ulm university universität
uulm

Verified Methods for State and Parameter Estimators for Nonlinear Uncertain Systems with Applications in Engineering

DISSERTATION

zur Erlangung des akademischen Grades eines

DOKTOR-INGENIEURS

(DR.-ING.)

der Fakultät für Ingenieurwissenschaften
und Informatik der Universität Ulm

von

MARCO KLETTING

AUS DORNSTADT-TEMMENHAUSEN

Gutachter:	Prof. Dr. Eberhard P. Hofer Dr. Èric Walter
Amtierender Dekan:	Prof. Dr. Ing. Michael Weber

Ulm, 15.05.2009

Preface

This dissertation resulted from my work as a research assistant at the Institute of Measurement, Control and Micro Technology of the Faculty of Engineering and Computer Science at the University of Ulm in Germany.
First of all, I would like to thank sincerely my doctor father Prof. Dr. Eberhard P. Hofer for his support, his encouragement, and guidance during my research. I am also very grateful to Dr. Eric Walter for being supervisor of my doctoral thesis and for his careful reading of the manuscript and for his valuable advices. Furthermore I would like to thank my former colleague Dr.-Ing. Andreas Rauh for the many precious discussions and suggestions and for the great collaboration during my time at the Institute. I would also like to thank Prof.-Dr.-Ing. Harald Aschemann for his support. Special thanks to my friend Dr.-Ing. Felix Antritter for a very fruitful cooperation and making it possible to extend my field of research. Also thanks to all of my students for their excellent work.
My deepest gratitude belongs to my beloved wife Jane. Her constant support helped me also through difficult times. And I also thank my three little children for being patient with their Dad, when he was too busy to play with them.

For my beloved wife Jane

"Many women have done excellently,
but you surpass them all." (Prov. 31:29)

Contents

List of Figures

List of Tables

List of Symbols and Abbreviations

This list of symbols contains the most important symbols used in this work. Exceptions or variations will be mentioned within the section where they occur.

System describing symbols

δ_k	additive measurement error at $t = t_k$
$\delta(t)$	additive measurement error
$\mathrm{diag}(x)$	diagonal matrix composed by the components of the vector x
e_k	time discretization error in time-step k
f	continuous-time dynamical system function in state space representation
$f^{(i)}$	i-th derivative of f with respect to time, i.e., $f^{(i)} = \frac{\partial^i f}{\partial t^i}$
h	nonlinear measurement equation
h_k	step size for the time interval $[t_k; t_{k+1}]$
$\mathfrak{I}$	identity matrix
k	time step k at $t = t_k$
m	dimension of the vector of measured values y
n_x	dimension of the state vectors x_k and $x(t)$
n_p	dimension of parameter vectors p_k and $p(t)$
n_q	dimension of parameter vectors q_k and $q(t)$
n_u	dimension of control variable vectors u_k and $u(t)$
n	dimension of the extended state vectors z_k and $z(t)$
$p(t)$	vector of uncertain system parameters of f
p_k	vector of uncertain system parameters of f at $t = t_k$
$\Delta p(t)$	vector of the variation rates of the vector of $p(t)$
$q(t)$	vector of uncertain parameters of the measurement model
q_k	vector of uncertain parameters of the measurement model at $t = t_k$
t	time variable
T_i^j	transition condition from model i to model j
$u(t)$	control variable
u_k	control variable at $t = t_k$
ν	order of Taylor series expansion with respect to time
$x(t)$	system state vector of f
x_k	system state vector at $t = t_k$
$x_{i,k}$, $x_i(t)$	i-th components of the vectors x_k and $x(t)$
ξ_k	point of time in the interval $[t_k; t_{k+1}]$
$y(t)$	vector of measured values
y_k	vector of measured values in time-step k
$z(t)$	extended state vector consisting of the system state vector x and uncertain system parameters p

z_k	extended state vector in time-step k
$z_{i,k}$	i-th component of the extended state vector in time-step k

Symbols related to interval arithmetic and Taylor model arithmetic

A_k	transformation matrix at $t = t_k$
$B(T)$	interval bound of the Taylor model (vector) T
$[B_k]$	coarse enclosure (Bounding-Box) of the extended state vector in the time-interval $[t_k; t_{k+1}]$
$[\delta_k]$	interval enclosure of $\delta(t)$ in time step k
$[e_k]$	interval enclosure of the time discretization error for $t \in [t_k; t_{k+1}]$
I_ρ	remainder interval for a Taylor model T_ρ
K	Krawczyk-operator
L_k	list length at $t = t_k$
L_{max}	maximum list length
$[p_k]$	interval enclosure of $p(t)$ in time step k
$\underline{p_k}$	infimum of the interval vector $[p_k]$
$\overline{p_k}$	supremum of the interval vector $[p_k]$
$[\Delta p(t)]$	interval bounds of the parameter variation rates
P_ρ	ρ-th order polynomial part of the Taylor model T_ρ
$\Phi([B])$	Picard operator for the determination of the coarse enclosure $[B]$
ρ	order of Taylor expansion in time and initial state variables
$\mathfrak{z}$	initial state vector in the Taylor model based integration of ordinary differential equations
$[\mathfrak{z}]$	domain interval vector, bounding the initial state vector $\mathfrak{z}$
T_ρ	Taylor model (vector) of order ρ
$T_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}})$	ρ-th order Taylor model (vector) in time-step k with respect to the vector $\mathfrak{z}$ at the expansion point $\hat{\mathfrak{z}}$
$T_{\rho,i,k}$	$i - th$ component of the Taylor model (vector) $T_{\rho,k}$
$\mathcal{T}_k$	list of Taylor models in time step k
$T^{(l)}_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}})$	the l-th Taylor model of $\mathcal{T}_k$
$[x_k]$	interval enclosure of $x(t)$ in time step k
$[z_k]$	interval enclosure of $z(t)$ in time step k
$\mathcal{Z}_k$	list of interval vectors in time step k
$[z^{(l)}_k]$	the l-th interval vector of $\mathcal{Z}_k$
$\mathscr{Z}_k$	exact solution set of the extended state vector $z(t)$ in time step k

