoting to the horizontal line drawn at $\eta = \exp(-1)$ with the limit values $\lim_{\eta \to \exp(-1)} |R(\lambda_{1(2)})| = \infty$. As can be seen in Tab. 4.2 and Tab. 4.3, the values of both the criteria corresponding to λ_1 and λ_2 increase considerably as η changes from 0.1 (Tab. 4.2) to 0.367 (Tab. 4.3), while the values of the criteria corresponding to $\lambda_{3,4}$ and $\lambda_{5,6}$ change only slightly. Thus, according to the criteria $|R(\lambda_i)|$ and h_{ei}, the significance of the roots depends on the mutual distance of the poles, which is not obviously true. However, as can be seen in Tab. 4.3, the residues $|R(\lambda_1)|$ and $|R(\lambda_2)|$ have the opposite signs and the absolute value of their sum $|R(\lambda_1) + R(\lambda_2)| = 0.66$ is comparable with the values of $|R(\lambda_1)|$ and $|R(\lambda_2)|$ in Tab. 4.2. According to this result, one could suggest evaluating the significance of the poles that are close to each other using $|R(\lambda_1) + R(\lambda_2)|$ instead of evaluating each pole separately using $|R(\lambda_1)|$ and $|R(\lambda_2)|$. In Fig. 4.47, we can also see the characteristic $|R(\lambda_1) + R(\lambda_2)|$ that continues as $|R(\lambda_1) + R(\lambda_2)| = \mathrm{Re}(2R(\lambda_{1,2}))$ as the poles become complex conjugate pair. If the complex pair has considerably small imaginary part, i.e., the poles are close to each other, the absolute value of $|R(\lambda_{1,2})|$ is very large as well. As it is seen in Fig. 4.47, $2\mathrm{Re}(R(\lambda_{1,2}))$ keeps low values even for $\eta \to \exp(-1)_+$. Thus, consequently, $|R(\lambda_{1,2})|$ is large as $\eta \to \exp(-1)_+$ because of the large imaginary parts of $R(\lambda_{1,2})$. Regarding the results achieved, using $\mathrm{Re}(2R(\lambda_{1,2}))$ to evaluate the significance of the complex conjugate poles seems to be better than to evaluate the significance using $|R(\lambda_{1,2})|$. However, neither the criterion $\mathrm{Re}(2R(\lambda_{i,i+1}))$ can be used to evaluate truly the significance of the complex poles since also the imaginary parts of $R(\lambda_{i,i+1})$ may influence considerably the dynamics of the modes corresponding to the complex conjugates poles .

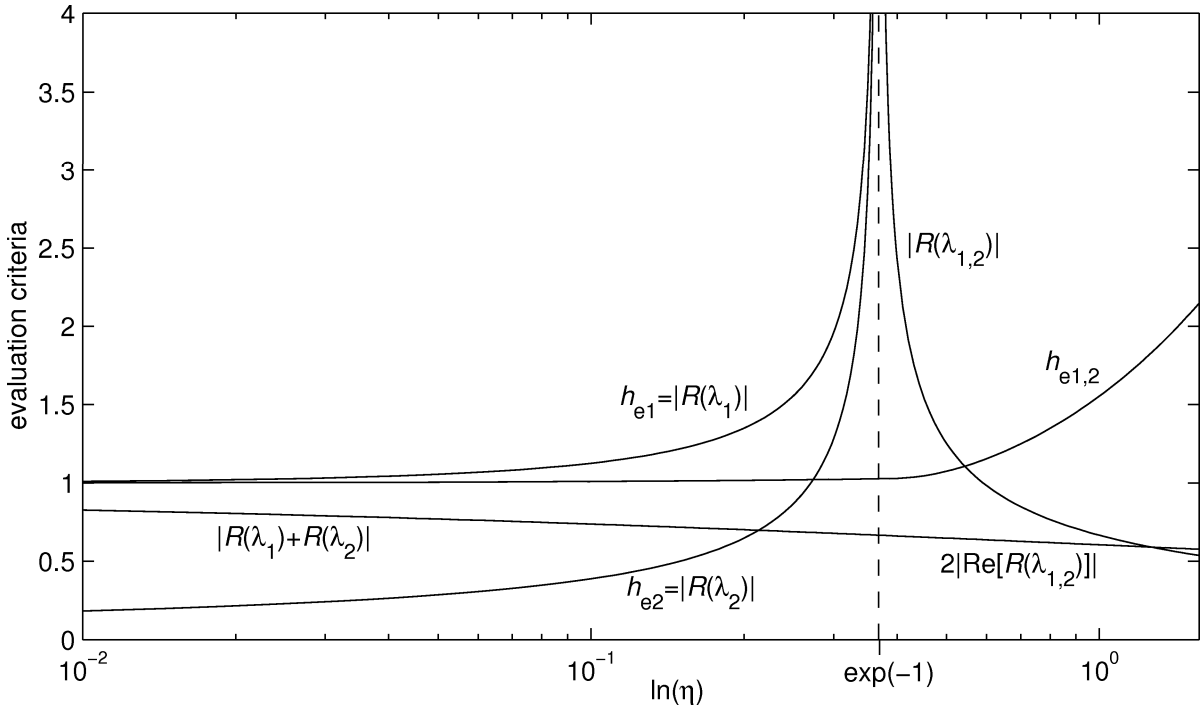

Fig. 4.47 The evaluating criteria of the significance of the dominant couple of poles of system (4.53)

Caption 4.3 The values of evaluating criteria applied to the most significant poles of (4.53) with different values of η

Tab 4.2 $\eta = 0.1$, see Caption 4.3

i	λ_i	$\lvert R(\lambda_i)\rvert$	h_{ei}
1	-1.12	1.12	1.12
2	-35.77	|-0.39|	0.39
3	-44.49 + 73.07j	0.12	0.19
5	-49.88 + 137.9j	0.07	0.12

Tab 4.3 $\eta = 0.367$, see Caption 4.3

i	λ_i	$\lvert R(\lambda_i)\rvert$	h_{ei}
1	-2.54	14.79	14.79
2	-2.92	|-14.13|	14.13
3	-8.42 + 20.33j	0.13	0.20
5	-9.99 + 37.82j	0.07	0.13

Tab. 4.4 $\eta = 0.368$, see Caption 4.3

i	λ_i	$\lvert R(\lambda_i)\rvert$	h_{ei}
1	-2.72 + 0.07j	39.06	1.03
3	-8.39 + 20.28j	0.13	0.20
5	-9.96 + 37.71j	0.07	0.13

Tab. 4.5 $\eta = 1$, see Caption 4.3

i	λ_i	$\lvert R(\lambda_i)\rvert$	h_{ei}
1	-0.32 + 1.34j	0.67	1.55
3	-2.06 + 7.59j	0.13	0.24
5	-2.65 + 13.95j	0.07	0.16

Much better results in evaluating the significance of two poles close to each other are achieved by evaluating the difference between the maximum and minimum of the weighting functions of the modes, i.e., the criterion h_{ei} given by (4.51). In Tab. 4.3 we can see that h_{e1} and h_{e2} are also much higher than $h_{e3,4}$ and $h_{e5,6}$. However, if we evaluate the weighting function of $H_1(s)+H_2(s)$ instead of evaluating the modes separately, we obtain the reasonable result $h_{e1,2}=1.03$, see also Tab. 4.4. In Fig. 4.47, we can see, that primarily, when

λ_2 is much farther from the imaginary axis than λ_1, i.e., λ_1 is the dominant pole, $h_{e1,2}$ holds the value of h_{e1}. If the values of h_{e1} and h_{e2} increase because the poles get close to each other, $h_{e1,2}$ holds almost constant value. Obviously, this is in the agreement with the true significance of the couple in the system dynamics. Even though the pole λ_1 moves to the left as η increases, i.e., its significance decreases, the pole λ_2 moves to the right, i.e., its significance increases. As can be seen, the significance of the couple keeps almost a constant level up to the value $\eta = \exp(-1)$. After the dominant couple of the poles becomes complex conjugate and $\eta \to \pi/2$, the value $h_{e1,2}$ given by (4.51) increases. It confirms the reliability of the evaluating criterion h_{ei} because as $\eta \to \pi/2$ the dominant couple gets very close to the imaginary axis (as well as to the *s*-plane origin) and its significance increases, see Tab. 4.5.

To sum up, the evaluation of the significance of the poles using criterion (4.51) seems to be a very valuable tool in the process of selecting the group of the dynamics determining poles. The advantage of the method consists in the possibility to evaluate the significance of a group of poles, which is performed by locating the global extrema of the weighting function of $\sum_{i=a}^{b} H_i(s)$. This feature is especially convenient in case of evaluating the group of close poles, for which the values of h_{ei}, $i = a..b$, are unreasonably large, as a rule.

4.4.4 Multiple poles

Since the Heaviside expansion can be applied only to the modes of the dynamics corresponding to the single poles, the evaluation of the significance of the multiple poles is not an easy task to work out. In case of classical systems with the polynomial numerator and denominator, the transfer function $G(s)$ is expanded into the partial fractions using the simple evaluation of the coefficients corresponding to the particular powers of *s*, see Angot, (1952). Having a multiple pole λ with the multiplicity *n*, the modes corresponding to this pole are of the form

$$\begin{aligned} &L^{-1}\left(\frac{A_1}{s-\lambda}+\frac{A_2}{(s-\lambda)^2}+\frac{A_3}{(s-\lambda)^3}\ldots+\frac{A_n}{(s-\lambda)^n}\right)= \\ &=A_1\exp(-\lambda t)+A_2 t\exp(-\lambda t)+\frac{A_3}{2}t^2\exp(-\lambda t)\ldots+\frac{A_n}{(n-1)!}t^{n-1}\exp(-\lambda t) \end{aligned} \tag{4.54}$$

As can be seen, the modes are of quite complicated forms which makes the pole significance evaluation even more difficult. Anyway, such a method based on evaluating the powers of *s* cannot be directly used for TDS since the denominator of the transfer function is quasipolynomial in which the powers of *s* are not multiplied by the coefficients but by the more complicated terms (exponential functions).

The solution of the problem I am going to propose is based on the fact that from the physical point of view the contribution of the multiple pole with the multiplicity *n* to the system dynamics is more or less equivalent to the contribution of the group of *n* single poles located in the vicinity of the position of the multiple pole. In the previous chapter I have shown how to evaluate the significance of such a group of single poles with small mutual distances. Using such an evaluation, the idea of evaluating the multiple poles is as follows. Firstly, let the multiple pole of the multiplicity *n* be compensated by a multiple zero with the same multiplicity and of the same value as the multiple pole. Secondly, let the multiple pole be substituted by a group of *n* single poles located close to the value of the multiple pole.

Finally, let the significance of the group of n single poles be evaluated using the criterion based on evaluating the weighting function of the system of transfer functions $\sum_{i=a}^{a+n} H_i(s)$, where $H_i(s)$ are the transfer functions of the modes corresponding to the introduced single poles. To figure out this idea, the multiple pole λ_m is to be compensated in the system transfer function and replaced by the modes corresponding to $\lambda_{s,i} = \lambda_m + \varepsilon_i$, $\varepsilon_i \neq \varepsilon_j$, $i = 1,2..n$, $j = 1,2..n$, $i \neq j$, thus

$$\frac{N(s)}{M(s)} \rightarrow \frac{N(s)(s-\lambda_m)^n}{M(s)\prod_{i=1}^{n}(s-\lambda_{s,i})} \tag{4.55}$$

where n is the multiplicity of the pole. The procedure of evaluating the significance of the multiple poles will be shown in Example 4.8.

Example 4.8

Let us consider the zero-free system with the following transfer function

$$G(s) = \frac{1}{[10s^2 + 3(\exp(-s) + \exp(-4.3s))s + 0.5(1 + \exp(-6s))](3s + \exp(-1.2s))^4} \tag{4.56}$$

A group of poles of system (4.56) that are the closest to the s-plane origin are shown in Fig. 4.48. In order to evaluate the significance of the poles, let us compensate the multiple pole of the value -0.787 ± 0.339j (with the multiplicity four) by the multiple zeros $\mu_{1..8} = -0.787 \pm 0.339\,\mathrm{j}$ and let us substitute the multiple complex conjugate poles by the poles that are located in their vicinities, e.g., $\lambda_{1,2} = -0.787 \pm 0.289\mathrm{j}$, $\lambda_{3,4} = -0.737 + 0.339\mathrm{j}$, $\lambda_{4,6} = -0.787 + 0.389\mathrm{j}$ and $\lambda_{7,8} = -0.8367 + 0.3394\mathrm{j}$. Thus, instead of the dynamics of system (4.56) we will evaluate the dynamics of the following system (which are very close to the dynamics of (4.56) in fact)

$$G_s(s) = G(s)\frac{((s-\mu_1)(s-\mu_2))^4}{\sum_{i=1}^{8}(s-\lambda_i)} \tag{4.57}$$

where $G(s)$ is original system (4.56) and $\mu_{1,2} = -0.787 \pm 0.339\,\mathrm{j}$. The poles of system (4.57) can be seen in Fig. 4.49 and the values of the poles ordered with respect to the value $h_{\mathrm{e}i}$ are in Tab. 4.6. Since the poles $\lambda_{1,2}..\lambda_{7,8}$ have quite small mutual distances, the weighting function of $\sum_{i=1}^{8} H_i(s)$ is to be evaluated instead of evaluating the weighting functions of $H_{i,i+1}(s)$, i=1, 3, 5, 7, separately. As it is seen in Tab. 4.6, if the modes are evaluated separately, the values of criterion $h_{\mathrm{e}i,i+1}$, i=1, 3, 5, 7 are much higher than the value of $h_{\mathrm{e}9}$, which corresponds to the closest pole to the s-plane origin (thus it is likely to be the most significant one). However, if the weighting function of $\sum_{i=1}^{8} H_i(s)$ is considered, the weighting functions of $H_{i,i+1}(s)$, i=1, 3, 5, 7, are superposed and the resultant weighting function and the value of $h_{\mathrm{e}1..8}$ are comparable with the weighting function of $H_9(s)$ and $h_{\mathrm{e}9}$,

respectively, see Fig. 4.50 and 4.51. As can be seen in Tab. 4.6, considering the criterion h_{ei}, the most important modes of system (4.57) are the modes corresponding to the group of the poles $\lambda_{1,2}..\lambda_{7,8}$, and λ_9. In fact, as can be seen in Fig. 4.54, where the step responses of the poles shown in Fig. 4.50 - Fig. 4.53 are gradually superposed, and according to the values $h_{e10,11}$ and h_{e12} also the poles $\lambda_{10,11}$ and λ_{12} are fairly important (but not so important as the poles $\lambda_i, i=1..9$). The values of h_{ei} corresponding to the remaining poles in Tab. 4.6 are much less than the value of h_{e12} corresponding to the less significant pole λ_{12} from the group of the decisive poles, which indicates their unimportance. Comparing the step response of the finite order system given by $\sum_{i=1}^{12} H_i(s)$ with the step response of original system (4.56), both shown in Fig. 4.54, we cannot see any differences. Thus, the group of the poles determining the dynamics of system (4.56) consists of the multiple pole (which we have substituted by the poles $\lambda_{1,2}..\lambda_{7,8}$), the real pole λ_9, the complex conjugate pole $\lambda_{10,11}$ (responsible for the slight oscillations in the step response) and the real pole λ_{12}.

Tab. 4.6 The poles of system (4.57) and the values of the residues and the significance evaluating criterion

i	λ_i	$R(\lambda_i)$	h_{ei}	
1	-0.787 + 0.289j	(-6.275 - 0.886j) 10	$1.259\ 10^2$	$h_{e1..8}=2.032$
3	-0.737 + 0.339j	(4.368 - 3.642j) 10	8.749 10	
5	-0.787 + 0.389j	(1.943 + 1.965j) 10	4.087 10	
7	-0.837 + 0.339j	(-0.137 + 2.960j) 10	9.661	
9	-0.2899	1.957	1.957	
10	-0.072 + 0.462j	$(7.175 + 6.248j)\ 10^{-2}$	$2.756\ 10^{-1}$	
12	-1.113	$-7.0850\ 10^{-2}$	$7.085\ 10^{-2}$	
13	-0.355 + 1.789j	$(1.098 + 1.771j)\ 10^{-5}$	$4.986\ 10^{-5}$	
15	-0.787 + 0.339j	$(0.875 + 2.253j)\ 10^{-8}$	$2.066\ 10^{-8}$	
17	-0.787 + 0.339j	$(-0.883 - 1.979j)\ 10^{-8}$	$2.019\ 10^{-8}$	
19	-0.787 + 0.339j	$(1.230 - 1.613j)\ 10^{-8}$	$2.461\ 10^{-8}$	
21	-0.787 + 0.339j	$(-1.941 + 0.362j)\ 10^{-8}$	$3.893\ 10^{-8}$	

As can be seen in Tab. 4.6, also the multiple poles are evaluated, i.e., $\lambda_{15+2i,16+2i} = -0.787 \pm 0.339\,\mathrm{j}, i=0..3$. Obviously, the poles are very well compensated by the zeros because the significance-evaluating criterion acquires very small values indicating the insignificance of these poles in the system dynamics. It should be noted that to achieve such a result, the values of the poles have to be computed with high precision.

To conclude, it has been shown in Example 4.8 that the compensation of the multiple poles and their substitution by the distinct poles that are close to the multiple poles is reasonable and it is a possible approach to evaluate the significance of the multiple poles.

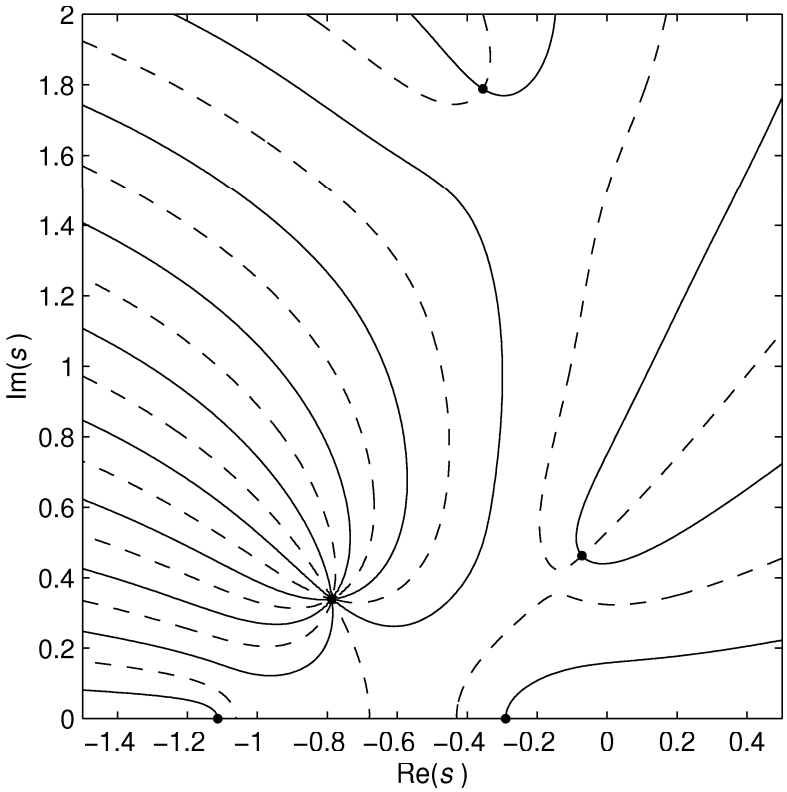

Fig. 4.48 The poles of system (4.56), Re($M(s)$)=0 - solid, Im($M(s)$)=0 - dashed, $M(s)$ -denominator of (4.56)

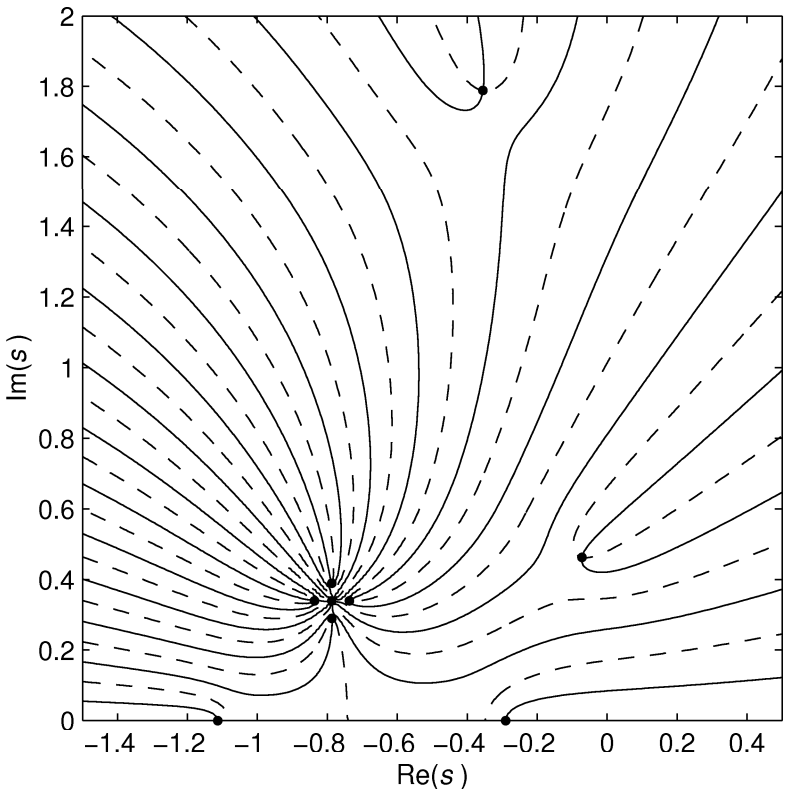

Fig. 4.49 The poles of system (4.57), in which the multiple poles are compensated by the zeros and substituted by a group of distinct poles located in the vicinity of the multiple pole. Re($M(s)$)=0 - solid, Im($M(s)$)=0 - dashed, $M(s)$ - denominator of (4.57)

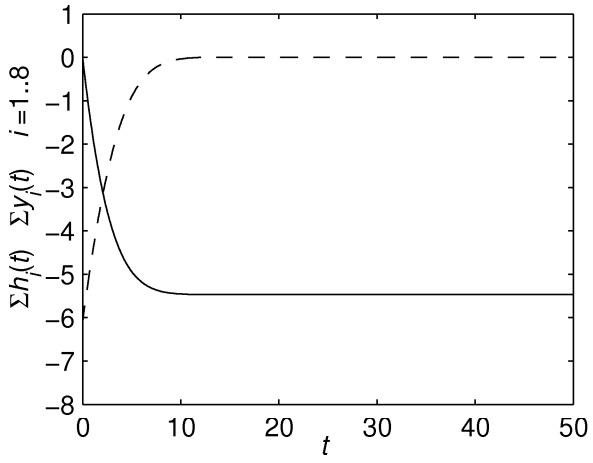

Fig. 4.50 Responses (s.f.=3) of $\sum_{i=1}^{8} H_i(s)$, see Caption 4.2

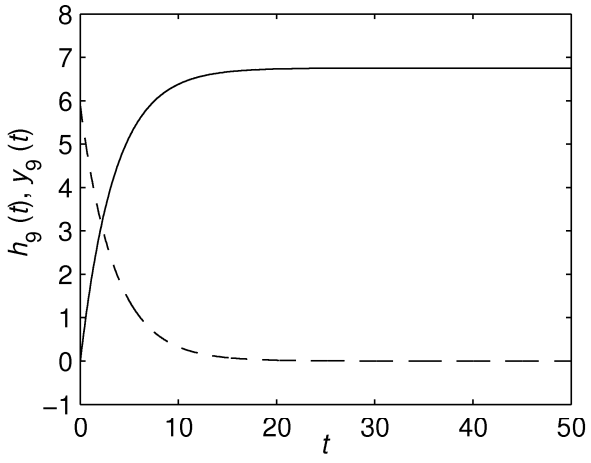

Fig. 4.51 Responses (s.f.=3) of $H_9(s)$, see Caption 4.2

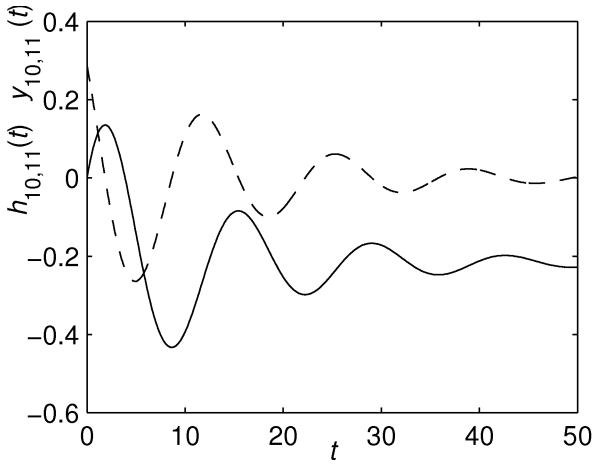

Fig. 4.52 Responses (s.f.=2) of $H_{10,11}(s)$, see Caption 4.2

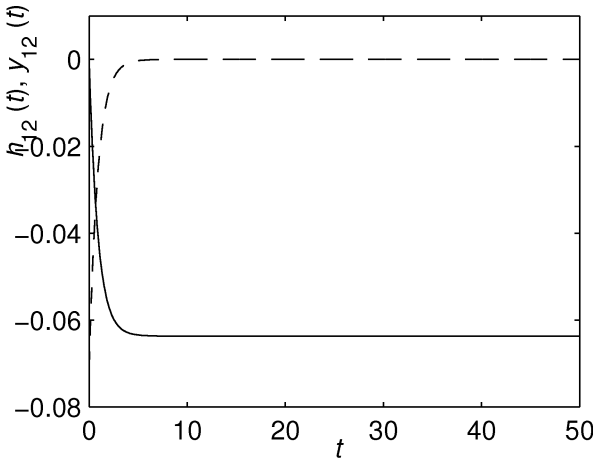

Fig. 4.53 Responses (s.f.=1) of $H_{12}(s)$, see Caption 4.2

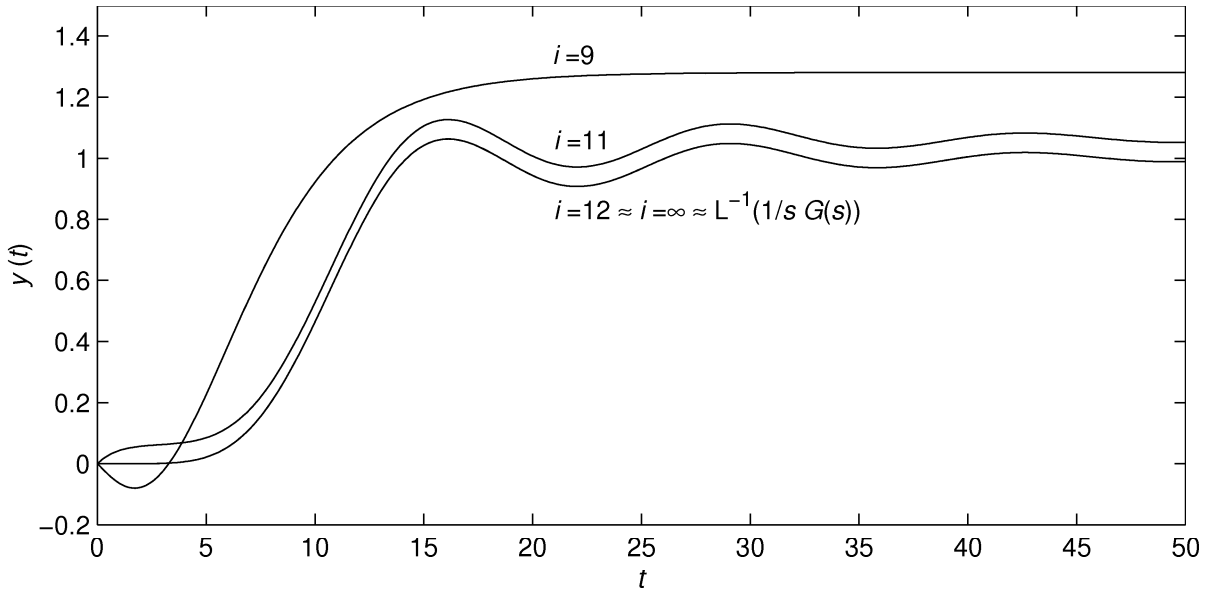

Fig. 4.54 Comparison of the step response of $G(s)$ given by (4.56) (TDS) with the step responses of the systems given by the sums of the first i transfer functions $H_i(s)$ of expansion (4.45) applied to system (4.57)

4.4.5 Global pole significance evaluation

Suppose we have MIMO TDS transformed into the input output transfer functions

$$G_{kl}(s) = \frac{N_{kl}(s)}{M(s)} \tag{4.58}$$

where $N_{kl}(s)$, $M(s)$ are given by (1.43) and (1.44), respectively (for retarded systems $\mathbf{H}_i = \mathbf{0}$, $i = 1..N$), and the indexes k and l denote the particular input-output relation. Each of transfer functions (4.58) may be expanded into (4.45) in which the modes are represented by the transfer functions $H_{kl_i}(s)$ of form (4.46) (for the real poles) and (4.47) for the complex conjugate pairs of poles), respectively. Considering whole system (4.58), the evaluation of the pole significance is not easy, since the poles (which are common for all the transfer functions $G_{kl}(s)$) have different roles in the dynamics of each of the transfer functions $G_{kl}(s)$ depending on the distribution of the zeros of $G_{kl}(s)$. As can be seen, all the transfer functions $G_{kl}(s)$ have the common denominator $M(s)$. Therefore, in order to evaluate the global significance of the poles, instead of the transfer functions $G_{kl}(s)$, let us expand the transfer function

$$G_g(s) = \frac{1}{M(s)} \tag{4.59}$$

into the Heaviside series (4.45). The convenience of evaluating (4.59) is not only in spearing the effort in the pole significance evaluation. As we have seen in Example 4.7, e.g., the pole λ_{22}, which is actually the closest one to the s-plane origin, is insignificant in the dynamics because it is compensated by the zero μ_1. Obviously, such an evaluation of the pole λ_{22} may be quite unrobust. If the system parameters change, the zero-pole compensation is not complete and the significance of the pole λ_{22} is likely to rise significantly. On the other hand, the evaluation of the poles based on (4.59) is robust since (4.59) does not have any zeros to influence the significance of the poles. Another reason of using (4.59) in evaluating the pole significance is given by the fact that in the control design methods based on pole placement, which are used to obtain safely stabilised damped system dynamics, the role of zeros in the dynamics is not considered.

Example 4.9

Consider system (4.52) from Example 4.7. In Tab. 4.7, we can see the poles of the system ordered with respect to the criterion h_{ei} applied to the weighting functions of the partial fractions of system (4.59) where $M(s)$ is given by the denominator of (4.52). Obviously the significance evaluation of the poles of the zero-free system gives quite different results compared to the significance evaluation of the poles of the system (4.52). First of all, the pole λ_{22} has been evaluated as the second most important. In fact the pole λ_{22} is the most important pole, since $\lambda_{3,4}$ and $\lambda_{1,2}$ are quite close to each other and if evaluated as such poles, i.e., the weighting function $M_{1,2}(s) + M_{3,4}(s)$ is evaluated, $h_{e1,4} = 0.331$, which is close to h_{e22}. Also most of the other poles have different positions in Tab. 4.7 than in Tab. 4.1. However, the changes in their positions are not so different as the change of the position of the pole λ_{22}. As can be seen in Fig. 4.36, the most significant poles according to Tab. 4.7, i.e., λ_{22}, $\lambda_{3,4}$ and $\lambda_{1,2}$ are the closest poles to the s-plane origin. This result points out the good features of the evaluation based on the expansion of (4.59).

Tab. 4.7 The poles of system (4.59) with $M(s)$ is given by the denominator of (4.52), the values of the residues and the significance evaluating criterion

i	λ_i	$R(\lambda_i)$	h_{ei}
3	-0.128 + 0.089j	(-2.525 + 2.111j) 10^{-1}	5.109 10^{-1}
22	-0.050	3.117 10^{-1}	3.117 10^{-1}
1	-0.081 + 0.176j	(9.583 - 1.237j) 10^{-1}	2.395 10^{-1}
7	-0.321 + 0.472j	(7.352 + 0.338j) 10^{-4}	1.693 10^{-3}
5	-0.516 + 0.352j	(1.937 - 3.928j) 10^{-4}	4.058 10^{-4}
14	-0.049 + 0.636j	(0.517 + 1.599j) 10^{-4}	5.197 10^{-4}
9	-0.656 + 0.242j	(-0.512 - 1.3589j) 10^{-4}	1.169 10^{-4}
11	-0.730 + 0.124i	(-6.713 - 4.512j) 10^{-5}	1.343 10^{-4}
13	-0.754	-6.726 10^{-5}	6.726 10^{-5}
16	-0.050 + 1.257j	(1.272 - 4.849j) 10^{-7}	1.793 10^{-6}
18	-0.315 + 1.252j	(6.688 - 2.722j) 10^{-8}	1.965 10^{-7}
20	-0.050 + 1.885j	(-4.438 - 5.265j) 10^{-9}	2.491 10^{-8}
23	-0.331 + 2.086j	(0.872 + 2.044j) 10^{-10}	5.285 10^{-10}

4.5 Gradient based state variable feedback control, direct pole placement

The application field in which the designed quasipolynomial mapping based rootfinder and the evaluation of the pole significance may be utilized are the frequency based control design methods. In this section, according to the fourth objective of the thesis, I am going to investigate the potentials of the control method of TDS based on state variable feedback control, see Zítek and Vyhlídal, (2000,2002a, 2002b). Consider coefficient feedback from the state variables (1.127) to control a retarded TDS, i.e., a system of form (1.15). Such a coefficient feedback does not accomplish a state feedback in fact if applied to TDS because the vector $\mathbf{x}(t)$ does not represent the system state any more. In spite of this fact, feedback (1.127) has proved to be an efficient tool to design robustly stable dynamics in TDS. Closing feedback (1.127), the characteristic equation of the feedback system is as follows

$$M(s,\mathbf{K}) = \det\left[s\mathbf{I} - \mathbf{A}(s) + \mathbf{B}(s)\mathbf{K} \right] = 0 \qquad (4.60)$$

Suppose the original spectrum of poles, i.e., the eigenvalues of the system matrix $\mathbf{A}(s)$, is $\mathrm{Sp}(\mathbf{A}(s)) = \{\lambda_i\}, i = 1..\infty$ and the characteristic quasipolynomial of the original system is $M_0(s) = \det[s\mathbf{I} - \mathbf{A}(s)]$. Closing feedback (1.127) the characteristic quasipolynomial of the system changes from $M_0(s)$ to $M(s)$ given by (4.60). Such a change of the quasipolynomial result in the change of the spectrum of the system poles into $\mathrm{Sp}(\mathbf{A}(s) - \mathbf{B}(s)\mathbf{K}) = \{\sigma_i\}, i = 1..\infty$. Since quasipolynomial (4.60) is linear with respect to $\mathbf{K}$ (Vaněček, 1990), (Zítek and Vyhlídal, 2002a,b), the following relationship holds between the original $M_0(s)$ and the feedback system quasipolynomial $M(s,\mathbf{K})$

$$M(s,\mathbf{K}) = M_0(s) + \sum_{j=1}^{n} \frac{\partial M(s,\mathbf{K})}{\partial K_j} K_j \tag{4.61}$$

where the gradient derivatives of $M(s, \mathbf{K})$ are independent on $\mathbf{K}$. Obviously these derivatives are variable in s too, owing to the exponential functions in $M(s,\mathbf{K})$.

Consider the original system has undesirable dynamics, i.e., the group of the most significant poles of the original system brings about too slow or less damped character of the dynamics or the system is even unstable. The aim of introducing the feedback from the state variables is to place the most significant system poles into prescribed new positions $s = \sigma_i, i = 1,2,..., \; i_{\max} \le n$ (n is system order), which are chosen to endow the system with more favourable dynamics. For any prescription of σ_i the following relationship holds

$$M(\sigma_i,\mathbf{K}) = 0 = M_0(\sigma_i) + \sum_{j=1}^{r} K_j \left[\frac{\partial M(s,\mathbf{K})}{\partial K_j} \right]_{s=\sigma_i} \tag{4.62}$$

i.e., a set of linear algebraic equations with the unknown parameters $K_1, K_2, ..., K_r$. In fact, equations (4.62) corresponds only to the prescribed real poles σ_i. If a prescribed pole is complex $\sigma_i = \beta_i + \mathrm{j}\omega_i$, equation (4.62) has to be split into two equations

$$\mathrm{Re}(M(\sigma_i,\mathbf{K})) = 0 = \mathrm{Re}(M_0(\sigma_i)) + \sum_{j=1}^{r} K_j \,\mathrm{Re}\left(\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j} \right]_{s=\sigma_i} \right) \tag{4.63}$$

$$\mathrm{Im}(M(\sigma_i,\mathbf{K})) = 0 = \mathrm{Im}(M_0(\sigma_i)) + \sum_{j=1}^{r} K_j \,\mathrm{Im}\left(\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j} \right]_{s=\sigma_i} \right) \tag{4.64}$$

arising from the separation process of the real and imaginary parts of each of the elements of (4.62). Prescribing a complex pole $\sigma_i = \beta_i + \mathrm{j}\omega_i$ to the feedback system, also its complex conjugate pole $\sigma_i = \beta_i - \mathrm{j}\omega_i$ is being assigned automatically, which justifies the fact of having two equations (4.63) and (4.64) for one prescribed complex pole.

In tuning the feedback coefficients, not only the single σ_i but also multiple poles can be prescribed to the feedback system. If one of the prescribed σ_i, e.g., σ_a is a multiple pole with the multiplicity d, only one condition of the forms either (4.62) (for real pole) or (4.63) and (4.64) (imaginary pole), with $s = \sigma_a$ can be used in the set of equation for computing $\mathbf{K}$. The missing d-1 conditions are to be provided as

$$\left[\frac{d^{\nu} M(s,\mathbf{K})}{ds^{\nu}} \right]_{s=\sigma_a} = \left[\frac{d^{\nu} M_0(s)}{ds^{\nu}} \right]_{s=\sigma_a} + \sum_{j=1}^{r} K_j \left[\frac{d^{\nu}}{ds^{\nu}} \left[\frac{\partial M(s,\mathbf{K})}{\partial K_j} \right] \right]_{s=\sigma_a} = 0 \tag{4.65}$$

$v = 1, 2, ..., d-1$. If the prescribed pole is real, d-1 equations of form (4.65) are to be used in the final set of equations. If the pole is complex, analogously to the case of distinct poles, equations (4.65) has to be split into real and imaginary parts

$$\mathrm{Re}\left[\frac{d^v M(s,\mathbf{K})}{ds^v}\right]_{s=\sigma_a} = \mathrm{Re}\left[\frac{d^v M_0(s)}{ds^v}\right]_{s=\sigma_a} + \sum_{j=1}^{r} K_j \,\mathrm{Re}\left[\frac{d^v}{ds^v}\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]\right]_{s=\sigma_a} = 0 \qquad (4.66)$$

$$\mathrm{Im}\left[\frac{d^v M(s,\mathbf{K})}{ds^v}\right]_{s=\sigma_a} = \mathrm{Im}\left[\frac{d^v M_0(s)}{ds^v}\right]_{s=\sigma_a} + \sum_{j=1}^{r} K_j \,\mathrm{Im}\left[\frac{d^v}{ds^v}\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]\right]_{s=\sigma_a} = 0 \qquad (4.67)$$

In case of prescribing the poles as distinct ones, evaluating the partial derivatives in equations (4.62), (4.63) and (4.64) and substituting the prescribed poles into the equations, the following system of equations results

$$\mathbf{S}\mathbf{K} = \mathbf{m} \qquad (4.68)$$

$\mathbf{K} \in \mathrm{R}^r\ (r \le n)$, $\mathbf{S} \in \mathrm{R}^{q \times r}$ and $\mathbf{m} \in \mathrm{R}^q\ (q \le n)$ where

$$\mathbf{S} = \begin{bmatrix} S_{k,j} \\ S_{Rl,j} \\ S_{Il,j} \end{bmatrix},\ S_{k,j} = \left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]_{s=\sigma_k} \text{ for real } \sigma_k,$$
$$S_{Rl,j} = \mathrm{Re}\left(\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]_{s=\sigma_l}\right),\ S_{Il,j} = \mathrm{Im}\left(\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]_{s=\sigma_l}\right) \text{ for complex } \sigma_l \qquad (4.69)$$

and

$$\mathbf{m} = \begin{bmatrix} m_k \\ m_{Rl} \\ m_{Il} \end{bmatrix},\ m_k = M_0(\sigma_k),\ m_{Rl} = \mathrm{Re}(M_0(\sigma_l)),\ m_{Il} = \mathrm{Im}(M_0(\sigma_l)) \qquad (4.70)$$

where $k = 1..q_\mathrm{r}$, q_r is the number of prescribed real poles, $l = 1..q_\mathrm{c}$, q_c is the number of prescribed imaginary poles, $q = q_\mathrm{r} + 2q_\mathrm{c}$, and j=1..r, r is the number of feedback loops from the state variables. If r<n, vector $\mathbf{x}(t)$ is not complete in the feedback (1.127). If any of the prescribed poles are multiple, the matrices **S** and **m** are obtained in the analogous way as in the case of prescribed single poles. The only difference consists in involving the terms of equations (4.65), (4.66) and (4.67) instead of those of (4.62), (4.63) and (4.64) in assembling the matrices **S** and **m**.

Obviously, the maximum number of the poles that might be prescribed using the state variable feedback control is equal to the number of the state variables, i.e., to the system order n. Thus, having $q = n$, the system of equations (4.68) may be solved as

$$\mathbf{K} = \mathbf{S}^{-1}\mathbf{m} \qquad (4.71)$$

if the matrix **S** is nonsigular. However, more numerically stable techniques of solving system of equation (4.68) are the iterative methods, e.g., Gauss-Seidel method, see the practical guide (Barrett, et al., 1994).

In general, there may be less than n significant poles with undesirable positions in the spectrum of the original system. Provided that the other poles are much farther to the left from the imaginary axis, it is reasonable to prescribe new positions only to these poles with undesirable positions. One possibility to solve the task of $q < n$ consists in using only $r = q$ feedback loops, which reduce the problem to solve set of equations (4.68). Another possibility, which is likely to result in more robust dynamics of the feedback system, consists in using all the feedback loops which are available. To obtain the feedback coefficients, set of underdetermined equations (4.68), $r>q$, is to be solved. Using the Moore-Penrose inverse $\mathbf{S}^{+}$ of **S** (Ben-Israeland Greville, 1977), (Penrose, 1955), the feedback coefficients are given by

$$\mathbf{K} = \mathbf{S}^{+}\mathbf{m} \tag{4.72}$$

see also Michiels, et. al., (2002). The Moore-Penrose generalized matrix inverse is a unique matrix pseudoinverse, which provides the solution with the minimal norm

$$\| \mathbf{K} \|_2 = \sqrt{K_1^{\,2} + K_2^{\,2} + ... + K_r^{\,2}} \tag{4.73}$$

Apparently the desired eigenvalue positions are to be prescribed with respect to λ_i, $i = 1,2,...$ constituting the group of the most significant system poles. It is of little sense to assign the insignificant system poles because they cannot affect the actual system behaviour.

The crucial issue of pole assignment in TDS (1.15) is the following. Although equation set (4.72) may be solved for arbitrary set of given $\sigma_i, i = 1,2,...,n$, in fact the region where these prescribed values may be taken from is rather restricted. Obviously, the prescribed $\sigma_i, i = 1,2,...,n$ have to correspond to the eigenvalues λ_i with the largest values of the criterion h_{ei} given by (4.51). Obviously, the set of n eigenvalues is only a little part of the whole spectrum. Thus an infinite set of the rest of eigenvalues is placed spontaneously. To get the prescribed σ_i actually determining the system dynamics, it is necessary that the assigned eigenvalues constitute the set of most significant poles of the system dynamics being designed. Basically, it means that σ_i must not be prescribed too fast, i.e. too far to the left with respect to the original positions of the poles. If such too fast eigenvalue is prescribed, the consequence is that one or more of spontaneously placed eigenvalues takes over the role of the significant poles in the new spectrum. Often such spontaneously placed pole causes the instability of the system. To avoid safely the case of such a pole placement failure, it is necessary to try repeatedly a sequence of the prescribed new positions of the dominant system poles with the stepwise increasing sizes. The critical size of this shifting is when firstly a spontaneously placed dominant pole with undesirable position appears.