Abbreviations

ASM1	Activated Sludge Model No. 1
BWTP	Biological waste water treatment plant
DAE	differential algebraic equation
IWA	International Water Association
NISTR	Non-isothermal stirred tank reactor
ODE	Ordinary differential equation
PDE	partial differential equation
SIVIA	set inversion via interval analysis
VSPE	Verified state and parameter estimator

1 Introduction

In most applications in control engineering a measurement of all state variables is either impossible or avoided for reasons of cost-effectiveness or because of the lack of reliable sensors. Especially in nonlinear control, however, also the knowledge of non-measured system state variables is required. Also process monitoring of safety critical processes requires often knowledge about state variables which are not accessible by measurements. Therefore, state observers are employed to compute estimates for the whole state vector. For the observer design in addition to the system model a measurement model which takes into account measurement errors and sensor parameters is needed.
For the mathematical description of nonlinear dynamical systems and measurement processes state space representations are well suited. In control engineering, a state space representation is a mathematical model of a physical system described by a set of input, output and state variables which are related by first-order differential equations. The internal state variables are the smallest possible subset of system variables that can represent the entire state of the system at any given time.
The modeling of dynamical system is usually affected by uncertainty. For example the system parameters may be uncertain due to manufacturing tolerances in technical systems. Often also the initial states of the state variables are not exactly known.
In engineering both stochastic uncertainty and set-valued uncertainty are used. Depending on the situation and the system the appropriate representation of the uncertainty has to be employed. Stochastic uncertainty requires knowledge about the probability density function of the corresponding uncertainty. On the other hand the representation of an uncertainty as set-valued uncertainty requires guaranteed bounds for their range.

From the theory of dynamical systems different concepts of state and parameter estimation are known. In recursive state and parameter estimation non-measured state variables and parameters are reconstructed with the help of the measured values and a model describing the sensor characteristics under consideration of the system dynamics. The consideration of several measurements at different points of time, the integration of several sensors and the measurements of different physical values makes it possible not only to reconstruct the non-measured state variables but also to decrease the uncertainty of the estimates. In case of knowledge about conservative upper and lower bounds of set-valued model- and measurement uncertainty interval methods are able to compute tight and at the same time verified enclosures of the complete state and parameter vector. In contrast to stochastic estimation approaches like the often applied extensions of Kalman-filters for nonlinear systems with standard normal distribution uncertainty, interval methods are a suitable approach for a verified estimation in case of interval uncertainty. The main challenge for the implementation of algorithms based on interval methods for verified state and parameter estimation is the reduction of overestimation, which occurs if so called naive interval methods are applied.

The verified state and parameter estimation consists of a recursive application of prediction and correction steps. The prediction step corresponds to a verified integration of the system model describing the system dynamics between two points of time at which measured data is available. Verified integration means that the obtained results guarantee to enclose the solution of the flow of the differential equation regarding all uncertainty including also time discretization and round off errors. In the correction step the state variables and parameters are reconstructed from the measurements and the measurement equation. The resulting estimates are intersected with the results from the prediction step. The obtained set after the intersection is the enclosure used for the next prediction step. The resulting enclosures are guaranteed enclosures of those state variables and parameters, which comply with the system and measurement model and all uncertainty. In order to give guaranteed estimations for a given system in the real world, the system and measurement models have to be accurate. To cope with modeling errors beside parameter uncertainty – e.g. errors in the model structure itself – it must be possible to consider such errors in form of intervals in the system and measurement equations.

1.1 State of the Art

Concepts of verified state and parameter estimators for nonlinear systems were first developed for discrete time systems by Kieffer [30]. In publications of Jaulin, Walter, Kieffer and Ditrit [27, 30, 31] the state vector and the vector of uncertain parameters are enclosed by multiple interval vectors. The components of an interval vector are the corresponding interval enclosures for each state variable and uncertain parameter of the state vector and the vector of uncertain parameters.
Alamo developed a verified state and parameter estimator based on zonotopes[1] [1].
The first verified estimator for continuous-time systems was proposed by Jaulin in [26]. Here, the sets are again described by multiple interval vectors. Like verified estimators for discrete time systems, verified estimators for continuous-time systems also calculate enclosures of the state vector at discrete points of time. However, for the truncation errors in the integration process guaranteed bounds are calculated additionally.
A verified estimator for cooperative systems[2], where the state variables are enclosed by a single interval vector, was presented by Gouze in [19].
In [32] Kieffer introduced also a verified estimator for cooperative systems. In contrast to [19] multiple interval vectors are used for the state enclosures. An algorithm, which uses preconditioning techniques, without interval splitting in the prediction step, was proposed by Raissi in [58]. The application of preconditioning techniques leads to parallelepiped enclosures.

[1] A zonotope is the the Minkowski sum of line segments.
[2] Cooperative systems have a monotonic behavior. In order to determine guaranteed upper and lower bounds, only calculations with the lower and upper bounds of the uncertainty and not with the whole intervals are required.

1.2 Goals of this Contribution

In this dissertation two different types of verified state and parameter estimators for continuous time systems are considered. The first approach employs interval vectors and it combines interval splitting and merging techniques with preconditioning techniques [36]. In the following, this type of estimator will be called *interval observer*. The second approach is based on Taylor models, which was proposed the first time in [37]. A Taylor model consists of a multivariate polynomial part and an interval remainder bound. In the following, this method will be designated as *Taylor model observer*. The concepts are applied to practical relevant systems from the field of engineering.
Both estimation concepts are implemented in C++ to enable a fair comparison. The interval observer uses the toolboxes PROFIL/BIAS and FADBAD++. PROFIL/BIAS [39] provides interval operations and FADBAD++ [72] is required for automatic differentiation. The Taylor model observer implementation is based on the C++ interface of COSY-INFINITY [45].
A further contribution is the application of interval methods for a verified robustness analysis of a controlled system including a traditional observer concept [33]. The goal of this robustness analysis is to find the set of parameter values and values for the initial conditions for which the controlled system with the traditional observer concept fulfills given robustness requirements. In this thesis also an application of verified state and parameter estimators in closed loop control is presented. Consistency techniques based on backward integration for the reduction of overestimation are also described.

1.3 Overview

This dissertation is structured as follows. Chapter 2 gives a detailed description for the applications which are studied in this work. In Chapter 3 an introduction to interval analysis and Taylor model arithmetic is given. In Chapter 4 algorithms for the verified integration of systems with interval uncertainty are presented. They are the basis of the prediction step. First concepts using interval enclosures for the state vector and the parameter vector are discussed. Also algorithms for systems with model switching characteristics are presented. Then methods employing Taylor model enclosures are introduced. In Chapter 5 the two different approaches for a verified state and parameter estimation which have been mentioned above are introduced. The underlying algorithms and corresponding estimation results are given separately for each of both concepts. A comparison of the approaches on the basis of estimation results is presented in the final section of Chapter 5.
Chapter 6 describes the application of interval methods for a verified robustness analysis of a controlled system including a traditional observer concept. The estimations in Chapter 5 are carried out for open loop systems. Chapter 7 shows an application of verified state and parameter estimators in closed loop control. The conclusion and outlook on future research are given in Chapter 8.

2 Applications

In this section the main applications which are studied in this dissertation are described. Academic examples are explained in the corresponding section, where they have been used.

2.1 Non-Isothermal Stirred Tank Reactor

Consider a jacketed non-isothermal stirred tank reactor (NISTR) [13] as depicted in Fig.2.1, in which cyclopentenol is produced from cyclopentadiene by an acid catalytic electrophilic addition of water. The reaction takes place under 15 bar in a nitrogen inert continuous operated stirred tank reactor.
In case the jacket temperature has to be known, because it should not exceed a given value for safety reasons, but the measurement is not possible or too costly, the temperature has to be estimated. The same holds, if a concentration cannot be measured but knowledge about it is crucial for the operation of the reactor.
The reactor with the volume V_R is equipped with a jacket cooling. The input volume stream only contains the reactant cyclopentadiene in form of a diluted solution. Because of the reaction ability of both cyclopentadien and cyclopentenol, cyclopentanediol emerges as a side-product and dicyclopentadiene is produced as a by-product in a parallel reaction. The complex reaction mechanisms which have first been described by *van de Vusse* can be summarized by the following reaction scheme:

$$\begin{aligned} A \overset{k_1}{\rightarrow} B \overset{k_2}{\rightarrow} C \ , \\ 2A \overset{k_3}{\rightarrow} D. \end{aligned} \tag{2.1}$$

Component A represents the reactant cyclopentadiene, B is the desired product cyclopentenol, component C is the unwanted side product cyclopentanediol, and D is dicyclopentadiene, the product of the undesirable parallel reaction. The temperature in the reactor is denoted by v, the jacket temperature by v_K. The input stream $\dot{V}$ contents only component A with the concentration c_{a0}. The total reaction is considered to be constant in volume. For the process relevant concentrations c_a of component A and c_b of component B the differential balance equations

$$\begin{aligned} \dot{c}_a &= \frac{\dot{V}}{V_R}(c_{a0} - c_a) - k_1(v)c_a - k_3(v)c_a^2 \ , \\ \dot{c}_b &= -\frac{\dot{V}}{V_R}c_b + k_1(v)c_a - k_2(v)c_b \end{aligned} \tag{2.2}$$