Example 4.10

Consider retarded TDS (1.20) with the following functional matrices

$$\mathbf{A}(s) = \frac{1}{100}\begin{bmatrix} -23\exp(-s) & \exp(-15s) & 13\exp(-2.2s) & 4\dfrac{\exp(-3.1s)-\exp(-6.3s)}{3.2s} \\ \dfrac{\exp(-2s)-\exp(-4s)}{2s} & -8\exp(-4.7s) & 2\exp(-3.5s) & \exp(-2.5s) \\ 10\exp(-2.7s) & 10\dfrac{\exp(-2.5s)-\exp(-6s)}{3.5s} & -40\exp(-6.3s) & 6\exp(-7.5s) \\ 3\exp(-0.3s) & 20\exp(-4s) & 9\exp(-5.5s) & -27\dfrac{\exp(-1.5s)-\exp(-7.3s)}{5.8s} \end{bmatrix}$$

$$\mathbf{B}(s) = \frac{1}{100}[37\exp(-6.5s) \quad 2\exp(-3.2s) \quad 70\exp(-2s) \quad 13\exp(-5s)]^T, \quad \mathbf{C} = [1\ 1\ 1\ 1] \qquad (4.74)$$

First of all, let us investigate the significance of the rightmost poles using the criterion based on evaluation of the weighting functions of the particular system modes whose transfer functions are obtained from generalized Heaviside expansion (4.45) of $1/M(s)$, where $M(s) = \det[s\mathbf{I} - \mathbf{A}(s)]$. A part of the spectrum of the system poles are shown in Fig. 4.55 and in Fig. 4.56, which is the detailed view of the poles closest to the s-plane origin.

Tab. 4.8 The poles of the system with matrices (4.74), i.e., the eigenvalues of the matrix $\mathbf{A}(s)$, the values of residues and the values of the significance evaluating criterion

i	λ_i	$R(\lambda_i)$	h_{ei}
1	0.0485 + 0.2869j	(-2.521+1.797j) 10	∞
3	-0.0627	5.7927 10^1	5.793 10
4	-0.0803 + 0.3075j	(-0.1413 -2.242j) 10	4.368 10
6	-0.3017 + 0.1393j	(-2.426-3.128j)	5.163 10
8	-0.1773 + 1.2208j	(8.413-2.180j) 10^{-2}	2.766 10
10	-0.2695 + 2.2185j	(5.369-3.468j) 10^{-3}	2.021 10^{-2}
12	-0.5411 + 1.1945j	(-3.942+4.126j) 10^{-3}	1.085 10^{-2}
14	-0.6497 + 1.5882j	(-6.485-7.476j) 10^{-4}	2.140 10^{-3}
16	-0.7188 + 2.0694j	(1.827+1.110j) 10^{-4}	5.495 10^{-4}
18	-0.7671 + 2.9787j	(1.788-0.317j) 10^{-4}	5.172 10^{-4}
20	-0.8071 + 2.8606j	(0.114+1.134j) 10^{-4}	2.090 10^{-4}

The values of the system rightmost poles with the values of the corresponding residues and with the evaluation criterion h_{ei} given by (4.51) are in Tab. 4.8. According to the evaluation criterion h_{ei}, obviously, besides the unstable pole for which $h_{ei} = \infty$, there are other seven poles, those of i = 3..9, for which the criterion h_{ei} acquires considerably higher values than for the other poles in Tab. 4.8., i.e., those of i = 10,.., 21.

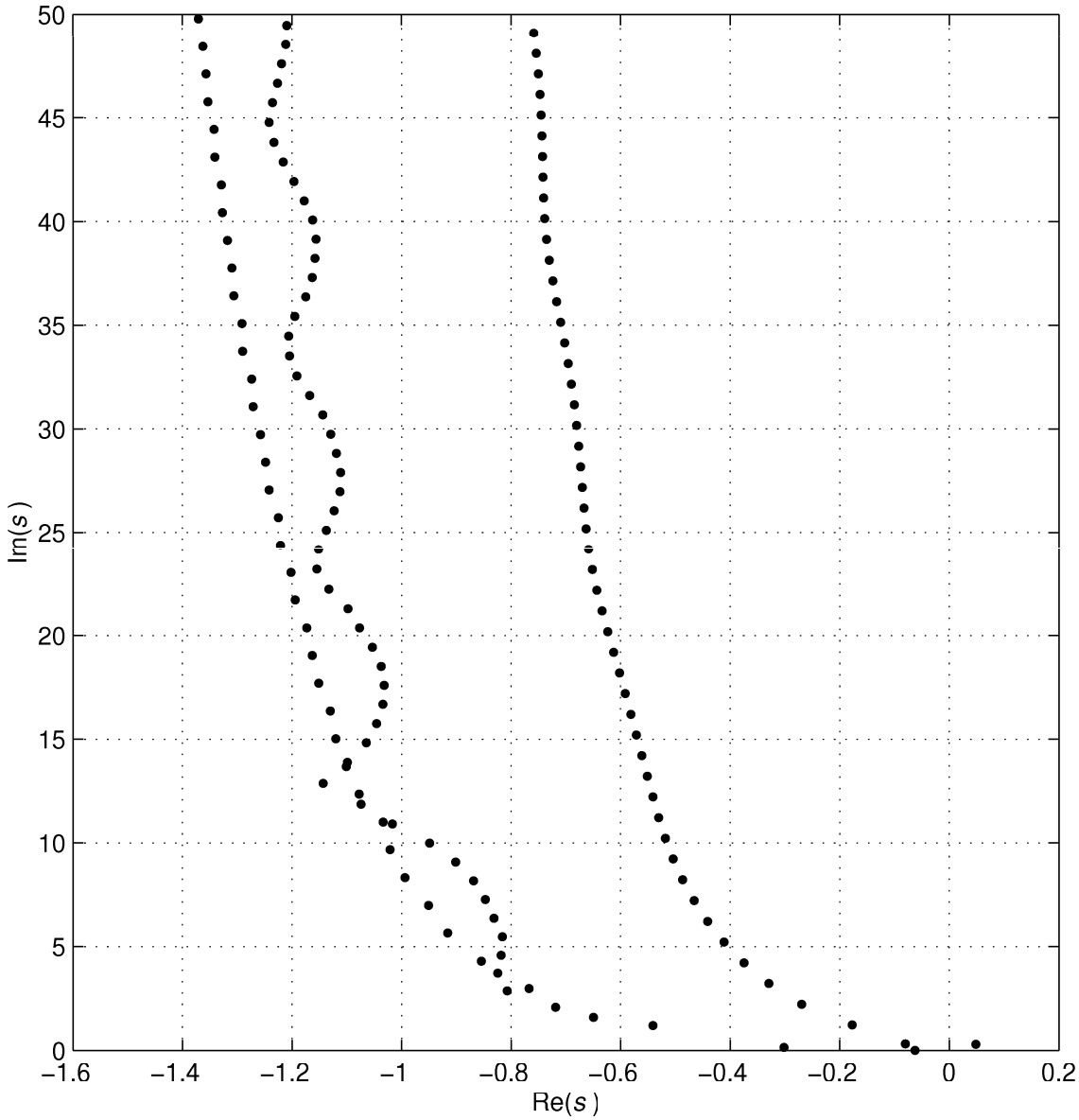

Fig. 4.55 The poles of system with matrices (4.74), i.e.,
the eigenvalues of the matrix $\mathbf{A}(s)$

Suppose the same number of feedback parameters and prescribed poles is considered. The system is of the fourth order which offers the opportunity to prescribe directly one of the following sets: four real, two real and one complex conjugate or two complex conjugate poles of the system. Using formula (4.72), the number of prescribed system poles can be lower. The aim of the pole assignment procedure is to stabilize the system and to achieve fast, damped and robust dynamics of the system. The crucial issue of this task is to prescribe such a set of the system poles which will assure favourable dynamics. Simultaneously, none of the non-prescribed poles brings about an unfavourable mode to the dynamics. Therefore, prescribing the system poles, the positions of the non-prescribed spontaneously placed poles have to be checked using the quasipolynomial mapping based rootfinder given by the Algorithm 3.1. Four of the possible sets of prescribed system poles are given in Tab. 4.9. For each set, the positions of the most significant system poles and the step responses of the system are shown in Fig. 4.57 and Fig. 4.58, respectively.

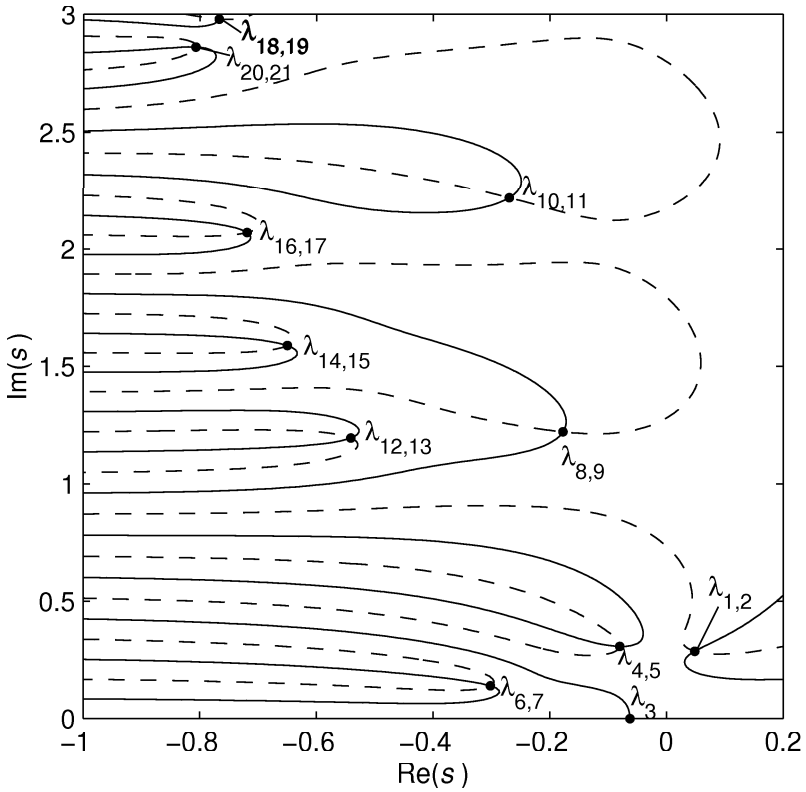

Fig. 4.56 The poles of system with the matrices (4.74), i.e., the eigenvalues of the matrix $\mathbf{A}(s)$, detail of Fig. 4.54

Tab. 4.9. Sets of prescribed system poles and the resultant feedback gains

k_s	Prescribed roots	$\mathbf{K}_{k_s}$	c_{k_s}
1	4 × -0.1	[0.0647 0.31885 -0.4224 -0.0679]	0.0279
2	2 × -0.1, -0.1+0.15j	[0.0610 0.3520 -0.3805 -0.1126]	0.0788
3	2 × -0.15, -0.06+0.15j	[0.0657 0.3563 -0.3630 -0.1222]	0.1455
4	-0.04+0.2j, -0.1+0.15j	[0.0607 0.3971 0.3237 -0.1352]	0.3296

Parameter c in the fourth column of Tab. 4.9 represents the compensation of the static gain of the input-output relation, which is changed by applying the state feedback. To achieve unchanged static behaviour of the system, the static compensation gain is given by the relation

$$c = \lim_{s \to 0} \frac{M(s,K)}{M_0(s)} \tag{4.75}$$

The feedback settings shown in Tab. 4.9 differ from each other in the character of the prescribed eigenvalues. Both real and complex eigenvalues, not only single but also multiple eigenvalue options, have been chosen in the prescribed sets of σ_i. Note that each of these options resulted from a series of several attempts to place the prescribed σ_i into more favourable positions.

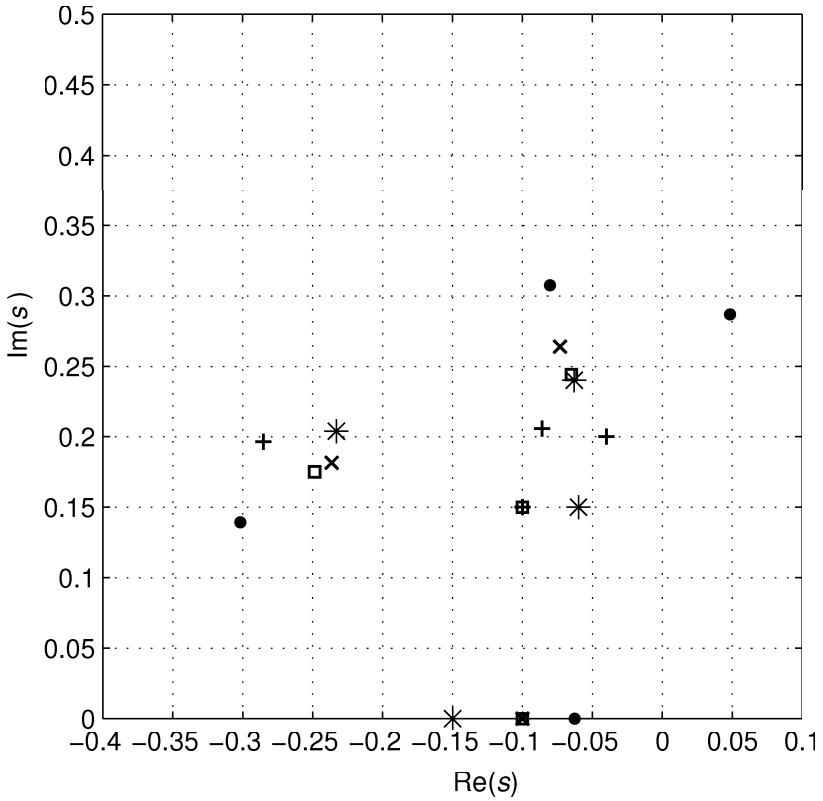

Fig. 4.57 The most significant poles of the feedback system with the feedback gain settings given in Tab. 4.9, black circles - original system poles, ×- $k = 1$, squares - $k = 2$, asterisks - $k = 3$, +- $k = 4$.

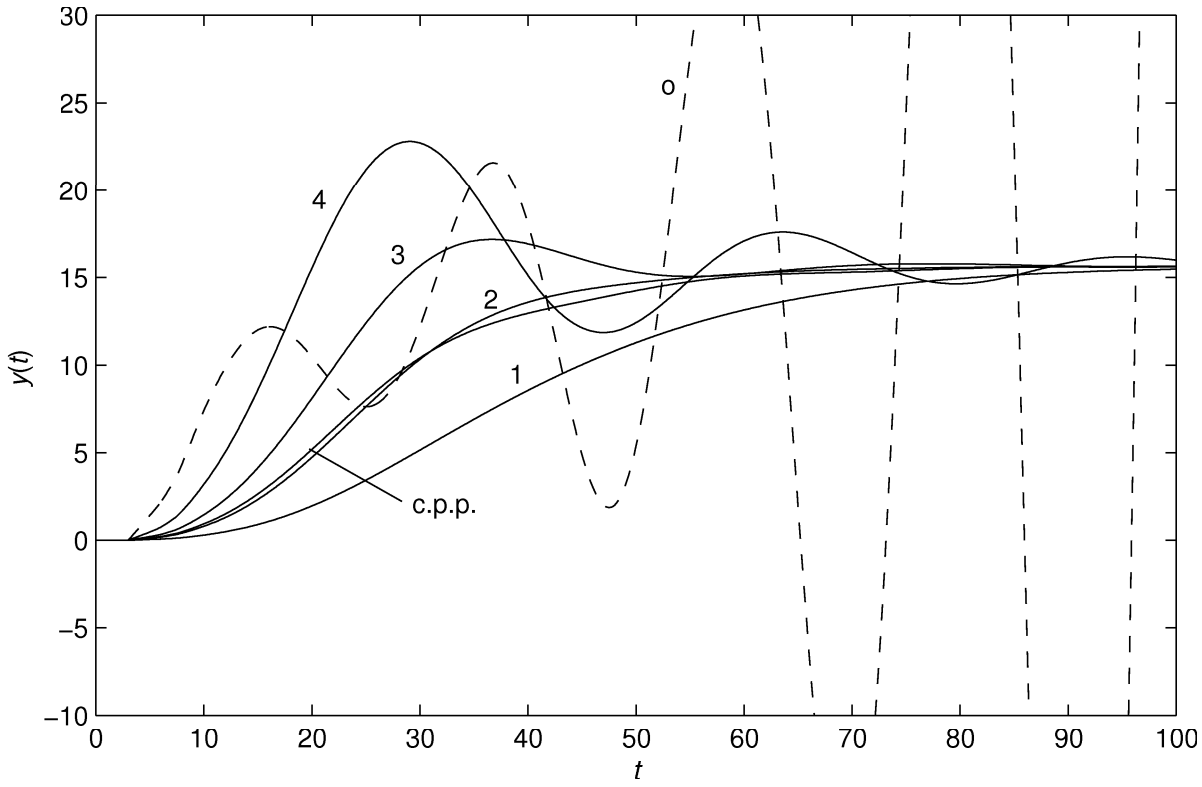

Fig. 4.58 Step responses of the original system (o - dashed) and of the feedback system with the feedback gain settings given in Tab. 4.9, c.p.p. - result of continuous pole placement procedure, see Example 4.12

Comparing step responses of the feedback system with the settings from Tab. 4.9 in Fig. 4.58, it can be seen that the first option (one eigenvalue with multiplicity four) leads to a monotonous but rather slow response. The second option is considerably faster and in spite of the prescribed complex pair, the response is aperiodic from the practical point of view. A small overshoot with a further acceleration of the response is obtained in option $k = 3$ and more oscillating response results if option $k = 4$ is used. The obtained feedback tuning is to be approved as apt only after checking the resulting system spectrum. The spectra of the options $k_{\mathrm{S}} = 1$ through $k_{\mathrm{S}} = 4$ are displayed in Fig. 4.57. Apparently, in all of the options the prescribed σ_i have become the dominant eigenvalues of the system designed and none of the spontaneously placed eigenvalues has taken over this role.

4.6 Continuous pole placement using gradient based state variable feedback control design

As has been shown in section 1.3.5, an ingenious pole placement method for stabilizing a retarded TDS using continuous shifting of the rightmost poles of the system to the left, has been worked out by Michiels, et al., (2002). Prescribing small shifting of the poles from the current positions, the increments of the feedback gain coefficients in $\Delta\mathbf{K}$ can be computed (by means of the sensitivity matrix defined by (1.142)) from equation (1.147). As will be shown, such a continuous shifting can also be performed using the gradient based feedback control design.

As has been shown in the previous section, if the prescribed poles are complex, the equations corresponding to the poles are given by (4.63) and (4.64). Substituting $s = \beta + \mathrm{j}\omega$ into the characteristic quasipolynomial of the closed loop system, i.e., into (4.60), yields

$$\begin{aligned} M(\beta + \mathrm{j}\omega, \mathbf{K}) &= \det\left[(\beta + \mathrm{j}\omega)\mathbf{I} - \mathbf{A}(\beta + \mathrm{j}\omega) + \mathbf{B}(\beta + \mathrm{j}\omega)\mathbf{K} \right] = \\ &= R(\beta, \omega, \mathbf{K}) + \mathrm{j}I(\beta, \omega, \mathbf{K}) \end{aligned} \tag{4.76}$$

Prescribing only the real parts of the complex poles, i.e., β_i, we have the following equations

$$R(\beta_i, \omega_i, \mathbf{K}) = 0 \tag{4.77}$$

$$I(\beta_i, \omega_i, \mathbf{K}) = 0 \tag{4.78}$$

for each of the prescribed β_i with the variables $K_1, K_2, ..., K_r$ and ω_i to be computed. Unlike equation (4.60), equations (4.77) and (4.78) are linear neither with respect to $K_1, K_2, ..., K_r$ nor with respect to ω_i because ω_i are here considered as the unknown variables). This nonlinearity has the inconvenient consequence of loosing the possibility to place the poles arbitrarily. Consider the actual setting of the feedback coefficients is $\tilde{\mathbf{K}}$ and the complex poles $\lambda_i = \beta_i + \mathrm{j}\omega_i$ correspond to this setting. Suppose we displace the real parts of the poles $\beta_i \to \beta_i + \Delta\sigma, i = 1, 2, ..., q_{\mathrm{c}}$ (q_{c} is the number of prescribed complex poles). Provided that $\Delta\sigma$ are small, approximate values of $\Delta\omega_i$ and ΔK_j can be obtained as the solutions of the following set of equations

$$R(\beta_i + \Delta\sigma, \omega_i, \mathbf{K}) + \sum_{j=1}^{r} \Delta K_j \left[\frac{\partial R(\beta, \omega, \mathbf{K})}{\partial K_j} \right]_{\substack{\beta = \beta_i + \Delta\sigma \\ \omega = \omega_i \\ \mathbf{K} = \tilde{\mathbf{K}}}} + \Delta\omega_i \left[\frac{\partial R(\beta, \omega, \mathbf{K})}{\partial \omega} \right]_{\substack{\beta = \beta_i + \Delta\sigma \\ \omega = \omega_i \\ \mathbf{K} = \tilde{\mathbf{K}}}} = 0 \tag{4.79}$$

$$I(\beta_i+\Delta\sigma_i,\omega_i,\mathbf{K})+\sum_{j=1}^{r}\Delta K_j\left[\frac{\partial I(\beta,\omega,\mathbf{K})}{\partial K_j}\right]_{\substack{\beta=\beta_i+\Delta\sigma\\ \omega=\omega_i\\ \mathbf{K}=\tilde{\mathbf{K}}}}+\Delta\omega_i\left[\frac{\partial I(\beta,\omega,\mathbf{K})}{\partial\omega}\right]_{\substack{\beta=\beta_i+\Delta\sigma\\ \omega=\omega_j\\ \mathbf{K}=\tilde{\mathbf{K}}}}=0 \quad (4.80)$$

which result from linearizing (4.77) and (4.78), respectively. Thus, analogously to (4.68), we can write the system of equations

$$\mathbf{S}\begin{bmatrix}\Delta\mathbf{K}\\ \Delta\boldsymbol{\omega}\end{bmatrix}=\mathbf{m} \quad (4.81)$$

$\Delta\mathbf{K}\in\mathbb{R}^r$, $\Delta\boldsymbol{\omega}\in\mathbb{R}^{q_c}$, $\mathbf{S}\in\mathbb{R}^{q\times(r+q_c)}$ and $\mathbf{m}\in\mathbb{R}^q$ where

$$\begin{bmatrix}\Delta\mathbf{K}\\ \Delta\boldsymbol{\omega}\end{bmatrix}=\left[\Delta K_1\,\Delta K_2\ldots\Delta K_r\,\Delta\omega_1\,\Delta\omega_2\ldots\Delta\omega_{q_c}\right]^{\mathrm{T}} \quad (4.82)$$

$$\mathbf{S}=\begin{bmatrix}
S_{1,1} & S_{1,2} & \ldots & S_{1,r} & 0 & 0 & \ldots & 0\\
S_{2,1} & S_{2,2} & \ldots & S_{2,r} & 0 & 0 & \ldots & 0\\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots\\
S_{q_r,1} & S_{q_r,2} & \ldots & S_{q_r,r} & 0 & 0 & \ldots & 0\\
S_{R1,1} & S_{R1,2} & \ldots & S_{R1,r} & S_{R,\omega_1} & 0 & \ldots & 0\\
S_{R1,1} & S_{R1,1} & \ldots & S_{R1,r} & 0 & S_{R,\omega_2} & \ldots & 0\\
\vdots & \vdots & & \vdots & \vdots & & \ddots & \vdots\\
S_{Rq_c,1} & S_{Rq_c,2} & \ldots & S_{Rq_c,r} & 0 & 0 & 0 & S_{R,\omega_{q_c}}\\
S_{I1,1} & S_{I1,1} & \ldots & S_{I1,1} & S_{I,\omega_1} & 0 & \ldots & 0\\
S_{I2,1} & S_{I2,2} & \ldots & S_{I2,r} & 0 & S_{I,\omega_2} & \ldots & 0\\
\vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots\\
S_{Iq_c,1} & S_{Iq_c,2} & \ldots & S_{Iq_c,r} & 0 & 0 & 0 & S_{I,\omega_{q_c}}
\end{bmatrix} \quad (4.83)$$

$$S_{k,j}=\left[\frac{\partial M(s,\mathbf{K})}{\partial K_j}\right]_{\substack{s=\sigma_k+\Delta\sigma\\ \mathbf{K}=\tilde{\mathbf{K}}}},S_{Rl,j}=\left[\frac{\partial R(\beta,\omega,\mathbf{K})}{\partial K_j}\right]_{\substack{\beta=\beta_l+\Delta\sigma\\ \omega=\omega_l\\ \mathbf{K}=\tilde{\mathbf{K}}}},S_{Il,j}=\left[\frac{\partial I(\beta,\omega,\mathbf{K})}{\partial K_j}\right]_{\substack{\beta=\beta_l+\Delta\sigma\\ \omega=\omega_l\\ \mathbf{K}=\tilde{\mathbf{K}}}}$$

$$S_{R,\omega_l}=\left[\frac{\partial R(\beta,\omega,\mathbf{K})}{\partial\omega}\right]_{\substack{\beta=\beta_l+\Delta\sigma\\ \omega=\omega_l\\ \mathbf{K}=\tilde{\mathbf{K}}}},S_{I,\omega_l}=\left[\frac{\partial I(\beta,\omega,\mathbf{K})}{\partial\omega}\right]_{\substack{\beta=\beta_l+\Delta\sigma\\ \omega=\omega_l\\ \mathbf{K}=\tilde{\mathbf{K}}}} \quad (4.84)$$

and

$$\mathbf{m}=\begin{bmatrix}m_k\\ m_{Rl}\\ m_{Il}\end{bmatrix},\ m_k=M(\sigma_k+\Delta\sigma,\tilde{\mathbf{K}}),\ m_{Rl}=R(\beta_l+\Delta\sigma,\omega_l,\tilde{\mathbf{K}}),\ m_{Il}=I(\beta_l+\Delta\sigma,\omega_l,\tilde{\mathbf{K}}) \quad (4.85)$$

where $k = 1..q_r$, q_r is the number of prescribed displacements of the real poles σ_k, $k = 1..q_c$, q_c is the number of prescribed displacements of the complex poles, i.e., the real parts of the poles, $q = q_r + q_c$, and j=1..r, r is the number of feedback loops from the state variables. Thus prescribing the sufficiently small displacements $\Delta\sigma$ from the current right-most poles σ_k and $\beta_l + j\omega_l$, the feedback increments ΔK_j and the displacements in imaginary parts of the complex poles $\Delta\omega_l$ can be computed from set of equations (4.81). If $r = q$ the set of equations can be solved in a classical method for solving the linear system of equations, see, e.g., Barrett, et al., (1994). If $r < q$ the underdetermined set of equations can be solved using the Moore-Penrose inversion, thus

$$\begin{bmatrix} \Delta\mathbf{K} \\ \Delta\boldsymbol{\omega} \end{bmatrix} = \mathbf{S}^{+}\mathbf{m} \tag{4.86}$$

Analogously to the algorithm for continuous pole placement based on the sensitivity functions (Algorithm 1.1), we can write the algorithm for the rightmost pole shifting using described gradient based feedback design.

Algorithm 4.1 Continuous pole placement using gradient based method

A. Start with $q = 1$
B. Compute the rightmost system poles using the mapping based rootfinder given by Algorithm 3.1
C. Assemble matrices (4.83) and (4.85) for system of equations (4.81)
D. Move q rightmost poles for which set of equations (4.81) has been assembled in direction to the left and find solution of (4.81).
E. Monitor the position of the rightmost poles of the system with the computed feedback settings. If necessary, increase the number of controlled poles q. Stop when stability is reached or when the available degrees of freedom of the controller do not allow $\sup(\mathrm{Re}(\lambda_i))$, $i = 1..\infty$ to be further reduced. In the other case, go to step B.

Unlike Algorithm 1.1, using the method described above, we obtain not only the changes of the coefficients ΔK_j but also the displacements of the imaginary parts of the poles $\Delta\omega_l$ being shifted. This fact may be useful in the task of accelerating the continuous pole placement procedure. Since the shifts of the poles that are controlled, i.e., the shifts of the real parts of the poles $\Delta\sigma$, are prescribed and $\Delta\omega_k$ result from (4.86), the approximate positions of the poles are known and only few steps of Newton's method is sufficient to obtain new positions of the poles that are controlled if the feedback gains change from $\tilde{\mathbf{K}}$ to $\tilde{\mathbf{K}} + \Delta\mathbf{K}$. However, note that also the positions of the uncontrolled poles are to be monitored. Therefore, in spite of the rather slower running of the procedure, it is advisable to compute the whole set of the rightmost poles in each step of Algorithm 4.1.

It is also important to note that from the numerical stability point of view, there should always be minimum distances between the neighbouring poles that are ashifted. If two of these poles are too close to each other, equation (4.86) becomes ill-conditioned. In the limit case that two poles being shifted are identical, matrix (4.83) becomes singular. Even though

the Moore-Penrose inversion of such a singular matrix exists, Algorithm 4.1 is likely to brake down. Unlike the algorithm presented by Michiels, et al, (2002), the gradient based feedback control offers the possibility to deal with the prescription of the multiple poles. However, from the numerical point of view, keeping the poles distinct is safer. The application of Algorithm 4.1 will be shown in the following examples.

Example 4.11

First, in order to compare the results achieved using the continuous pole placement method based on the sensitivity function introduced by Michiels, et al, (2002), with the method described above in Algorithm 4.1, consider the simple TDS of form (1.20) with only the input delay.

$$\mathbf{A} = \begin{bmatrix} -0.08 & -0.03 & 0.2 \\ 0.2 & -0.04 & -0.005 \\ -0.06 & 0.2 & -0.07 \end{bmatrix}, \quad \mathbf{B}(s) = \exp(-5s) \begin{bmatrix} -0.1 \\ -0.2 \\ 0.1 \end{bmatrix} \tag{4.87}$$

which was used as an introductory example in the paper of Michiels, et al, (2002). The system with matrices (4.87) have only three poles, $\lambda_1 = 0.108$ and $\lambda_{2,3} = -0.149 \pm 0.201$, i.e., the eigenvalues of the matrix **A**. However, if feedback (1.127) with nonzero coefficients is closed, the number of the poles of the closed loop system is infinite thanks to introducing the delay from matrix $\mathbf{B}(s)$ into the feedback system dynamics, i.e., to the matrix $\mathbf{A} - \mathbf{B}(s)\mathbf{K}$.

Since the pole λ_1 is positive, the system with matrices (4.87) is unstable. In order to stabilize the system, let us apply the stabilization procedure given by Algorithm 4.1, based on the continuous pole placement using gradient based feedback design. The result of pole shifting can be seen in Fig. 4.59. In the figure, we can see the changes of the real parts of the poles closest to the *s*-plane origin during the stabilization procedure. According to Algorithm 4.1, starting with $q = 1$ only the rightmost pole is controlled, i.e., continuously shifted to the left. Unlike in paper of Michiels, et al, (2002) we start the procedure with $\mathbf{K} = [0\,0\,0]$. It implies that the system has only three poles $\lambda_1 = 0.108$ and $\lambda_{2,3} = -0.149 \pm 0.202\mathrm{j}$ at the beginning of the stabilization procedure. However, thanks to the changes of the feedback coefficients which acquire nonzero values after applying the first step of the stabilization procedure, the exponential function corresponding to the input delay become a part of the system dynamics matrix. Thus, after the first step of the procedure, besides the slightly shifted λ_1 and $\lambda_{2,3}$ the system has infinitely many poles distributed in a pole chain typical for the retarded systems (i.e., the chain consisting of the poles departing from the *s*-plane origin to the infinities in both real and imaginary axes with the imaginary parts increasing much faster than the real parts of the poles). As the pole λ_1 is being shifted to the left, the chain of poles is moving towards the imaginary axis, see Fig. 4.59. As can be seen, the only real pole of the chain λ_4 gets close to the pole λ_1 around iteration 140 and from iteration 147 the pole λ_4 becomes also the controlled pole, i.e., shifted continuously to the left together with the other real pole λ_1. Note that a certain minimum distance of the controlled poles is kept in order to ensure numerically robust computation. At iteration 166, the complex conjugate pair $\lambda_{2,3}$ approaches the real axis and splits into two real poles λ_2 and λ_3 from which the former is controlled from iteration 191. The pole shifting procedure is stopped as the pole λ_3 gets close to the controlled group of poles because only three real parts of the poles may be controlled using three feedback loops from the state variables. In Fig. 4.60, we can see the evolution of

the feedback coefficients in the continuous pole placement procedure. From the implementation point of view, only the final set of the feedback gains $\mathbf{K} = [-0.470 \;\; -0.498 \;\; -0.600]$ is important, which provides rightmost system poles $\lambda_1 = -0.122$, $\lambda_4 = -0.139$, $\lambda_2 = -0.164$, $\lambda_3 = -0.177$. The distribution of the rightmost system poles can be seen in Fig. 4.61 in which the mentioned chain, which has got into this final position undergoing the journey from $-\infty$ (in the real axis), can be seen as well.

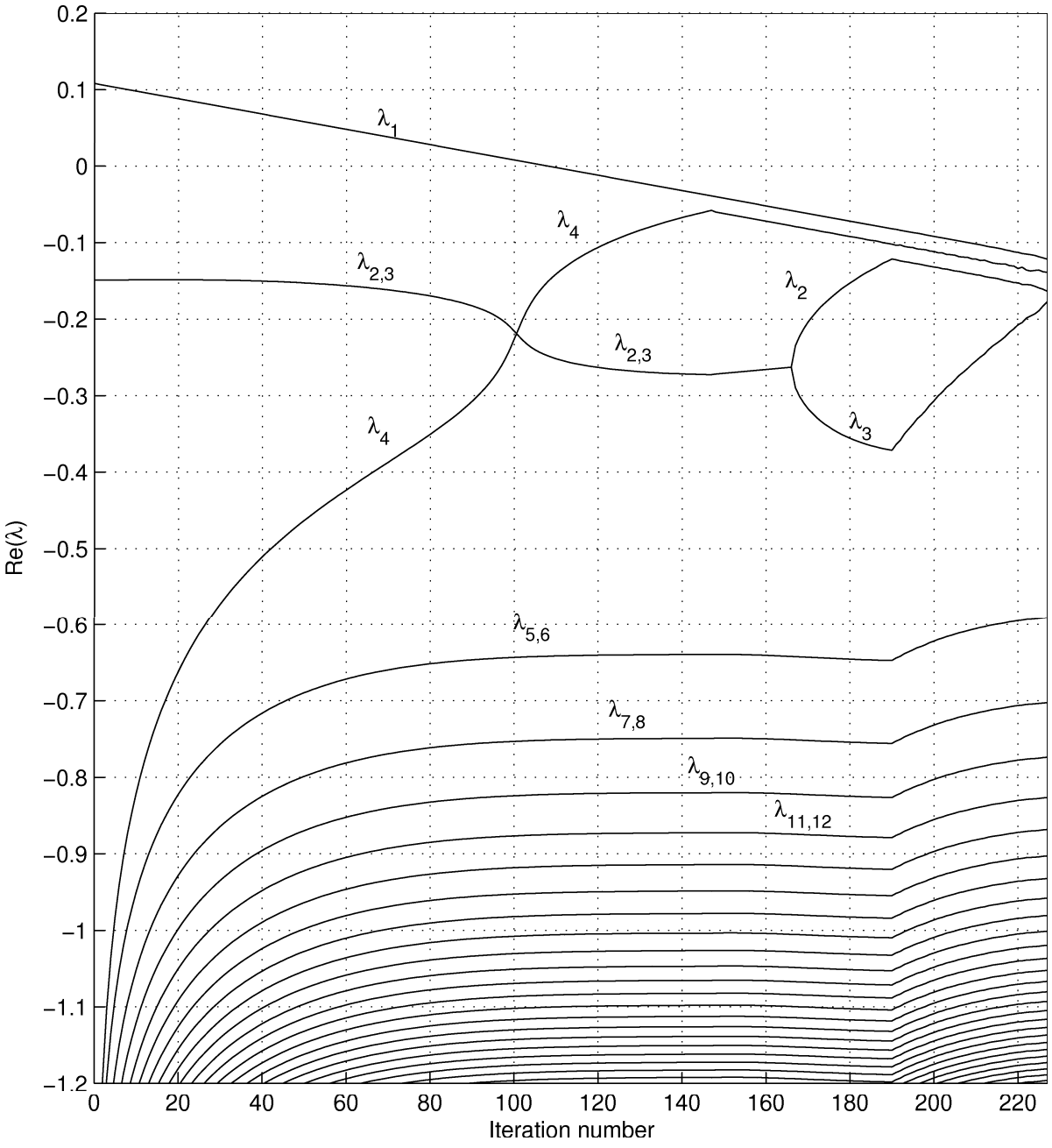

Fig. 4.59 Evolution of the real parts of the of the retarded system given by matrices (4.87) with the feedback from the state variables during the continuous pole placement procedure performed according to Algorithm 4.1.

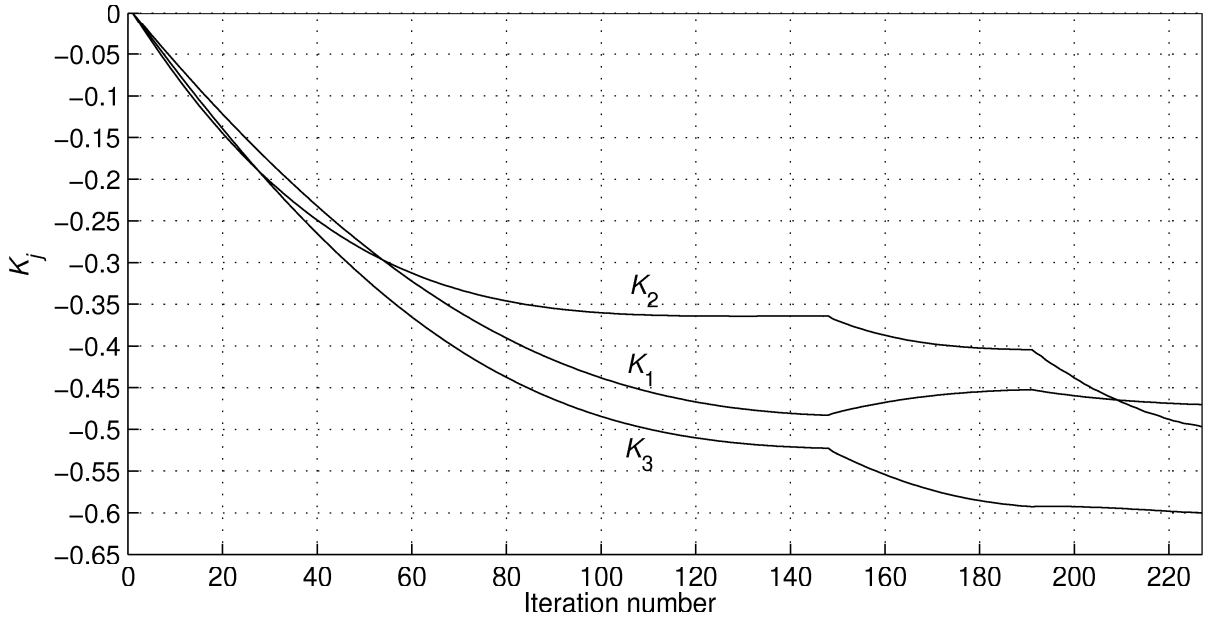

Fig. 4.60 Evolution of the feedback gain coefficients during the continuous pole placement procedure applied to system given by matrices (4.87) performed according to Algorithm 4.1.

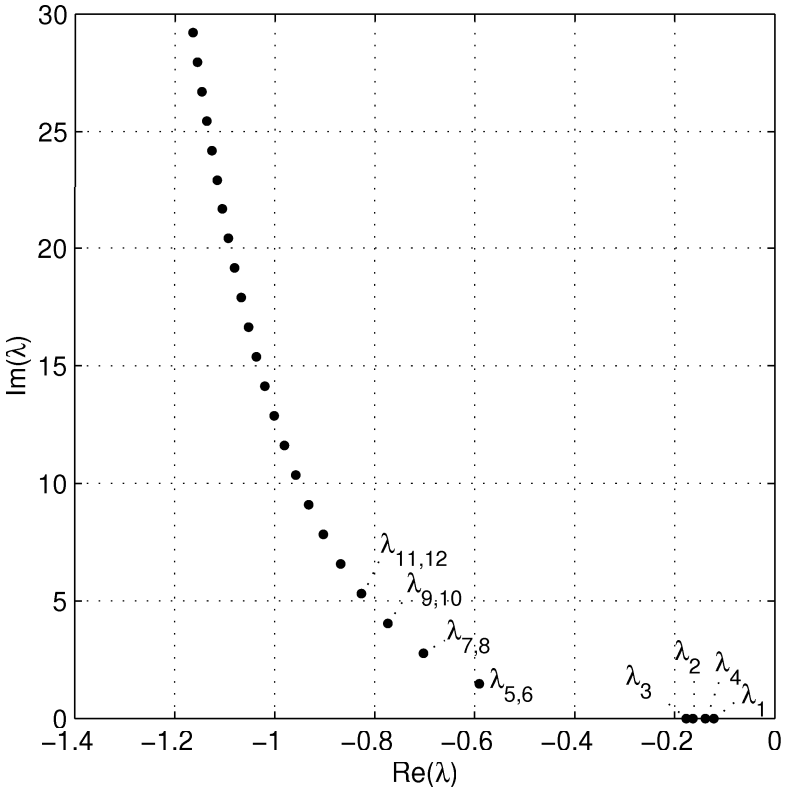

Fig. 4.61 Resultant distribution of the rightmost poles of the system given by matrices (4.87) with the feedback from the state variables after applying the continuous pole placement procedure.

Using the continuous pole placement, the stable system dynamics has been obtained. However, the system dynamics determined by this group of four poles will be rather slow. Further using the continuous pole placement procedure proposed in Algorithm 1.1 and

Algorithm 4.1 can not be used to shift the poles more to the left because no more than three poles can be controlled. The only possibility to shift the poles more to the left is to decrease the distances between the poles, which is not, however, safe from the numerical point of view. The way that serves for solving the task consists in applying the pole placement method introduced in section 4.4 based on the direct prescribing the poles. Using the small changes in the positions of the prescribed poles, the risk of obtaining an unacceptable result, i.e., the result in which the non-prescribed poles take over the role of the most significant ones, is low. For example, obviously, it seems to be natural to substitute three real poles λ_1, λ_4 and λ_2 by a triple pole with the position identical with the position of λ_3. Performing the task, we obtain only slightly changed feedback gain matrix $\mathbf{K} = [-0.470 \;\; -0.499 \;\; -0.601]$ and a new position of the pole $\lambda_3 = -0.184$. The new distribution of the dominant poles offers the possibility to shift the triple pole further to the left and to obtain the faster system dynamics. Prescribing $\lambda_{1,4,2} = -0.150$ results in $\mathbf{K} = [-0.471 \;\; -0.503 \;\; -0.602]$ and place the fourth pole into $\lambda_3 = -0.150$. Further shifting of the triple pole to the left is not possible because the pole λ_3 would take over the role of the dominant pole. The only possibility to make the closed loop dynamics faster than that given by $\lambda_{1,4,2,3} = -0.150$ is to prescribe the complex poles instead of the real ones. Such an acceleration of the dynamics resulting from prescribing the complex poles closer to the *s*-plane origin than the obtained supremum of $\mathrm{Re}(\lambda_i), i = 1..\infty$ will be shown in the application example in chapter 5.

Example 4.12

In order to show that Algorithm 4.1 can be applied also to a quite complicated system with many different delays (lumped and distributed), let us apply the algorithm to the system from Example 4.10 given by matrices (4.74). The evolution of the real parts of the poles closest to the *s*-plane origin during the continuous pole placement procedure, performed according to Algorithm 4.1, can be seen in Fig. 4.62 and the evolution of the feedback gains can be seen in Fig. 4.63. The starting distribution of the poles is that seen in Fig. 4.55 and Fig. 4.56. The region on which the continuous pole placement is performed is $\mathcal{D} = [-1, 0.1] \times [0, 2]$. As can be seen in Fig. 4.62, until iteration 11, only the poles being continuously shifted to the left are the poles $\lambda_{1,2}$. From this iteration, also the pole λ_3 is being controlled. Other three poles, i.e., λ_6 and $\lambda_{4,5}$, join the group of the controlled poles at iteration 51. The apparent discontinuities of the pole trajectories at this iteration are caused by the sudden shift of the poles λ_6 and $\lambda_{4,5}$ exceeding considerably the prescribed shifting step, which is done in order to keep the minimum distance between the poles being shifted. The pole shifting procedure stops at iteration 95 as the group of poles gets close to the pole $\lambda_{8,9}$ because no more than four real parts of the poles can be controlled. The resultant feedback of the continuous pole placement is $\mathbf{K} = [-0.0637 \;\; 0.3882 \;\; -0.3567 \;\; -0.1565]$, which correspond to the rightmost poles $\lambda_{1,2} = -0.0371 \pm 0.2386\mathrm{j}$, $\lambda_3 = -0.0769$, $\lambda_6 = -0.1172$, $\lambda_{4,5} = -0.1559 \pm 0.1631\mathrm{j}$, for the distribution of more poles see Fig. 4.64. In Fig. 4.58, the corresponding step response of the system can be seen. Comparing the step response with the responses obtained as the results of direct pole placement suggested in section 4.4, see Example 4.10, we can see that three of the responses in Fig. 4.58 are faster.