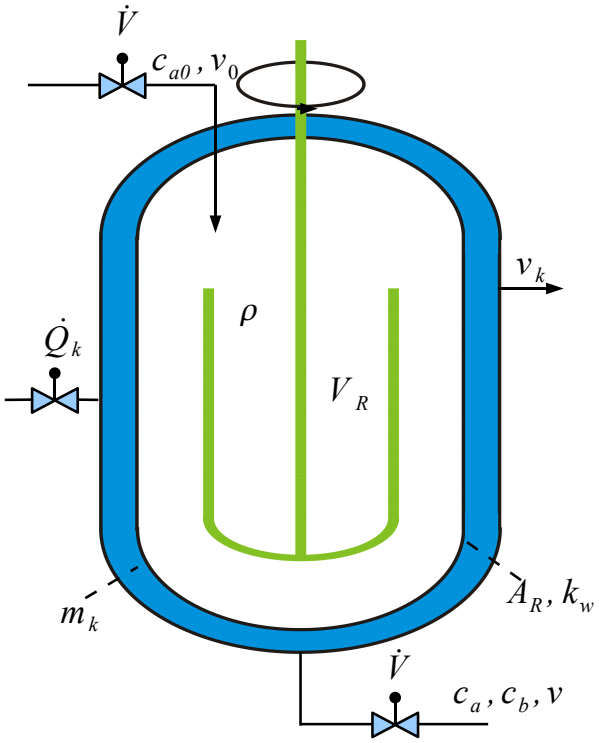


Figure 2.1: Non-isothermal stirred tank reactor.

are obtained, assuming an ideal stirred tank reactor.

The heat-balance equation for the inner reactor temperature v is

$$\dot{v} = \frac{\dot{V}}{V_R}(v_0 - v) + \frac{k_w A_r}{\rho C_p V_R}(v_K - v) - \frac{1}{\rho C_p}\left(k_1(v)c_a \Delta H_{R_{AB}} + k_2(v)c_b \Delta H_{R_{BC}} + k_3(v)c_a^2 \Delta H_{R_{AD}}\right) , \tag{2.3}$$

where the ΔH_{R_i} denote the reaction enthalpies for the released energy and v_0 denotes the temperature of the reactor input. An external heat exchanger revokes the heat stream $\dot{Q}_K$ out of the cooling medium of the reactor jacket. For the temperature v_K the following balance equation

$$\dot{v_K} = \frac{1}{m_K C_{pk}}(\dot{Q}_K + k_w A_R(v - v_K)) \tag{2.4}$$

is obtained. The reaction rate coefficients k_i, which are related to the educt of the particular partial reaction, are temperature dependent, according to

$$k_i(v) = k_{i,0} e^{\frac{-E_i}{v+273.15}} \quad , i = 1, 2, 3 \ . \tag{2.5}$$

Furthermore

$$k_1(v) = k_2(v) \tag{2.6}$$

holds [13]. The E_i in the exponential term represent the activation energies. In summary, the following system of four coupled nonlinear differential equations are obtained:

$$\begin{aligned}
\dot{c}_a &= \frac{\dot{V}}{V_R}(c_{a0} - c_a) - k_1(v)c_a - k_3(v)c_a^2 \ , \\
\dot{c}_b &= -\frac{\dot{V}}{V_R}c_b + k_1(v)c_a - k_2(v)c_b \ , \\
\dot{v} &= \frac{\dot{V}}{V_R}(v_0 - v) + \frac{k_w A_r}{\rho C_p V_R}(v_K - v) \ , \\
&\quad - \frac{1}{\rho C_p}(k_1(v)c_a\Delta H_{R_{AB}} + k_2(v)c_b\Delta H_{R_{BC}} + k_3(v)c_a^2\Delta H_{R_{AD}}) \ , \\
\dot{v_K} &= \frac{1}{m_K C_{pk}}(\dot{Q}_K + k_w A_R(v - v_K)) \ .
\end{aligned} \tag{2.7}$$

The nominal values of the parameters are listed in Tab. 2.1.

Table 2.1: Nominal values of the NISTR system parameters.

Parameter	Physical meaning	Nominal value
$\dot{V}$	influent flow rate of Component A	141.9 l/h
V_R	volume of the reactor	10 l
$\dot{Q}_K$	power of the heat exchanger	-1113.5 kJ/h
c_{A0}	influent concentration of component A	5.1 mol/l
v_0	temperature in the inflow	104.9 °C
$\Delta H_{R_{AB}}$	reaction enthalpy for $A \rightarrow B$	4.2 $\frac{\text{kJ}}{\text{kgK}}$
$\Delta H_{R_{BC}}$	reaction enthalpy for $B \rightarrow C$	-11 $\frac{\text{kJ}}{\text{kgK}}$
$\Delta H_{R_{AD}}$	reaction enthalpy for $A \rightarrow D$	-41.85 $\frac{\text{kJ}}{\text{kgK}}$
C_{pk}	heat capacity of the cooling	2 $\frac{\text{kJ}}{\text{kgK}}$
m_K	fluid mass in the cooling circuit	5 kg
$k_{1,0}$	multiplicative coefficient in reaction rate coefficient k_1	$1.287 \cdot 10^{12}$ h^{-1}
$k_{2,0}$	multiplicative coefficient in reaction rate coefficient k_2	$1.287 \cdot 10^{12}$ h^{-1}
$k_{3,0}$	multiplicative coefficient in reaction rate coefficient k_3	$9.043 \cdot 10^{9}$ h^{-1}
ρ	average density of the reactor content	0.9342 kg/m^3
C_p	average heat capacity of the reactor content	3.01 $\frac{\text{kJ}}{\text{kgK}}$
k_w	heat transfer coefficient	4032 $\frac{\text{kJ}}{\text{hK}m^2}$
A_R	area of the cooling jacket	0.215 m^2
E_1	activation energy in reaction rate coefficient k_1	9758.3 K
E_2	activation energy in reaction rate coefficient k_2	9758.3 K
E_3	activation energy in reaction rate coefficient k_3	8560 K

For the concentrations the following restrictions hold additionally:

$$c_a \geq 0, c_b \geq 0 \ . \tag{2.8}$$

The concentration c_b and the reactor temperature v are measured [13]. The concentration c_a and the jacket temperature v_k have to be estimated.

2.2 Mechanical Positioning System

This example is used to illustrate the influence of friction forces in dynamical systems. The system is affected by static and sliding friction. The system equation with the nonlinear friction characteristic $F_f(x_2)$ and the driving force $F_a(t)$ is given by

$$\begin{aligned} \dot{x_1} &= x_2 \ , \\ \dot{x_2} &= \frac{1}{m}(F_a(t) - F_f(x_2)) \ . \end{aligned} \tag{2.9}$$

The friction characteristic is illustrated in Fig. 2.2 and is assumed to be symmetric. F_s denotes the static friction force and μ the sliding friction coefficient.

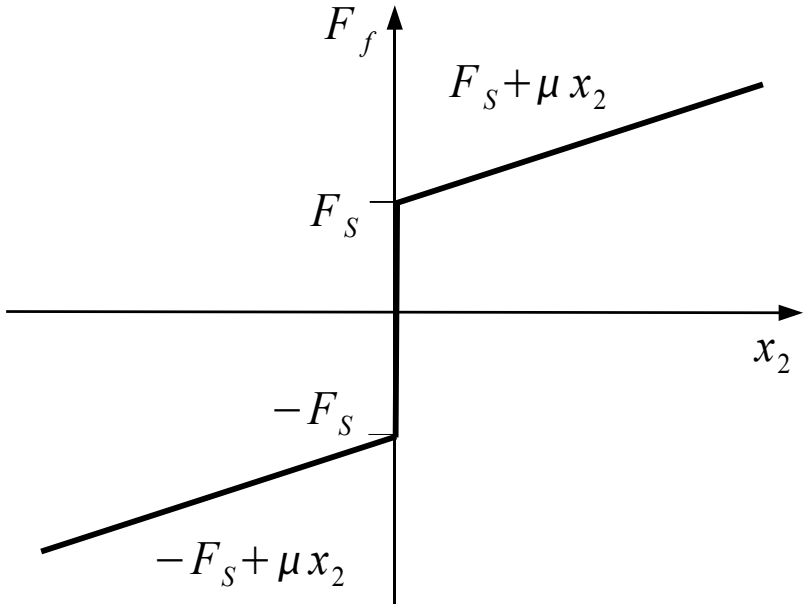


Figure 2.2: Friction characteristic.

If the system is in motion, the resulting sliding friction force is given by

$$F_f(x_2) = \begin{cases} -F_s + \mu \cdot x_2 & \text{for } x_2 < 0 \ , \\ +F_s + \mu \cdot x_2 & \text{for } x_2 > 0 \ . \end{cases} \tag{2.10}$$

The static friction force in the idle state is given by

$$F_f(x_2) = \begin{cases} F_f(x_2) = F_s & \text{for } x_2 = 0 \wedge F_a \geq F_s \ , \\ F_f(x_2) = F_a & \text{for } x_2 = 0 \wedge |F_a| < F_s \ , \\ F_f(x_2) = -F_s & \text{for } x_2 = 0 \wedge F_a \leq -F_s \ . \end{cases} \tag{2.11}$$

The static friction leads to a discontinuity in the friction characteristic at $x_2 = 0$. For $x_2 = 0$ the system remains in the idle state as long as the driving force $F_a(t)$ is below the positive friction force F_s and above the negative static friction force $-F_s$. When this is no longer the case the system is set in motion.
On the other hand the system only goes back into the idle state if the velocity is zero – $x_2 = 0$ – and the driving force $F_a(t)$ is smaller than the static friction. More complicated friction models have been studied in [7, 49, 69].

2.3 Double Pendulum

The third system under consideration is a double pendulum [73], which is depicted in Fig. 2.3. The double pendulum is a well known application for the illustration of chaotic motions.

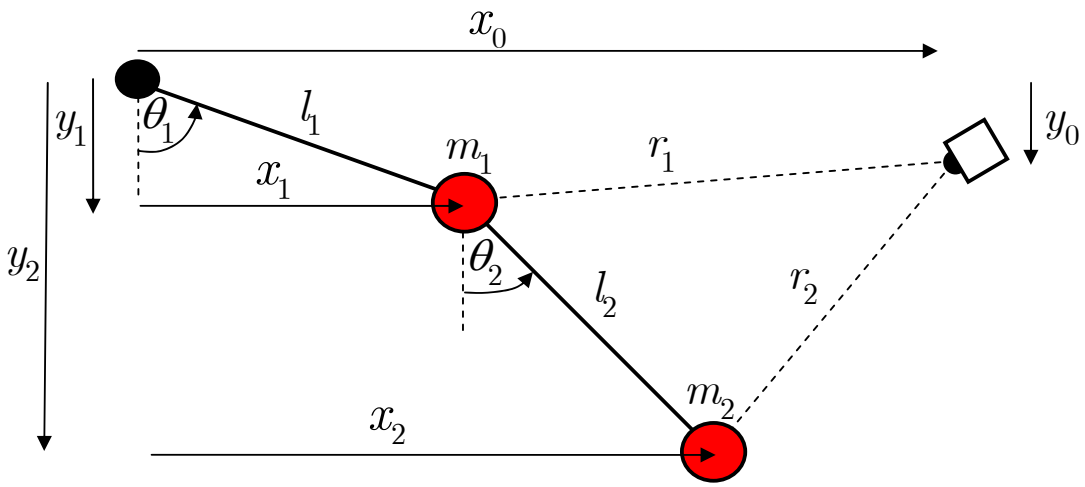


Figure 2.3: Double pendulum.