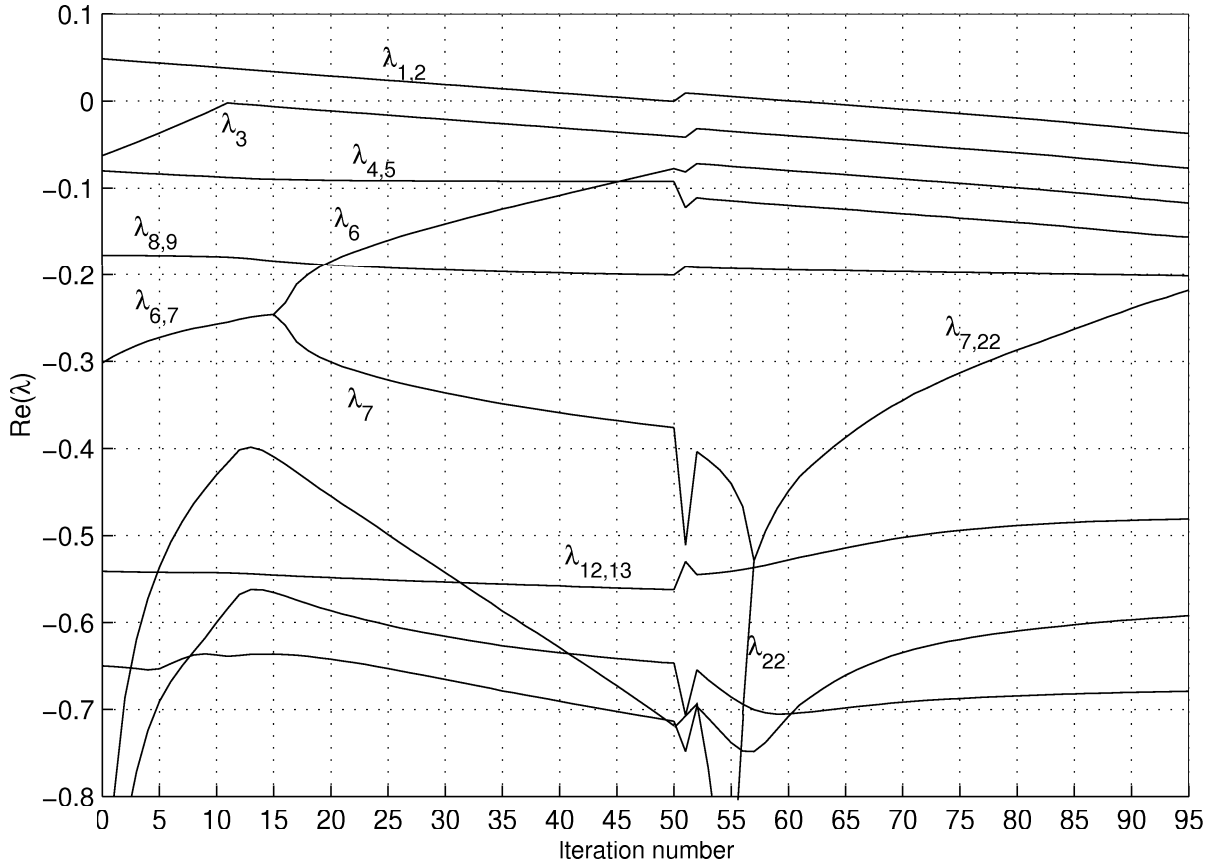

Fig. 4.62 Evolution of the real parts of the poles during the continuous pole placement procedure given by Algorithm 4.1 applied to system with matrices (4.74)

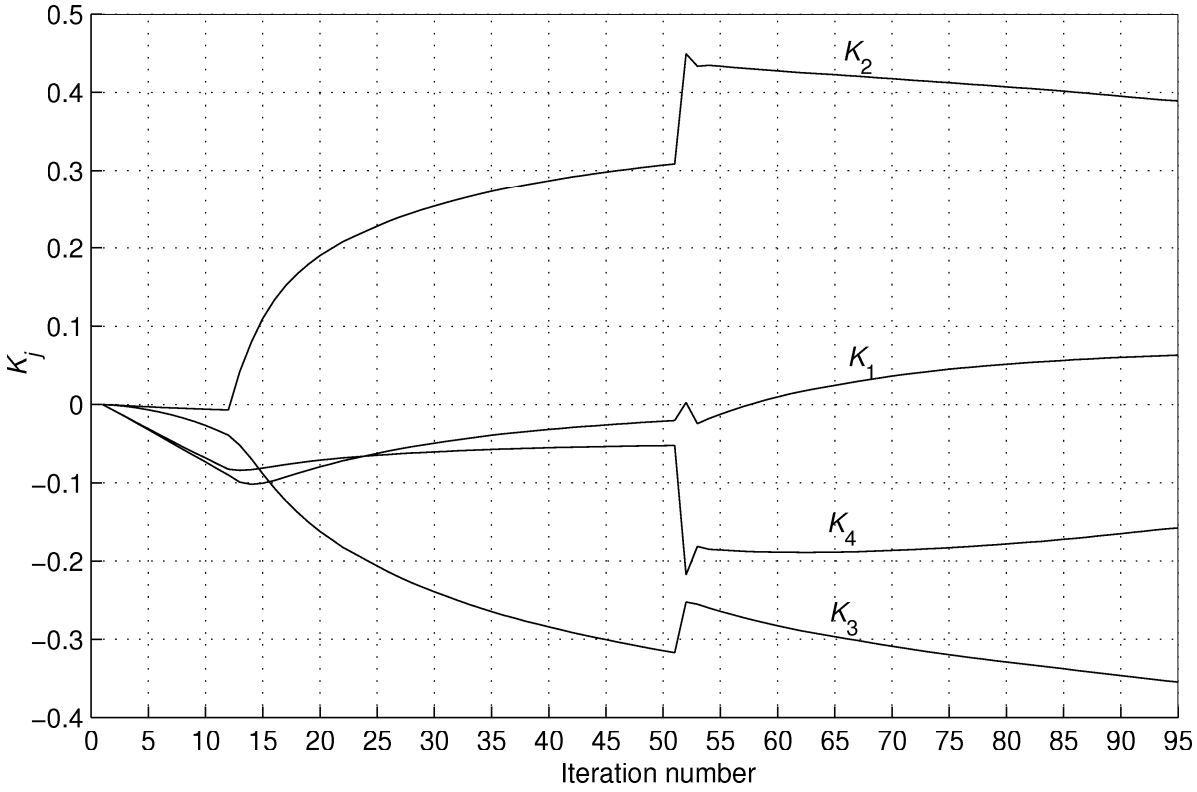

Fig. 4.63 Evolution of the feedback gains during the continuous pole placement procedure

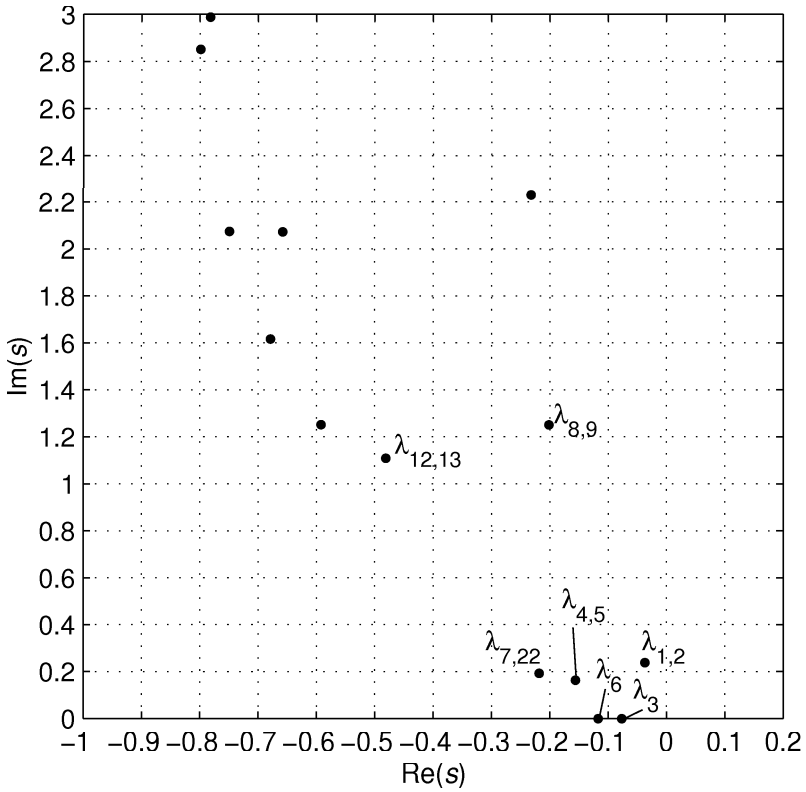

Fig. 4.64 Poles of the closed feedback system with the feedback coefficients resulted from the continuous pole placement

Obviously, the continuous pole placement does not provide the feedback system dynamics close to the optimal dynamics (if the optimal is considered well damped and fast dynamics). It is partly given by keeping the minimum distances between the neighbouring poles that are controlled in order to perform the numerical robustness of the computation, which stops the shifting before reaching the minimum of $\sup(\mathrm{Re}(\lambda_i)), i = 1..\infty$. Another drawback of the continuous pole placement method is given by the fact that only the real parts of the system are controlled while the imaginary parts are out of the control.

4.7 Pole placement applied to neutral systems

Although in sections 4.5 and 4.6 we have assumed that the system for which the pole placement is accomplished is a retarded system, the introduced pole placement algorithms may be used for neutral systems of the form

$$\frac{d}{dt}\left[\mathbf{x}(t)-\sum_{i=1}^{N}\mathbf{H}_i\mathbf{x}(t-\eta_i)\right]=\int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau)+\int_0^T d\mathbf{B}(\tau)\mathbf{u}(t-\tau) \tag{4.88}$$

as well. However, the potentials of using the coefficient feedback (1.127) to stabilize or positively change the neutral system dynamics are rather restricted. Stabilization of a neutral system by feedback (1.127) is possible if and only if difference equation (1.144) associated to (4.88) is strongly stable, see section 1.3.6 (for complete stabilization of a neutral system see Hale and Verduyn Lunel, (2002), Salamon, (1984), O'Connor and Tarn, (1983a, 1983b) or Pandolfi, (1976)), i.e., the roots of the exponential polynomial

$$M_e(s) = \det\left[\mathbf{I} - \sum_{i=1}^{N} \mathbf{H}_i \exp(-s\eta_i)\right] \tag{4.89}$$

(essential spectrum) are located to the left from the stability boundary and do not cross the boundary for any small changes in delays η_i. Provided that such a condition is satisfied neutral system (4.88) might be stabilizable by means of the coefficient feedback. The characteristic function of system (4.88) with the closed coefficient feedback is

$$M(s,\mathbf{K}) = \det\left[s\left[\mathbf{I} - \sum_{i=1}^{N} \mathbf{H}_i \exp(-s\eta_i)\right] - \mathbf{A}(s) + \mathbf{KB}(s)\right] \tag{4.90}$$

and the characteristic function of original system (4.88) is

$$M_0(s) = \det\left[s\left[\mathbf{I} - \sum_{i=1}^{N} \mathbf{H}_i \exp(-s\eta_i)\right] - \mathbf{A}(s)\right] \tag{4.91}$$

As in the case of retarded system, due to the linearity of (4.90) with respect to **K,** equation (4.61) holds also for (4.90) and the pole placement may be accomplished by solving the set of equations analogous to (4.72) in which the matrices **S** and **m** are given by (4.69) and (4.70), respectively, where $M(s,\mathbf{K})$ and $M_0(s)$ are given by (4.90) and (4.91), respectively. Also the continuous pole placement method given by Algorithm 4.1 may be used to attempt to stabilize neutral system (4.88), (provided that the essential system spectrum is strongly stable). The extension of the continuous pole placement method to the class of neutral systems is possible because of the use of rootfinding algorithm introduced in chapter 3.4, given by Algorithm 3.1, by means of which we can compute also the poles of neutral systems. Application of Algorithm 4.1 to a system with strongly stable essential spectrum will be shown in the following example.

Example 4.13

Consider the neutral system of the form

$$\frac{d}{dt}\left[\mathbf{x}(t) - \mathbf{H}_1\mathbf{x}(t-\eta_1) - \mathbf{H}_2\mathbf{x}(t-\eta_2)\right] = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t-\tau) \tag{4.92}$$

with the matrices

$$\mathbf{H}_1 = \begin{bmatrix} 0 & 0.2 & -0.4 \\ -0.5 & 0.3 & 0 \\ 0.2 & 0.7 & 0 \end{bmatrix},\ \mathbf{H}_2 = \begin{bmatrix} -0.3 & -0.1 & 0 \\ 0 & 0.2 & 0 \\ 0.1 & 0 & 0.4 \end{bmatrix},\ \mathbf{A} = \begin{bmatrix} -4.8 & 4.7 & 3 \\ 0.1 & 1.4 & -0.4 \\ 0.7 & 3.1 & -1.5 \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} 0.3 \\ 0.7 \\ 0.1 \end{bmatrix}$$

and the delays $\eta_1 = 0.7$, $\eta_2 = 1.7$ and $\tau = 0.5$. Using the mapping based quasipolynomial rootfinder, a part of the spectrum of the poles of system (4.92), i.e., the roots of the system characteristic function

$$M(s) = \det\left[s(\mathbf{I} - \mathbf{H}_1\exp(-s\eta_1) - \mathbf{H}_2\exp(-s\eta_2)) - \mathbf{A}\right] \tag{4.93}$$

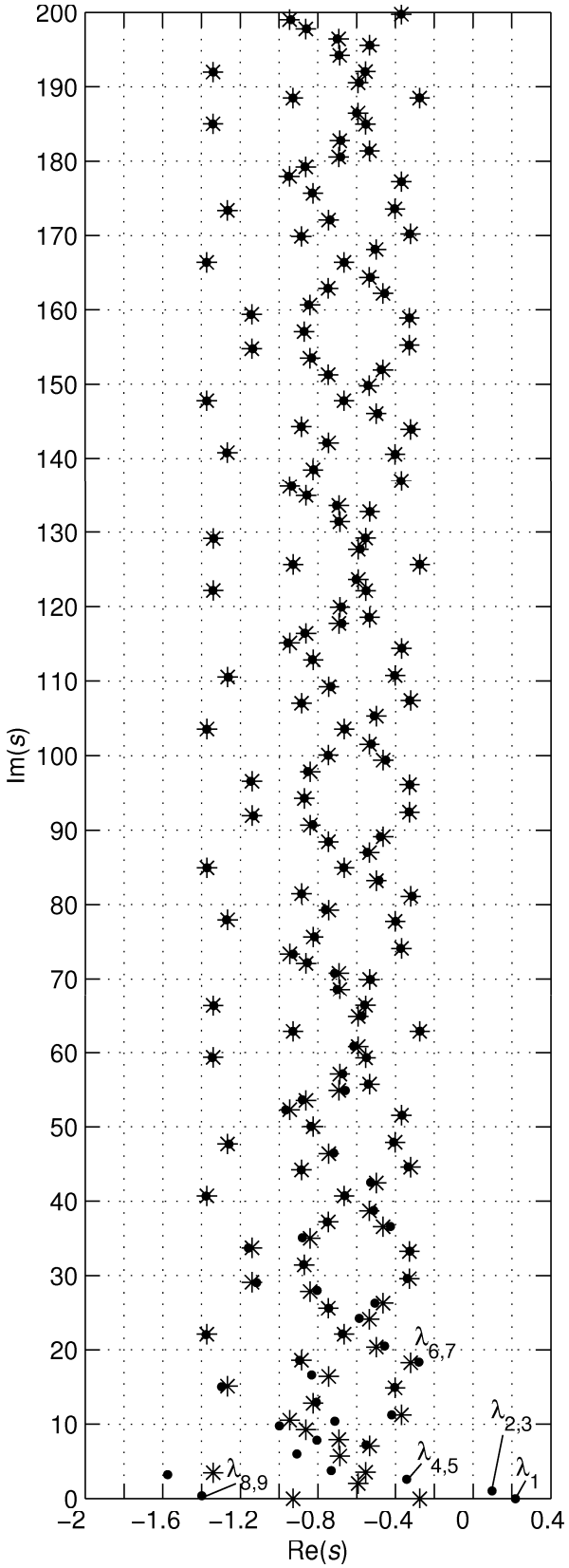

Fig. 4.65 Spectra of neutral system (4.92), asterisks - essential spectrum, roots of equation (4.94), black circles - spectrum of poles of system (4.92)

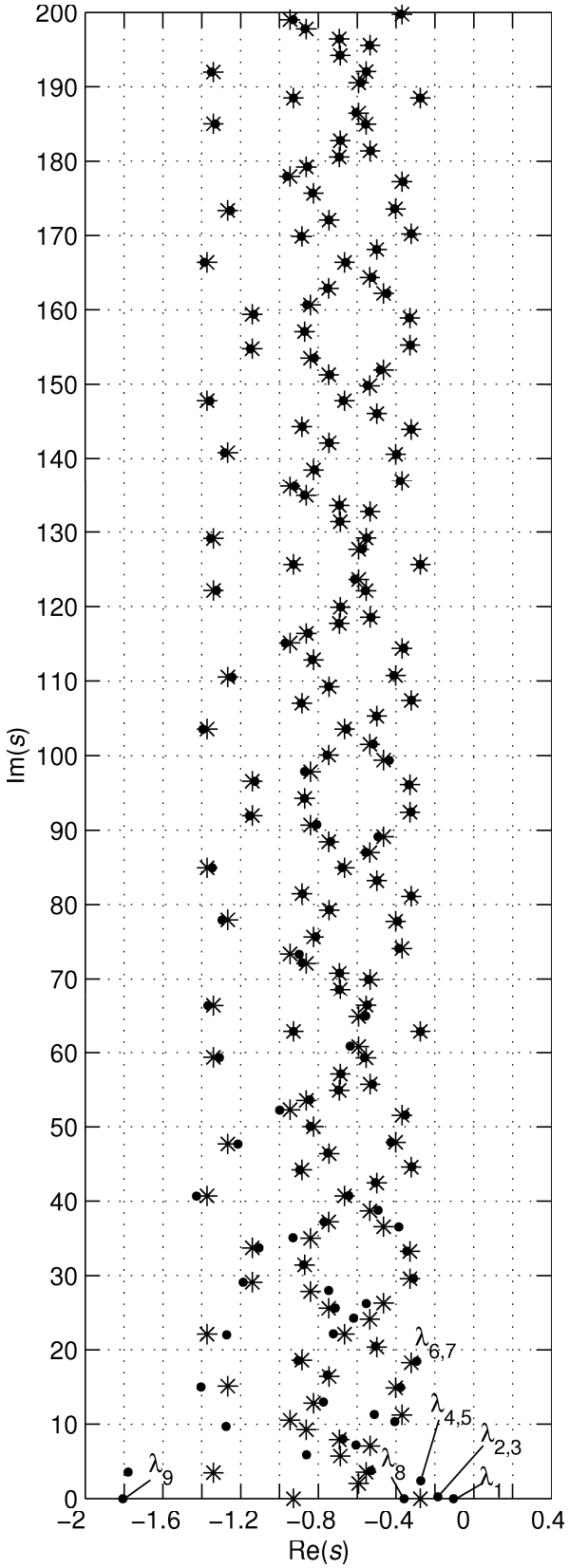

Fig. 4.66 Spectra of neutral system (4.92) with feedback (1.127), asterisks - essential spectrum, roots of equation (4.94), black circles - spectrum of poles of system (4.92) with feedback (1.127)

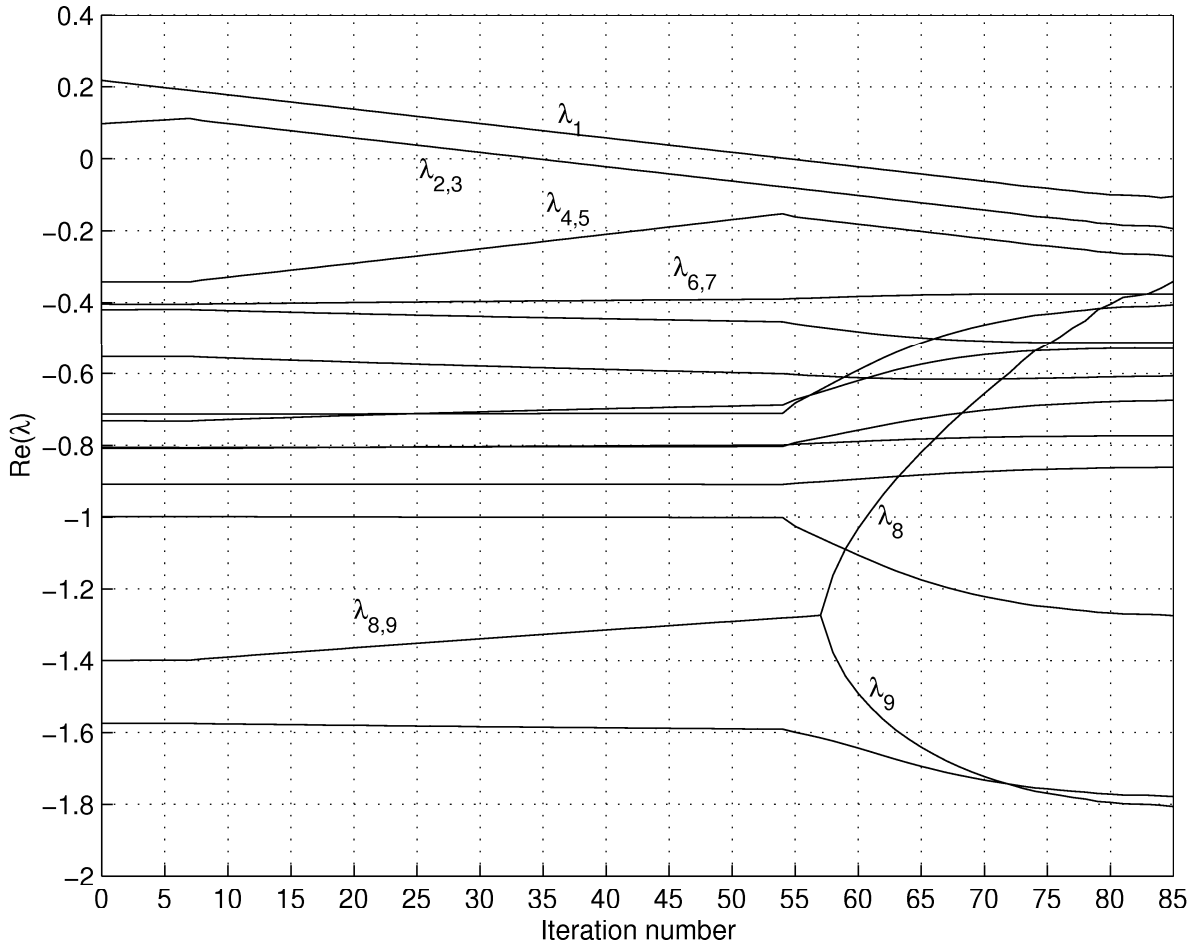

Fig. 4.67 Evolution of the real parts of the poles corresponding to neutral system (4.92) with the feedback from the state variables during the continuous pole placement performed according to Algorithm 4.1.

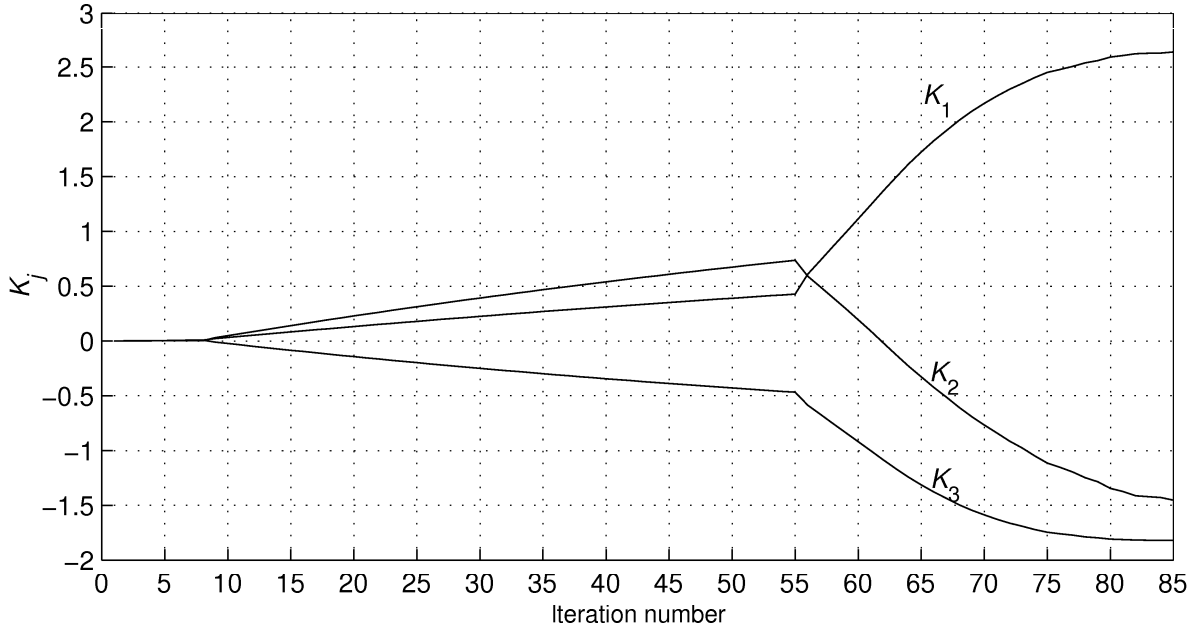

Fig. 4.68 Evolution of the feedback gain coefficients during the continuous pole placement applied to neutral system (4.92) performed according to Algorithm 4.1.

The part of the poles nearest to the real axis is seen in Fig. 4.65, marked with black circles. As can be seen, the system has three unstable poles $\lambda_1 = 0.218$ and $\lambda_{2,3} = 0.0976 \pm 1.0396\text{j}$. Before applying the continuous pole placement algorithm, the stability of the essential spectrum of the system, given as the solutions of the following equation

$$\det\left[\mathbf{I} - \mathbf{H}_1 \exp(-\eta_1 s) - \mathbf{H}_2 \exp(-\eta_2 s)\right] = 0 \tag{4.94}$$

has to be checked. The part of the essential spectrum of the neutral system can also be seen in Fig. 4.65, marked by the asterisks. As can be seen, the spectrum is stable on the whole region which is shown in Fig 4.65. Due to the periodicity of the essential spectrum with the period 62.83j, no root of the essential spectrum crosses the stability boundary neither in higher range of ω. In Fig. 4.65 we can observe the typical feature of the neutral systems, i.e., the spectrum of neutral system converges to the essential spectrum as the imaginary parts of the poles increase. Note that complete stability analysis of the neutral system should involve a robust stability test of the system essential spectrum (which is omitted here – the system with strongly stable essential spectrum has been chosen) to decide whether or not the spectrum is sensitive to the changes in the delays, see Hale and Verduyn Lunel, (2002).

Instead of deeper investigation whether or not the system is stabilizable using (1.127), we directly apply the continuous pole placement given by Algorithm 4.1. As can be seen in Fig. 4.66, the method gives positive result, i.e., the resultant feedback matrix $\mathbf{K} = [2.653 \;\; -1.493 \;\; -1.823]$ stabilize the system by shifting the unstable poles to the positions $\lambda_1 = -0.1048$ and $\lambda_{2,3} = -0.1946 \pm 0.2527\text{j}$. The evolution of the continuous pole placement can be seen in Fig. 4.67 (the real parts of the poles closest to the real axis) and in Fig. 4.68 (the coefficient feedback gains). Only the pole λ_1 is controlled until iteration 7 from which also $\lambda_{2,3}$ are controlled. At iteration 55 the couple $\lambda_{4,5}$ joins the group of the controlled poles. The algorithm stops at iteration 85, because the pole λ_8 gets risky close to the controlled group of poles. The spectrum of the poles of the feedback neutral system with K given by the result of continuous pole placement can be seen in Fig. 4.66. Comparing the spectrum with the spectrum of the system without the feedback, we can see that only the positions of the poles that are close to the real axis of the complex plane changed considerably, while the positions of the poles more distant from the real axis changed only slightly (the changes decay with increasing imaginary parts of the poles). This phenomenon is given by the fact that both the systems have common essential spectrum to which the poles of both the systems converge as their imaginary parts increase.

Because of the mutual distances between the neighbouring controlled poles kept to ensure more robust numerical computation during the stabilization procedure, the pole placement which results from the continuous shifting of the poles does not guarantee the fastest possible system dynamics. Using the direct pole placement method given by solving (4.72) we can further change the positions of the most significant poles. However, it should be noted, that the changes of the positions of the dominant poles should not be too large and each of the attempts has to be followed by checking the positions of the unprescribed poles. The result of several prescription the poles can be seen in Tab. 4.10, in which, besides the prescribed poles, we can see also the resultant coefficients of the feedback gain matrix **K** and the compensation *c* given by (4.75) which compensates the change of the system static gain caused by introducing the feedback.

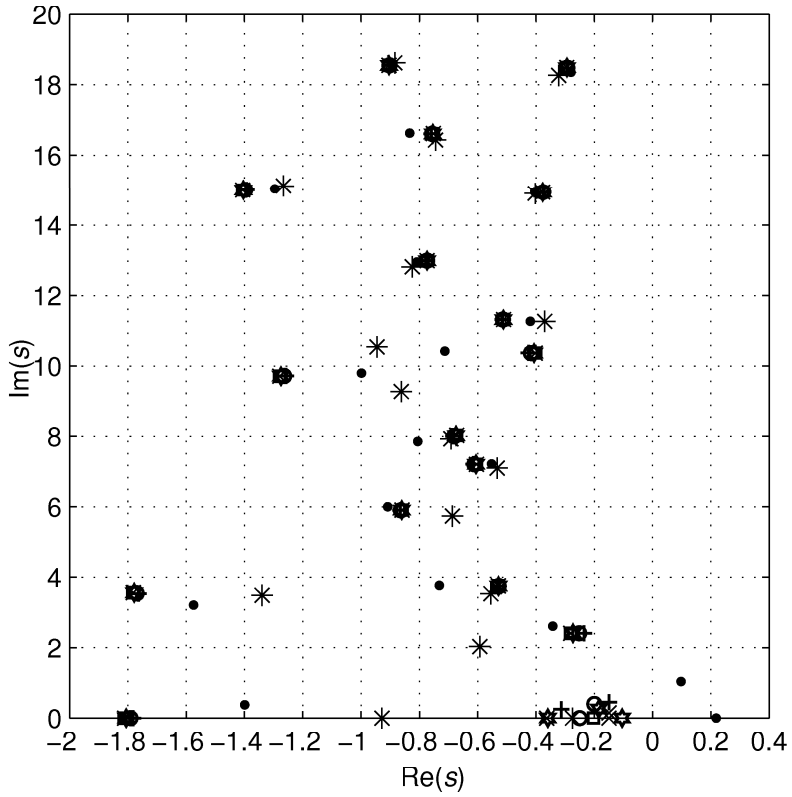

Fig. 4.69 The most significant poles of the feedback system with the feedback gain settings from Tab. 4.10, asterisks - essential spectrum, black circles - original system poles, six-pointed stars - result of continuous pole placement, squares - $k_S = 2$, × - $k_S = 3$, empty circles - $k_S = 4$, + - $k_S = 5$.

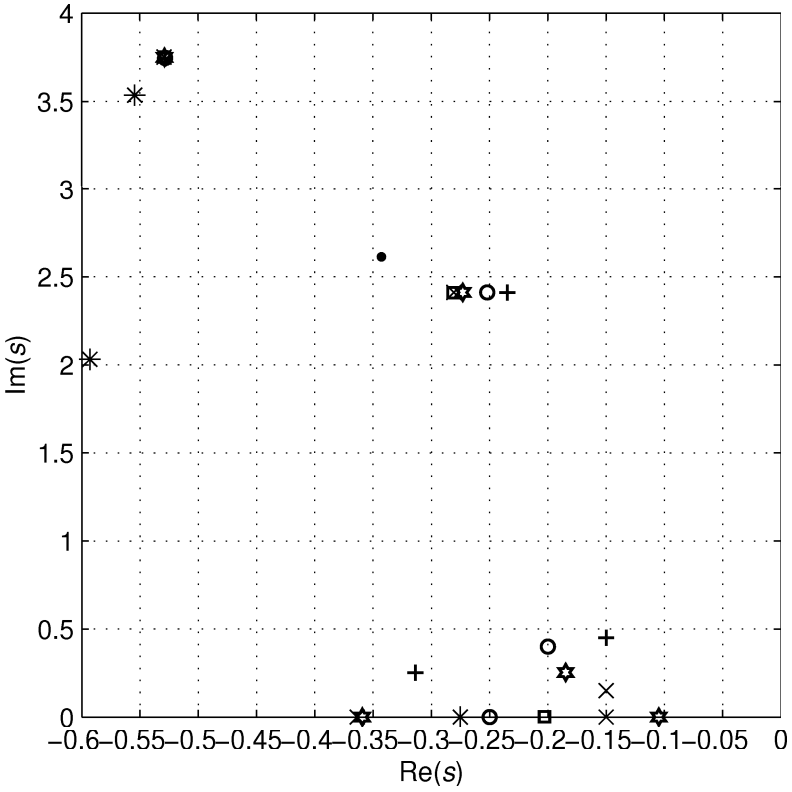

Fig. 4.70 Detailed view of Fig. 4.69

Tab. 4.10 Sets of prescribed system poles and the resultant feedback gains

k_s	Prescribed roots	$\mathbf{K}_{k_s}$	c_{k_s}
1	c.p.p.	[2.653 -1.493 -1.823]	0.01332
2	$3 \times$ -0.203	[2.678 -1.577 -1.822]	0.00625
3	-0.15, -0.15+0.15j	[2.670 -1.567 -1.817]	0.00906
4	-0.2+0.4j, -0.25	[2.601 -1.313 -1.824]	0.04261
5	-0.15+0.45j	[2.573 -1.179 -1.826]	0.12846

In Fig. 4.69, we can see the spectra of the group of poles closest to the real axis of the *s*-plane corresponding to the result of continuous pole placement and the successive direct prescribing the poles according to Tab. 4.10. As can be seen, the positions of the poles that are farther from the complex plane origin change only slightly with respect to the prescribed poles. The position of the prescribed poles and of the poles that are close to them can be seen in Fig. 4.70.

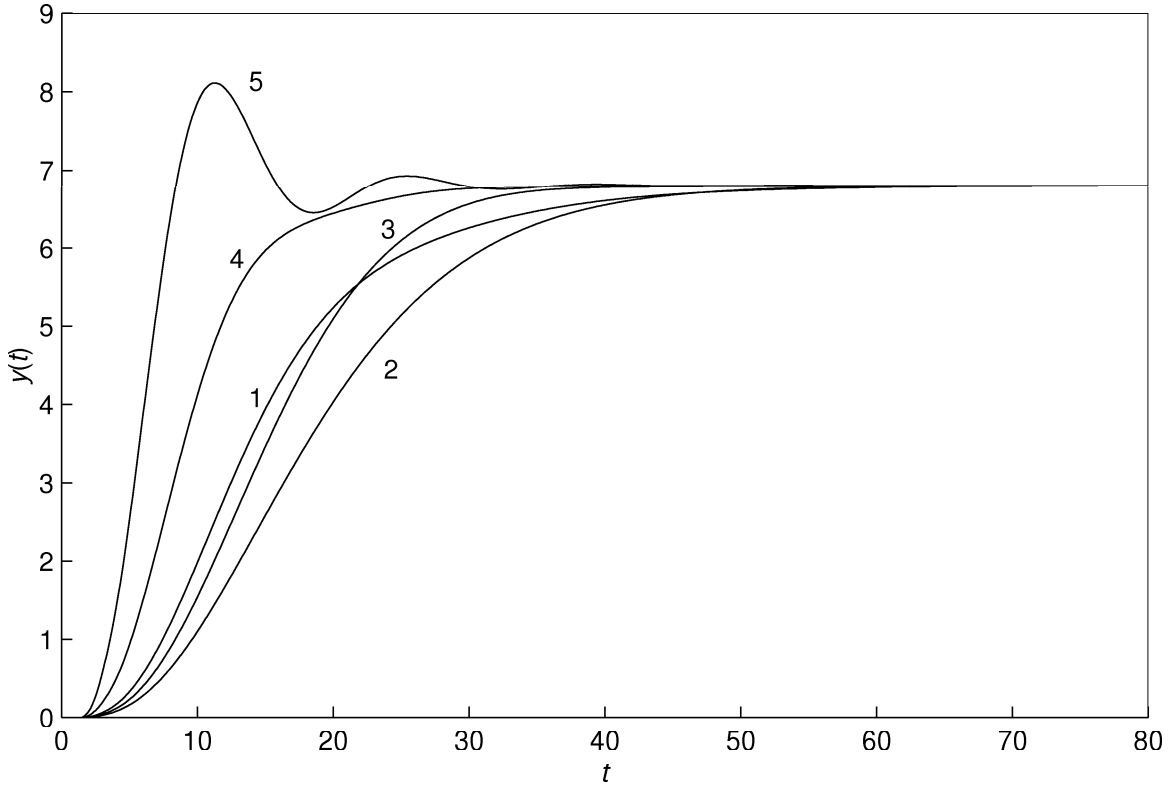

Fig. 4.71 Step responses of neutral system (4.92) with feedback (1.127) with the settings from Tab. 4.10

In Fig. 4.71 we can see the step responses of the neutral system with the feedback setting resulted from the continuous pole placement and from the direct pole placement according to Tab. 4.10. Even though the continuous pole placement safely stabilizes the system, further correction of the pole positions using the direct prescribing the poles is reasonable and provides the possibility to achieve more favourable system dynamics.

4.8 Strategy of Pole placement method applied to TDS

As has been shown in Examples 4.10, 4.11, 4.12 and 4.13, the direct prescription of the poles using the coefficient feedback from the state variables of the system explained in section 4.4 is a quite powerful control design tool. Although the poles cannot be placed or shifted arbitrarily, as it is in case of classical systems, the strategy described in section 4.5 is likely to converge to a satisfactory result. Even though the pole placement approach is rather heuristic, the pole placement is controlled by the requirements for the system as it is also in case of the finite order systems. On the other hand, from the application point of view, the dynamics can be prescribed arbitrarily neither in case of using the pole placement method for classical finite order system (1.1). In case that the real system with distributed parameters or transportation phenomenon involved in the dynamics is described using a finite order model, the description of the dynamics is more or less rough, covering only rather narrow frequency range of the system dynamics. Prescribing too fast dynamics to such a feedback system may also result in instability or undesired features of the dynamics caused by the modes that are not involved in the finite dimensional model.

As a very promising tool in designing the coefficient feedback from the state variables seems to be the combination of the continuous and direct pole placement methods. The continuous pole placement applied first gets the poles close to the minimal possible $\sup(\mathrm{Re}(\lambda_i)), i = 1..\infty$, i.e., place the pole as much to the left as possible. Then the direct pole placement is used taking into account the distribution of the poles resulted from the continuous pole placement as the starting distribution of the poles. Using the direct pole placement the most significant poles are to be gradually slightly shifted to improve the dynamics. Both the algorithms for pole placement are available on the CD enclosed as Matlab functions, see Appendix 1.

5. REAL PLANT APPLICATION EXAMPLE

5.1 Model of laboratory heat transfer system

In this chapter, I am going to investigate the features of the dynamics of the laboratory heating system which has been developed at the Institute of Instrumentation and Control Engineering in order to test the anisochronic approaches in control and modelling. The other important topics of this chapter are methods for computing the approximations of the poles of TDS and the application of the pole placement methods using the coefficient feedback in control of the laboratory plant.

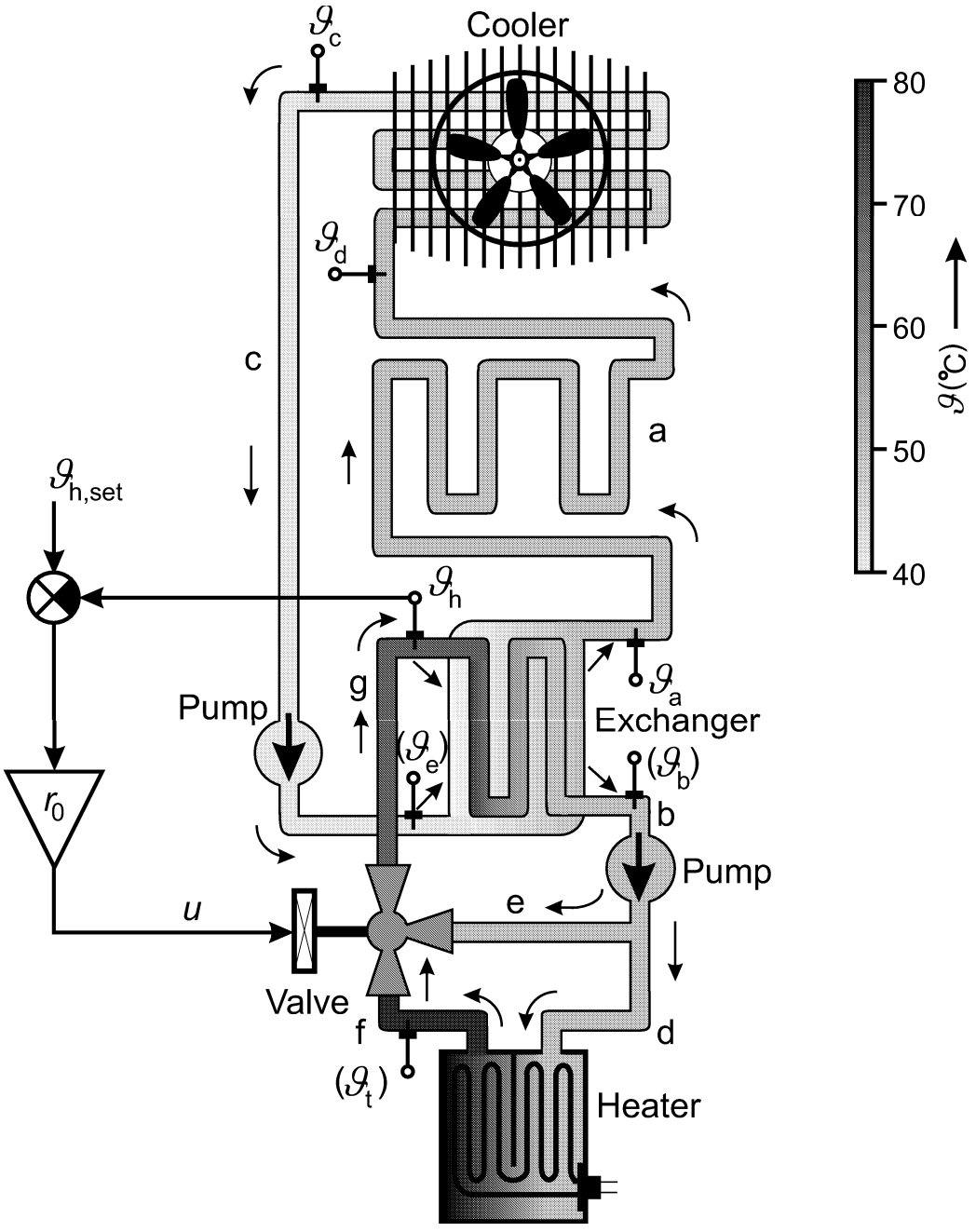

Fig. 5.1 Scheme of the laboratory heating system, the approximate temperature distribution over the plant at the chosen operational point is visualized by shade of grey

The scheme of the laboratory system is sketched in Fig. 5.1. The heating system consists of two heating circuits with the circulation of the heat medium (water) accomplished by two pumps (one in each circuit). The heat source of the system is an electric heater, located in the primary circuit. The heat exchange between the two circuits, which is controlled by the mixing valve, takes place in the multi-plate heating exchanger. The last important component of the system is an air-water cooler located in the secondary circuit. As can be seen in Fig. 5.1, the components of the system are connected by the piping lines that provide the most important delays in the system. For the technical data of the system and its components, see Appendix 2.

Let us consider four temperatures measured on the system, i.e., ϑ_h - mixed up hot water at the input of the exchanger (delayed outlet water temperature of the mixing valve), ϑ_a - water/water exchanger outlet, ϑ_d, ϑ_c - air/water cooler inlet and outlet, respectively. The control input of the system is the signal u adjusting the position of the valve cone by means of the servomechanism, i.e., the mixing ration of the water flow rates coming from the heater and the exchanger, and thereby the temperature ϑ_h. The other system inputs are the propeller velocity in the cooler and the heating performance of the heater.

Obviously, the transportation phenomenon encountered mainly in the piping lines plays the significant role in the laboratory heating system. However, time delay relations will not only be used to model the transportation phenomenon. In order to obtain as low order model of the system as possible, the main system units will be described by linear first order anisochronic model (4.6), which can be used to fit the higher order dynamics of the units. Before starting the procedure of assembling the model of the laboratory system from the local models of the system units, let us note that the model will be valid only for a certain operational point for which the parameters of the model will be identified. It is given by the fact that the laboratory heating system is considerably nonlinear and its linear approximation can not cover its static and dynamic features for the whole range of the settings of the inputs. Therefore, the variables of the model will be in the form of their increments from their values corresponding to the operational point.

Since the efficiency of the multi-plate heat exchanger is very high, the heat loss during heat exchange can be neglected and the logarithmic temperature gradient may be approximated by the formula

$$\Delta\overline{\vartheta} = \vartheta_h - \frac{1+q}{2}\vartheta_a - \frac{1-q}{2}\vartheta_e \tag{5.1}$$

where ϑ_e is the other inlet temperature of the exchanger and $q = m_2/m_1$, where m_1, m_2 are the flow rates in the parts of the exchangers, i.e., the flow rates in the heating circuits. Balancing the heat exchange, transportation phenomenon (i.e., $\vartheta_e(t) = \vartheta_c(t-\tau_e)$, if the heat loos on the piping line is neglected) and heat accumulation, the model of the system unit may be considered in the following form

$$T_a \frac{d\Delta\vartheta_a(t)}{dt} = K_a\left[\Delta\vartheta_h(t) - \frac{1+q}{2}\Delta\vartheta_a(t) - \frac{1-q}{2}\Delta\vartheta_c(t-\tau_e)\right] - \left[\Delta\vartheta_a(t) - \Delta\vartheta_c(t-\tau_e)\right] \tag{5.2}$$

where T_a is the accumulation constant, K_a is the heat transfer coefficient, and τ_e is the time delay.

As the second unit of the system let the longest pipeline be considered connecting the heat exchanger with the cooler. The simplest model of the unit is pure time shifting of the pipeline-input temperature, i.e., $\vartheta_d(t) = \vartheta_a(t-\tau_h)$. Physically more relevant model of the

long pipeline is the model based on a distributed delay, i.e., $\vartheta_{\mathrm{d}}(t) = \int_0^{\tau_{\mathrm{d}}} dr(\tau)\vartheta_{\mathrm{a}}(t-\tau)$, where $r(\tau)$ is the distribution of the delay. However, since the aim of the model design is to obtain as simple model as possible and the distributed delay is not a convenient model unit to deal with, the following first order model will be used to approximate the dynamics of the distributed delay in the piping line

$$T_{\mathrm{d}} \frac{d\Delta\vartheta_{\mathrm{d}}(t)}{dt} = -\Delta\vartheta_{\mathrm{d}}(t) + K_{\mathrm{d}}\Delta\vartheta_{\mathrm{a}}(t-\tau_{\mathrm{d}}) \tag{5.3}$$

where T_{d} is the time constant, τ_{d} is the delay and K_{d} is the static gain coefficient. Using (5.3) to model the longest pipeline unit is convenient because it provides the temperature $\vartheta_{\mathrm{d}}(t)$ as the state variable of the system.

The last unit of the secondary heating circuit is the cooler. Let the system input controlling the propeller velocity is considered constant, the unit may be described by the mentioned first order anisochronic model (4.6)

$$T_{\mathrm{c}} \frac{d\Delta\vartheta_{\mathrm{c}}(t)}{dt} = -\Delta\vartheta_{\mathrm{c}}(t-\eta_{\mathrm{c}}) + K_{\mathrm{c}}\Delta\vartheta_{\mathrm{d}}(t-\tau_{\mathrm{c}}) \tag{5.4}$$

where T_{c} is the time constant, τ_{c} and η_{c} are the delays and K_{c} is the static gain coefficient.

A markedly nonlinear system unit is the mixing valve. Even though the characteristic of the valve is linear, its outlet temperature of the water is given by

$$\vartheta_{\mathrm{v}} = v(u)\vartheta_{\mathrm{t}} + (1-v(u))\vartheta_{\mathrm{b}} \tag{5.5}$$

where ϑ_{t} is the outlet water temperature of the heater, ϑ_{b} is the outlet water temperature of the exchanger and v is the mixing ratio given by the actual value of the system input u. Instead of building the model using linearized relation of (5.5), which would be supposedly valid only for relatively small Δu, let the proportional feedback from $\vartheta_{\mathrm{h}}(t)$ be introduced (with $r_0 = 0.25\mathrm{V}/^{\circ}\mathrm{C}$, found in the experimental way using Ziegler-Nichols rules), see Fig. 5.1, and the arising control loop be described using the first order anisochronic model

$$T_{\mathrm{h}} \frac{d\Delta\vartheta_{\mathrm{h}}(t)}{dt} = -\Delta\vartheta_{\mathrm{h}}(t-\eta_h) + K_{\mathrm{b}}\Delta\vartheta_{\mathrm{a}}(t-\tau_{\mathrm{b}}) + K_{\mathrm{u}}\Delta\vartheta_{\mathrm{h,set}}(t-\tau_{\mathrm{u}}) \tag{5.6}$$

where $\vartheta_{\mathrm{h,set}}(t)$ is the set-point value of the temperature $\vartheta_{\mathrm{h}}(t)$, T_{h} is the time constant, τ_{b}, τ_{u} and η_{h} are the delays and K_{b} and K_{u} are the static gains. In fact, equation (5.6) is the approximation of the large part of the primary heating circuit. Omitting the model of the heater is possible because the heator is controlled by a thermostat, which keeps its outlet water temperature almost constant independently of the actual values of the system inputs. The model (5.6) is valid if the heat exchange performance of the exchanger is close to 100%, which is satisfied in case of the used multi-plate exchanger. Thus, to model the back-flow of the water in the primary heating circuit, the temperature $\vartheta_{\mathrm{a}}(t)$ is used in the model instead of $\vartheta_{\mathrm{b}}(t)$ (where $\vartheta_{\mathrm{b}}(t)$ is the other outlet temperature of the exchanger) because $\vartheta_{\mathrm{a}}(t) \approx \vartheta_{\mathrm{b}}(t)$. In fact, validity of (5.6) is also limited only for small Δu, because the length of the delay τ_{u} depends on the actual flow rates in the inlet branches of the mixing valve. However, the contribution of the back-flow effect to the primary circuit dynamics is relatively small. This

fact enlarges the range of Δu for which (5.6) satisfactory describes the primary heating circuit.

To sum up, the model of the system consists of four functional differential equations (5.2), (5.3), (5.4) and (5.6). Performing the Laplace transform of the equations (considering the zero initial conditions) the model of the laboratory heating system may be written in the form

$$s\mathbf{x}(s) = \mathbf{A}(s)\mathbf{x}(s) + \mathbf{B}(s)\Delta\vartheta_{\mathrm{h,set}}(s) \tag{5.7}$$

where $\mathbf{x}(s) = \left[\Delta\vartheta_{\mathrm{h}}(s) \;\; \Delta\vartheta_{\mathrm{a}}(s) \;\; \Delta\vartheta_{\mathrm{d}}(s) \;\; \Delta\vartheta_{\mathrm{c}}(s)\right]^T$ is the vector of the state variables, and

$$\mathbf{A}(s) = \begin{bmatrix} \dfrac{-\exp(-\eta_{\mathrm{h}}s)}{T_{\mathrm{h}}} & \dfrac{K_{\mathrm{b}}\exp(-\tau_{\mathrm{b}}s)}{T_{\mathrm{h}}} & 0 & 0 \\ \dfrac{K_{\mathrm{a}}}{T_{\mathrm{a}}} & \dfrac{-(1+0.5K_{\mathrm{a}}(1+q))}{T_{\mathrm{a}}} & 0 & \dfrac{(1-0.5K_{\mathrm{a}}(1-q))\exp(-\tau_{\mathrm{e}}s)}{T_{\mathrm{a}}} \\ 0 & \dfrac{K_{\mathrm{d}}\exp(-\tau_{\mathrm{d}}s)}{T_{\mathrm{d}}} & \dfrac{-1}{T_{\mathrm{d}}} & 0 \\ 0 & 0 & \dfrac{K_{\mathrm{c}}\exp(-\tau_{\mathrm{c}}s)}{T_{\mathrm{c}}} & \dfrac{-\exp(-\eta_{\mathrm{c}}s)}{T_{\mathrm{c}}} \end{bmatrix}$$

$$\mathbf{B}(s) = \begin{bmatrix} \dfrac{K_{\mathrm{u}}\exp(-\tau_{\mathrm{u}}s)}{T_{\mathrm{h}}} & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$$

Since all the system state variables are measured, they all might be considered as the system outputs, i.e., $\mathbf{C} = \mathbf{I}$. However, since a control of the variable $\vartheta_{\mathrm{c}}(t)$ will be worked out in section 5.5, let $\vartheta_{\mathrm{c}}(t)$ be considered as the only output of the system, i.e., $\mathbf{C} = [0001]$.

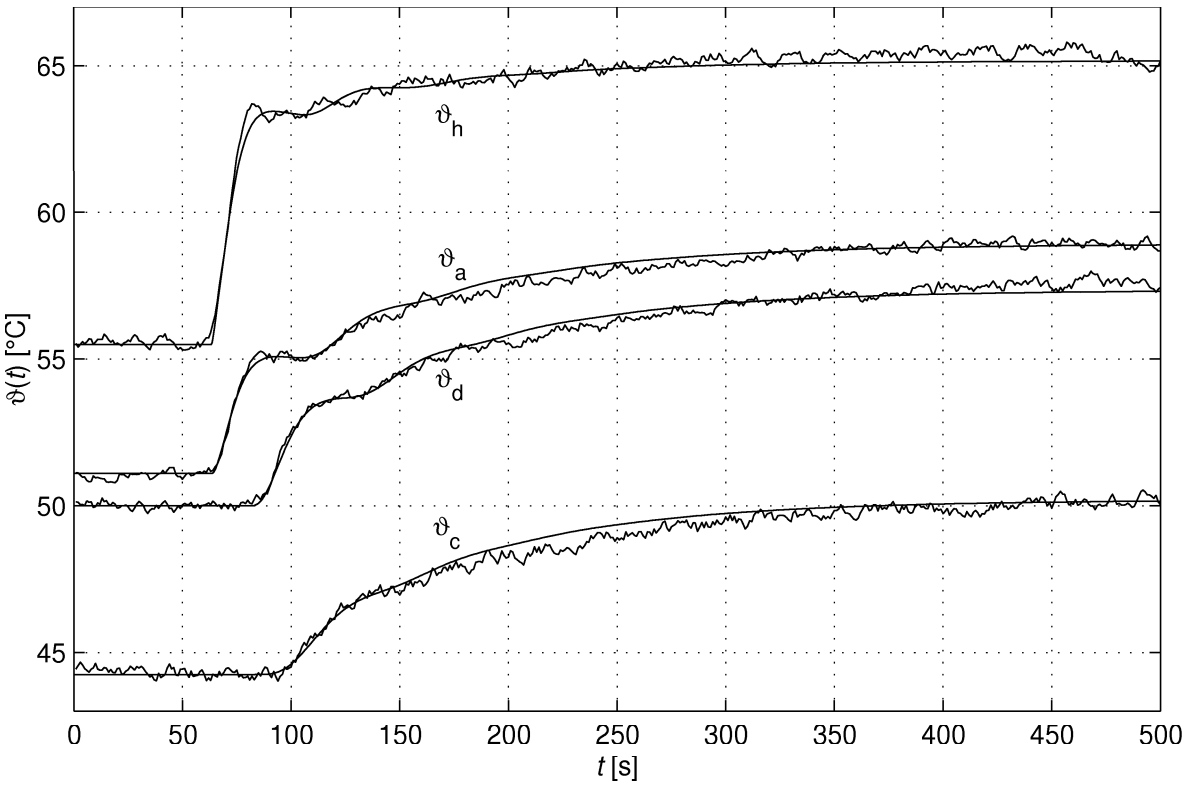

Fig. 5.2 Step response comparison of the laboratory system (influenced by noise) and its anisochronic model (5.7) (smooth), $\Delta\vartheta_{\mathrm{h,set}}(50\mathrm{s}) = 20^{\circ}\mathrm{C}$

The parameters that assure a quite good approximation of the system step response performed in the vicinity of the operational point for which the model is assumed to approximate the system dynamics are the following:

$T_h = 14\text{s}$, $K_h = 0.24$, $K_u = 0.39$, $\eta_h = 6.5\text{s}$, $\tau_h = 40\text{s}$, $\tau_u = 13.2\text{s}$

$T_a = 3\text{s}$, $K_a = 1$, $\tau_e = 13\text{s}$, $q = 1$ ($m_1 = m_2 = 0.08$ m^3/hour)

$T_d = 3\text{s}$, $K_d = 0.94$, $\tau_d = 18\text{s}$

$T_c = 25\text{s}$, $K_c = 0.81$, $\eta_c = 9.2\text{s}$, $\tau_c = 2.8\text{s}$

The comparison of the step responses of the system and its model can be seen in Fig. 5.2. Generally, the model whose parameters are derived from a single step response of the system does approximate the model only roughly on a bounded frequency range. However, in this application example, the aim of which is not a thoroughgoing analysis of the system dynamics but the demonstration of the approaches and the methods worked out in the thesis, let the model with the parameters obtained be considered as sufficient. Nevertheless, the step responses are so close to each other that the model is likely to be a good approximation of the system.

5.2 Analysis of the laboratory plant dynamics in the vicinity of the operational point

Having the linear model of the laboratory heating system in the form of retarded system, let us analyse the modes of the system dynamics. First of all, let us transform model (5.7) into the form of the transfer function

$$G(s) = \frac{N(s)}{M(s)} \exp(-s\tau) \tag{5.8}$$

According to (1.41) and (1.42) the numerator and denominator of (5.8) result in

$$N = 4.04\,10^{-5} \tag{5.9}$$

$$M(s) = s^4 + \sum_{i=0}^{3} Q_i(s) s^i \tag{5.10}$$

where

$$Q_0(s) = 2.721 \cdot 10^{-4} \exp(-15.7s) - 1.036 \cdot 10^{-4} \exp(-40.3s) - 3.265 \cdot 10^{-5} \exp(-49.2s)$$

$$Q_1(s) = 6.803 \cdot 10^{-4} \exp(-6.5s) + 3.810 \cdot 10^{-3} \exp(-9.2s) + 2.313 \cdot 10^{-3} \exp(-15.7s) -$$
$$- 1.450 \cdot 10^{-3} \exp(-33.8s) - 8.163 \cdot 10^{-4} \exp(-40s) - 2.286 \cdot 10^{-4} \exp(-49.2s)$$

$$Q_2(s) = 0.095 + 0.058 \exp(-6.5s) + 0.032 \exp(-9.2s) + 2.857 \exp(-15.7s) -$$
$$- 5.714 \cdot 10^{-3} \exp(-40s)$$

$$Q_3(s) = 0.810 + 0.071 \exp(-6.5s) + 0.040 \exp(-9.2s)$$

and the input delay $\tau = 34\text{s}$. Since the numerator part N of (5.8) is constant, the system does not have any zeros. The character of the input-output dynamics is given only by poles of (5.8), i.e., the roots of $M(s)$ given by (5.10). Using the quasipolynomial mapping based rootfinder explained in section 3.4 given by Algorithm 3.1 with the parameters $\Delta_s = 0.003\,\text{s}^{-1}$ and $\varepsilon_N = 10^{-9}$ s^{-1}, the spectrum of system poles can be seen in Fig. 5.3 and Fig. 5.4.

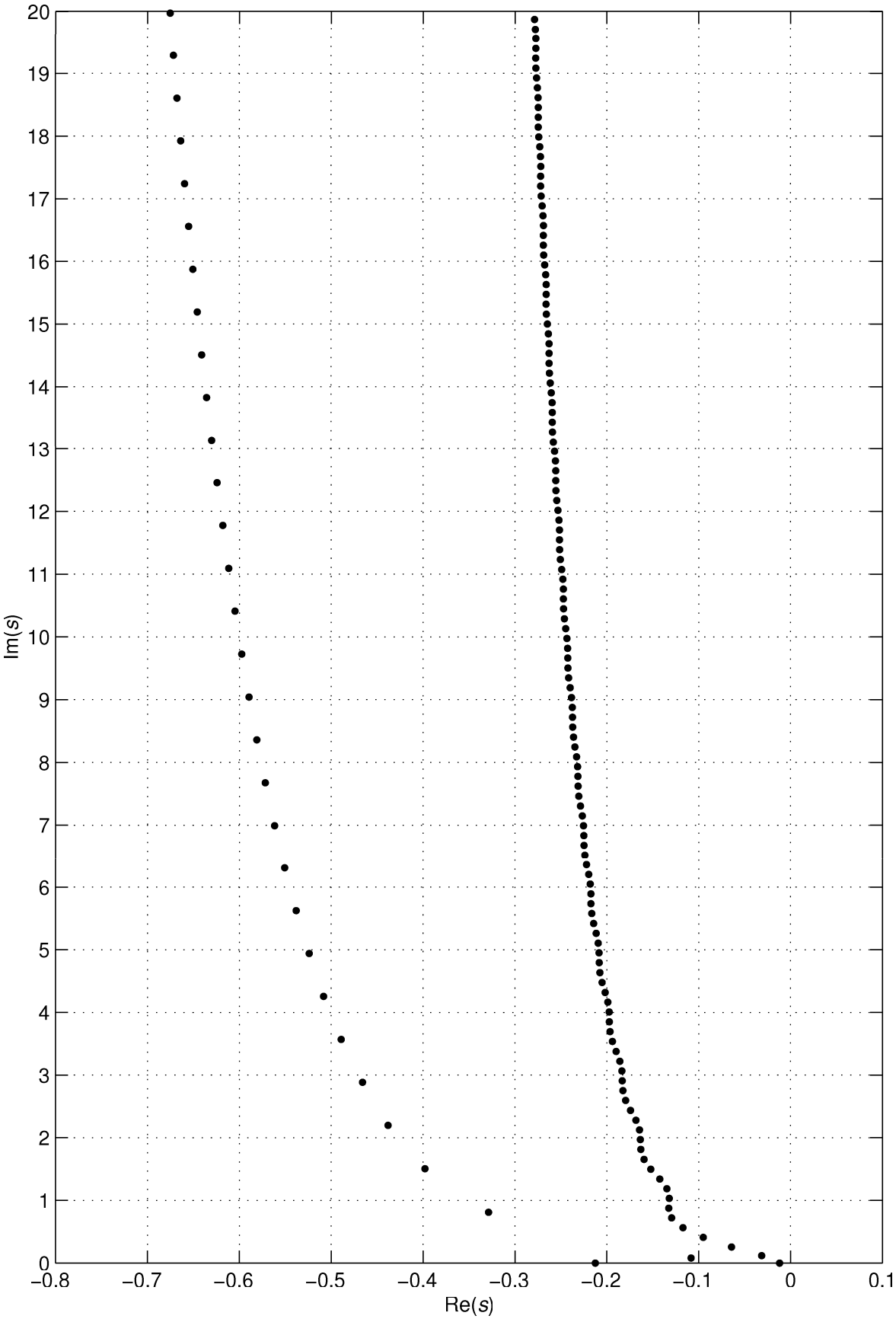

Fig. 5.3 Poles of system (5.7), i.e., roots of quasipolynomial (5.10)

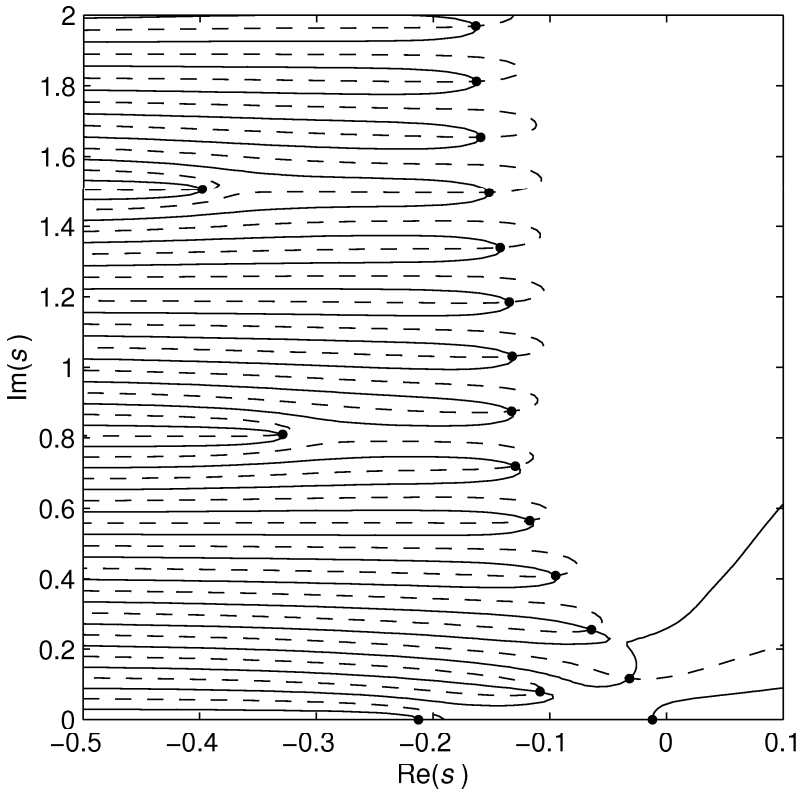

Fig. 5.4 Mapping quasipolynomial (5.10), Re($M(s)$) - solid, Im($M(s)$) - dashed. Roots of the quasipolynomial given as the intersections of the contours.

In order to evaluate the significance of the poles, let us expand transfer function (5.8) into Heaviside series (4.45) and let us evaluate the weighting functions of the partial fractions $H_i(s)$, i=1,2,..., of $M(s)$. Separating the dead time from the system input-output dynamics and applying the generalized Heaviside expansion (4.45), the following expression holds

$$G(s) = G_{\mathrm{f}}(s)\exp(-s\tau) = \sum_{i=1}^{\infty} H_i(s)\exp(-s\tau) = \sum_{i=1}^{\infty} \frac{R(\lambda_i)}{s-\lambda_i}\exp(-s\tau) \qquad (5.11)$$

where λ_i are the poles of model (5.7) and $R(\lambda_i)$ are the corresponding residues. In Tab. 5.1, we can see the values of 34 system poles closest to the s-plane origin with the corresponding values of the residues. The poles are ordered with respect to their significance that has been evaluated using the method based on the absolute value of the difference between the maxima and minima of the weighting functions of $H_i(s)$ (given by (4.46) and (4.47), respectively) for t>0, i.e., using criterion (4.51). As can be seen in Tab. 5.1, according to the criterion, the most significant poles of the system are $\lambda_{1,2}$ and λ_3. Also the following two couples of poles, i.e., $\lambda_{4,5}$ and $\lambda_{6,7}$, may be considered as the significant according to the chosen criterion (but less significant than $\lambda_{1,2}$ and λ_3). In Fig. 5.6 and in the detailed view in Fig. 5.7 we can see the step responses of $G_{\mathrm{f}}(s)$, $H_{1,2}(s)$, $H_3(s)$, $H_{4,5}(s)$ and $H_{6,7}(s)$, (see (5.11)). Considering the step response of the system $H_{1,2}(s)+H_3(s)$ (the sum of the responses of $H_{1,2}(s)$ and $H_3(s)$, which is very close to the response of $G_{\mathrm{f}}(s)$) the response of $H_3(s)$ is its determining part while the response of $H_{1,2}(s)$ only form the

Tab. 5.1 The poles of system (5.7) and the values of the residues and the significance evaluating criterion

i	λ_i [s^{-1}]	$R(\lambda_i)$	h_{ei}
1	-0.0316 + 0.1167j	(-1.718+0.617j) 10^{-3}	4.921 10^{-3}
3	-0.0121	3.801 10^{-3}	3.801 10^{-3}
4	-0.1083 + 0.0791j	(-3.142-5.757) 10^{-4}	7.928 10^{-4}
6	-0.0643 + 0.2553j	(1.094+0.957j) 10^{-4}	3.817 10^{-4}
8	-0.0951 + 0.4088j	(1.653+0.562j) 10^{-5}	5.170 10^{-5}
10	-0.1171 + 0.5648j	(4.356+2.189j) 10^{-6}	1.443 10^{-5}
12	-0.2125	-9.317 10^{-6}	9.317 10^{-6}
13	-0.1295 + 0.7197j	(2.178+1.329j) 10^{-6}	7.607 10^{-6}
15	-0.1327 + 0.8755j	(1.422+0.5287j) 10^{-6}	4.858 10^{-6}
17	-0.1322 + 1.0311j	(7.770+2.072j) 10^{-7}	2.675 10^{-6}
19	-0.1347 + 1.1852j	(4.587+1.150j) 10^{-7}	1.602 10^{-6}
21	-0.1425 + 1.3400j	(2.959+0.635j) 10^{-7}	1.038 10^{-6}
23	-0.1521 + 1.4972j	(1.99+0.198j) 10^{-7}	6.934 10^{-7}
25	-0.1595 + 1.6553j	(1.185+0.020j) 10^{-7}	4.133 10^{-7}
27	-0.1630 + 1.8128j	(7.581+0.974j) 10^{-8}	2.687 10^{-7}
29	-0.1636 + 1.9692j	(6.039+1.514j) 10^{-8}	2.190 10^{-7}
31	-0.3288 + 0.8097j	(2.087+1.088j) 10^{-9}	5.906 10^{-9}
33	-0.3979 + 1.5064j	(-3.959-0.713j) 10^{-11}	1.172 10^{-10}

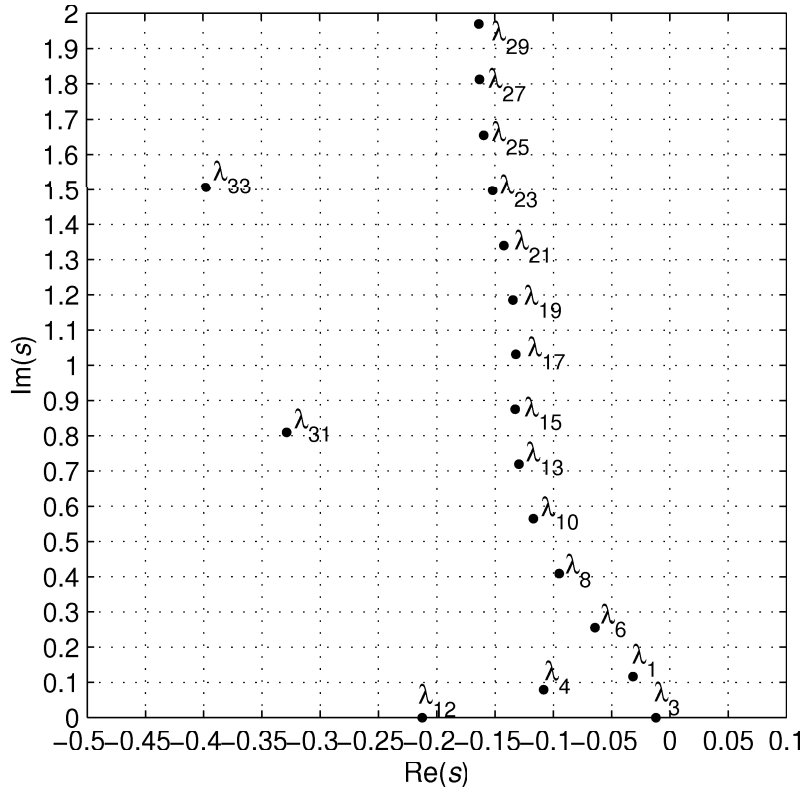

Fig. 5.5 Poles of system (5.7), i.e., roots of quasipolynomial (5.10), detail of Fig. 5.3.

final shape of the response of $H_{1,2}(s)+H_3(s)$ in order to fit the response of $G_f(s)$. This result suggests that the mode of $H_3(s)$ is the most important mode of the system dynamics. However, according to criterion (4.51), the mode corresponding to λ_3 is slightly less significant than the mode that corresponds to $\lambda_{1,2}$. Obviously the mode corresponding to λ_3 plays more important role in the step response and supposedly also in the low frequency range of the dynamics. Nevertheless, it does not mean that this mode is more important than the mode of $\lambda_{1,2}$. In fact, it is not the task to decide which mode is the most important one in the dynamics, but to select the group of the dynamics determining modes. Obviously, in this case, the modes corresponding to $\lambda_{1,2}$ and λ_3 belongs to this group. Let us look at the contribution of the other modes to the response. The contribution of the parts $H_{4,5}(s)$ and $H_{6,7}(s)$ is seen in Fig. 5.6. As can be seen, the approximation of the response of $G_f(s)$ becomes outstanding if it is approximated by the response of $H_{1,2}(s)+H_3(s)+H_{4,5}(s)+H_{6,7}(s)$. In Fig. 5.8, we can see the frequency responses of system (5.7) and its approximations $H_3(s)\exp(-34s)$ and $(H_{1,2}(s)+H_3(s))\exp(-34s)$. As can be seen the former approximation is quite good only in the low frequency range while the latter approximation, which involves both the dominant modes, covers the whole frequency range of the TDS frequency response displayed ($\omega \in [0,1]\,\mathrm{s}^{-1}$). Even better approximation of the response would be obtained, if also the modes $H_{4,5}(s)$ and $H_{6,7}(s)$ were involved. To sum up, from the infinitely many poles of system (5.7), only seven poles that are closest to the *s*-plane origin, i.e., $\lambda_{1,2}$, λ_3, $\lambda_{4,5}$ and $\lambda_{6,7}$, are really significant. The contribution of the other poles to the dynamics may be considered as negligible.

Now, let us show why the system input delay representing the dead time is to be separated from the transfer function of the system in analysing the significance of the modes. Applying the generalized Heaviside expansion directly to the transfer function of system (5.7) we obtain (the input delay is not separated in this case).

$$G(s)=\sum_{i=1}^{\infty}\overline{H}_i(s)=\sum_{i=1}^{\infty}\frac{\overline{R}(\lambda_i)}{s-\lambda_i} \tag{5.12}$$

The poles of the system ordered with respect to the values of criterion (4.51) applied to the transfer functions $\overline{H}_i(s)$ are in Tab. 5.2. The values of the residues $\overline{R}(\lambda_i)$ are also in Tab. 5.2. As can be seen, in this case, the ordering of the poles in the table, i.e., the significance of the poles, is different. The most significant poles (according to the criterion (4.51)) are the poles $\lambda_{4,5}$, λ_{12}, $\lambda_{1,2}$ and perhaps $\lambda_{6,7}$ and $\lambda_{8,9}$. Such a result is not in accordance with the analysis of the step and frequency responses we have performed and which has shown that the most significant poles in the system dynamics are λ_3 and $\lambda_{1,2}$. However, if we look at the comparison of the step response of system (5.7) (with not separated input delay) and its approximations given as the sums of the step responses of $\overline{H}_i(s)$ corresponding to the most significant poles in Tab. 5.2, we can see that the step responses of the approximations are getting close to the system step response with increasing number of $\overline{H}_i(s)$ involved in the approximation. The agreement of the dynamics of model (4.51) with the dynamics of its approximation given by $\overline{H}_{4,5}(s)+\overline{H}_{12}(s)+\overline{H}_{1,2}(s)+\overline{H}_3(s)$ is

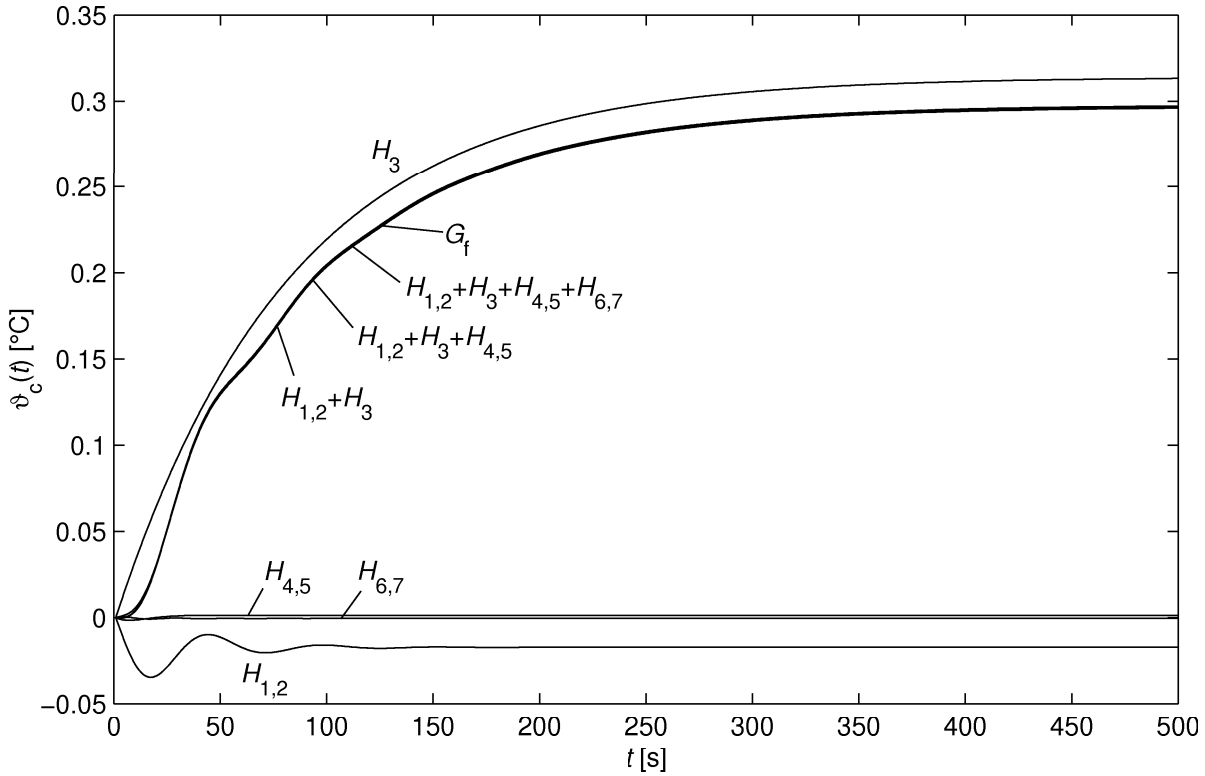

Fig. 5.6 Step responses of the system (5.7) with separated dead time, given by transfer function $G_{\mathrm{f}}(s)$, see (5.11), and of the transfer functions $H_i(s)$ and their sums approximating $G_{\mathrm{f}}(s)$. $\Delta\vartheta_{\mathrm{h,set}}(1) = 1^{\circ}\mathrm{C}$

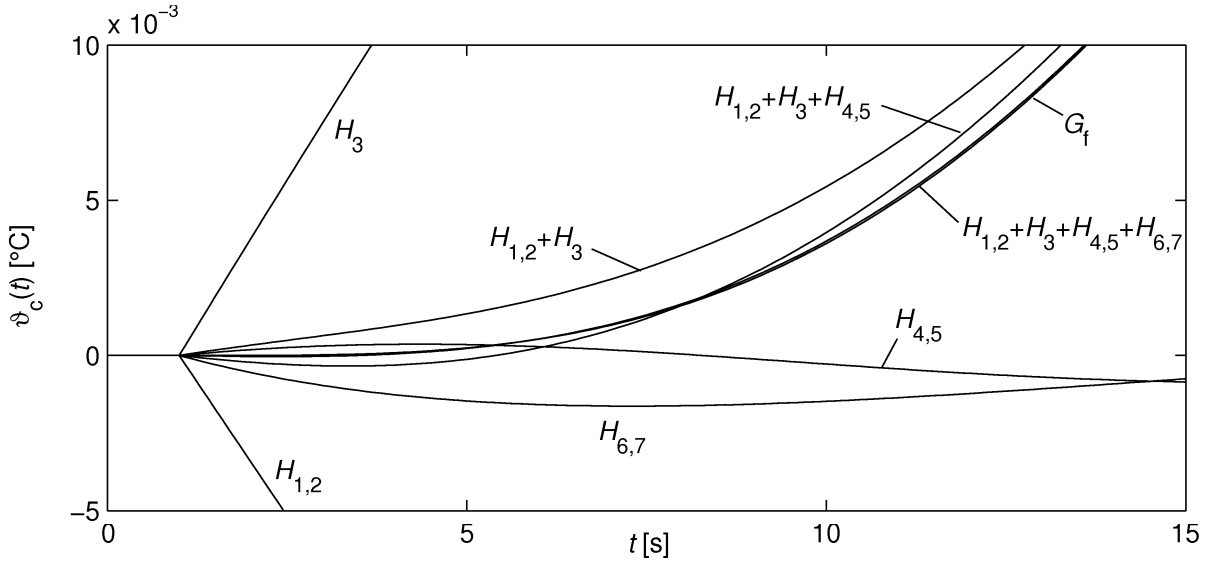

Fig. 5.7 Detail of Fig. 5.6

demonstrated also by the mutual correspondance of their frequency responses. Even though also the system with not separated input delay may be approximated by a finite number of transfer functions $\overline{H}_i(s)$ resulting from the Heaviside expansion, analysis of the weighting functions of $\overline{H}_i(s)$ does not truly evaluate the significance of the system modes. It is given by the fact that also the dead time has to be approximated using the series expansion. It rather

mixes up the evaluated significance of the system modes in the series expansion dynamics. To conclude, the first step in evaluating the input-output dynamics should be the separation of the input lumped delay from the system dynamics. It should be noted that if the input delay is not lumped, it should not be separated, because it may bring about significant zeros to the system dynamics, see section 4.2.4.

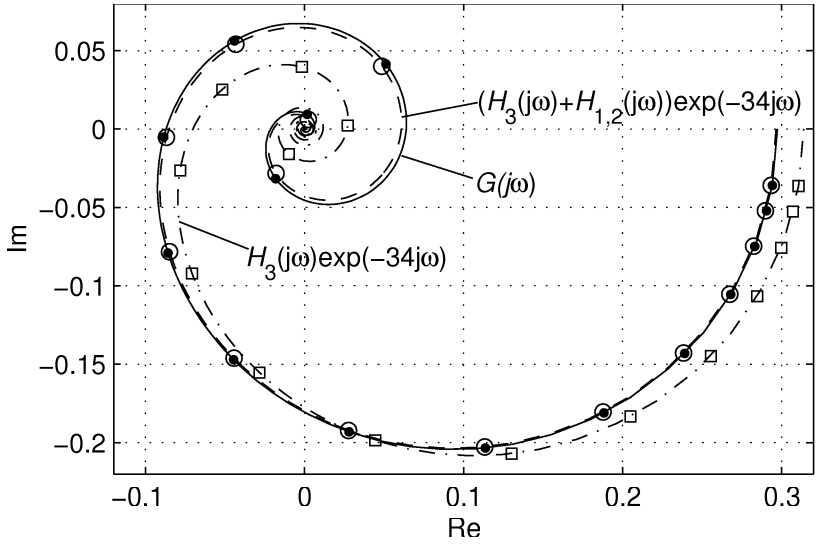

Fig. 5.8 Frequency responses of system (5.7) and its approximations arising from the use of finite number of transfer functions resulting from the Heaviside expansion of $G(s)$ given by (5.11)

Tab. 5.2 The poles of system (5.7) and the values of the significance evaluating criteria

i	λ_i [s^{-1}]	$\overline{R}(\lambda_i)$	$\overline{h}_{ei}$
4	-0.1083 + 0.0791j	(0.121+2.608j) 10^{-2}	1.4601 10^{-2}
12	-0.2125	-1.281 10^{-2}	1.2810 10^{-2}
1	-0.0316 + 0.1167j	(2.073-4.926j) 10^{-3}	1.1516 10^{-2}
3	-0.0121	5.737 10^{-3}	5.7370 10^{-3}
6	-0.0643 + 0.2553j	(-0.137-1.288j) 10^{-3}	2.5454 10^{-3}
8	-0.0951 + 0.4088j	(2.375-3.737j) 10^{-3}	1.0667 10^{-3}
10	-0.1171 + 0.5648j	(2.599+0.294j) 10^{-4}	8.0480 10^{-4}
13	-0.1295 + 0.7197j	(0.733+1.952) 10^{-4}	4.6955 10^{-4}
15	-0.1327 + 0.8755j	(-0.580+1.256j) 10^{-4}	3.8170 10^{-4}
31	-0.3288 + 0.8097j	(-0.574-1.584j) 10^{-4}	2.8244 10^{-4}
17	-0.1322 + 1.0311j	(-6.988+1.707j) 10^{-5}	2.3470 10^{-4}
19	-0.1347 + 1.1852j	(-3.23-3.274j) 10^{-5}	1.3574 10^{-4}
23	-0.1521 + 1.4972j	(3.027-1.810j) 10^{-5}	1.1592 10^{-4}
21	-0.1425 + 1.3400j	(0.787-3.763j) 10^{-5}	1.1477 10^{-4}
25	-0.1595 + 1.6553j	(2.578+0.753j) 10^{-5}	9.2541 10^{-5}
33	-0.3979 + 1.5064j	(-2.160+2.108j) 10^{-5}	7.3147 10^{-5}
27	-0.1630 + 1.8128j	(0.478+1.892j) 10^{-5}	5.8351 10^{-5}
29	-0.1636 + 1.9692j	(-1.202+1.087j) 10^{-5}	5.4172 10^{-5}

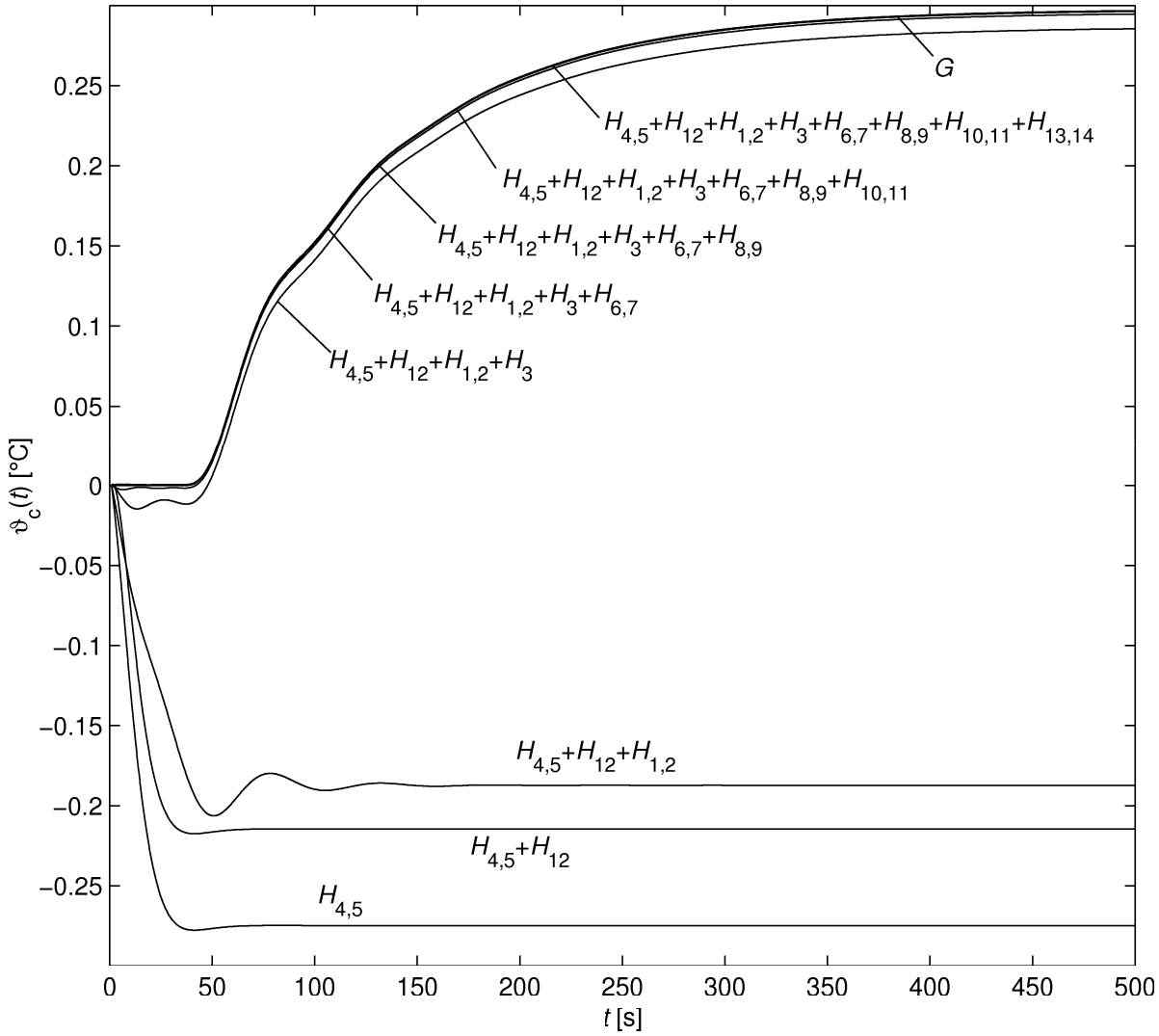

Fig. 5.9 Step response of system (5.7), given by transfer function $G(s)$, see (5.12), and the step responses of transfer functions $\overline{H}_i(s)$ and their sums approximating $G(s)$.

$$\Delta\vartheta_{\mathrm{h,set}}(1) = 1^{\circ}\mathrm{C}$$

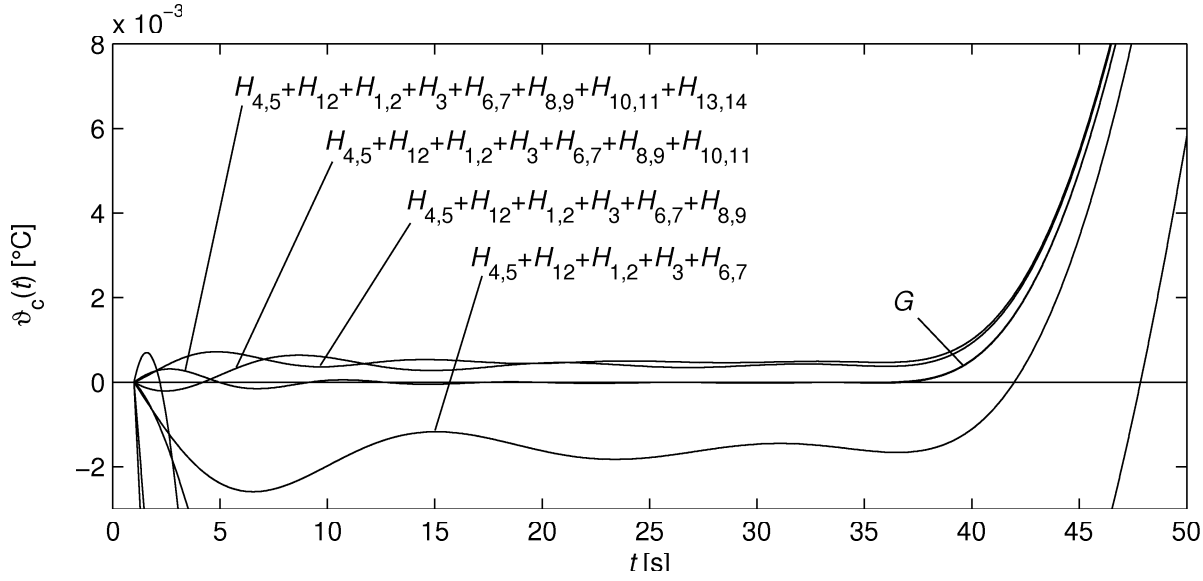

Fig. 5.10 Detail of Fig. 5.9

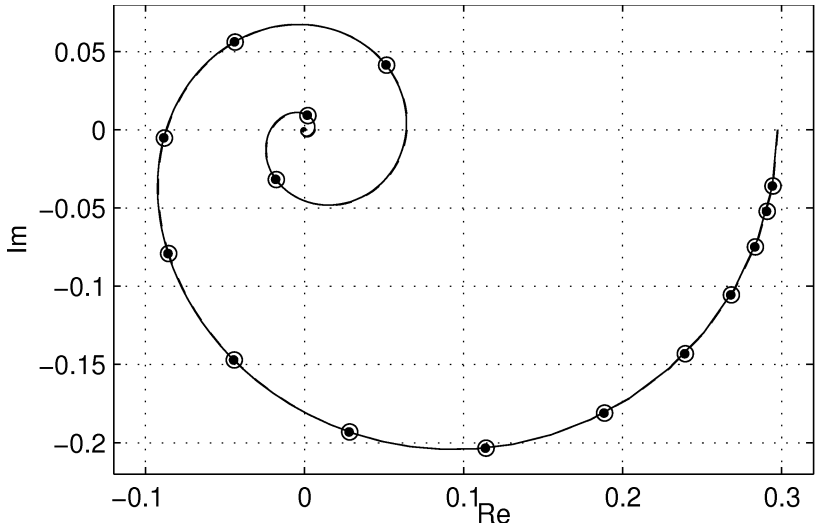

Fig. 5.11 Frequency responses of the system (5.7) (black circles) and its approximations arising from the use of six transfer functions resulting from the Heaviside expansion of $G(s)$, see (5.12), i.e., $\overline{H}_{4,5}(s)+\overline{H}_{12}(s)+\overline{H}_{1,2}(s)+\overline{H}_{3}(s)$, (empty cirles)

5.3 Comparing spectra of poles of laboratory plant model computed by the methods based on discretization

In sections 1.2.7 - 1.2.9, I have made a review of methods used to compute the approximations of the rightmost poles of retarded systems. These methods are based either on discretization of the solution operator or on discretization of the infinitesimal generator of the semigroup. In this section, I am going to apply these methods to compute the approximations of the rightmost poles of model of laboratory plant (5.7) using the following numerical discretization methods, Euler implicit, Trapezoidal and Runge-Kutta Radau IIA, (for references see chapter 1.2.7-1.2.9). The problem I am going to concentrate on is the accuracy of the approximation and the number of the system poles that are satisfactory approximated. The values of the computed approximations of the poles are compared with the poles computed using the quasipolynomial based rootfinder given by Algorithm 3.1, i.e., with the results presented in section 5.2 in Tab. 5.1 and shown in Fig. 5.3, Fig. 5.4 and Fig. 5.5. In order to obtain more precise values of the system poles λ_i, the parameter determining the end of the Newton's iteration has been chosen $\varepsilon_N = 10^{-15}$, which guarantees high acuracy of the computed poles.