The position x_1 and x_2 of the ends of each pendulum are given by

$$\begin{aligned} x_1 &= l_1 \cdot \sin(\theta_1) \ , \\ y_1 &= -l_1 \cdot \cos(\theta_1) \ , \\ x_2 &= l_1 \cdot \sin(\theta_1) + l_2 \cdot \sin(\theta_2) \ , \\ y_2 &= -l_1 \cdot \cos(\theta_1) - l_2 \cdot \cos(\theta_2) \ . \end{aligned} \tag{2.12}$$

The two forms of energy of the system are the potential energy $\mathcal{E}_1 = E_{pot}(\theta_1, \theta_2)$ and the kinetic energy $\mathcal{E}_2 = E_{kin}(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2)$. For the deviation of the differential equations, which describe this system the $Lagrange - Equations$ $\mathcal{L} = \mathcal{E}_2 - \mathcal{E}_1$ have to be solved. The potential energy is given by

$$\begin{aligned} \mathcal{E}_1 &= m_1 g y_1 + m_2 g y_2 \\ &= -(m_1 + m_2) g l_1 \cos(\theta_1) - m_2 g l_2 \cos(\theta_2) \ , \end{aligned} \tag{2.13}$$

the kinetic energy $\mathcal{E}_2$ is described by

$$\begin{aligned}\mathcal{E}_2 &= \frac{1}{2}m_1 v_1^2 + \frac{1}{2}m_2 v_2^2 \\ &= \frac{1}{2}m_1 l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2[l_1^2\dot{\theta}_1^2 + l_2^2\dot{\theta}_2^2 + 2l_1 l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_1-\theta_2)] \ .\end{aligned} \tag{2.14}$$

The Lagrangian $\mathcal{L}$ is then

$$\begin{aligned}\mathcal{L} = \mathcal{E}_2 - \mathcal{E}_1 &= \frac{1}{2}m_1 l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2[l_1^2\dot{\theta}_1^2 + l_2^2\dot{\theta}_2^2 \\ &\quad +2l_1 l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_1-\theta_2)] - [(m_1+m_2)gl_1\cos(\theta_1) - m_2 g l_2\cos(\theta_2)] \\ &= \frac{1}{2}(m_1+m_2)l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2 l_2^2\dot{\theta}_2^2 + m_2 l_1 l_2\dot{\theta}_1\dot{\theta}_2\cos(\theta_1-\theta_2) \\ &\quad +(m_1+m_2)gl_1\cos(\theta_1) + m_2 g l_2\cos(\theta_2) \ .\end{aligned} \tag{2.15}$$

Therefore for θ_1

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_1} &= m_1 l_1^2\dot{\theta}_1 + m_2 l_1^2\dot{\theta}_1 + m_2 l_1 l_2\dot{\theta}_2\cos(\theta_1-\theta_2) \ , \\ \frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_1} &= (m_1+m_2)l_1^2\ddot{\theta}_1 + m_2 l_1 l_2\ddot{\theta}_2\cos(\theta_1-\theta_2) - m_2 l_1 l_2\dot{\theta}_2\sin(\theta_1-\theta_2)(\dot{\theta}_1-\dot{\theta}_2) \\ \frac{\partial\mathcal{L}}{\partial\theta_1} &= -l_1 g(m_1+m_2)\sin(\theta_1) - m_2 l_1 l_2\dot{\theta}_1\dot{\theta}_2\sin(\theta_1-\theta_2)\end{aligned} \tag{2.16}$$

holds. The Euler-Lagrangian differential equation

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_1} - \frac{\partial\mathcal{L}}{\partial\theta_1} = 0 \tag{2.17}$$

is then

$$(m_1+m_2)l_1^2\ddot{\theta}_1 + m_2 l_1 l_2\ddot{\theta}_2\cos(\theta_1-\theta_2) + m_2 l_1 l_2\dot{\theta}_2^2\sin(\theta_1-\theta_2) + l_1 g(m_1+m_2)\sin(\theta_1) = 0 \tag{2.18}$$

and dividing by l_1 yields

$$(m_1+m_2)l_1\ddot{\theta}_1 + m_2 l_2\ddot{\theta}_2\cos(\theta_1-\theta_2) + m_2 l_2\dot{\theta}_2^2\sin(\theta_1-\theta_2) + g(m_1+m_2)\sin(\theta_1) = 0 \ . \tag{2.19}$$

Similar, for θ_2 the following expressions are obtained

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_2} &= m_2 l_2^2\dot{\theta}_2 + m_2 l_1 l_2\dot{\theta}_1 + \cos(\theta_1-\theta_2) \ , \\ \frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_2} &= m_2 l_2^2\ddot{\theta}_2 + m_2 l_1 l_2\ddot{\theta}_1\cos(\theta_1-\theta_2) - m_2 l_1 l_2\dot{\theta}_1\sin(\theta_1-\theta_2)(\dot{\theta}_1-\dot{\theta}_2) \ , \\ \frac{\partial\mathcal{L}}{\partial\theta_2} &= m_2 l_1 l_2\dot{\theta}_1\dot{\theta}_2\sin(\theta_1-\theta_2) - l_2 m_2 g\sin(\theta_2) \ .\end{aligned} \tag{2.20}$$