First, let us consider the autonomous system associated with (5.7) (which has the same spectrum of poles as (5.7))

$$\frac{d\mathbf{x}(t)}{dt} = \int_0^T d\mathbf{A}(\tau)\mathbf{x}(t-\tau) \tag{5.13}$$

where the matrix $\mathbf{A}(\tau)$ is represented by the matrix $\mathbf{A}(s)$ from laboratory plant model (5.7), i.e., $\mathbf{A}(s) = \int_0^T \exp(-s\tau)d\mathbf{A}(\tau)$. Let us transform the right-hand side of (5.13) into form (1.87). Considering the values of the time constants and the lengths of the delays in system (5.7), let the sampling period be chosen h = 1s. Since the largest delay of system (5.7) is 40s, $H = 40$ in (1.87). The first method that is used to perform the discretization is the Euler implicit method. Disretizing the infinitesimal generator of the semigroup (i.g.s.), the discrete

approximation of the system is of form (1.106) where $\mathcal{A}_1 \in \mathbb{R}^{164\times164}$ is given by (1.117), $h = 1$s and $s = 1$ (for $s=1$ R-K Radau IIA approximation given by (1.117) is equivalent to Euler implicit aproximation). The spectrum of $\mathcal{A}_1$ can be seen in Fig. 5.12 and the comparison of the rightmost eigenvalues of $\mathcal{A}_1$ (asterisks) and the poles of system (5.7) (black circles) can be seen in Fig. 5.13. The rightmost eigenvalues of $\mathcal{A}_1$ approximate directly the rightmost poles of the system. As can be seen in Fig. 5.13, and in Tab. 5.3 (where the absolute values of the error of the pole approximations can be seen), the approximation is rather poor. Only few of the poles are satisfactory approximated. Besides the approximation of the most significant poles λ_3, $\lambda_{1,2}$, $\lambda_{4,5}$ and $\lambda_{6,7}$ the approximation errors are large.

Discretizing the solution operator (s.o.) using the Euler implicit method, the discrete approximation of system (5.13) acquires form (1.90), where the matrix $\mathbf{\Phi} \in \mathbb{R}^{164\times164}$ is given by (1.96), if $\beta_1 = 1$ and $\beta_0 = 0$. The eigenvalues z_l, $l = 1..160$ of $\mathbf{\Phi}$ that are the roots of the characteristic polynomial $M(z) = \det[z\mathbf{I} - \mathbf{\Phi}]$ determining the spectrum of the discrete approximation of $\mathcal{T}^*(1\text{s})$ are displayed in Fig. 5.14. As can be seen, asymptotic stability condition (1.60), i.e., all the eigenvalues z_l are located inside the circle with the radius equal to one, is satisfied. It implies that also the discrete approximation of (5.7) is stable. Unlike in case of discretizing the i.g.s. the spectrum of the s.o. does not directly approximate the poles of the TDS. According to (1.59), the spectrum of the solution operator is given by $\sigma(\mathcal{T}^*(h)) = \{\exp(\lambda_i h)\} = \{\exp(\lambda_i)\}$ which implies that the eigenvalues z_l has to be transformed into the s-plane using (1.78) (note that four-quadrant inverse tangent is used). The final approximations of the rightmost poles of the system can be seen in Fig. 5.15. As can be seen, the rightmost system poles are being approximated much better than in case of using the method based on discretization of the i.g.s.. More poles are approximated with the satisfactory approximation errors, see Tab. 5.3. On the other hand the approximation errors of the most significant poles are comparable with the errors achieved in the previous case.

The second method used to approximate the rightmost poles is the trapezoidal numerical method. Using the method the following matrices are obtained: $\mathcal{A}_1 \in \mathbb{R}^{164\times164}$ given by (1.115) in case of approximating the i.g.s. and $\mathbf{\Phi} \in \mathbb{R}^{164\times164}$ given by (1.96), $\beta_1 = 0.5$ and $\beta_0 = 0.5$ in case of approximating the s.o.. The whole spectrum of the eigenvalues of $\mathcal{A}_1$ can be seen in Fig. 5.16 and the detail of the rightmost eigenvalues in Fig. 5.17. Comparing the positions of the rightmost eigenvalues of $\mathcal{A}_1$ obtained using the Euler method with the rightmost eigenvalues $\mathcal{A}_1$ obtained using the Trapezoidal method, see Tab 5.1, more poles of TDS are approximated well if the latter method is used. As can be seen in Tab. 5.1, also the approximation errors are much less if the later method is used. Particularly the position of the pole λ_3 is approximated with very high accuracy. Also the rightmost pole approximations resulted from the discretization of s.o. using the Trapezoidal method, which can be seen in Fig. 5.19, are more precise than in case of using the implicit Euler discretization method. Note that the pole approximations have also been obtained transforming the eigenvalues of $\mathbf{\Phi}$, the spectrum of which is seen in Fig. 5.18, into the s-plane using (1.78).

The last method used to approximate the i.g.s. and the s.o. is Runge-Kutta Radau IIA with $s = 3$. Using the method the following matrices are obtained: $\mathcal{A}_1 \in \mathbb{R}^{484\times484}$ given by (1.117) in case of approximating the i.g.s. and $\mathbf{\Phi} \in \mathbb{R}^{484\times484}$ given by (1.104), in case of approximating the s.o.. The whole spectrum of the eigenvalues of $\mathcal{A}_1$ can be seen in Fig. 5.20 and the detail of the rightmost eigenvalues in Fig. 5.21. As can be seen in Tab. 5.3, using the

numerical method R-K Radau IIA to discretize i.g.s, the pole approximations are comparable with the pole approximations obtained using R-K Radau IIA to discretize the s.o., see Fig. 5.22 for the original spectrum of the approximation of the s.o. and Fig. 5.23 for the righmost part of the spectrum transformed into the s-plane using (1.78). Moreover, as can be seen in Fig. 5.23, transforming the eigenvalues of $\mathbf{\Phi}$ into the s-plane using (1.78) provides the resultant imaginary parts of the poles in $\mathrm{mod}(\pi h)$. This unpleasant feature is given by evaluating the argument of the complex eigenvalues. The problem arises from the fact that the results of the four-quadrant inverse tangent used in (1.78) are bounded by $[-\pi/h,\pi/h]$ while the boundary of the imaginary parts of the eigenvalues are $[-3\pi/h,3\pi/h]$ in case the method R-K Radau IIA with $s=3$ is used. Obtaining the spectrum of the pole approximations in $\mathrm{mod}(\pi h)$ is inconvenient since the imaginary part of each of the poles has to be further treated to find the real value of the pole approximation. The result of such a procedure can be seen in Fig. 5.25.. The pole approximations that are located in the interval $[0,\pi/h\mathrm{j}]$ in Fig. 5.25, they are located in this interval also in the original spectrum seen in Fig. 5.24, obtained transforming the eigenvalues of $\mathbf{\Phi}$. The pole approximations that are located in the interval $[\pi/h\mathrm{j},2\pi/h\mathrm{j}]$ in Fig. 5.25, they are located in the interval $[-\pi/h\mathrm{j},0]$ in Fig. 5.24. Thus, the sign of the imaginary parts of the poles is to be changed and the value $\pi/h\mathrm{j}$ is to be added to each of the poles. Finally, the pole approximations that are located in the interval $[2\pi/h\mathrm{j},3\pi/h\mathrm{j}]$ in Fig. 5.25, these are located in the interval $[0,\pi/h\mathrm{j}]$ in Fig. 5.24. In this case, the value $2\pi/h\mathrm{j}$ is to be added to each of the poles. On one hand, the procedure of placing the root approximations into the correct positions is rather tedious, see Engelborghs and Roose, (1988). However, on the other hand, considerably more poles are well approximated.

Tab. 5.3 Absolute values of the error approximation of the system poles from Tab. 5.1,

i	Euler, h = 1s		Trapezoidal, h = 1s		R-K, Radau IIA, h =1s		Trapezoidal
	i.g.s.	s.o.	i.g.s.	s.o.	i.g.s.	s.o.	s.o., h = 0.33s
1	$6.111\ 10^{-3}$	$2.660\ 10^{-3}$	$1.191\ 10^{-4}$	$8.318\ 10^{-5}$	$3.073\ 10^{-5}$	$3.073\ 10^{-5}$	$1.394\ 10^{-5}$
3	$2.957\ 10^{-5}$	$4.345\ 10^{-5}$	$8.033\ 10^{-8}$	$2.283\ 10^{-7}$	$1.401\ 10^{-7}$	$1.401\ 10^{-7}$	$5.074\ 10^{-8}$
4	$1.104\ 10^{-2}$	$1.278\ 10^{-2}$	$6.315\ 10^{-4}$	$6.595\ 10^{-4}$	$3.973\ 10^{-4}$	$3.973\ 10^{-4}$	$6.930\ 10^{-5}$
6	$3.281\ 10^{-2}$	$5.760\ 10^{-3}$	$1.600\ 10^{-3}$	$3.614\ 10^{-4}$	$1.362\ 10^{-4}$	$1.362\ 10^{-4}$	$6.480\ 10^{-5}$
8	$8.130\ 10^{-2}$	$7.156\ 10^{-3}$	$6.205\ 10^{-3}$	$4.462\ 10^{-4}$	$9.297\ 10^{-5}$	$9.356\ 10^{-5}$	$1.073\ 10^{-4}$
10	$1.516\ 10^{-1}$	$8.609\ 10^{-3}$	$1.605\ 10^{-2}$	$6.002\ 10^{-4}$	$3.242\ 10^{-4}$	$3.281\ 10^{-4}$	$1.342\ 10^{-4}$
12	$1.093\ 10^{-2}$	$7.150\ 10^{-3}$	$1.081\ 10^{-4}$	$6.961\ 10^{-4}$	$3.355\ 10^{-4}$	$3.355\ 10^{-4}$	$1.147\ 10^{-4}$
13	$2.431\ 10^{-1}$	$1.127\ 10^{-2}$	$3.358\ 10^{-2}$	$8.870\ 10^{-4}$	$6.572\ 10^{-4}$	$6.738\ 10^{-4}$	$1.275\ 10^{-4}$
15	$3.348\ 10^{-1}$	$1.681\ 10^{-2}$	$6.135\ 10^{-2}$	$2.199\ 10^{-3}$	$5.390\ 10^{-4}$	$5.801\ 10^{-4}$	$3.024\ 10^{-4}$
17		$2.383\ 10^{-2}$	$1.028\ 10^{-1}$	$4.750\ 10^{-3}$	$5.471\ 10^{-4}$	$4.074\ 10^{-4}$	$6.932\ 10^{-4}$
19		$2.982\ 10^{-2}$	$1.623\ 10^{-1}$	$7.248\ 10^{-3}$	$1.199\ 10^{-3}$	$8.340\ 10^{-4}$	$1.088\ 10^{-3}$
21		$3.318\ 10^{-2}$	$2.457\ 10^{-1}$	$8.679\ 10^{-3}$	$1.279\ 10^{-3}$	$6.782\ 10^{-4}$	$1.328\ 10^{-3}$
23		$3.478\ 10^{-2}$	$3.605\ 10^{-1}$	$9.740\ 10^{-3}$	$9.919\ 10^{-4}$	$5.997\ 10^{-4}$	$1.454\ 10^{-3}$
25		$3.706\ 10^{-2}$		$1.170\ 10^{-2}$	$1.487\ 10^{-3}$	$1.147\ 10^{-3}$	$1.606\ 10^{-3}$
27		$4.124\ 10^{-2}$	$6.557\ 10^{-2}$	$1.545\ 10^{-2}$	$3.545\ 10^{-3}$	$1.078\ 10^{-3}$	$1.951\ 10^{-3}$
29		$4.694\ 10^{-2}$		$2.127\ 10^{-2}$	$7.419\ 10^{-3}$	$8.613\ 10^{-4}$	$2.610\ 10^{-3}$
31	$3.545\ 10^{-1}$	$5.705\ 10^{-2}$		$1.633\ 10^{-2}$	$8.359\ 10^{-3}$	$8.369\ 10^{-3}$	$1.573\ 10^{-3}$
33		$9.909\ 10^{-2}$		$4.859\ 10^{-2}$	$2.597\ 10^{-2}$	$2.515\ 10^{-2}$	$3.993\ 10^{-3}$

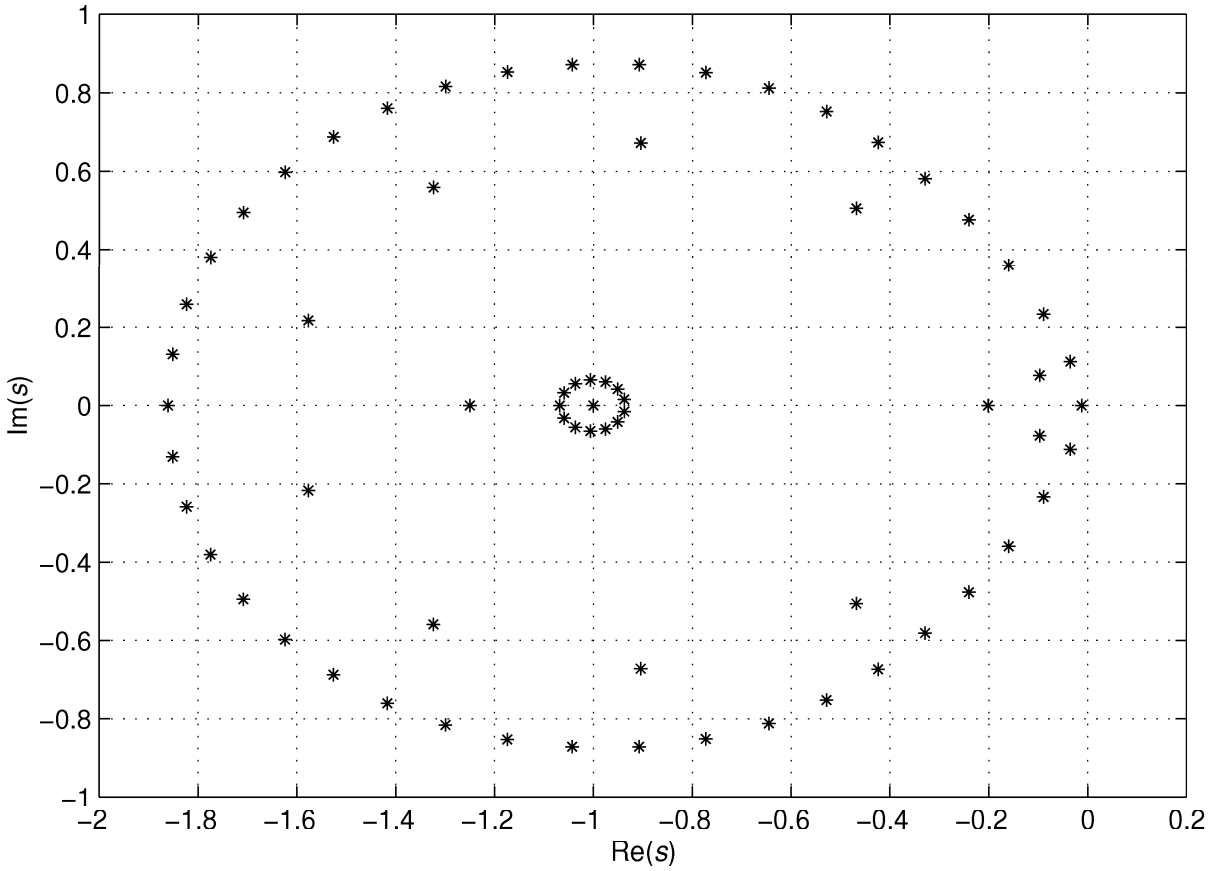

Fig. 5.12 Whole spectrum of the eigenvalues of matrix $\mathcal{A}_1$, resulting from discretizing the infinitesimal generator of the semigroup using Euler implicit method

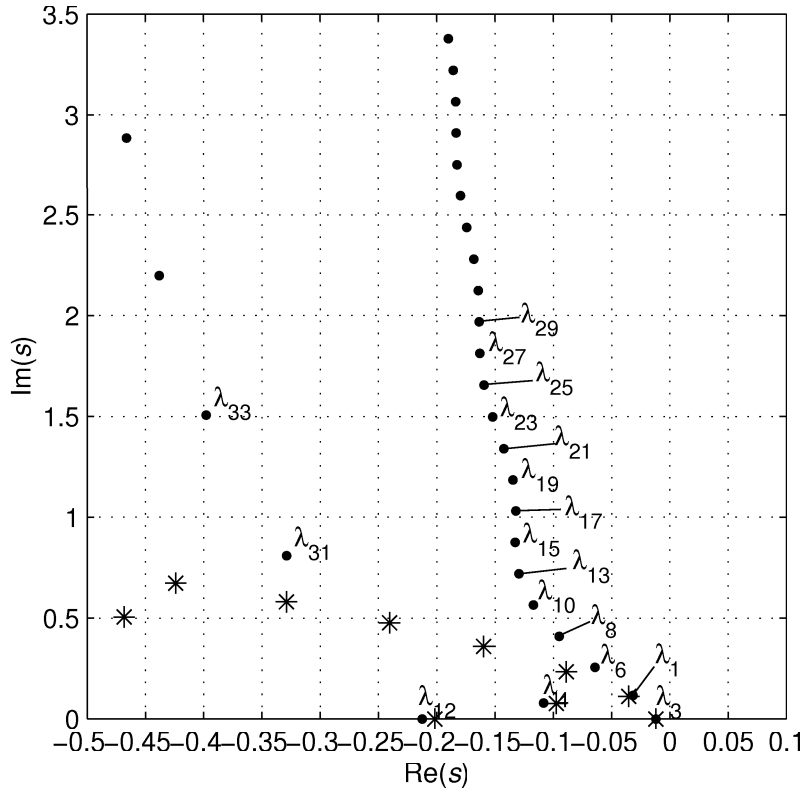

Fig. 5.13 Comparison of the rightmost eigenvalues of $\mathcal{A}_1$ - asterisks (detail of Fig. 5.12, approximations of the system poles) with the real values of the system poles - black circles

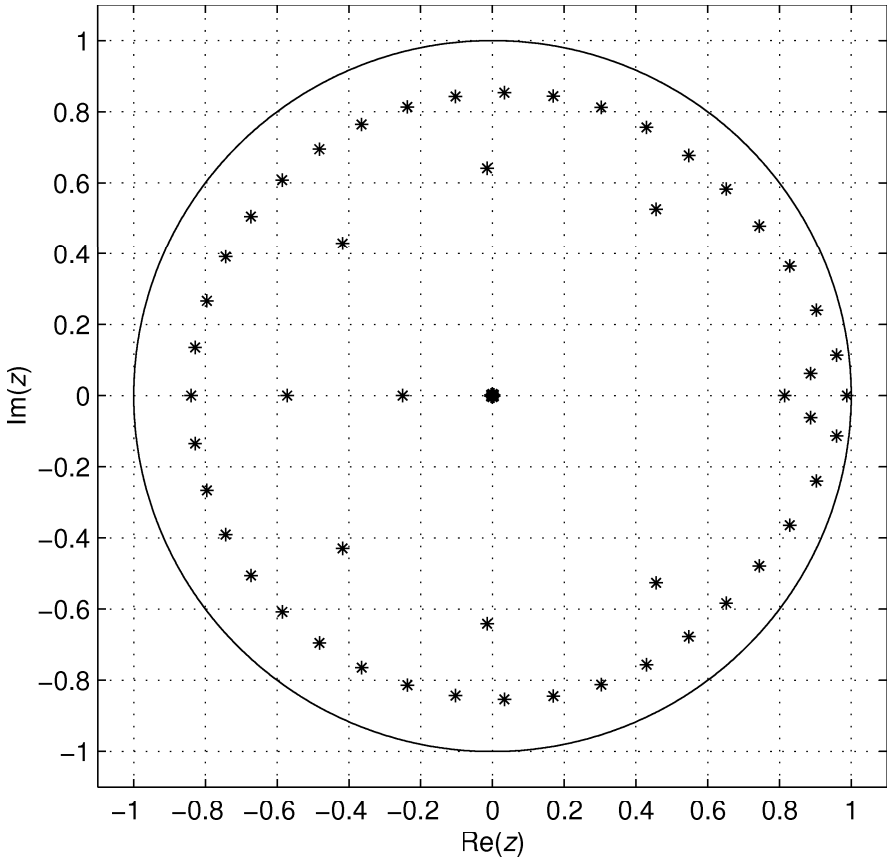

Fig. 5.14 Whole spectrum of the eigenvalues of matrix $\mathbf{\Phi}$, resulting from discretizing the solution operator using Euler implicit method

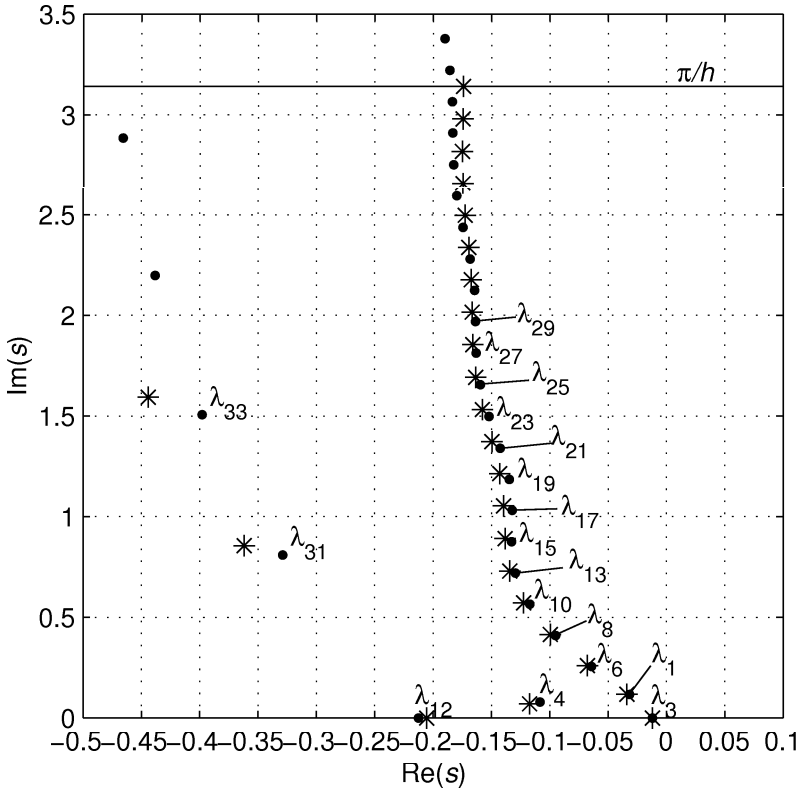

Fig. 5.15 Comparison of the right most eigenvalues of $\mathbf{\Phi}$ transformed into s-plane (from Fig. 5.14) - asterisks, with the real values of the system poles - balck circles

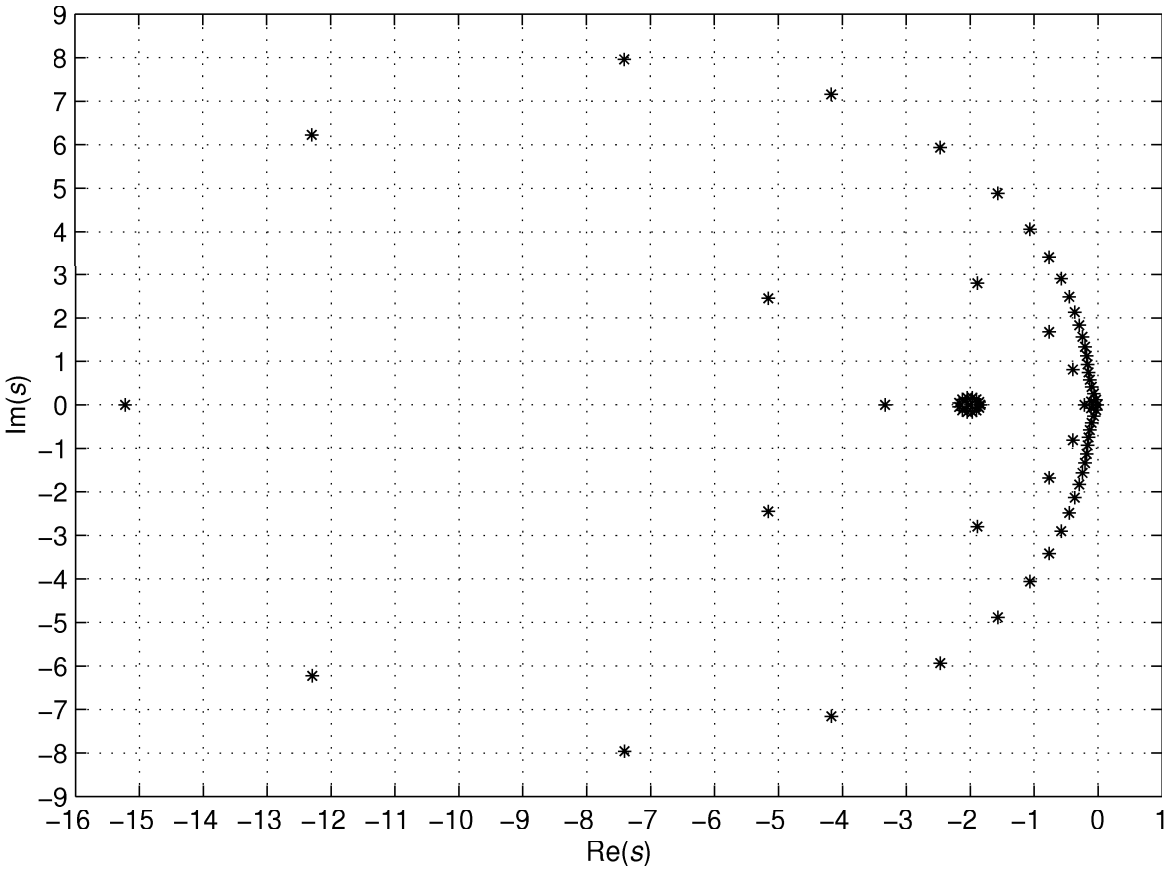

Fig. 5.16 Whole spectrum of eigenvalues of the matrix $\mathcal{A}_1$, resulting from discretizing the infinitesimal generator of the semigroup using Trapezoidal method

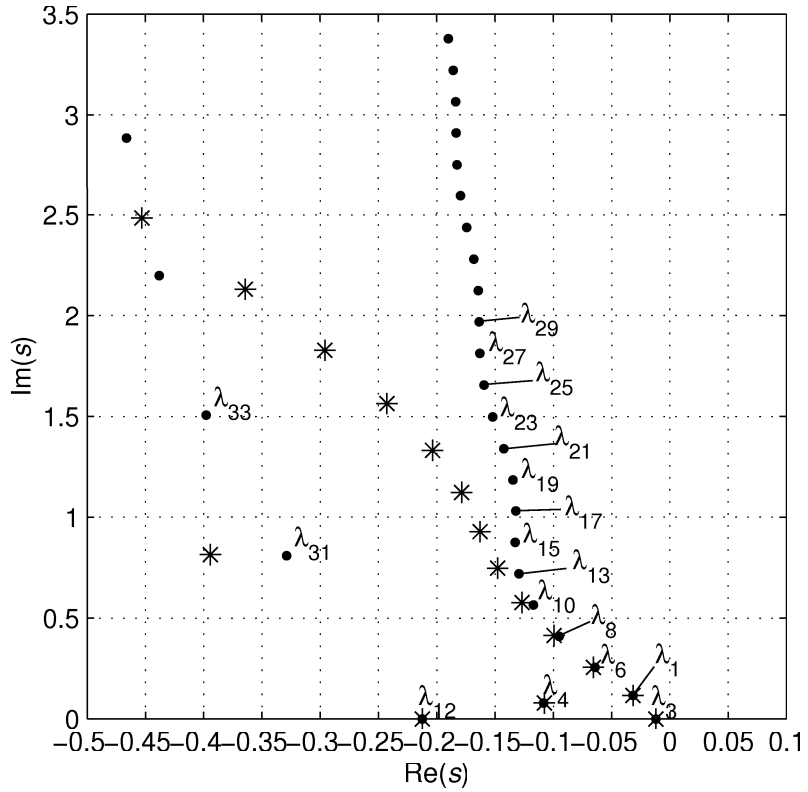

Fig. 5.17 Comparison of the right most eigenvalues of $\mathcal{A}_1$ - asterisks (detail of Fig. 5.16, approximations of the system poles) with the real values of the system poles - black circles

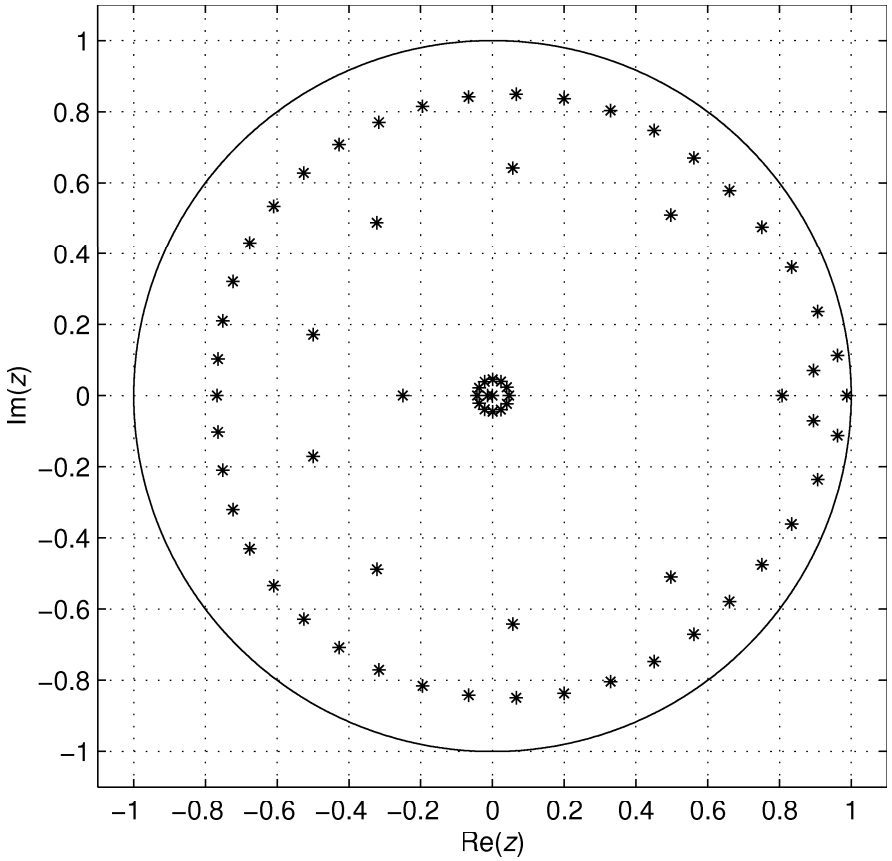

Fig. 5.18 Whole spectrum of the eigenvalues of matrix $\mathbf{\Phi}$, resulting from discretizing the solution operator using Trapezoidal method

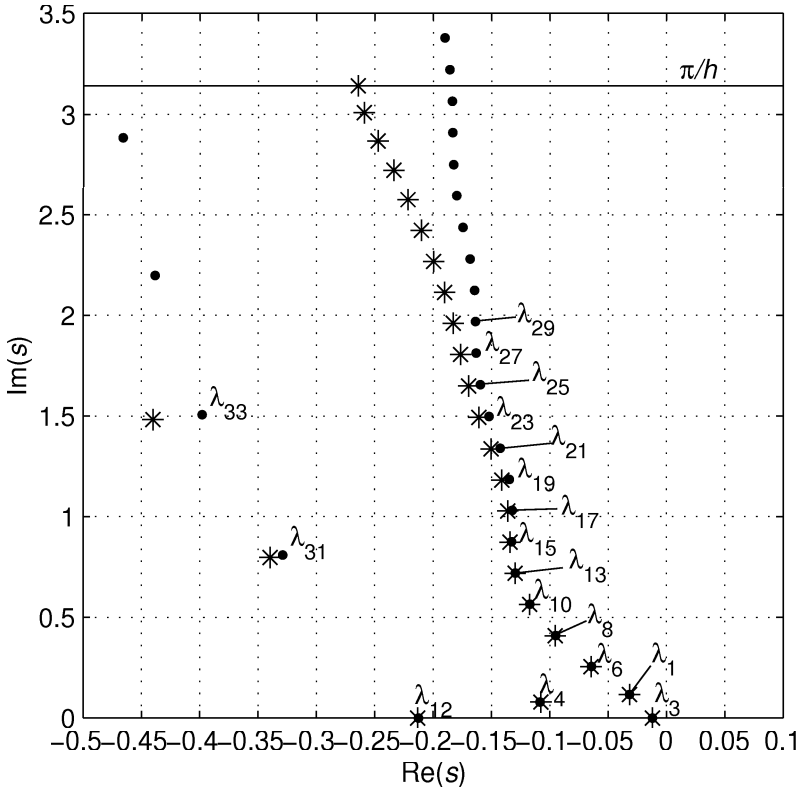

Fig. 5.19 Comparison of the right most eigenvalues of $\mathbf{\Phi}$ transformed into s-plane (from Fig. 5.18) - asterisks, with the real values of the system poles black circles

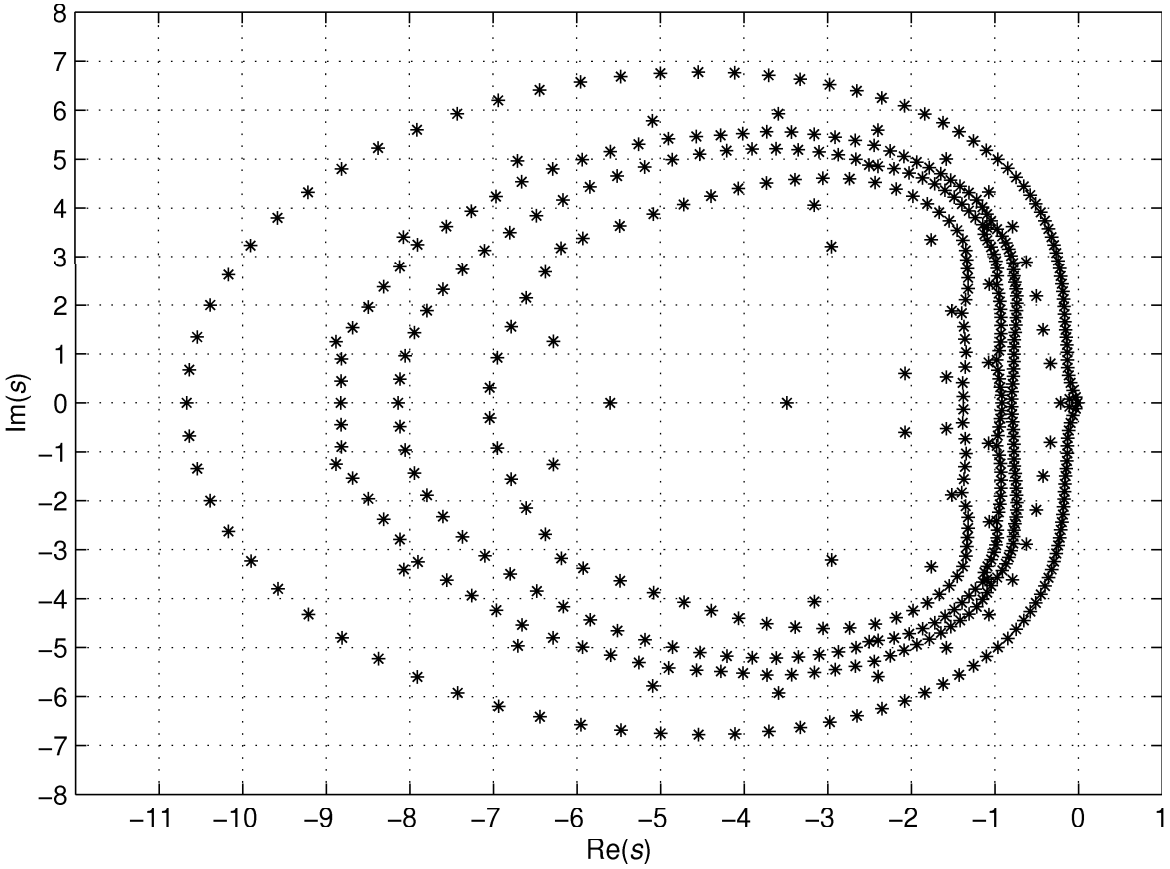

Fig. 5.20 Whole spectrum of the eigenvalues of the matrix $\mathcal{A}_1$, resulting from discretizing the infinitesimal generator of the semigroup using R-K method Radau IIA

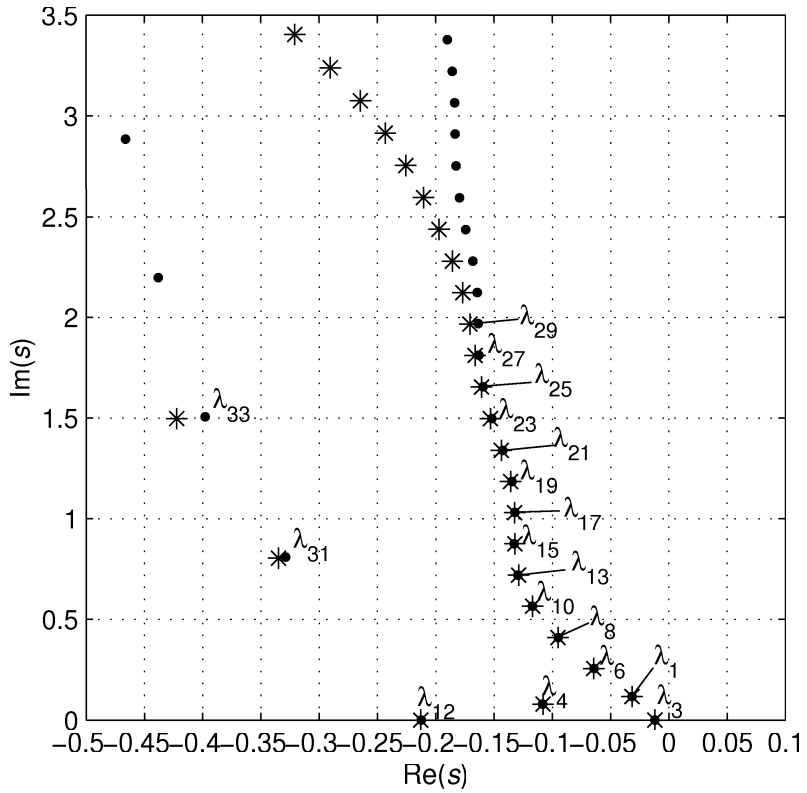

Fig. 5.21 Comparison the right most eigenvalues of $\mathcal{A}_1$ (detail of Fig. 5.20, approximations of the system poles) - asterisks, with the real values of the system poles - black circles