The Euler-Lagrangian differential equation

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\theta}_2} - \frac{\partial\mathcal{L}}{\partial\theta_2} = 0 \tag{2.21}$$

is then

$$m_2 l_2^2 \ddot{\theta}_2 + m_2 l_1 l_2 \ddot{\theta}_1 \cos(\theta_1 - \theta_2) - m_2 l_1 l_2 \dot{\theta}_1^2 \sin(\theta_1 - \theta_2) + l_2 m_2 g \sin(\theta_2) = 0 \tag{2.22}$$

and dividing by l_2 yields

$$m_2 l_2 \ddot{\theta}_2 + m_2 l_1 \ddot{\theta}_1 \cos(\theta_1 - \theta_2) - m_2 l_1 \dot{\theta}_1^2 \sin(\theta_1 - \theta_2) + m_2 g \sin(\theta_2) = 0 \ . \tag{2.23}$$

From the two Euler-Lagrangian differential equations derived above, the following system of first order differential equations is obtained when setting $\theta_3 = \dot{\theta_1}$ and $\theta_4 = \dot{\theta_2}$:

$$\begin{aligned}
\dot{\theta_1} &= \theta_3 \ , \\
\dot{\theta_2} &= \theta_4 \ , \\
\dot{\theta_3} &= \frac{-(g\sin(\theta_1)m_1 + g\sin(\theta_1)m_2 + m_2 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2^2 - \cos(\theta_1 - \theta_2)m_2 g \sin(\theta_2))}{l_1(m_1 + m_2 - m_2 \cos^2(\theta_1 - \theta_2))} \\
&\quad - \frac{(\cos(\theta_1 - \theta_2)m_2 l_1 \sin(\theta_1 - \theta_2)\dot{\theta}_1^2)}{l_1(m_1 + m_2 - m_2 \cos^2(\theta_1 - \theta_2))} \ , \\
\dot{\theta_4} &= \frac{(\cos(\theta_1 - \theta_2)g\sin(\theta_1)m_1 + \cos(\theta_1 - \theta_2)g\sin(\theta_1)m_2 + \cos(\theta_1 - \theta_2)m_2 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2^2)}{l_2(m_1 + m_2 - m_2 \cos^2(\theta_1 - \theta_2))} \\
&\quad + \frac{(-m_1 g \sin(\theta_2) + m_1 l_1 \sin(\theta_1 - \theta_2)\dot{\theta}_1^2 - g m_2 \sin(\theta_2) + m_2 l_1 \sin(\theta_1 - \theta_2)\dot{\theta}_1^2)}{l_2(m_1 + m_2 - m_2 \cos^2(\theta_1 - \theta_2))} \ .
\end{aligned} \tag{2.24}$$

The friction forces are neglected here. The chaotic behavior of this system results especially from the disregard of the velocity proportional friction forces.

A distance sensor, which has been installed at a fixed position, measures the distance of the balls at the end of each pendulum (see Fig. 2.3). For r_1 and r_2

$$\begin{aligned}
r_1 &= \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \ , \\
r_2 &= \sqrt{(x_2 - x_0)^2 + (y_2 - y_0)^2}
\end{aligned} \tag{2.25}$$

holds. Rewriting these equation in terms of θ_1 and θ_2 yields

$$\begin{aligned}
r_1 &= \sqrt{(l_1 \sin(\theta_1) - x_0)^2 + (-l_1 \cos(\theta_1) - y_0)^2} \ , \\
r_2 &= \sqrt{(l_1 \sin(\theta_1) + l_2 \sin(\theta_2) - x_0)^2 + (-l_1 \cos(\theta_1) - l_2 \cos(\theta_2) - y_0)^2} \ .
\end{aligned} \tag{2.26}$$

Thus, θ_1 and θ_2 can be reconstructed from the measurements of r_1 and r_2. The angular velocities θ_3 and θ_4 have to be estimated.

2.4 Biological Waste Water Treatment Plant

The task of a biological waste water treatment plant (BWTP) [25,29] is to remove the culture medium characteristic of waste water and to eliminate polluting substances or reduce them below a required measure. Modern wastewater treatment plants are complex biochemical systems consisting of several activated sludge tanks, also called aeration tanks, and settler

tanks. Purification takes place in the aeration tanks, where heterotrophic and autotrophic bacteria consume biodegradable matter. The ASM1 (Activated Sludge Model No. 1) of the IWA (International Water Association) is a suitable mathematical description of wastewater treatment processes. It consists of a system of 13 coupled nonlinear ordinary differential equations for the aeration tank together with a multi-layer model for the settler. Purification processes include reduction of organic matter (substrate) and removal of nitrogen fractions from the wastewater. They are characterized by additive nonlinear kinetics describing the growth rates of bacteria. Uncertainties of the system parameters in the ASM1 are caused by disturbances in composition and amount of the influent wastewater as well as changing temperature and weather conditions. These uncertainties significantly influence the system dynamics and, hence, also the performance of the complete wastewater treatment plant. In this thesis a subsystem of the ASM1 is considered including only reduction of organic matter as described by the block diagram in Fig. 2.4.