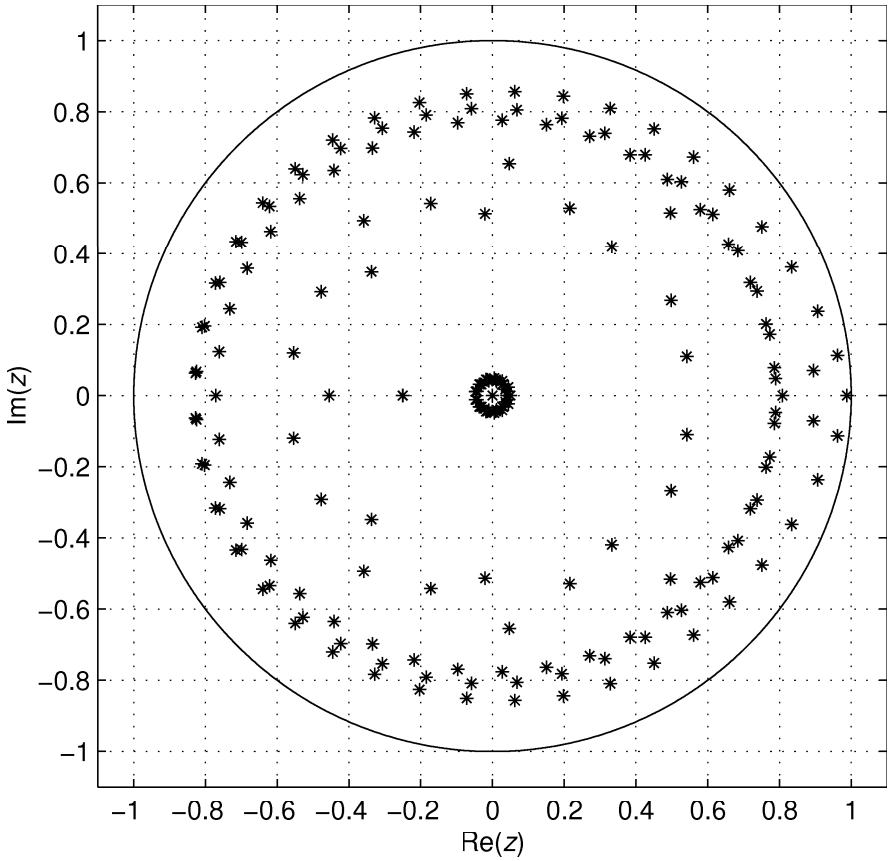

Fig. 5.22 Whole spectrum of the eigenvalues of matrix $\mathbf{\Phi}$, resulting from discretizing the solution operator using R-K method Radau IIA

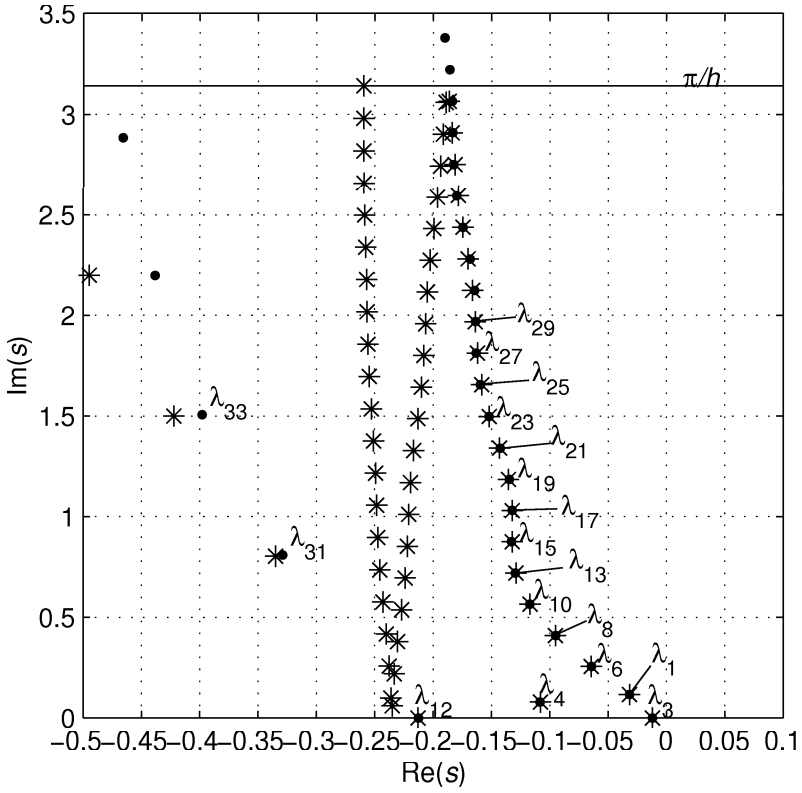

Fig. 5.23 Comparison of the right most eigenvalues of $\mathbf{\Phi}$ transformed into s-plane (from Fig. 5.22) -asterisks, with the real values of the system poles - black circles

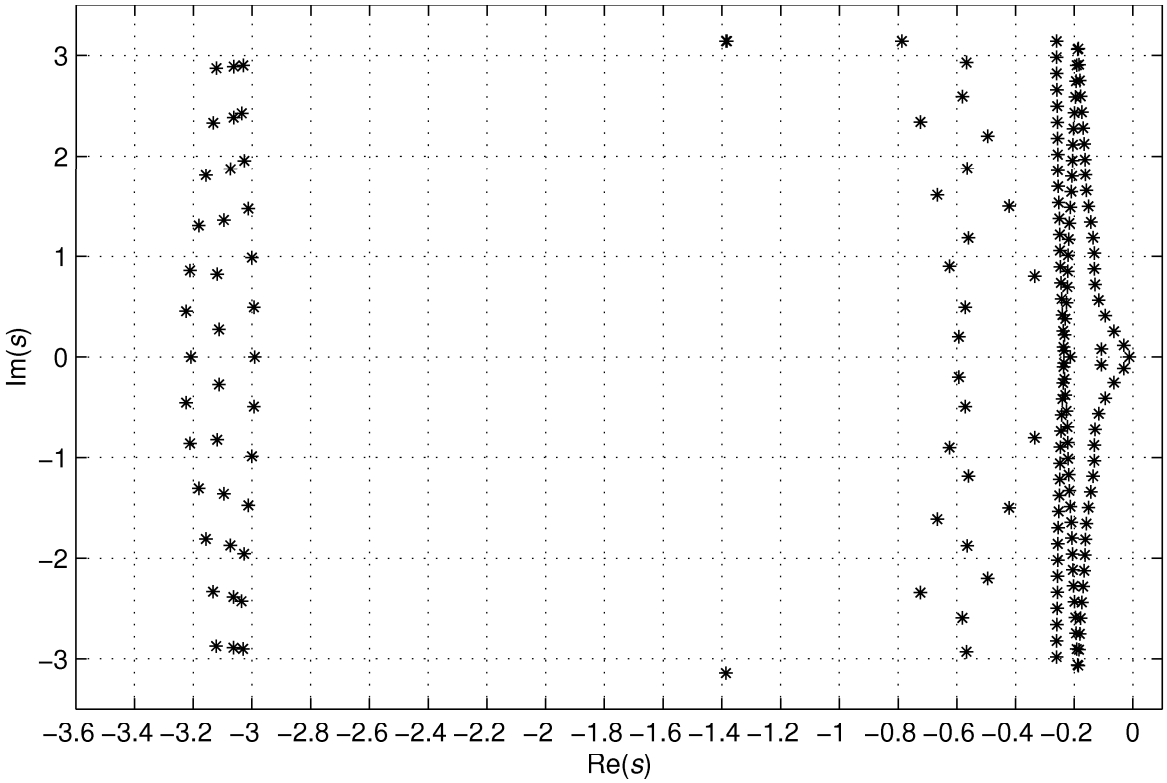

Fig. 5.24 Whole spectrum of the eigenvalues of matrix $\mathbf{\Phi}$ resulting from discretizing the solution operator using R-K method Radau IIA, transformed into s-plane

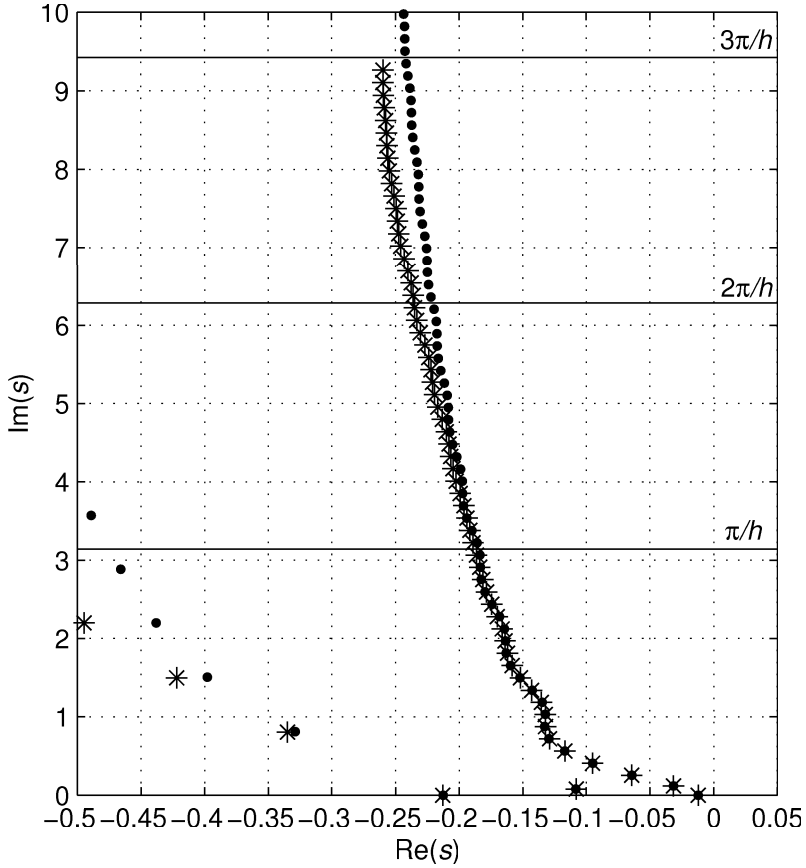

Fig. 5.25 The constructed spectrum of the approximation of the rightmost poles from Fig. 5.23

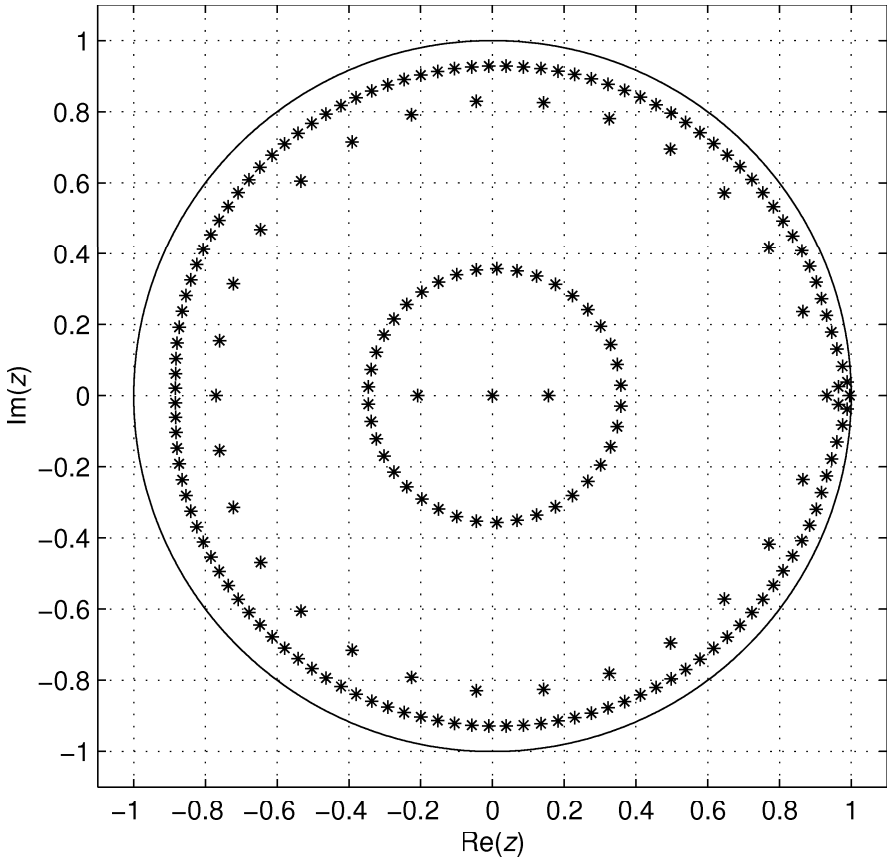

Fig. 5.26 Whole spectrum of the eigenvalues of matrix $\mathbf{\Phi}$, resulting from discretizing the solution operator using Trapeziodal method, $h = 0.33$s

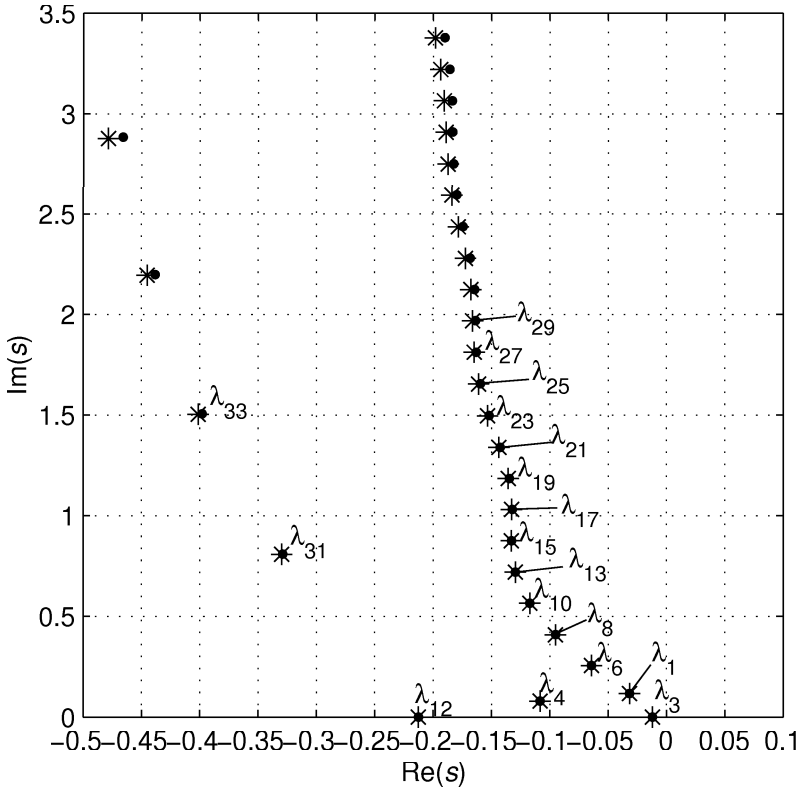

Fig. 5.27 Comparison of the right most eigenvalues of $\mathbf{\Phi}$ transformed into s-plane (from Fig. 5.26, $h = 0.33$s) - asterisks, with the real values of the system poles - black circles

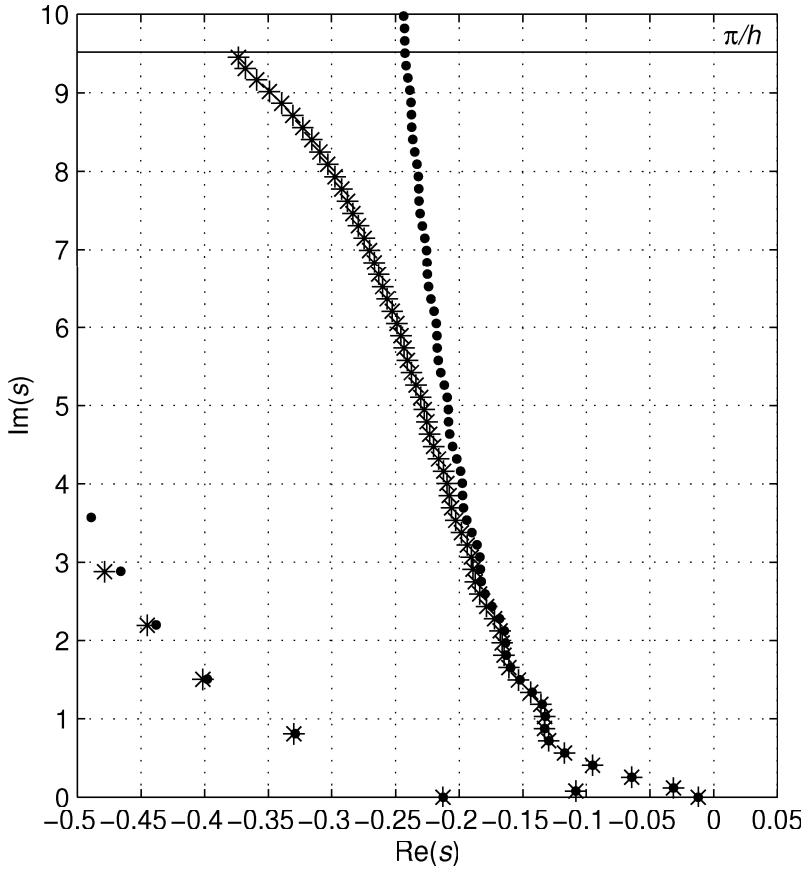

Fig. 5.28 Comparison of the right most eigenvalues of Φ transformed into *s*-plane (from Fig. 5.26, $h = 0.33$s), with the real values of the system poles, enlarged region

In Fig. 5.26 we can see the spectrum of discretized s.o. using the Trapezoidal numerical method for $h = 0.33$s, i.e., the discretization step is approximately one third of the originally used discretization step $h = 1$s. The rightmost eigenvalues of $\mathbf{\Phi} \in \mathrm{R}^{492\times492}$, which results from the use of Trapezoidal method, transformed into the *s*-plane can be seen in Fig. 5.27. The approximation errors of the poles for $h = 0.33$s can be seen in the last column of Tab. 5.3. As can be seen, the accuracy of the approximation of the rightmost poles is equivalent to the approximation of the poles obtained using R-K Radau IIA method with $h = 1$s. As can be seen in the enlarged region of the *s*-plane seen in Fig. 5.28, also some of the poles with larger imaginary parts are quite well approximated. On one hand, the approximation of these poles is not so good as in the case of using R-K Radau IIA method to discretize the s.o., compare Fig. 5.28 and Fig. 5.25. However, on the other hand, the imaginary parts of the pole approximations do not have to be recalculated as it is necessary to do in case of using R-K Radau IIA method to discretize the s.o..

To sum up, the accuracy of the approximations of the poles varies with respect to the numerical method used for the discretization as well as with respect to h. Considering the number of poles being approximated and the accuracy of the approximation, the best results have been obtained using R-K Radau IIA method to discretize the s.o.. The equivalent accuracy of the approximation of the rightmost poles has been obtained using the same method to discretize the i.g.s., but the number of poles being satisfactory approximated was lower. On the other hand, the method based on discretization of the i.g.s. is convenient because the rightmost eigenvalues of the resultant matrix are approximate directly the system rightmost poles. Also the Trapezoidal numerical method has provided good results. Taking into consideration that the sizes of the matrices $\mathcal{A}_1$ and $\mathbf{\Phi}$ were one third of the sizes of the

matrices in case of using R-K Radau IIA method, the approximations of the right most pole were quite good. Equalising the sizes of the matrices $\mathcal{A}_1$ and $\mathbf{\Phi}$ in both the methods, i.e., using h = 0.33s for discretization based on Trapezoidal numerical method, the results obtained are very close.

As has been mentioned in sections 1.2.7 -1.2.9 and shown in the application example, the methods based on the discretization provide only the approximation of the rightmost poles of the retarded systems. In some cases the results obtained may be rather confusing, especially if the s.o. is discretized, which provides the imaginary parts of the poles in $\mathrm{mod}(\pi/h)$. The eigenvalues corresponding to the approximations of the poles may be mixed with the other eigenvalues of the matrices $\mathcal{A}_1$ and $\mathbf{\Phi}$, respectively. Also the asymptotic features of the chains of the poles are not clearly seen in the spectrum of the poles. As has been shown in section 5.2, see Fig. 5.3 the quasipolynomial mapping based rootfinder I have introduced in the thesis (given by Algorithm 3.1) provides more transparent result. The approximations of the poles do not decay with the increasing modulus of the poles. Moreover arbitrary part of the spectrum can be computed using the rootfinder. On the other hand, it should be noted that the application of the rootfinder is limited to the low order TDS. As has been shown in section 3.4.4, the algorithm may be used only if the quasipolynomial, i.e., the characteristic function of the system, is not ill conditioned. If the quasipolynomial is ill conditioned, using a method based on discretization is reasonable because the approximations of the poles are computed directly from the matrices, which is numerically more robust.

5.4 Approximation of the poles of laboratory plant model using its δ-model

In section 1.2.10, I have outlined the basic ideas of the method for computing the rightmost poles of the system based on the δ-model corresponding to retarded TDS. In this section, I am not going to study the accuracy of the approximation of the poles obtained using the method based on the δ-model because it is likely to be equivalent to the results obtained using the discretization of the solution operator. In this section I am going to demonstrate the basic feature of the eigenvalues of the finite order δ-model which converge to the poles of the original continuous time model as $h \to 0$. Let us use the Trapezoidal rule in the discrete-time integrator $I(\delta)$, which implies $I(\delta) = (0.5h\delta + 1)/\delta$ and the right-hand side of system (5.13) being discretized into form (1.87). Let us gradually decrease the discretization step h starting from h = 1s with the decrement $\Delta h = 0.0005$ s. The results of such a procedure can be seen in Fig. 5.29 and Fig. 5.30. First, let us look at the distribution of the poles of the δ-model for h = 1s, which can be seen in Fig. 5.18 (for h = 1s is the spectrum of δ-model identical with the spectrum of the discrete model that results from discretizing the s.o. using trapezoidal method). As can be seen in Fig. 5.18, there are two groups of poles. The first group is close to the centre -1/h of the δ-plane stability region given by the circle with the radius 1/h. The poles of this group do not correspond to the approximation of the poles of the continuous time model. The same group of poles can also be seen in Fig. 5.12-5.26 and the number of poles in this group depends on the numerical method being used. This group of poles is distributed in a circle with the centre in the centre of the δ-plane stability region. Obviously, this group of poles is moving to the left and the radius of the circle is increasing as h is being decreased. This phenomenon can be seen in Fig. 5.29 and 5.30. The poles of the other group of the poles of δ-model approximate the poles of the continuous time model. As can be seen in Fig. 5.29 and 5.30, these poles move continuously as h decreases. In Fig. 5.31 we can see the trajectories of these poles for $h \in [1, 0.05]$ s. As can be seen the poles of the second group for h = 1s converge to the poles of (5.7) as h decreases. The other trajectories have the origins close to the real axis.

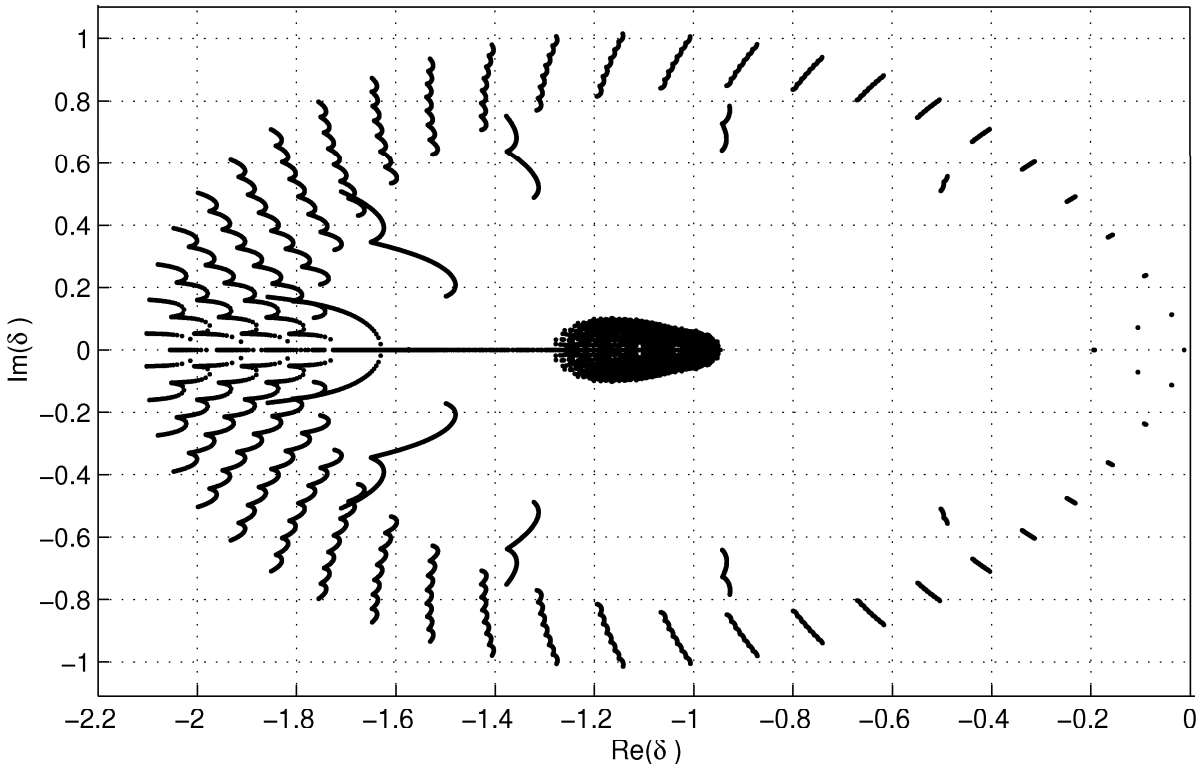

Fig. 5.29 The positions of the poles of the δ-model corresponding to the laboratory heating system described by (5.7) as the function of the discretization step h = (1-0.0005l)s, l=0,..300

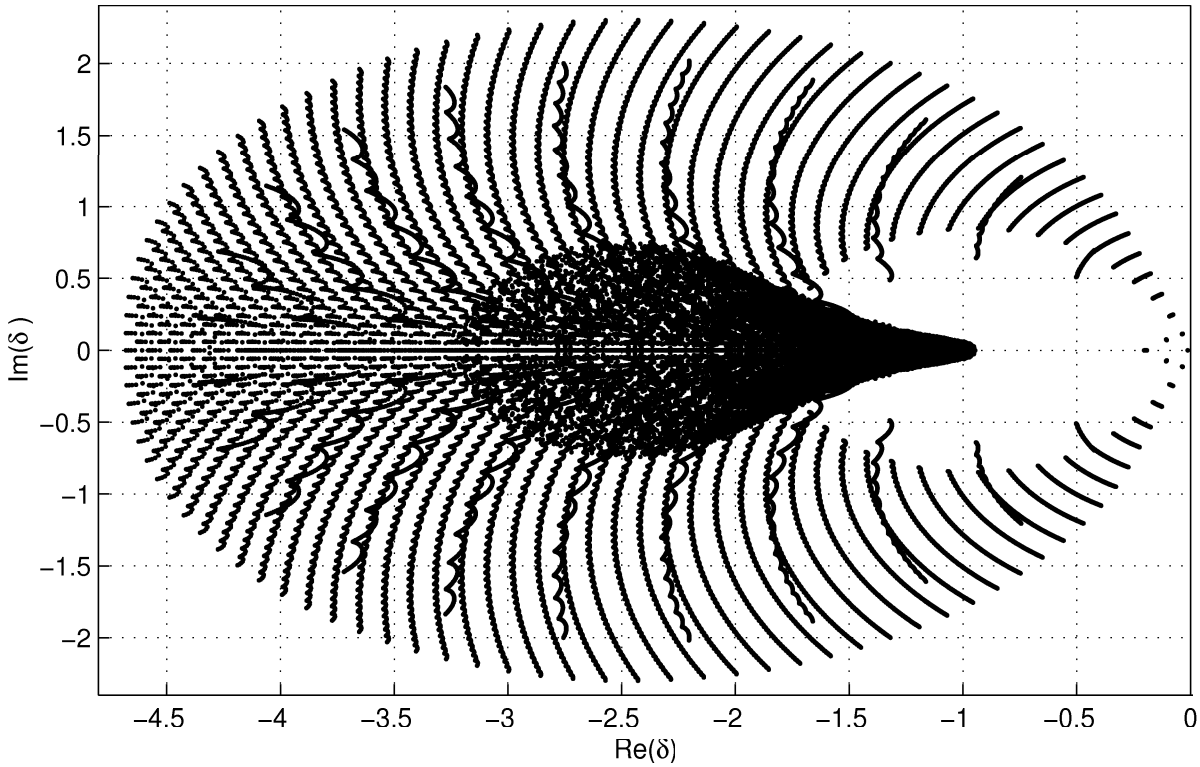

Fig. 5.30 The positions of the poles of the δ-model corresponding to the laboratory heating system described by (5.7) as the function of the discretization step h = (1-0.0005l)s, l=0,..1200

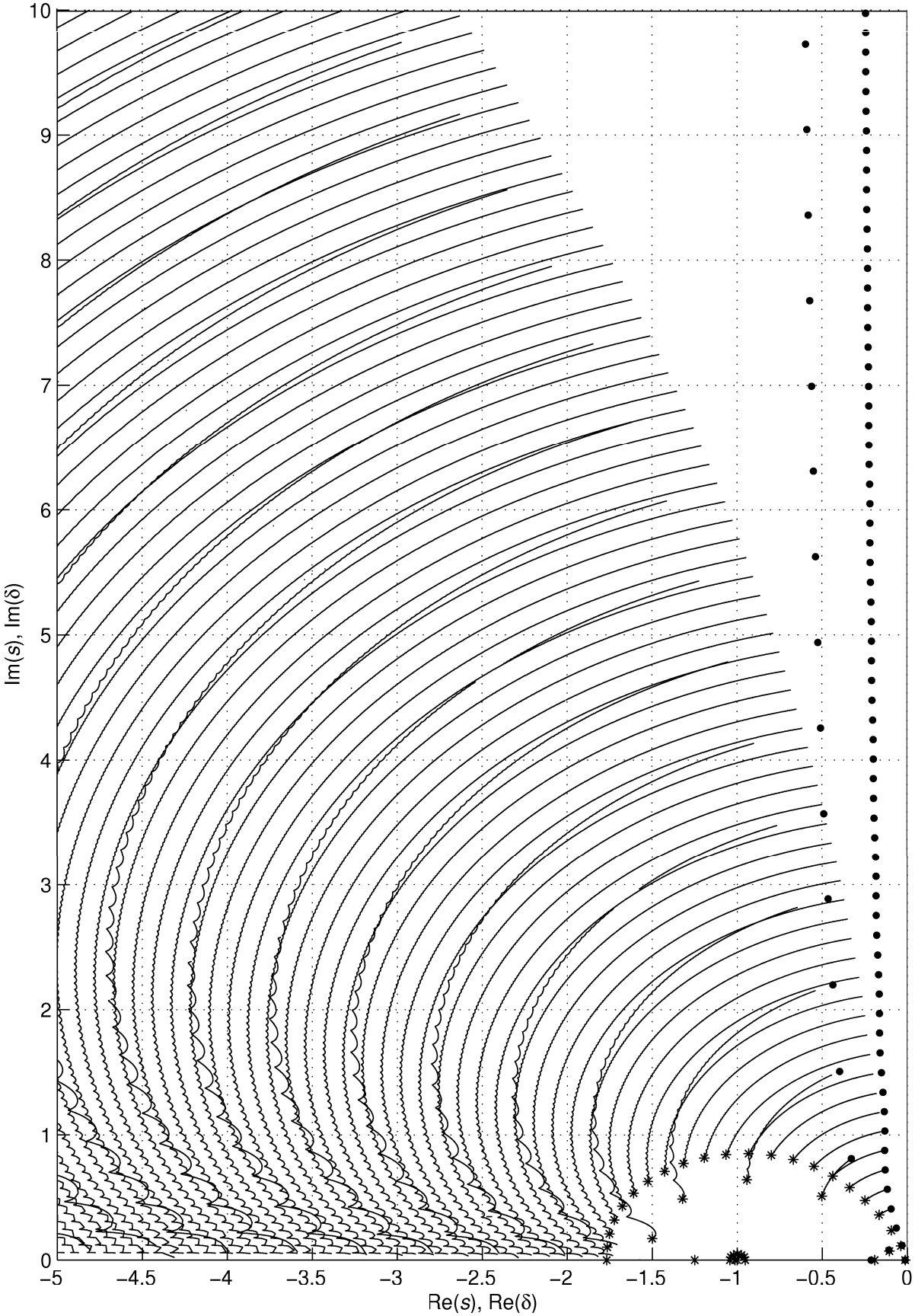

Fig. 5.31 Trajectories of the poles of δ-model corresponding to the laboratory heating system described by (5.7) that converge to the poles of model (5.7) as the functions of $h \in [1,0.05]$ s.

Black circles - poles of the continuous system, Asterisks - poles of δ-model with $h = 1$s

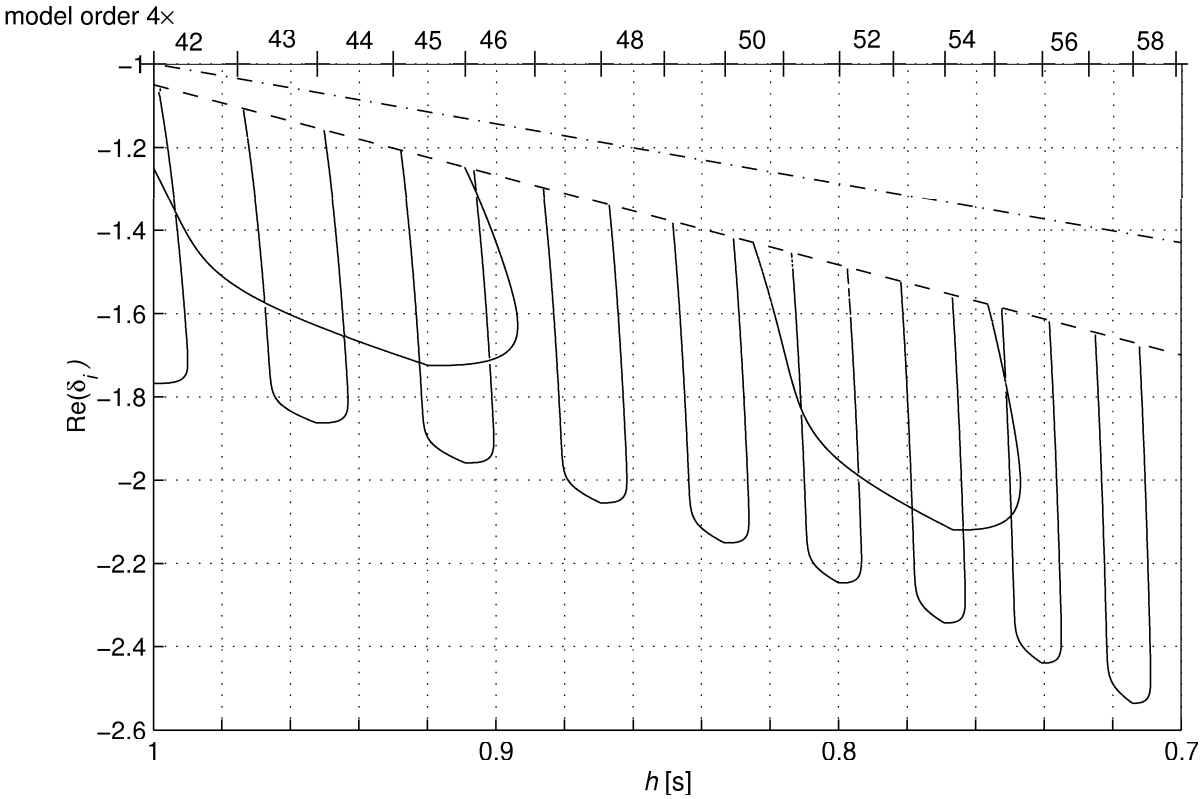

Fig. 5.32 The process of emerging of the real poles of the δ-model given by increasing the order caused increasing the sampling period h

In fact, the trajectories have the origins at the real axis and they obviously converge to the poles of (5.7). Let us investigate in detail the process of emerging of the new poles, i.e., the new trajectories. The procedure of emerging of the new pole-trajectories as h is getting smaller can be seen in Fig. 5.32. Obviously, the number of poles of the δ-model depends on h. For h = 1s, the order of the δ-model is 164 (considering $\tau_{\max} = \tau_{\mathrm{b}} = 40\mathrm{s}$, n = 4 (n is the order of TDS (5.7)), and according to (1.88) providing the approximation of the right-hand side of the model in form (1.87)). As h decreases, the order of the δ-model, i.e., the number of its poles, depends on $\mathrm{int}(\tau_{\max}/h)$, see Fig. 5.32. If the increment of $\mathrm{int}(\tau_{\max}/h)$ is one (considering the continuous change of h from the starting value h = 1s) the increment of order of the δ-model is four, i.e., four new poles appear in the spectrum of the δ-model. First, these new poles appear in the part of the spectrum that is close to the centre of δ-plane stability region. However, some of these new poles leave this group and become the approximations of the poles of continuous TDS, i.e., generate one of the trajectories seen in Fig. 5.31. The procedure of such a process is seen in Fig. 5.32. In this figure, we can see the process of emerging the real poles from the primary group of poles (group of poles that is close to the centre of δ-plane stability region, the position of the centre of δ-plane stability region is marked by the dash-dotted line and the boundary of the group is marked by the dashed line) as h decreases. In agreement with Fig. 5.18, for h = 1s, there are two real poles located to the left from the border of the primary group of poles in Fig. 5.32. Let us look at the behaviour of these and other real poles as h is being decreased. One of these poles is moving to the left and the other, which is more distant from the δ-plane origin, is moving slightly to the right as h decreases. The third real root emerges from the primary group of poles as the order of the δ-model increase to 168, i.e., as h becomes less than 1s. As can be seen, this pole is moving to the left until its position is identical with another real pole in the rightmost point of the

trajectory seen in Fig. 5.32. In such rightmost points of the real pole trajectories, the poles become complex conjugate pairs and further follows one of the trajectories seen in Fig. 5.31. As can be seen in Fig. 5.32, another real pole emerges from the group of primary poles as the order of δ-model increase to 172. This pole is moving first to the left and then slightly back to the right. Finally the pole acquires the same position as the pole that emerged from the primary group of poles as the order of δ-model changed to 176. The poles leave the real axis as the complex conjugate pair and follow another trajectory seen in Fig. 5.31. As can be seen in Fig. 5.32, in this way, eleven couples of poles have left the real axis to become approximations of the poles of TDS as h changed from 1s to 0.7s.

In this section, I have demonstrated the characteristic feature of δ-model, i.e., the convergence of the spectrum to the spectrum of the original continuous TDS if $h \to 0$. As can be seen in Fig. 5.31, the accuracy of the approximation is the best for the poles located close to the s-plane origin, i.e., for the poles that are likely to be the most important in the system dynamics. As can be seen, the poles of the δ-model move continuously as h is continuously changed and their trajectories really converge to the poles of original TDS (5.7). The advantage of using δ-model is given by the fact that the poles of the δ-model that are closest to the complex plane origin may be considered as the direct approximations of the poles of the original TDS. The accuracy of the approximation increases as h decreases. From the practical point of view, there is a certain limit for decreasing h because the value of h determines the order of the δ-model. For example the ends of the trajectories seen in Fig. 5.31 correspond to the poles of the δ-model for h = 0.05s, providing its order equal to 3204. It is more reasonable not to use such a short h because a quite good approximation of the poles is obtained even for higher values of h. If it is necessary to obtain better approximation of the poles, the poles of δ-model may be transformed into s-plane by (1.122).

5.5 State variable feedback control applied to laboratory heating system

In this section I am going to apply the state variable feedback control method to perform the control of the output of system (5.7), i.e., the temperature $\vartheta_c(t)$. In order to achieve tracking the set-point value $\vartheta_{c,set}(t)$ by the controlled variable $\vartheta_c(t)$ with zero control error in the steady state of the control system, let additional state equation be introduced

$$\frac{dI(t)}{dt} = \Delta\vartheta_{c,set}(t) - \Delta\vartheta_c(t) \tag{5.14}$$

which is in fact the integration part of the feedback controller used to eliminate the control error $e(t) = \vartheta_{c,set}(t) - \vartheta_c(t)$ in the steady state. Thus the system matrices change into

$$\mathbf{A}(s) = \begin{bmatrix} \frac{-\exp(-\eta_h s)}{T_h} & \frac{K_b \exp(-\tau_b s)}{T_h} & 0 & 0 & 0 \\ \frac{K_a}{T_a} & \frac{-(1+0.5K_a(1+q))}{T_a} & 0 & \frac{(1-0.5K_a(1-q))\exp(-\tau_e s)}{T_a} & 0 \\ 0 & \frac{K_d \exp(-\tau_d s)}{T_d} & \frac{-1}{T_d} & 0 & 0 \\ 0 & 0 & \frac{K_c \exp(-\tau_c s)}{T_c} & \frac{-\exp(-\eta_c s)}{T_c} & 0 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

$$\mathbf{B}(s) = \left[\frac{K_\mathrm{u} \exp(-\tau_\mathrm{u})}{T_\mathrm{h}} \quad 0 \quad 0 \quad 0 \quad 0 \right]^\mathrm{T} \tag{5.15}$$

and the vector of the state variables is extended into

$$\mathbf{x}(s) = \left[\Delta\vartheta_\mathrm{h}(s) \quad \Delta\vartheta_\mathrm{a}(s) \quad \Delta\vartheta_\mathrm{d}(s) \quad \Delta\vartheta_\mathrm{c}(s) \quad I(s) \right]^T$$

Let the control low be considered in the form of the proportional feedback loops from the state variables, i.e.,

$$\Delta\vartheta_{\mathrm{h,set}}(t) = -[K_1 \quad K_2 \quad K_3 \quad K_4 \quad K_5][\Delta\vartheta_\mathrm{h}(t) \quad \Delta\vartheta_\mathrm{a}(t) \quad \Delta\vartheta_\mathrm{d}(t) \quad \Delta\vartheta_\mathrm{c}(t) \quad I(t)]^\mathrm{T} \tag{5.16}$$

The scheme of the whole controlled system (the dynamics of the slave control loop is already involved in model of system (5.7)) is seen in Fig. 5.33

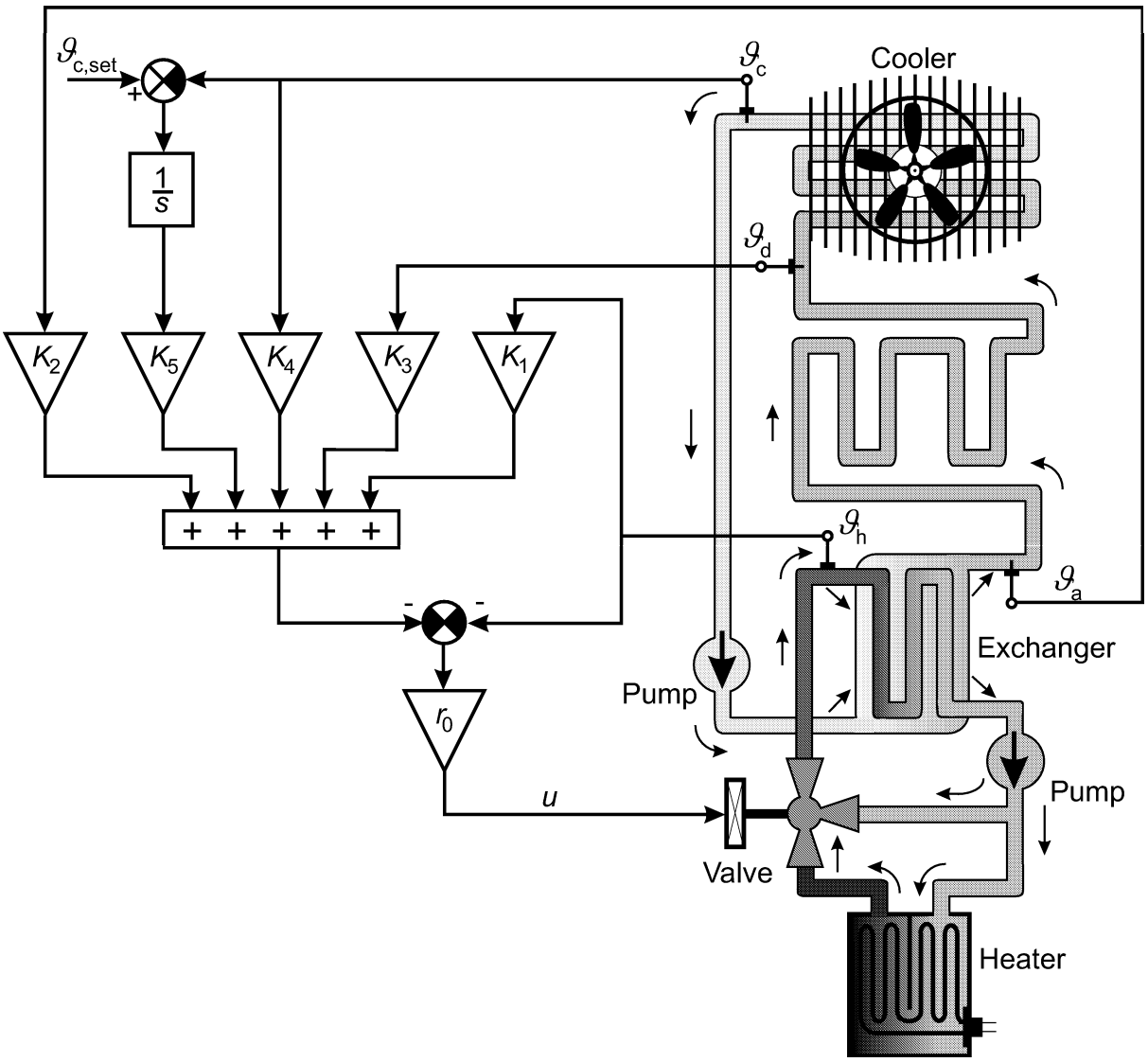

Fig. 5.33 Scheme of the state variable feedback control of the laboratory heating system

As the first step in designing the values of the feedback parameters, let the continuous pole placement method given by Algorithm 4.1 be applied. The method gets the poles close to

the minimum of $\sup(\mathrm{Re}(\lambda_i)), i = 1..\infty$ which is good starting position for direct pole placement method. Introducing the additional state equation (5.14) does not change much the original spectrum. Only one more pole $\lambda_{35} = 0$ appear in the spectrum while the positions of the other poles (the poles of original system (5.7)) remain unchanged. It is given by the fact that the whole system described by matrices (5.15) is given by serial connection of models (5.7) and (5.14). Thus the spectrum of the system given by matrices (5.15) is given us the union of both the spectra. The spectrum of the poles of the system with matrices (5.15) is seen in Fig. 5.34.

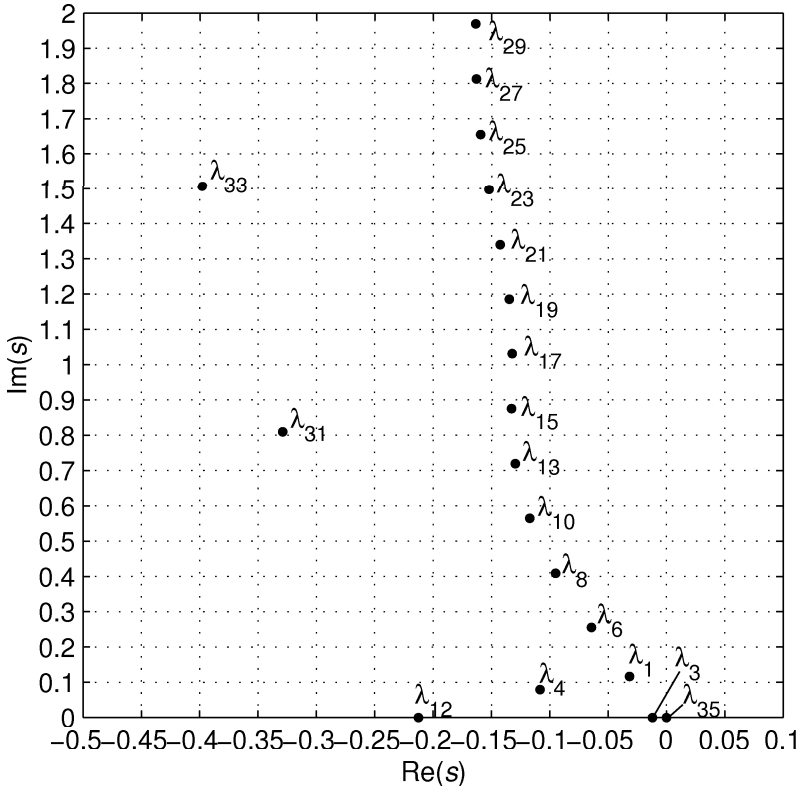

Fig. 5.34 Poles of the system given by matrices (5.15)

Obviously, the prime task of the pole placement procedure is to move the pole $\lambda_{35} = 0\mathrm{s}^{-1}$ from the stability boundary to the left. In Fig. 5.35 we can see the evolution of the real parts of the poles closest to the *s*-plane origin during the continuous pole placement procedure given by Algorithm 4.1 performed on the region $\mathcal{D} = [-0.5, 0.1] \times [0, 2]$. In Fig. 5.36 we can see the evolution of the feedback gains during the procedure. As can be seen, until iteration 5, only the pole λ_{35} is being shifted to the left. From this iteration also the pole λ_3 is controlled. At iteration 48, the group of poles that are continuously shifted to the left is enlarged by involving the poles λ_{12} and $\lambda_{1,2}$. The whole group of these poles is further continuously shifted to the left until iteration 139 at which the procedure stops. At this iteration the group gets close to the poles $\lambda_{6,7}$, $\lambda_{8,9}$ and λ_4 which can not be all controlled using only five feedback gain parameters. The final values of the feedback coefficient gains are $\mathbf{K} = [0.245 \quad 1.101 \quad 3.181 \quad 3.348 \quad -0.119]$. Using the coefficients resulting from the continuos pole placement procedure, the significant part of the spectrum of the feedback system is $\lambda_{35} = -0.0413\mathrm{s}^{-1}$, $\lambda_3 = -0.0502\mathrm{s}^{-1}$, $\lambda_{12} = -0.0594\mathrm{s}^{-1}$, $\lambda_{1,2} = -0.0683 \pm 0.1278\mathrm{j}\ \mathrm{s}^{-1}$,

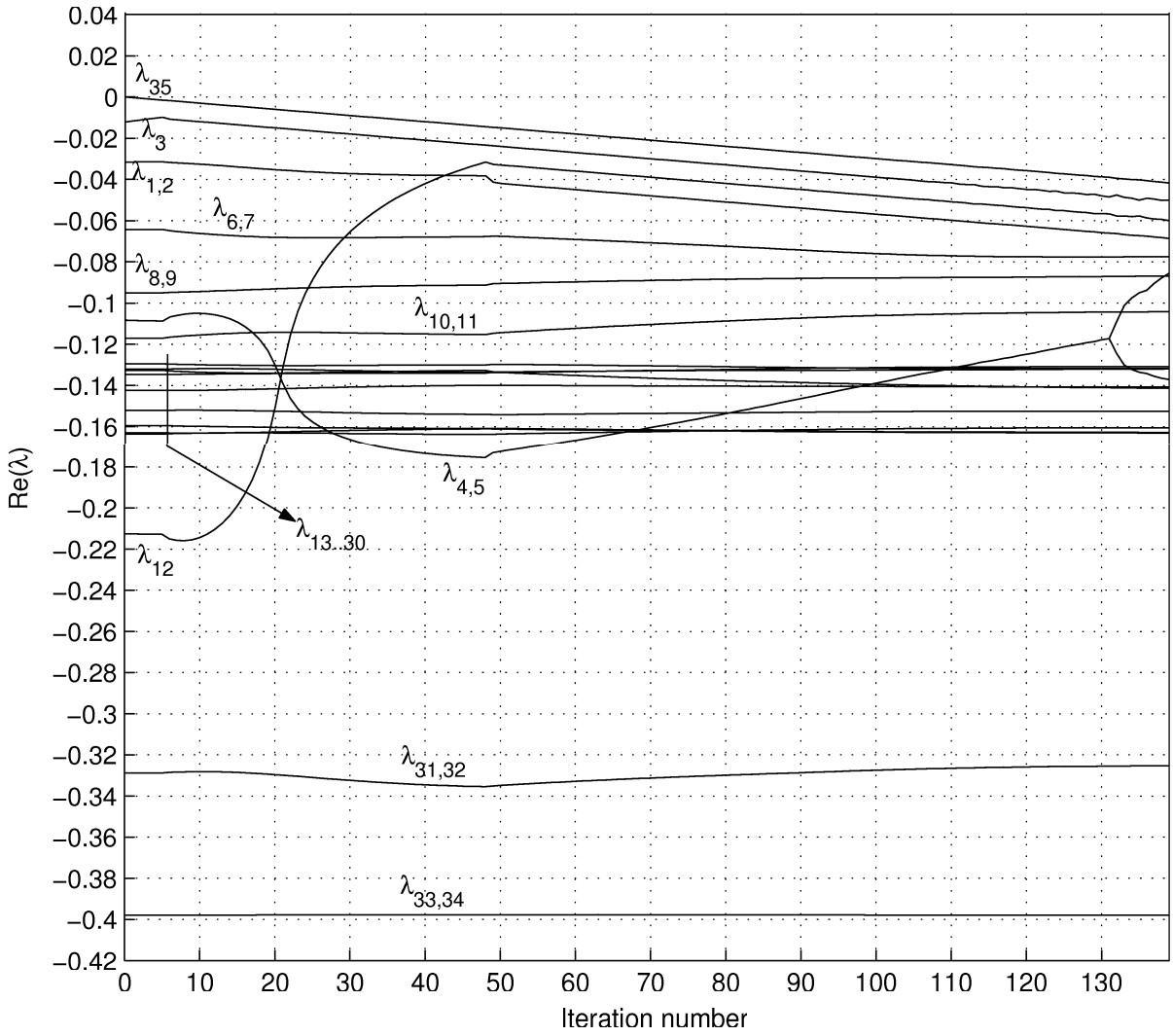

Fig. 5.35 The evolution of the real parts of the poles during the continuous pole placement given by Algorithm 4.1 applied to laboratory plant model given by matrices (5.15)

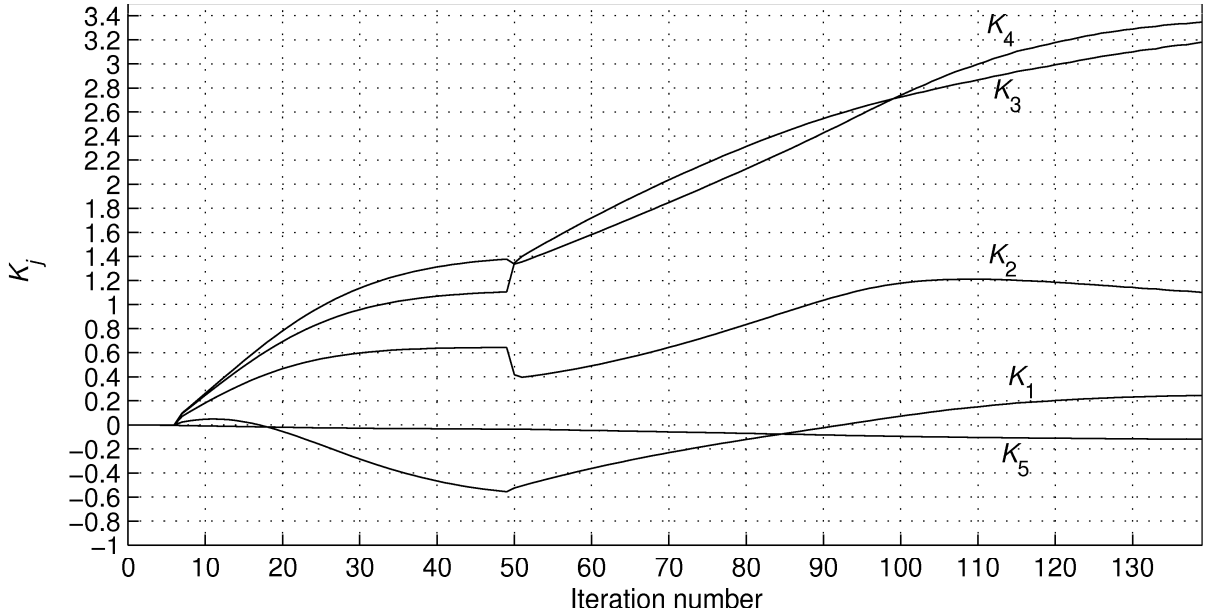

Fig. 5.36 The evolution of the feedback gains during the continuous pole placement

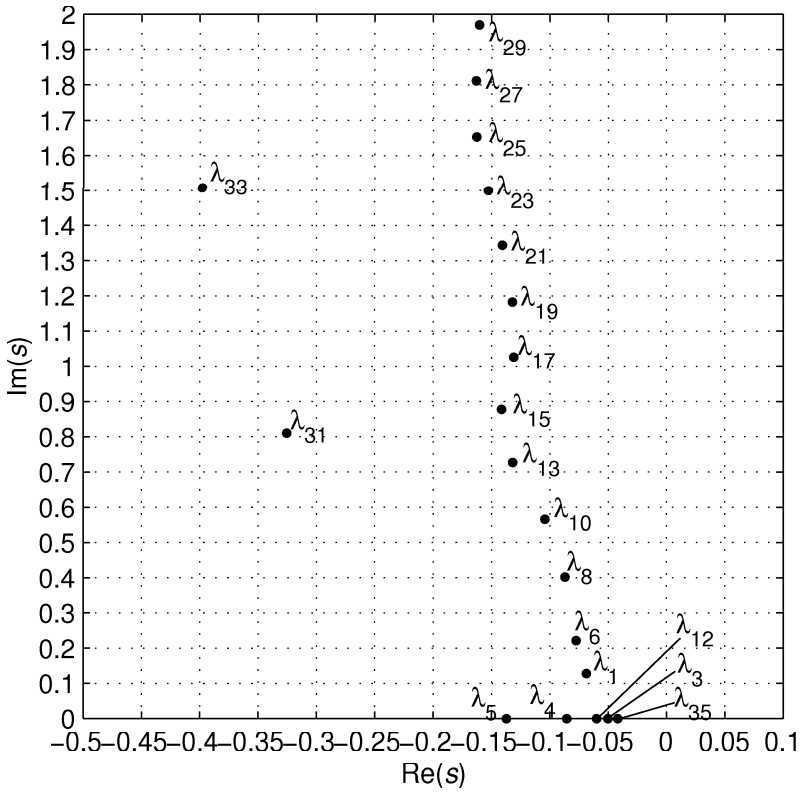

Fig. 5.37 Poles of the closed feedback system with the feedback coefficients resulted from the continuous pole placement

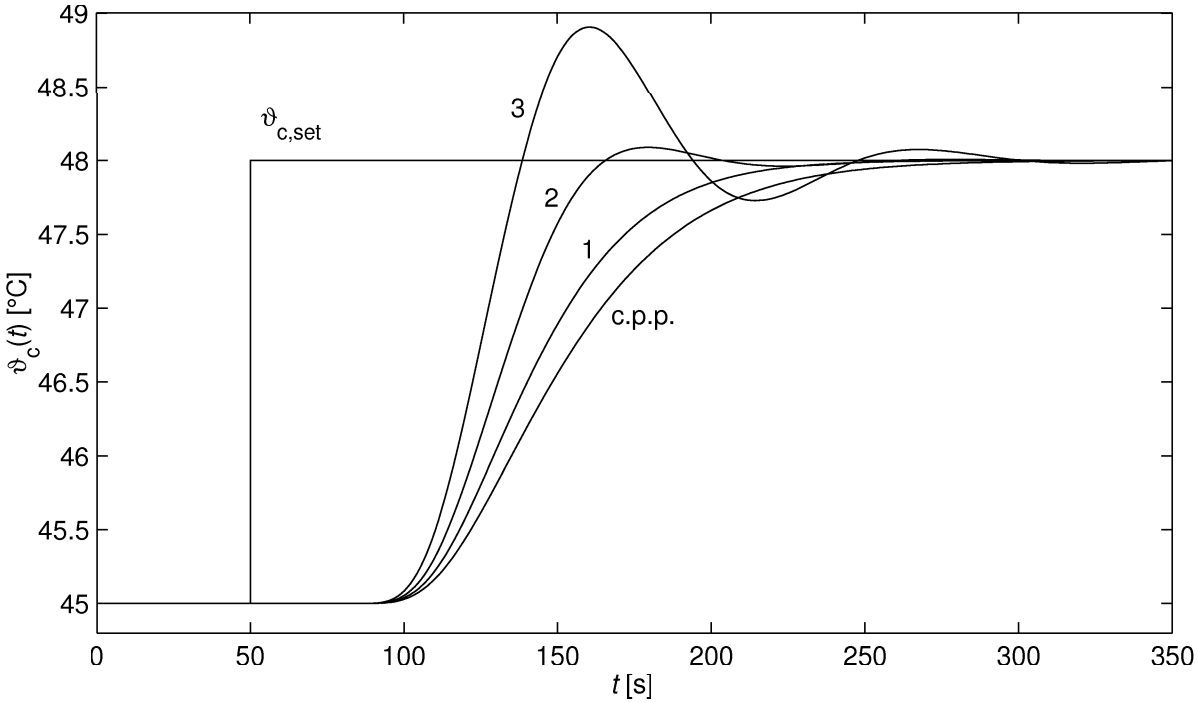

Fig. 5.38 Comparison of the set-point responses of the laboratory system with the feedback from the state variables with various settings of the feedback coefficients, c.p.p. - the feedback coefficients resulted from the continuous pole placement, k = 1, 2, 3, the feedback coefficient settings according to Tab. 5.4

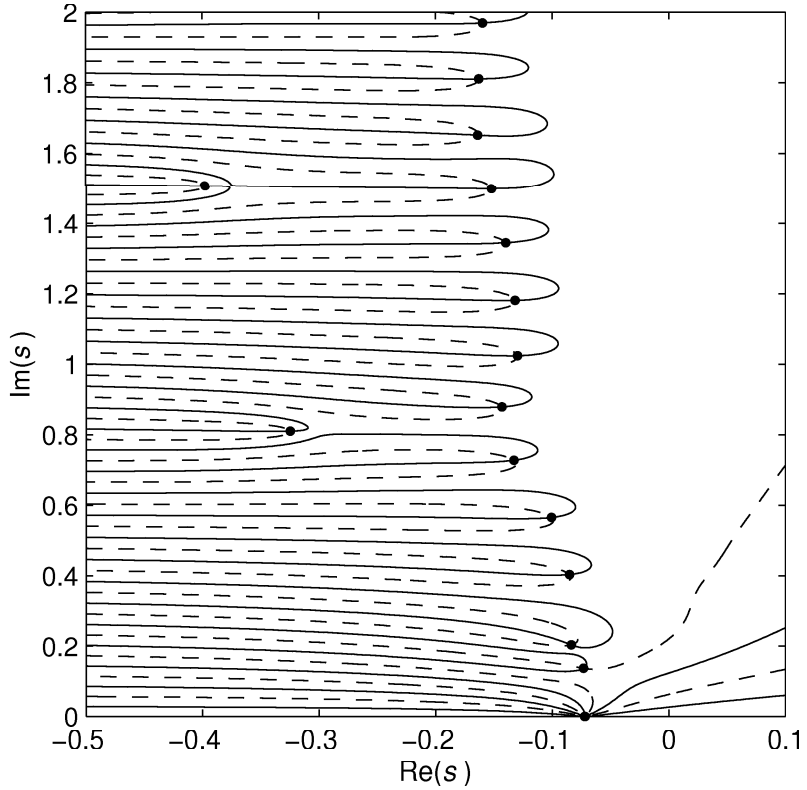

Fig. 5.39 Poles of the closed feedback system with the feedback coefficients seen in Tab. 5.4, $k = 1$

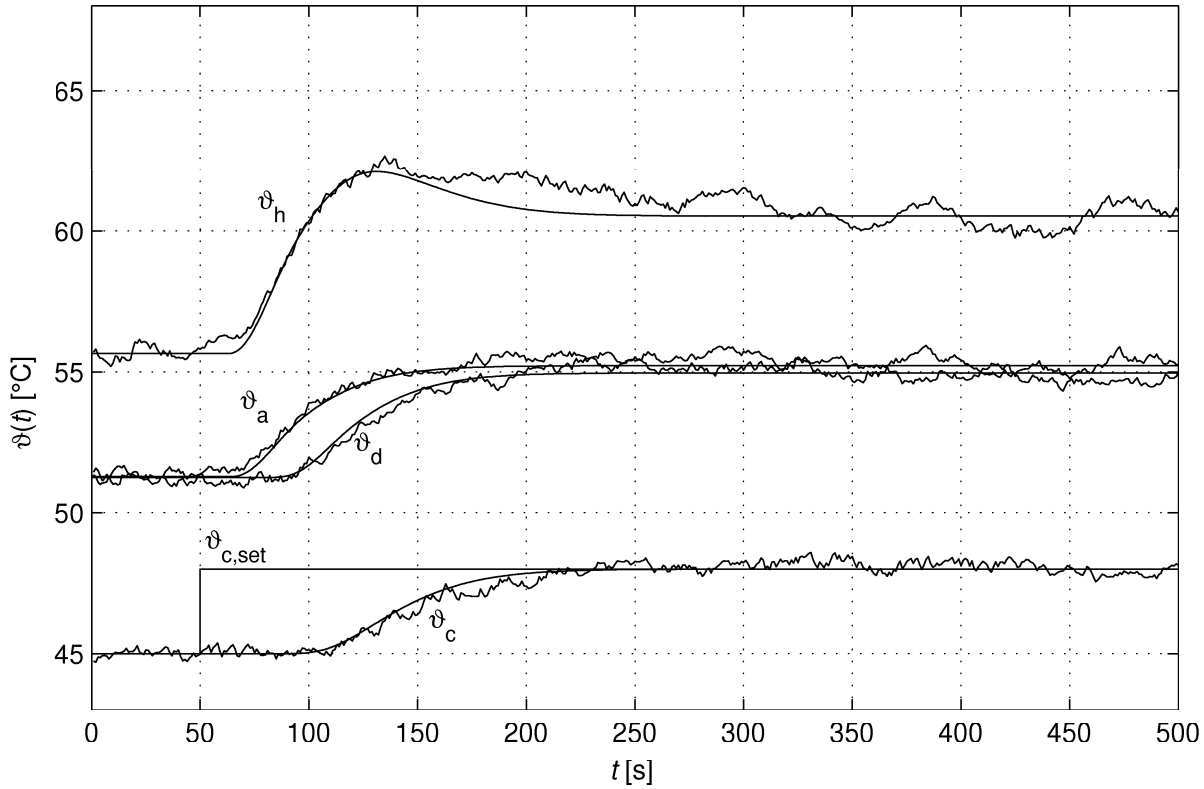

Fig. 5.40 Comparison of the simulated (smooth) and the measured (influenced by noise) set-point responses of the laboratory heating system feedback coefficients seen in Tab. 5.4, $k = 1$

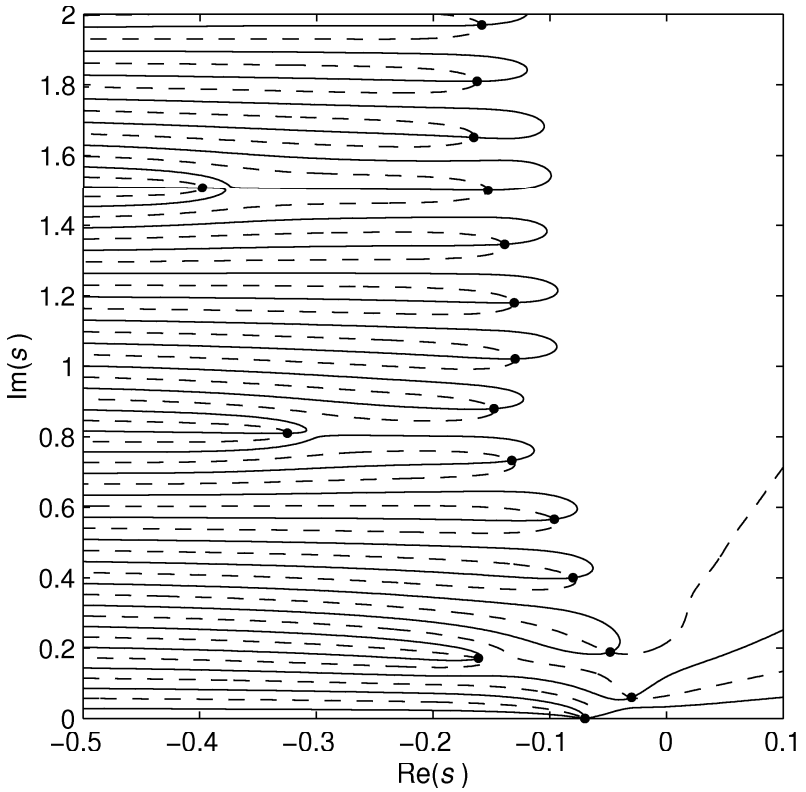

Fig. 5.41 Poles of the closed feedback system with the feedback coefficients seen in Tab. 5.4, $k = 2$

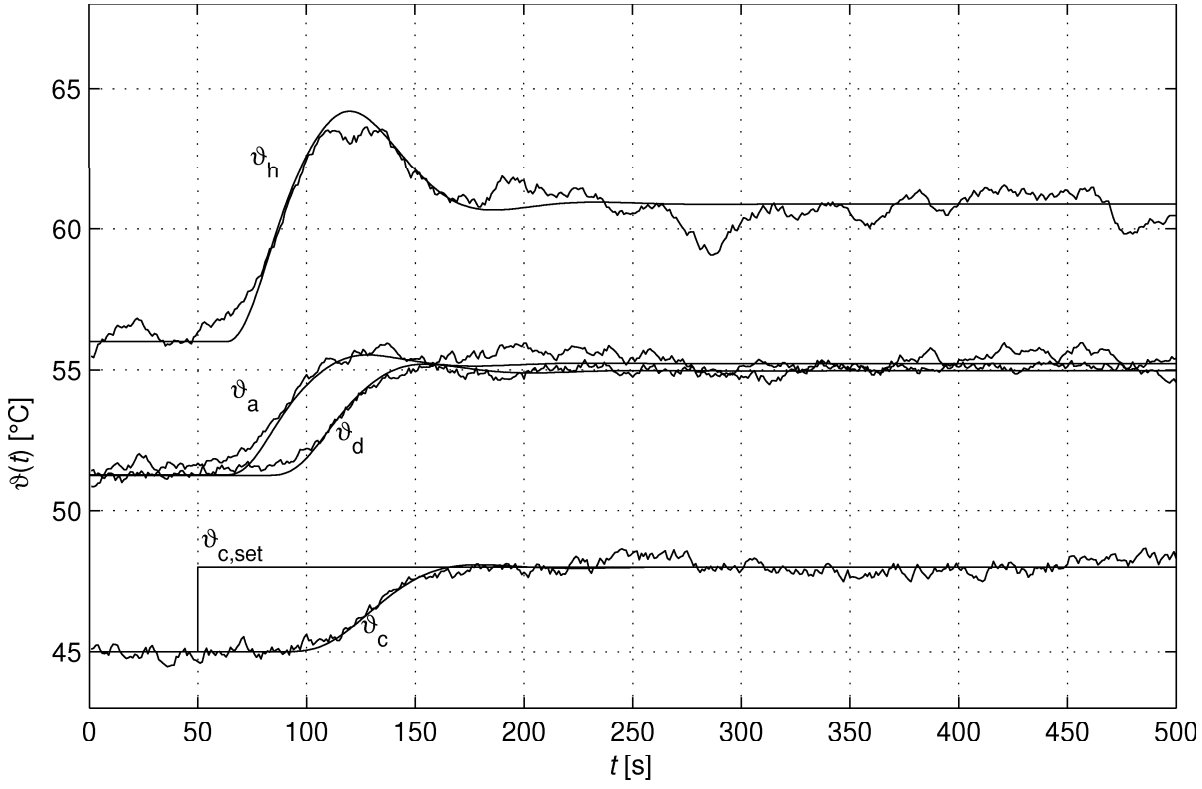

Fig. 5.42 Comparison of the simulated (smooth) and the measured (influenced by noise) set-point responses of the laboratory heating system, feedback coefficients seen in Tab. 5.4, $k = 2$

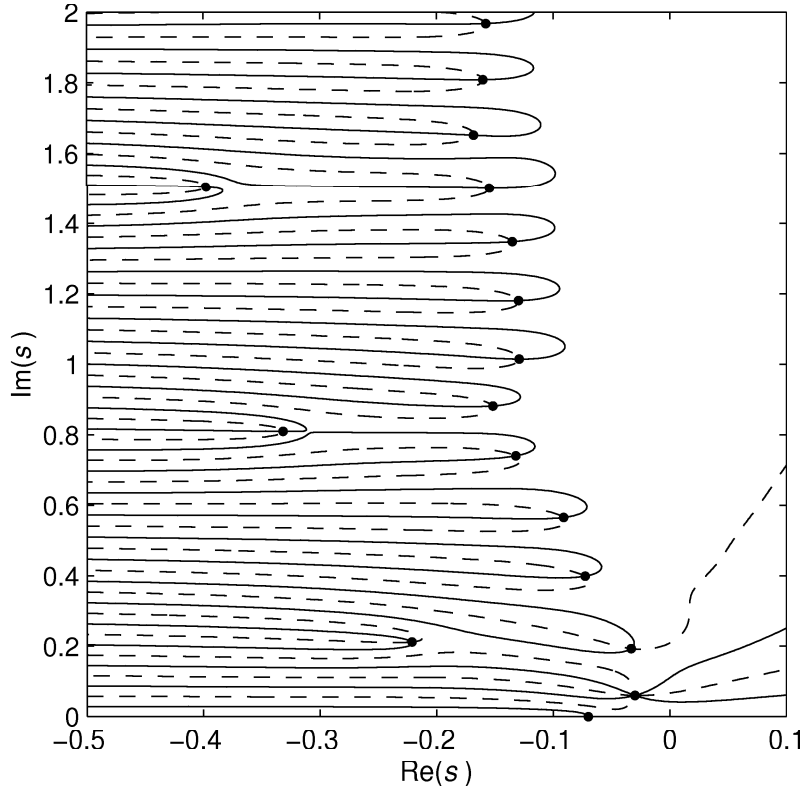

Fig. 5.43 Poles of the closed feedback system with the feedback coefficients seen in Tab. 5.4, $k = 3$

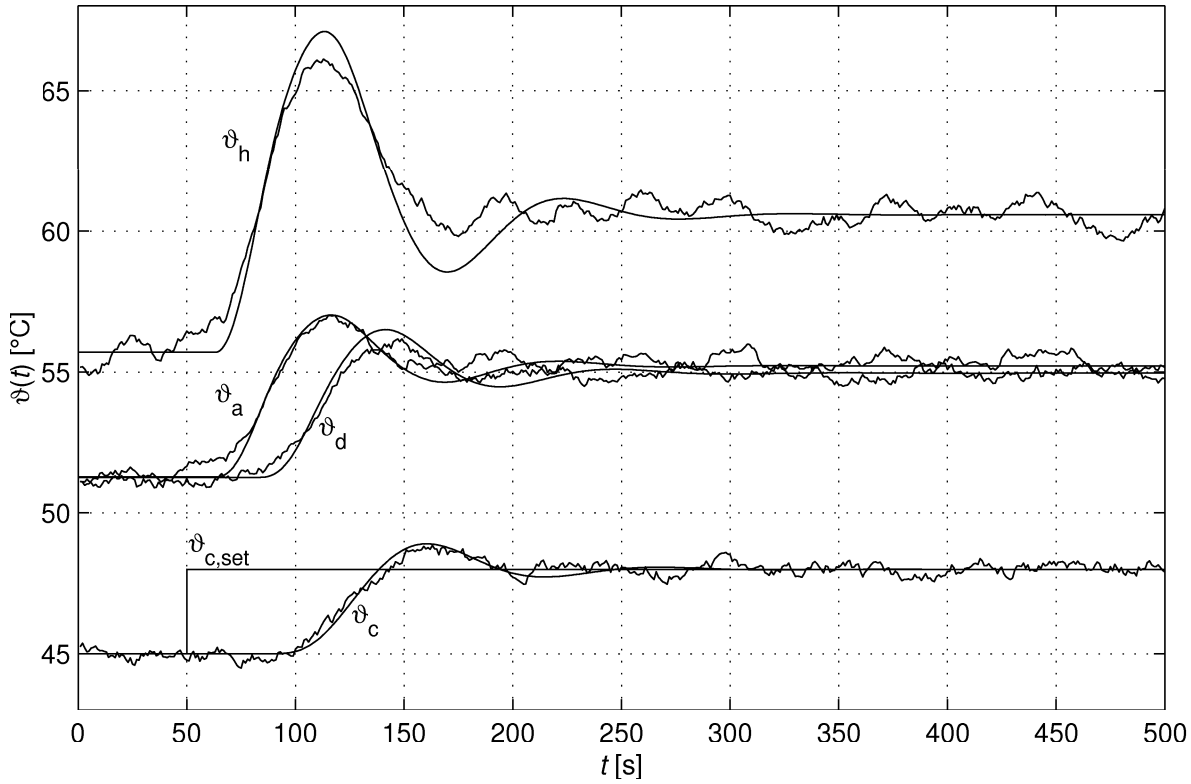

Fig. 5.44 Comparison of the simulated (smooth) and the measured (influenced by noise) set-point responses of the laboratory heating system, feedback coefficients seen in Tab. 5.4, $k = 3$

Tab. 5.4 Sets of prescribed system poles and the resultant feedback gains

k	Prescribed roots [s^{-1}]	$\mathbf{K}_k$
1	5×-0.072	$[0.277 \quad 2.028 \quad 3.841 \quad 3.871 \quad -0.159]$
2	$3\times -0.07, -0.03+0.06j$	$[0.263 \quad 2.230 \quad 5.523 \quad 4.513 \quad -0.225]$
3	$-0.07, 2\times(-0.03+0.06j)$	$[-0.298 \quad 4.091 \quad 7.690 \quad 4.576 \quad -0.352]$
	c.p.p	$[0.245 \quad 1.101 \quad 3.181 \quad 3.348 \quad -0.119]$

$\lambda_{6.7} = -0.0776 \pm 0.2212j\,s^{-1}$, $\lambda_{8,9} = -0.0869 \pm 0.4022j\,s^{-1}$, $\lambda_4 = -0.0878 s^{-1}$, $\lambda_5 = -0.1365 s^{-1}$, see Fig. 5.35. Because of the distances between the neighbouring poles being controlled (kept to ensure the robust numerical computation) the $\sup(\mathrm{Re}(\lambda_i)) = -0.0413 s^{-1}, i = 1..\infty$ is not the absolute minimum of $\sup(\mathrm{Re}(\lambda_i)), i = 1..\infty$ which can be achieved by means of the coefficient feedback from the state variables. However, supposedly, the actual pole distribution is quite close to the supremum.

On the basis of the pole distribution achieved using continuous pole placement procedure, let us apply the direct pole placement method to further modify the feedback system dynamics. Prescribing $\sigma_{1..5} = -0.072\,s^{-1}$ in (4.72), we obtain feedback gains $\mathbf{K} = [0.277 \quad 2.028 \quad 3.841 \quad 3.871 \quad -0.159]$ which results in the spectrum of the poles seen in Fig. 5.39. Let us note that σ_i denotes the prescribed spectrum in the direct pole placement while λ_i is the spectrum being modified to improve the system dynamics (in this example, it is the spectrum resulted from the continuous pole placement). As can be seen, prescribing the multiple pole, $\sup(\mathrm{Re}(\sigma_i)) = -0.072 s^{-1}, i = 1..\infty$ has been considerable shifted to the left with respect to the original $\sup(\mathrm{Re}(\lambda_i)) = -0.0413 s^{-1}, i = 1..\infty$. In Fig. 3.38 we can see the comparison of the set-point responses of the closed feedback system with several settings of the feedback coeffitients. As can be seen, the feedback gain settings, i.e., the result of continuous pole placement and the result of prescribing $\sigma_{1..5} = -0.072\,s^{-1}$, result in overdamped set point responses. The other two responses in Fig. 5.38 (which are more favourable considering the speed of the control error rejection as the evaluating criterion) result if also the complex conjugate pairs are prescribed to the closed loop system dynamics. The response in Fig. 5.38 marked by 2 is the set-point response of the closed system with feedback coefficient $\mathbf{K} = [0.263 \quad 2.230 \quad 5.523 \quad 4.513 \quad -0.225]$ resulted if the poles $\sigma_{1..3} = -0.07\,s^{-1}$ and $\sigma_4 = -0.03 + 0.06j\,s^{-1}$ are prescribed. The corresponding spectrum of the other poles that are close to the *s*-plane origin are seen in Fig. 5.41. The response in Fig. 5.38 marked by 3 is the set-point response of the closed system with feedback coefficient $\mathbf{K} = [-0.298 \quad 4.091 \quad 7.690 \quad 4.576 \quad -0.352]$ resulted if the poles $\sigma_{1..2} = -0.07\,s^{-1}$ $\sigma_3 = -0.03 + 0.06j\,s^{-1}$ and $\sigma_4 = -0.03 + 0.06j\,s^{-1}$ are prescribed. The corresponding spectrum of the other poles close to the *s*-plane origin are seen in Fig. 5.43. As can be seen in Fig. 5.38, prescribing the complex conjugate poles closer to the imaginary axis than the real poles being prescribed considerably accelerate the set point responses. As has been shown, using continuous pole placement method to shift the poles as much to the left as possible is convenient. Even though the feedback system dynamics given as the result of the continuous pole placement is not optimal as a rule, it provides very good starting pole distribution for applying direct pole placement. It would be very difficult to achieve the adequately good

result starting the direct prescription of the poles from the pole distribution of the system without the feedback, seen in Fig. 5.34.

In Fig. 5.40, Fig. 5.42 and Fig. 5.44, we can see the comparisons of the simulated set-point responses of the model with the set-point responses performed and measured on the laboratory heating system with the coefficient feedback gains designed. Even though the parameters of the model of the heating system have been identified only on the basis of one measured step response performed in the operational point, see Fig. 5.2, the set-point responses performed in the vicinity of the operational point are very close to the simulated set-point responses. As can be seen in these figures, not only the dynamics of the system output temperature is modelled well, but also the responses of the other temperatures show very good equivalence between the model and the system dynamics in the vicinity of the operational point in which vicinity the linear plant anisochronic model is valid.

5.6 Summary

In this part of the thesis I have shown the advantages of using the delays in modelling the systems with distributed parameters and the transportation phenomenon involved. The dynamics of the laboratory system is described by fourth order anisochronic model. Each of the state variables is the temperature measured on the laboratory system. Thus the model provides the state variables that are the available physical quantities. This fact considerably simplifies the application of state feedback control because no observer is needed to estimate the state variables. Even though the system is infinite dimensional, i.e., it has infinitely many poles, it has been shown in section 5.2 that the dynamics of the system is determined by three poles closest to the *s*-plane origin. It has also been shown, that the dynamics of the anisochronic can be adequately described by fourteenth order delay free model given as the sum of the first fourteen transfer functions of (5.12). In secion 5.3, the numerical methods for computing the rightmost poles based on discretization of the infinitesimal generator of the semigroup and the solution operator are applied to the system model. The resultant spectra are compared with the results achieved using the quasipolynomial mapping based rootfinder designed in the thesis. In section 5.4, the methods of continuous and direct pole placement are combined in designing the state variable feedback control. Three of the resultant feedback settings are applied to the laboratory system and the measured set-point responses are compared with the simulated responses.

Remark: Note that for computing the eigenvalues of the large matrices, the Matlab function *eig*, which uses the subroutines of Lapack, see Anderson, et al., (1999), has been used.

6. SUMMARY OF CONTRIBUTIONS, CONCLUSIONS AND FURTHER DIRECTIONS

This thesis provides several contributions to the analysis and control synthesis of linear time delay systems (TDS). The first contribution consists in the coherent overview of the methods used to approximate the rightmost poles of the retarded systems. In sections 1.2.7 - 1.2.9, the methods available in the various papers have been adapted to the unified description of TDS with more than one lumped delays in the state variables (some of the methods have been originally designed only for a class of retarded systems, e.g., for systems with single delay). As the numerical methods used for the approximation I have chosen basic linear multi-step methods, i.e., Euler methods (both implicit and explicit) and Trapezoidal method. As the representation of the advanced method used for discretizing the TDS the stiffly acurate method Runge - Kutta, Radau IIA is presented. The methods were used to discretize both the solution operator and the infinitesimal generator of the semigroup. In the last chapter of this thesis, in section 5.3, the methods were used to approximate the rightmost poles of the model of the laboratory heating system. The accuracy of the pole approximations were studied revealing the potentials in approximating the poles of TDS.

6.1 Algorithms for computing quasipolynomial roots

The **first objective** of this thesis, i.e., the **design of the algorithm for computing the roots of the quasipolynomials**, has been solved in chapter 3, where I have introduced two algorithms for computing the roots of quasipolynomials. The **first algorithm** is based on the extension of **Weyl's construction** with the **argument principle** used to compute the number of poles in a particular suspect region of the complex plane (the region in which we suspect some roots being located). For the evaluation of the argument increment along the boundaries of the suspect regions, the graphical based evaluation is used. Although the algorithm designed is quite reliable, its application may be rather cumbersome if the number of the roots in the suspect region is large. Therefore, in section 3.4, I have designed an alternative **rootfinding algorithm based on the mapping of the quasipolynomial function**. The algorithm is original, based on mapping the zero-level contours of the real and imaginary parts of the quasipolynomial and locating the intersection points of the contours. In the first part of section 3.4, the idea of the algorithm is explained. In section 3.4.1, some of the algorithms for **contour plotting** are outlined. For the practical realization of the mapping based rootfinder, the function *contour* available in Matlab has been chosen. In section 3.4.1, the algorithm used in function *contour* is analyzed. In section 3.4.2, the algorithm for automatic location of the intersection points of the contours is introduced. The very positive feature of the mapping based rootfinder is the capability to deal with the multiple roots in the same way as with the single roots. The only problem which occurs in the process of locating the multiple roots is the mismatching of the contours close to the point corresponding to a multiple root, which may result in evaluating incorrect multiplicity of the root. This inconvenient feature could be eliminated using more advanced method for mapping the contours that involve and perform the requirement so that the contours were smooth. To sum up, the algorithm is more powerful than Weyl's construction based algorithm. The mapping based rootfinder may be used to compute the roots of **polynomials**, **quasipolynomials** and the **exponential polynomials** which implies that the algorithm can be used to compute poles and zeros of both **retarded** and **neutral** systems. Moreover, also the essential spectrum of the neutral system can be computed. The algorithm can be used to locate the roots in an arbitrarily placed region in the complex plane (not only the rightmost roots may be computed). The accuracy of the root approximations does not decay with the increasing magnitudes of the

poles. Such a rather broad applicability of the algorithm is one of its key merits. Moreover, no special form of the function being analyzed is required. Another merit of the algorithm is provided by the possibility to check the positions of the roots visually (roots are given as the intersection points of the contours). The most important drawback of the mapping based rootfinder is the incapability to deal with the ill-conditioned (quasi)polynomials. This problem has been discussed in section 3.4.4. However, this feature is the inherent feature of the non-iterative polynomial rootfinding algorithms. In my experience, the applicability of the mapping based rootfinder is comparable with the algorithm used in Matlab function *roots*. The incapability of dealing with the ill-conditioned (quasi)polynomials restricts the applicability of the mapping based rootfinder to the **low degree (quasi)polynomials** (let us say up to $n<20$). It is due to the fact that the higher degree (quasi)polynomials are likely to be ill-conditioned. On the other hand, note that the anisochronic approach provides models with low degree characteristic functions for which the mapping based rootfinder gives good results. The crucial issue of the implementation of the mapping based rootfinder is the suitably chosen step (increment) of the grid. This problem has been discussed in section 3.4.5. The step (increment) of the grid should be chosen according to the requirement for the mapped contours being smooth. The step of the grid also determines the accuracy of the pole approximations. In order to enhance considerably the accuracy of the pole approximations, **Newton's method** is used for each of the poles located in the suspect region. In section 3.4.6, the algorithm is summarized in Algorithm 3.1 and the aspects of the practical implementation (performed in Matlab) are discussed. The algorithm is realized as the command function *aroots*. Most of the algorithms for analysis and control synthesis of TDS presented in this thesis are build on this mapping based rootfinder.

6.2 Features of low order anisochronic models

The other objectives of this thesis stated in chapter 2 are solved in chapter 4. First, according to the **second objective** of this thesis, the features of the **first order anisochronic model** are investigated. Especially, the potentials of approximating the dominant poles using the first order anisochronic model with delay in denominator are studied. The results achieved in section 4.2.3 show that two parameters of the denominator, i.e., time constant and time delay, allows the **dominant couple of poles** to be placed arbitrarily in the complex plane. Taking into account that the system dead time may be approximated by the numerator delay, the first order anisochronic model may be used to approximate the dynamics of the plants conventionally described by considerably higher order delay-free models. Besides the investigation of the features of the first order anisochronic model, also the identification method for assessing the parameters of the model is introduced. The identification method used is the modification of the well-known method of Åström and Hägglund based on the critical parameters of the closed loop system obtained from the relay feedback test. I have chosen this method for identifying the parameters of the anisochronic first order model because the method has proved to be efficient in real plant applications. In section 4.2.4, the first order anisochronic model is further extended to approximate also the effect of the **dominant zeros**. The extension is performed by involving an exponential polynomial in the numerator of the transfer function of the anisochronic first order model. By means of the parameters of the exponential polynomial, the dominant zeros may be placed arbitrarily in the complex plane. Using this model, the dynamics of plants with zero-effect, e. g., non-minimum phase systems can be approximated. The drawback of involving the exponential polynomial in the numerator of the model is given by the fact that besides the dominant zeros being assigned, infinitely many zeros with large imaginary parts (distributed in a vertical strip of the complex plane) are introduced into the model dynamics. On the other hand, as has been

shown in the examples in sections 4.2.4 and 4.3, the model may be used to approximate quite broad class of plant dynamics. The infinitely many zeros in the vertical strip of the complex plane may bring about difficulties to the control design if it is based on model of the plant. In section 4.3, I have studied the applicability of the anisochronic first order model with the exponential polynomial in the numerator in the **internal model control** (IMC) design. Using this control method, the spectrum of the zeros of the numerator exponential polynomial becomes the essential spectrum of the closed loop dynamics, which acquires neutral form. Thus, to design a robust closed loop system dynamics, also the features of the essential spectrum have to be investigated in order to achieve robustly stable closed loop dynamics. In section 4.3.2, the **robust IMC design** based on the first order anisochronic model with the exponential polynomial in the numerator has been introduced. This result has been achieved beyond the framework of the objectives of this thesis. However, in this control design example, it has been shown that the specific features of TDS have to be taken into account in the control design, which needs certain modifications as a rule.

6.3 Criterion for evaluating the significance of the poles of TDS

The third objective of this thesis deals with the basic feature of TDS - with the infinite spectrum of TDS poles. In section 4.4, it has been shown that only a specific group of poles are significant in the system input-output dynamics. In general, the most important poles of the stable TDS are the poles closest to the origin of the complex plane. However, their significance is not directly determined by the distances of the poles from the complex plane origin. Also the distribution of zeros of TDS influence the significance of the poles in the dynamics. In fact, the input-output dynamics are determined by the distribution of both the dominant poles and the dominant zeros. In section 4.4, I have designed an original **pole-significance evaluating criterion**. The criterion is based on the **generalized Heaviside expansion** of the input-output transfer function of TDS. First, the poles of the stable TDS that are closest to the complex plane origin, which are likely to be the dynamics determining poles, are computed using the mapping based algorithm. Note that the methods for computing the poles of TDS based on discretization cannot be used because for accomplishing the Heaviside expansion, we need very precise values of the poles. Thus, having computed the poles of TDS and performing the Heaviside expansion (provided that all the poles are single), the transfer functions resulting from the expansion correspond to the modes of TDS. The significance evaluating criterion evaluate the **weighting functions** of the transfer functions resulting from the Heaviside expansion. Particularly, the absolute values of the differences between the maxima and minima of the weighting functions are evaluated. The larger is the value of the criterion, the pole is more significant. Since the weighting function of the whole TDS is given as the superposition of the infinitely many weighting functions corresponding to the transfer functions resulting from the Heaviside expansion, obviously, the criterion is likely to evaluate truly the contribution of a particular mode to the system dynamics. The criterion is transparent, easy to apply and provides reliable evaluation of the significance of the poles if the mutual distances of the poles are not small. If there is a pole close to the pole being evaluated, the significance evaluating criterion acquires inadequately large value as a rule. This feature of the significance evaluating criterion, which has been studied in section 4.4.3, is induced by the physical equivalence of the contribution of the multiple pole and the group of distinct poles that are quite close to the value of the multiple pole. For the multiple poles, the Heaviside expansion formula provides residues equal to infinity, which implies the evaluating criterion being also equal to infinity. In section 4.4.3, it has been shown, that as the distance of two poles tends to zero, the values of evaluating criterion of both these poles tends to infinity. However, if the sum of the weighting functions corresponding to these poles is

evaluated, the criterion truly evaluates the significance of the poles. Thus if there are poles close to each other, they have to be evaluated as the group of poles, i.e., the sum of the corresponding weighting functions is analyzed. In section 4.4.4, I have designed a method to evaluate the **significance of the multiple poles**. The idea of the method is based on the physical equivalence of the multiple poles and the group of single poles that are close to the value of the multiple poles. In section 4.4.5, the criterion is modified in order to evaluate the **global significance of the poles of MIMO TDS**. The idea of this modification consists in ignoring the influence of the zeros to the system dynamics (each transfer function of MIMO TDS has its own spectrum of zeros while the poles are common for all the transfer functions). Thus only the transfer function with the numerator equal to one is expanded into the Heaviside series. Consequently the weighting functions of the transfer functions resulted from the expansion are evaluated. To conclude, the designed criterion for significance evaluation of the poles can be used to select the most significant poles of TDS in order to substitute its dynamics by the finite order approximation involving all the substantial modes of the TDS. However, originally, the method has been designed as the preceding step to the direct pole placement (particularly the method of global pole significance evaluation). Locating the most significant poles of the TDS dynamics provides the possibility to shift these poles in order to improve the system dynamics. The further development of the significance evaluating method based on the Heaviside expansion might consist in using a different criterion in evaluating the weighting functions. The criterion based on difference between the maximum and minimum of the weighting function, which is used in this thesis, provides higher values for the oscillatory modes of the dynamics, i.e., evaluates the oscillatory modes as more important than the adequately significant non-oscillatory modes. Thus, a criterion which evaluates whole the weighting function, e.g., an integration based criterion, might be used to further enhance the reliability of the significance evaluation method.

6.4 Pole placement based control design in TDS

In section 4.5, 4.6 and 4.7, according to the **fourth objective** of this thesis, I have investigated features of the methods for **pole placement** using the **proportional** feedback from the state variables applied to TDS. In section 4.5, the fundamentals of the **gradient based state variable feedback control design** are summarized. The algorithm presented arises from the linearity of the closed loop characteristic function with respect to the feedback gains and can be used for **direct pole placement**. The poles being prescribed may be both real and complex conjugate, either single or multiple. The values of the feedback gain coefficients result as the solution of the set of linear equations. The equations in the set are given by substituting the prescribed poles for the operator s in the closed loop characteristic function (for complex poles, the function has to be split into real and imaginary parts). The maximum number of poles which can be prescribed is restricted by the order of the system n, i.e., by the number of available feedback loops. However, since the Moore-Penrose generalized inversion is used to solve the set of equations, less than n poles may be prescribed to compute values of n feedback gain coefficients. Using the method for direct pole placement, the infinity of the spectrum of TDS has to be taken into consideration. The fact that we can prescribe the position only to n poles, while the other poles (infinitely many) are placed spontaneously, rather restricts the applicability of the method. Therefore, to obtain satisfactory result, the following procedure for pole placement using gradient based method is suggested. First, the poles of the original system are computed using, e.g., mapping based rootfinder. Then, the pole significance evaluating criterion is used to define the dynamics determining poles. The third step of the procedure consists in prescribing the new positions to these most significant poles in order to stabilize or improve the system dynamics. Note that the prescribed shifts of

the poles from their original positions should not be too large in order to preserve the dominance of the prescribed poles. In the fourth step of the procedure, applying the computed values of the feedback gain coefficients, the new spectrum of the feedback system poles has to be checked using the rootfinder. If some of the non-prescribed poles are placed into the undesirable position, the result of the procedure cannot be accepted and the whole procedure has to be repeated with the new values of the prescribed poles. If the dominance of the prescribed poles is preserved after the procedure, the poles may be further shifted if the result achieved is not satisfactory yet. In this way, the pole placement is accomplished in several steps. Even though this pole placement procedure is rather heuristic, it has proved to be efficient in modifying the TDS dynamics. The presented method for direct pole placement can be used for both retarded and neutral system with both lumped and distributed delays.

In section 4.6 I have modified the method of gradient based pole placement so that it might be used in the pole placement method known as **continuous pole placement** (see section 1.3.5). The method is especially suitable for stabilization of TDS. The idea of the method consists in shifting only the real parts of the rightmost poles and monitoring the positions of the other poles. Since only the real parts of the poles are prescribed, the characteristic function looses the linearity with respect to the parameters being computed (besides the feedback coefficients, also the imaginary part of the prescribed pole is computed) if the shifts are prescribed to a complex pole. Therefore, the characteristic function is linearized and the results achieved are accurate enough only if the prescribed shifts are small. The advantageous is the possibility to run the algorithm of the pole shifting automatically. At each step of the continuous pole placement procedure, the rightmost spectrum of the feedback system is computed using the mapping based rootfinder. The distribution of the feedback system poles resulting from the continuous pole placement procedure is close to the minimal supremum of the real parts of the poles. Such a result is convenient from the stability point of view. However dynamics determined by such a distribution of the poles have often undesirable features. Since only the real parts of the poles are controlled, the resultant dynamics may be too oscillatory. On the other hand, if all the rightmost poles are real, the resultant dynamics are overdamped. Therefore, in some cases, it is convenient apply the direct pole placement to further improve the feedback system dynamics having resulted from the continuous pole placement. The algorithm for continuous pole placement (given by Algorithm 4.1) may be applied to retarded systems and also to a class of neutral systems (with strongly stable essential spectrum). It should be noted that if the methods for pole placement based on the coefficient feedback from the state variables are used for stabilization, the class of stabilizable TDS is rather limited. For example TDS with more than n unstable poles and TDS whose rightmost poles are farther to the right from the stability boundary are highly likely not to be stabilizable using the gain feedback. Stabilizing such a TDS acquires a form of functional feedback from the state variables to be used. Also the methods for stabilizing the essential spectrum of the neutral systems should be investigated to extend the class of stabilizable TDS. The coefficient feedback from the state variables is convenient because it can be easily implemented (using. e.g., programmable controllers. However, neither implementation of the functional feedback (if the distribution of the feedback terms is not too complicated) acquires excessive effort. Thus, the further direction in the research dealing with the pole placement applied to TDS should be in designing the methods for robust functional (and easy to implement) feedback control.

6.5 Real plant application example

In the last chapter of this thesis, the dynamics synthesis and control design methods for TDS designed according to the objectives of this thesis are applied to the model of a

laboratory heating system. First the model of the laboratory plant is built using the anisochronic approach. The system is divided into subsystems and each of these subsystems is modeled using the linear first order anisochronic model. This approach provides the favorable feature that the state variables are identical with the measured outputs of each of the subsystems. In section 5.2, the poles of the system model are computed using the mapping based rootfinder. Consequently, the significance of the poles is evaluated using the criterion based on the Heaviside expansion of the system transfer function. In section 5.3, discretization methods for approximating the rightmost poles are compared with the result obtained using the mapping based rootfinder. The results presented reveal that the mapping based rootfinder is much better tool for obtaining the positions of the rightmost poles of lower order TDS (higher order TDS may have ill-conditioned characteristic function which rules out the applicability of the rootfinder). In section 5.4, another unique numerical result has been achieved. It demonstrates the continuity of the trajectories of poles of the δ-model with respect to the sampling period in approximating the poles of the continuous TDS. In section 5.5, the control of laboratory heating system has been designed using the coefficient feedback from the state variables. First, the continuous pole placement method is applied. The resultant dynamics are then further improved (the target of the design is to obtain fast and well-damped dynamics) applying the direct pole placement. Finally, three settings of the feedback gain coefficients are applied to the laboratory heating system and the measured set-point responses are compared with the simulated responses.

The mapping based rootfinder, the algorithms for computing rightmost poles based on discretization, the algorithm for evaluating the pole significance, and both the algorithms for pole placement using proportional feedback from the state variables are available on the CD enclosed, see Appendix 1.

To conclude, the main contribution of this thesis is the designed mapping based rootfinder. As has been shown the rootfinder is a powerful tool for investigation of the spectra of lower order TDS. Particularly, the applicability of the rootfinder to compute the spectra of the neutral systems is a unique result. As has been shown the knowledge of the spectrum of TDS is quite important (more that in case of delay free systems) in analyzing the system dynamics and designing the control. Note that most of the results presented in this thesis have already bee published. To sum up, **all the objectives stated in chapter 2 have been fulfilled**.

APPENDIX 1 - Matlab functions for TDS spectrum assessment and assignment available on CD enclosed

P=aroots(M,D,d,e)

- accomplishes Algorithm 3.1, compute the roots of function ((quasi)polynomial) M ($M(s)$) or eigenvalues of the (functional) matrix M ($\mathbf{A}(s)$) both in symbolic variable s, $D=[\beta_{\min}\ \beta_{\max}\ \omega_{\max}\ \omega_{\max}]$ ($\mathcal{D}=[\beta_{\min},\beta_{\max}]\times[\omega_{\min},\omega_{\max}]$) - suspect region of the complex plane, d (Δ_s) - step (increment) of the grid of nodes on D ($\mathcal{D}$), e (ε_N) - absolute value of the difference of two successive approximations of Newton's method
- P - vector of computed roots in the region D ($\mathcal{D}$)

R=aevalpoles(M,N,P)

- evaluate the significance of the distinct poles using criterion (4.51), M ($M(s)$) - denominator function, N ($N(s)$) - numerator function, both in variable s, P - vector of poles being evaluated
- R.p - poles ordered with respect to significance evaluating criterion (4.51), R.he -values of pole significance evaluating criterion (4.51) based on evaluating the weighting functions corresponding to the poles in R.p, R.r - values of residues corresponding to the poles in R.p

K=acontpp(H,A,B,Km,dp,ndp,D,d,e)

- accomplishes Algorithm 4.1, H ($\sum_{i=1}^{N}\mathbf{H}_i\exp(-s\eta_i)$) - matrix of difference part of neutral system, if system is retarded H=0, A ($\mathbf{A}(s)$) - functional matrix of system dynamics, B ($\mathbf{B}(s)$) - system input functional matrix, all in symbolic variable s. Km=[K_1 K_2 ... K_n]- mask of the feedback (n - order of TDS), if $K_i \neq 0$, feedback from the i^{th} state variable is used, dp ($\Delta\sigma$) - pole shifting increment, product of ndp and dp define the minimum distances (in real parts) of the poles being shifted), $D=[\beta_{\min}\ \beta_{\max}\ \omega_{\max}\ \omega_{\max}]$ - the region on which the continuous pole placement is performed, d (Δ_s) - increment of the grid of nodes on D ($\mathcal{D}$), e (ε_N) - absolute value of the difference of two successive approximations of Newton's method. Function aroots is used to compute the poles in D ($\mathcal{D}$).
- K.g - resultant feedback matrix, K.c - compensation of the feedback system static gain, K.p - values of the final pole distribution on D ($\mathcal{D}$),

K=adirpp(H,A,B,Km,Pp)

- accomplishes direct pole placement according to (4.72), H ($\sum_{i=1}^{N}\mathbf{H}_i\exp(-s\eta_i)$) - matrix of difference part of the neutral system, if the system is retarded, H=0, A ($\mathbf{A}(s)$) - functional matrix of system dynamics, B ($\mathbf{B}(s)$) - system input functional matrix, all in symbolic variable s. Km=[K_1 K_2 ... K_n]- mask of the feedback, if $K_i \neq 0$,

feedback from the i^{th} state variable is used, Pp=[σ_1 σ_2 ... σ_q] - prescribed poles, $q = q_r + 2q_c$, q_r - number of prescribed real poles, q_c - number of prescribed imaginary poles, $q \le n$ (n - order of TDS),

- K.g - resultant feedback matrix, K.c - compensation of the feedback system static gain,

P=apolesso(A,h,method,sr)

- accomplishes approximation of the rightmost eigenvalues of the functional matrix A (**A**(s)) in symbolic variable s (only lumped delays allowed) using the discretization of the solution operator. First, the matrices of form (1.88) are obtained from A for the discretization period h, then matrix **Φ** (see (1.90)) is obtained according to the chosen discretization method, method=1 - Euler explicit (**Φ** given by (1.94)), method=2 - Trapezoidal (**Φ** given by (1.96)), method=3 - Runge-Kutta Radau IIA with stage sr (s), (**Φ** given by (1.104))
- P.d - poles of the discrete system, P.p - approximation of the rightmost poles

P=apolesigs(A,h,method,sr)

- Accomplishes approximation of the rightmost eigenvalues of the functional matrix A (**A**(s)) in symbolic variable s (only lumped delays allowed) using the discretization of the infinitesimal generator of the semigroup. First, the matrices of form (1.88) are obtained from A for the discretization period h, then matrix $\mathcal{A}_h$ (see (1.106)) is obtained according to chosen discretization method, method=1 - Euler explicit ($\mathcal{A}_h$ given by (1.109)), method=2 - Trapezoidal ($\mathcal{A}_h$ given by (1.115)), method=3 - Runge-Kutta Radau IIA with stage sr (s) ($\mathcal{A}_h$ given by (1.118))
- P - approximation of the rightmost poles

Remark 1 The command line functions described use the functions of Symbolic Math Toolbox of Matlab.

Remark 2 The (quasi)polynomials M, N, and the functional matrices H, A, B are in the symbolic variable s which has to be defined before defining the (quasi)polynomials and matrices by the command - syms s

Remark 3 The Laplace transform of lumped delays τ_i are to be inserted in the form $\exp(-\text{tau} * s)$.

Remark 4 Demos explaining further the functions are available on the CD enclosed.

APPENDIX 2 - Technical data of the parts of the laboratory heating system

- Electric heater — Stiebel Eltron SHU 5S, 2 kW, capacity 5 l, max. temperature 85°C
- Multi-plate heat exchanger — Zilmet Z 1/8, 5 plates, heat transfer up to 3 kW, $\Delta\vartheta_{prim.}$=20°C
- Cooler — heating system of Škoda 120, exchanger Sofico, ventilator - nominal performance 4.6 kW, air flow rate 250 m^3 per hour/195 Pa (300 kg per hour)
- Mixing valve — Landis & Gyr, VXG44.15-0.4 (k_{VS}=0.4m^3 per hour)
- Pumps — Zirco - Wilo Z15
- Flow rate control valves — Danfos RTD-N20
- Piping lines — Supersan 15×1, (inner diameter ∅13 mm)

Tab A.1 Length of the piping lines, see Fig. 5.1

piping line	a	b	c	d	e	f	g
length [mm]	2740	570	995	490	400	360	750

REFERENCES

Ackerman, J.E., (1972), Der entwulf linearer regelungs systems im zustandstraum, r*egelungs-technik und prozessdatenverarbeitung*, vol. 7, pp. 297-300, (In German).

Anderson, E., Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, (1999) *LAPACK User's gide*, http://www.netlib.org/lapack/lug/lapack_lug.html, third edition, SIAM, Philadelphia.

Angot, A. (1952). *Compléments de mathématiques a l'usage des ingénieurs de l'electrotechnique et des télécommunications*. Masson, Paris, (In French).

Aramini, M., (1981), Implementation of an improved contour plotting algorithm, MSc. thesis University of Illinois at Urban-Champaign, http://www.ultranet.com/~aramini/ .

Åström, K. J. and Hägglund, T. A., (1984), Automatic tuning of simple regulators with specifications on phase and amplitude margins, *Automatica*, vol. 20, pp. 645-651.

Åström, K. J. and Hägglund, T. A., (1988a), *Automatic tuning of PID controllers*, Instrument Society of America, NC.

Åström, K. J. and Hägglund, T. A., (1998b), New auto-tuning design, *IFAC International Symposium on Adaptive Control of Chemical Processes*, ADCHEM'88, Lyngby, Denmark, pp. 141.

Åström, K., (1977), Frequency domain properties of Otto Smith regulator, *International Journal of Control*, vol. 26, pp. 307-314.

Avelar, C.E. and Hale, J.K., (1980), On the zeros of exponential polynomials, *Mathematical analysis and application*, 73, pp. 434-452.

Bai, Z., Demmel, J., Dongarra, J., Ruhe, A. and van der Vorst, H. - editors, (2000), *Templates for the solution of algebraic eigenvalue problems: A Practical Guide,* SIAM, Philadelphia, (@Book, http://www.cs.utk.edu/~dongarra/etemplates/index.html).

Barrett, R., Berry, M., Chan, T. F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C., Van der Vorst, H. (1994), *Templates for the solution of linear systems: building blocks for iterative methods, 2nd Edition*, SIAM, Philadelphia, PA (@Book, http://www.netlib.org/linalg/html_templates/Templates.html).

Bellen, A. and Maset, S., (2000), Numerical solution of constant coefficient linear delay differential equations as abstract Cauchy problem, *Numerical Mathematics*, vol. 84, pp. 351-374.

Bellman, R., and Cooke, K.L., (1963), *Differential-difference equation*, Academic Press, New York.

Bellman, R., and Dankis, J. M., (1954), *A survey of the mathematical theory of time lag, retarded control, and hereditary processes,* The Rand Corporation, R-256.

Ben-Israel, A. and Greville, T. N. E., (1977), *Generalized inverses: theory and applications*, New York: Wiley.

Bensoussan, A., Da Prato, G., Delfour, M. C., and Mitter, S. K., (1993), *Representation and control of infinite dimensional systems*, *system & control: foundation and application*, 2 volumes, Birkhauser, Boston.

Bini, D. H., (1996), Numerical computation of polynomial zeros by means of Alberth's method. *Numerical Algorithms*, vol. 13, no. 3-4.

Bini, D.H., Fiorentino, G., (1999), Numerical computation of polynomial roots using MP-solve, *manuscript*, available at ftp://ftp.dm.unipi.it/pub/mpsolve/.

Bini, D.H., Fiorentino, G., (2000), Design, analysis and implementation of multiprecision polynomial rootfinder, *Numerical Algorithms*, 23, pp. 127-173.

Breda, D., Maset, S. and Vermiglio, R. (2001). Numerical computation of characteristic roots for delay differential equations, In. *Proc. of 3rd IFAC Workshop on Time delay systems (TDS 2001)*, Santa Fe, New Mexico.

Butcher, J. C., (1987), *The numerical analysis of ordinary differential equations*, John Wiley & Sons.

Campbell, S. L. and Meyer, C. D. Jr., (1991), *Generalized inverses of linear transformations.* New York: Dover.

Comeau, A. R. and Hori, N., (1998), State-space forms for higher-order discrete-time models, *Systems & Control Letters*, vol. 34, pp. 23-31.

Cottafava, G. and Le Moli, G. (1969), Automatic contour plotting, *Communications of the ACM,* vol. *12*, no. 7, pp. 386-391.

Curtain, R. F., (1989), Representation of infinite-dimensional systems, in *Three decades of mathematical system theory*, LNCIS, Springer-verlag, Berlin 135, pp. 101-128.

Curtain, R. F., Bensoussan, A. and Lions, J. L., (Editors) (1993), *Analysis and optimization of systems: state frequency domain approaches for infinite-dimensional systems*: Proceedings of the 10th International Conference, Sophia-Antipolis, France, June 9-12, 1992, Springer-Verlag, Berlin, LNCIS 185.

Curtain, R.F., Pritchard, A. J., (1978), *Infinite dimensional linear system theory*, Lecture Notes in Control and information Science, Vol. 8, Springer-Verlag, Berlin.

Curtain, R.F., and Zwart, H., (1995), *An introduction to infinite-dimensional linear systems theory*, Springer Verlag, New York.

Davison, E. J., (1970), On pole assignment in linear systems with incomplete state feedback, *IEEE Transactions on Automatic Control*, vol. 15, pp. 384-351.

Delfour, M. C., (1981), The role of the structural operator and the quotient space structure in the theory of hereditary differential equations, in *Recent advances in differential equations*, Academic Press, new York (R. Conti, Ed.), pp. 111-135.

Delfour, M. C., (1997), State theory of linear differential systems, *Journal of Mathematical Analysis and Applications,* vol. 60, pp. 8-35.

Demmel, J., (1997), *Applied numerical linear algebra*, SIAM, Philadelphia.

Diekmann, O., S.A. Van Gils, S.M. Verduyn Lunel and H.O. Walther (1995) *Delay equations functional, complex and nonlinear analysis*. Springer Verlag, AMS Series, no. 110.

Duff, I. S. and Reid, J. K., (1993), Computing selected eigenvalues of large sparse matrices using subspace iteration. *ACM Transactions on Mathematical Software*, vol. 19, pp. 137-159.

Eijkhout, V., (1992), *Distributed sparse data structures for linear algebra equations*, Technical report CS 92-169, Computer Science Department, University of Tenessee, Knoxville, TN, LAPACK Working Note #50.

El'sgol'c, L. E. and Norkin, S. B., (1971), *Introduction to the theory of differential equations with deviating argument*, 2nd ed. Nauka, Moscov (in Russian). (1973), Mathematics in science and Eng., vol.105, Academic Press, New York.

Engelborghs, K. (2000). *DDE-BIFTOOL: a Matlab package for bifurcation analysis of delay differential equations*. TW Report 305, Department of Computer Science, Katholike Universiteit Leuven, Belgium. Available from http://www.cs.kuleuven.ac.be/~koen/delay/ddebiftool.shtml.

Engelborghs, K. and Roose, D. (1999), Numerical computation of stability and detection of Hopf bifurcation of steady state solutions of delay differential equations, *Advanced in Computational Mathematics,* vol. 10 (3-4), pp. 271-289.

Engelborghs, K., Luzyanina, T. and Roose, D. (2000). Numerical bifurcation analysis of delay differential, *Journal of Computational and Applied Mathematics*, vol. 125, pp. 265-275.

Engelborghs, K., Dambrine, M. and Roose, D. (2001), Limitation of a class of stabilization methods for delay systems, *IEEE Transactions on Automatic Control*, vol. 46, no. 2, pp. 336-339.

Engelborghs, K., Luzyanina, T. and Samaey, G., (2001b). *DDE-BIFTOOL v. 2.00: a Matlab package for bifurcation analysis of delay differential equations*. TW Report 330, Department of Computer Science, Katholike Universiteit Leuven, Belgium.

Engelborghs, K., Roose, D. (2002), On stability of LMS-methods and characteristic roots of delay differential equations, *SIAM Journal of Numerical Analysis*, vol. 40, number 2, pp. 629-650.

Fokkema, D. R., Sleijpen, G. L. G. and van der Vorst, H.A., (1998), Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils, SIAM Journal of Scientific Computing, vol. 20, no.1 , pp. 94-125.

Ford, N. J. and Wulf, V, (1998), Embedding of a numerical solution of a DDE into the numerical solution of a system of ODEs, *Technical Report*, Manchester Centre for Computational Mathematics, University of Manchester.

Fortune, S., (2001), Polynomial root finding algorithm using iterated eigenvalue computation, in *Proc. of ISSAC 2001,* pp. 121-128. Software Eigensolve available at http://cm.bell-labs.com/who/sjf/eigensolve.html.

George, A. and Liu, J., (1981), *Computer solution of large sparse positive definite systems*, Prentice-Hall, Englewood Cliffs, NJ.

Goodwin, G. C., Graebe, S. F., Salgado, M. E., (2001), *Control system design*, Prentice-Hall, Inc., New Jersey.

Gorecki, H., S. Fuksa, P. Grabowski and A. Korytowski (1989), *Analysis and synthesis of time delay systems*, Polish Scient. Publ. Warszawa.

Guglielmi, N and Hairer, E. (2001), *Implementing Radau IIA methods for stiff delay differential equations*, Computing 67, pp. 1-12.

Hairer, E, Lubich, C. and Roche, M., (1989), *The numerical solution of differential-algebraic systems by Runge-Kutta methods*, Lecture Notes in Mathematics, vol. 1409, Springer Verlag.

Hairer, E, Norsett, S. P., Wanner, G., (1987), *Solving ordinary differential equations I, nonstiff problems,* Springer Series in Comput. mathematics, vol. 8, Springer-Verlag.

Hairer, E, Wanner, G., (1996), *Solving ordinary differential equations II, stiff and differential-algebraic problems,* Springer Series in Comput. mathematics, vol. 14, Springer-Verlag.

Halanay, A., (1966), *Differential equations: stability, oscillations, time lags*, Academic Press, New York.

Hale, J. K., (1977), *Theory of functional differential equations*, Springer-Verlag, New York.

Hale, J. K., and Verduyn Lunel, (2001), Effects of small delays on stability and control, *Operator Theory: Advances and Applications*, 122, 275-301.

Hale, J. K., and Verduyn Lunel, S. M., (1993), *Introduction to functional differential equations,* Vol. 99 of Applied Mathematical Sciences, Springer Verlag New York Inc.

Hale, J. K., and Verduyn Lunel, S. M., (2002), Strong stabilization of neutral functional differential equations, *IMA Journal Mathematical Control and Information*, Vol. 19, no. 1, pp. 5-23.

Hang, C. C., Lee, T. H., and Ho, W. K., (1993), *Applied adaptive control*, ISA, NC.

Hassard, B. D., Kazarinoff, N. D. and Wan, Y. H., (1981*), Theory and applications of Hopf bifurcation*, Cambridge University Press, Cambridge.

Henrici, (1974), *Applied and computational complex analysis*, volume 1, John Wiley, New York.

Henry, D, (1974), Linear autonomous neutral functional differential equations, *Journal of Differential Equations*, vol. 15, pp.106-128.

Hlava, J., (1998), *Anisochronic internal model control of time delay systems, potentials and limits,* Ph.D. Thesis, Czech Technical University in Prague, Faculty of Mechanical Engineering.

Hong-Jiong, T. and Jiao-Xun, K., (1996), The numerical stability of linear multistep methods for delay differential equations with many delays, *SIAM Journal of Numerical Analysis*, vol. 33, no. 3, pp. 883-889.

Hout, K. J. in 't, (1992), The stability of a class of Runge-Kutta methods for delay differential equations, *Applied Numerical Mathematics*, vol. 9, p. 347-355.

Chow, S.-N. and Hale, J. K., (1982), *Methods of bifurcation theory*, Springer-verlag, New York Inc.

Chukwu, E. N., (1989), *Stability and time-optimal control of hereditary systems*, Academic Press, Boston.

Ichikava, K., (1985), Frequency domain pole assignment and exact model matching for delay systems, *International Journal of Control*, vol. 41, pp.1005-1024.

Kaashoek, M.A., and Verduyn Lunel, S.M., (1994) An integrability condition on the resolvent for hyperbolicity of the semigroup, *Journal of Differential Equations*, vol. 112, pp. 374-406 .

Kamen, E. W., (1978a), An operator theory of linear functional differential equations, *Journal of Differential Equations*, vol. 27, pp. 274-297.

Kamen, E. W., (1978b), *Lecture on algebraic system theory: linear systems over rings.* NASA Contractor Report 3016.

Kamen, E. W., (1975), On an algebraic theory of systems defined by convolution operators, *Math. System Theory*, vol. 9, pp. 57-74.

Knyazev, A, (2001), Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method. *SIAM Journal on Scientific Computing*, vol. 23, no. 2, pp. 517-541.

Kolmanovskii, V. B., and Myshkis, A. D., (1992), *Applied theory of functional differential equations*, Kluwert, Dordrecht, The Netherlands.

Kolmanovskii, V. B. and Nosov, V. R.,(1986), *Stability of functional differential equations*, vol. 180 of *Mathematicsin Science and Engineering*. Academic Press.

Kolmanovskii, V. B. and Shaikhet, L. E., (1996), *Control of systems with aftereffects*, American Math. Society, RI, vol. 157.

Kolmogorov, A. N., Fomin, S. V., (1999), *Elements of the theory of functions and functional analysis*, Dover Publications Inc., New York, (1972), third edition (in Russian), Nauka, Moscow.

Krasovskii, N. N., (1963), *Stability of motion*, Standford University Press.

Kuang, Y., (1993), *Delay differential equations with applications in population dynamics*, Academic Press, Boston.

Kuznetsov, Y. A., (1995), Elements of bifurcation theory, *Applied Mathematical Science*, vol. 112, Springer Verlag, New York.

Lehoucq, R. B., Sorensen, D. C. and Yang, C, (1998), *ARPACK Users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, SIAM, Phildelphia, http://www.caam.rice.edu/software/ARPACK/.

Levin, B. J., (1964), *Distribution of entire functions,* AMS, 5, Providence, Rhode Island.

Luzyanina, T. and Roose, D., (1996), Numerical stability analysis and computation of Hopf bifurcation points for delay differential equations, *Journal of Computational and Applied Mathematics*, vol. 72, pp. 379-392.

Luzyanina, T., Engelborghs, K., Lust, K. and Roose, D. (1997), Computation, continuation and bifurcation analysis of periodic solutions of delay differential equations *International Journal of Bifurcation and Chaos*, vol. 7, number 11, pp. 2547-2560.

Malek-Zaverei, M. and Jamshidi, M., (1987), *Time delay systems*: a*nalysis optimisation and applications*, North-Holland Systems and Control Series, vol. 9, Amsterdam, 1987.

Manitius, A. Z. and Olbrot, A. W., (1979), Finite spectrum assignment problem for systems with delays, *IEEE Transactions on Automatic Control*, vol. 24, no. 4, pp. 541-553.

Marsden, J. E. and McCracken, M., (1976), *The Hopf bifurcation and its applications*, Springer Verlag, New York, Heidelberg, Berlin.

Marshal, J. E., (1979), *Control of time delay systems*, IEE, Control Engineering Series, vol. 10, London.

Marshall, J.E., Górecki, H., Korytowski, A., Walton, K., (1992), *Time delay systems, stability and performance criteria with applications*, Ellis Horwood.

Mastinšek, M., (1994), Adjoints of solution semigroups and identifiability of delay differential equations in Hilbert spaces, *Acta Mathematica Universitatis Comeniane*, vol. 63/2, pp. 193-206.

Meerbergen, K. and Roose, D., (1996), Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems, *IMA J. Numer. Anal.*, 16, pp. 297-346.

Middleton and Goodwin, (1990), *Digital control and estimation*, Prentice-Hall.

Michiels, W. and Roose, D., (2002), Limitations of delayed state feedback: a numerical study, *International Journal of Bifurcation and Chaos*, vol.12, no.6, pp. 1309-1320.

Michiels, W., Engelborghs, K., Roose, D., (2001), Sensitivity to infinitesimal delays in neutral equations, *SIAM Journal on Control and Optimisation*, vol.40, no.4, pp.1134-1158.

Michiels, W., Engelborghs, K., Vansevenant, P., and Roose, D., (2002), Continuous pole placement for delay equations, *Automatica*, vol. 38, no. 6, pp. 747-761. (2000), in Proc. of 2nd IFAC Workshop on Linear Time Delay Systems, Ancona, Italy.

Minorski, N, (1942), Self excited oscillations in dynamical systems possessing retarded actions, *Journal of Applied Mechanics*, vol. 9, pp. 65-71.

Moore, G., and Spence, A., (1980), The calculation of turning points of nonlinear equations, *SIAM Journal of Numerical Analysis*, vol. 17, no. 4, pp. 567-576.

Morari, M. and Zafiriou, E., (1989), *Robust Process Control*, Prentice-Hall.

Myshkis, A. D., (1949), General theory of differential equations with delay, *Uspehi Mat. Nauk*. 22 (134), pp. 21-57, (in Russian). (1951), Amer. Math. Soc. Transl. no. 55.

Myshkis, A. D., (1955), *Lineare differentialgleichungen mit nacheilenden argumentom*, Deutscher Verlag, Wiss. Berlin. Translation of the 1951 Russian edition. (1972), *Linear differential equations with retarded arguments*, Nauka, Moscov, (in Russian).

Niculescu, S. I., (2001), *Delay effects on stability, a robust control approach*. Springer-Verlag London Limited.

O'Connor, D. A. and Tarn, T.J., (1983b), On stabilization by state feedback for neutral differential-difference equations, *IEEE Transactions of Automatic Control*, vol. 28, pp. 615-618.

O'Connor, D.A. and Tarn, T.J., (1983a), On the function space controllability of linear neutral systems, *SIAM Journal on Control and Optimization*, vol. 21,pp. 306-329.

Ogata, K, (1997), *Modern control engineering*, Prentice-Hall, Inc. third edition.

Ogunnaike, B. A. and Ray, W. H., (1979), Multivariable controller design for linear systems having multiple time delays, *AIChE Journal*, vol. 25, pp. 1043-1057.

Ogustoreli, M. N., (1996), *Time-lag control systems*, Academic Press, New York.

Palmor, Z. J., (1996), Time-delay compensation - Smith predictor and its modifications, *The Control Handbook* (W. S., Levine, Eds.), CRC Press, pp. 224-237.

Pan, V, (1997), Solving a polynomial equation: some history and recent progress, *SIAM Rewiev* 39:2, pp.187-220.

Pandolfi, L., (1976), Stabilization of neutral functional differential equations, *Journal of Optimization Theory and Applications*, vol. 20, pp. 191-204.

Pazy, A., (1983), *Semigroups of linear operators and applications to partial differential equations*, Springer-Verlag, New York.

Penrose, R. (1955), A generalized inverse for matrices, in *Proc. Cambridge Phil. Soc.* vol. 51, pp. 406-413.

Peters, G. and Wilkinson, J. H., (1971), Practical problems arising in the solution of polynomial equations, *Journal of the Institute of Mathematics and its Applications.*

Petrová, (2001), *Discrete approximation of the anisochronic models of time delay systems in delta transform*, Ph.D. Thesis, Czech Technical University in Prague, Faculty of Mechanical Engineering, (In Czech).

Pinney, E., (1958), *Differential-difference equations*, University California Press, Berkeley, CL.

Pontryagin, L., (1942), On zeros of some transcendental functions, *Izvestiya Ackademii Nauk SSSR*, vol. 6, pp. 115-131, (in Russian).

Preusser, A., (1984), Algorithm 626: TRICP- a contour plot program for triangular meshes, *ACM Transactions on Mathematical Software*, vol. 10, no. 4, pp. 473-475.

Preusser, A., (1986), Computing area filling contours for surfaces defined by piecewise polynomials, *Computer Aided Geometric Design*, vol. 3, pp. 267-279.

Rektorys, K, (ed.), (1994), *Survey of applicable mathematics, Volumes I and II,* Kluwer Academic Publishers, Dodrecht.

Saad, Y., (1992), *Numerical methods for large eigenvalue problems: theory and algorithms*, John Wiley, New York.

Salamon, D., (1984), *Control and observation of neutral systems*, Research Notes on Mathematics; 91, Pitman Publishing, London.

Scott, J.A., (1995), *An Arnoldi code for computing selected eigenvalues of sparse real unsymmetric matrices*, ACM Transactions on Mathematical Software, Vol. 21, no 4, pp. 432-475.

Seborg, D. E., Edgar, T. F. and Melichamp, D. A., (1989), *Process dynamics and control*, John Wiley & Sons, New York.

Schafer, R. D., (1996), *An introduction to nonassociative algebras*, Dover Pubns, New York.

Skogestad, S. and Postlethwaite, I., (1996), *Multivariable feedback control.* John Wiley and Sons.

Sleijpen, G. L. G. and Vorst, H.A., (1996), A Jacobi-Davidson iteration method for linear eigenvalue problems, *SIAM Journal of Matrix Analysis and Application*, vol. 17, pp. 401-425.

Smith, O. J. M., (1959), A controller to overcome the dead time, *Instrument Sociaty of American Journals* (*ISA*), vol. 6, pp. 28-33.

Snyder, W. V., (1978), Algorithm 531, contour plotting [J6], *ACM Transactions on Mathematical Software*, *vol.* 4, no. 3, pp. 290-294.

Sontag, E.D., (1998), *Mathematical control theory*, Second edition, Springer-Verlag, New York.

Stépán, G.,(1989), *Retarded dynamical systems, stability and characteristic function*, Research Notes in Math. Series, 210, Longman Scientific, UK.

Trembley J., and Bunt R., (1989), *Introduction to computer science, an algorithm approach*, McGraw Hill.

Vaněček, A., (1990), *Control systems theory*. Academia Praha, (In Czech).

Verduyn Lunel, S. M., (1995), About completeness for a class of unbounded operators, *Journal of Differential Equations*. vol. 120, 108-132.

Verduyn Lunel, S. M., (2001a), Parameter identifiability of differential delay equations, *International Journal of Adaptive Control and Signal Processing*, vol. 15, no. 6, pp. 655-678.

Verduyn Lunel, S. M., (2001b), Spectral theory for delay equations, *Operator Theory: Advances and Applications*, Vol. 129, pp. 465-508.

Vyhlídal, T., Zítek, P., (2001), Control system design based on a universal first order model with time delays, In: *Proceedings of International Conference on Advanced Engineering Design*. Glasgow: University of Glasgow, and In: *Acta Polytechnica*. vol. 41, no. 4-5, pp. 49-53.

Wang, Q. G., Lee, T. H. and Hang, C. C., (1993), Frequency-domain finite spectrum assignment for delay systems with multiple poles, *International Journal of Control*, vol. 58, pp. 735-738.

Wang, Q. G., Lee, T. H., Tan, K.G., (1999), *Finite spectrum assignment for time-delay systems*, Springer-Verlag, London.

Wang, Q. G., Lee, T. H. and Tan, K.G., (1995), Automatic tuning of finite spectrum assignment controllers for delay systems, *Automatica*, vol. 31, no. 3, pp. 477-482.

Watanabe, K., Ito, M. and Kaneko, M., (1983), Finite spectrum assignment for systems with multiple commensurate delays in state variables, *International Journal of Control*, vol. 38, no. 5, pp. 913-926.

Watanabe, K., Ito, M. and Kaneko, M., (1984), Finite spectrum assignment of systems with multiple commensurate delays in state and control, *International Journal of Control*, vol. 39, no. 5, pp. 1073-1082.

Watanabe, K, (1986), Finite spectrum assignment and observer for multivariable systems with commensurate delays, *IEEE Transactions on Automatic Control*, vol. 31, no. 6, pp. 543-550.

Weiner, R. and Strehmel, K. (1988), A type insensitive code for delay differential equations basing on adaptive and explicit Runge Kutta interpolation methods, *Computing*, vol. 40, pp. 255-265.

Wilf, H. S., (1978), A global bisection algorithm for computing the zeros of polynomials in the complex plane, *Journal of the ACM*, vol. 25, pp. 415-420.

Wilkinson, J. H. (1984), The perfidious polynomial, *Studies in Numerical Analysis*, M.A.A. Studies in Mathematics, 24, Gene H. Golub, ed., pp. 1-28, 1984.

Wilkinson, J. H., (1963) *Rounding errors in algebraic processes*. Notes on Applied Science No. 32, Her Majesty's Stationery Office, London.

Wilkinson, J. H., (1965), *The algebraic eigenvalue problem*, Oxford University Press.

Wonham, W. M., (1967), On pole assignment in multi-input controllable linear systems, *IEEE Transactions on Automatic Control*, vol. 12, no. 6.

Wulf, V. and Ford, N. J., (2000), Numerical Hopf bifurcation for a class of delay differential equations, *Journal of Computational and Applied Mathematics*, vol. 115, pp. 601-616.

Yanushevski, R. T., (1978), *Control of retarded systems*, Nauka, Moscow, (in Russian).

Zadeh, L. A. and Desoer, C. A., (1963), *Linear system theory-the state space approach*, New York: McGraw-Hill.

Zennaro, M., (1986), P-stability properties of Runge Kutta methods for delay differential equations, *Numerical Mathematics*, vol. 49, pp. 305-318.

Zítek and Petrová, (2002), Discrete approximation of anisochronic systems using delta transform, *In Proc. of Process Control - Rip 200*2. Pardubice, University of Pardubice.

Zítek, P. - Hlava, J., (1998), Anisochronic inverse-based control of time delay systems, *Control 98*. London: Institution of Electrical Engineers, pp. 1409-1414.

Zítek, P. (1997) Frequency domain synthesis of hereditary control systems via anisochronic state space. *International Journal of Control 66*, No. 4, 539-556.

Zítek, P. (1998a). Anisochronic state observers for hereditary systems. *Int. Journal of Control,* Vol. 71, No. 4, pp. 581- 599.

Zítek, P. and Vyhlídal, T. (2002a), Dominant eigenvalue placement for time delay systems, In *Proc. of Controlo 2002, 5th Portuguese Conference on Automatic Control, University of Aveiro, Portugal.*

Zítek, P. and Vyhlídal, T. (2002b), Dominant eigenvalue placement for time delay systems, *Technical report of Centre For Applied Cybernetics, CTU in Prague*.

Zítek, P., (1983), Anisochronic state theory of dynamic systems, *Acta Technica ČSAV*, vol. 4

Zítek, P., (1986). Anisochronic modelling and stability criterion of hereditary systems. *Problems of Control and Information Theory* 15, No. 6, pp. 413-423.

Zítek, P., (1998b). Time delay control system design using functional state models. *CTU Reports*, no.1, CTU Prague.

Zítek, P., Fišer, J., Vyhlídal, T. (2001), Sliding mode control of time delay systems *1st IFAC/IEEE Symposium On System Structure and Control 2001*. Praha : IFAC, pp. 21.

Zítek, P., Garagič, D, (1997), Anisochronic state observers in fault detection of hereditary control systems, In: *Fault Detection, Supervision and Safety for Technical Processes*. Kingston upon Hull : University of Hull, pp. 106-111.

Zítek, P., Hlava, J., (2001), Anisochronic internal model control of time-delay systems, In: *Control Engineering Practice*. 2001, vol. 9, no. 5, p. 501-516.

Zítek, P., Matějka, J., Petrová, R. (1995), State feedback control of hereditary systems via anisochronic state concept, In: *Proceedings - 10th Conference Process Control '95*. Bratislava : Slovak University of Technology, pp. 29-34 .

Zítek, P., Vyhlídal, T.(1999), Anisochronic state feedback: an approach to designing branched control scheme, In: *Proceedings - 12th Conference Process Control '99*. Bratislava Slovak University of Technology, 1999, vol. 1, pp. 1-5.

Zítek, P., Vyhlídal, T., (2000), State feedback control of time delay system: conformal mapping aided design In: *2nd IFAC Workshop on Linear Time Delay Systems*. Ancona, Universita di Ancona, pp. 146-151.

Zítek, P., Vyhlídal, T., Fišer, J., (2001), Anisochronic state feedback design compensating for system delays, In: *3rd IFAC Workshop on Time Delay Systems* [CD-ROM]. Santa Fe.