Heterotrophic bacteria need organic carbon compounds (substrate) for the synthesis of their cell components and for the covering of their energy demand. The aeration tank in Fig. 2.4 is considered as a continuous ideal stirred tank reactor which is fed with wastewater. The degradation via mineralization can approximately be described by a mass balance of the fast degradable substrate $S(t)$:

$$\dot{S} = D\,(S_W - S) - \rho_S, \quad \rho_S = \frac{\rho_H}{Y_H} \ , \tag{2.27}$$

where $D = \frac{Q_W}{V_A}$ is the dilution rate, which is the time-varying influent wastewater flow rate Q_W divided by the constant reaction volume V_A of the aeration tank, and S_W is the influent biodegradable substrate concentration in the waste water. The reaction rate is given in form of a Monod kinetic:

$$\begin{aligned} \rho_H &= \mu_H(S, S_O)X \ , \\ \mu_H\,(S, S_O) &= \hat{\mu}_H \frac{S}{S + K_S} \frac{S_O}{S_O + K_{OS}} \ . \end{aligned} \tag{2.28}$$

The change of the Substrate concentration $S(t)$ over time depends on the inflow and outflow and on the degradation through the heterotrophic bacteria with concentration $X(t)$. Y_H is the yield coefficient and $\hat{\mu}_H = \mu_{Hmax}$ the maximum specific growth rate. K_S and K_{OH} are the half saturation coefficients of the reaction kinetics. The balance equation for the heterotrophic bacteria results in:

$$\dot{X} = D\,(X_{in} - X) + (\mu_H - b_H)X \ , \tag{2.29}$$

where X_{in} is the influent bacteria concentration. The last term in this equation describes the net growth of the organism (growth minus decay). The oxygen which is used for mineralization given by the concentration $S_O(t)$ is fed in externally by the influent oxygen flow rate S_{OW} in the waste water and the control variable u_{O2}. The oxygen attrition rate is given by ρ_O. The oxygen balance results in

$$\dot{S_O} = D\,(S_{OW} - S_O) - \rho_O + (1 - \frac{S_O}{S_{O_{Sat}}})u_{O2} \ , \tag{2.30}$$

where S_{OW} is the influent oxygen concentration in the waste water. Oxygen can only be fed in up to a temperature and pressure dependent saturation concentration $S_{O_{sat}}$. The oxygen attrition rate ρ_O can be described by:

$$\rho_O = \rho_H(1 - Y_H)/Y_H \ . \tag{2.31}$$

The settler tank has also to be modeled. The inflow concentration in the aeration tank X_{in} is assumed to be depending only on the concentration X_{RS} of the heterotrophic bacteria in the feedback sludge, which is equivalent to the concentration X_{Set} in the settler. Hence, $X_{Set} = X_{in} = X_{RS}$. Assuming an ideal separation of the purified waste water and the activated sludge the balance equation for the settler is

$$V_{Set}\dot{X}_{Set} = ((Q_W + Q_{RS})\, X - (Q_{EX} + Q_{RS})\, X_{Set}) \ . \tag{2.32}$$

V_{Set} is the volume of the settler, Q_{RS} is the flow rate of the return sludge, and Q_{EX} is the flow rate of the excess sludge, which is removed from the waste water in the settler.
These equations can be summarized by a set of four nonlinear coupled differential equations:

$$\begin{aligned}
\dot{S} &= \frac{Q_W}{V_A}(S_W - S) - \mu_H(S, S_O)\frac{1}{Y_H}X, \\
\dot{X} &= -\frac{Q_W}{V_A}X + \frac{Q_{RS}}{V_A}(X_{Set} - X) + (\mu_H(S, S_O) - b)\, X, \\
\dot{S}_O &= \frac{Q_W}{V_A}(S_{OW} - S_O) - \mu_H(S, S_O)\frac{1 - Y_H}{Y_H}X + \frac{\rho_{O2}}{V_A}\left(1 - \frac{S_O}{S_{O,sat}}\right)u_{O2}, \\
\dot{X}_{Set} &= \frac{((Q_W + Q_{RS})\, X - (Q_{EX} + Q_{RS})\, X_{Set})}{V_{Set}} \ .
\end{aligned} \tag{2.33}$$

The four state variables represent the concentration S of biologically degradable organic matter (substrate), the concentration X of substrate consuming bacteria in the aeration tank, the concentration S_O of dissolved oxygen in the aeration tank, and the concentration X_{Set} of bacteria in the settler (see Tab. 2.2). For the state variables the following restrictions must hold additionally,

$$\begin{aligned}
&S \geq 0, X \geq 0, X_{Set} \geq 0, \\
&0 \leq S_O \leq S_{O,Sat}.
\end{aligned} \tag{2.34}$$

These inequalities point out the physical restrictions, namely, all concentrations have to be non-negative for all times. Furthermore, the concentration S_O of dissolved oxygen is limited by the saturation concentration $S_{O,Sat}$. The nominal values of the system parameters are shown in Tab. 2.3. The BWTP is a stiff system, since the time constant for the oxygen concentration is much smaller than the time constants of the other state variables.

The oxygen concentration S_O and the substrate concentration S are measured [25, 29], both bacteria concentrations X and X_{Set} have to be estimated. Knowledge about the state variables is especially important in safety critical systems like a BWTP. For example the substrate concentration S in the plant has to be below a specified value after a given time-span. S can be measured, but since these measurements are usually affected by uncertainty the combination of the measurements and the knowledge about the dynamical behavior of the system in the estimation process leads often to an improved estimation of S reducing the uncertainty of the measurement.

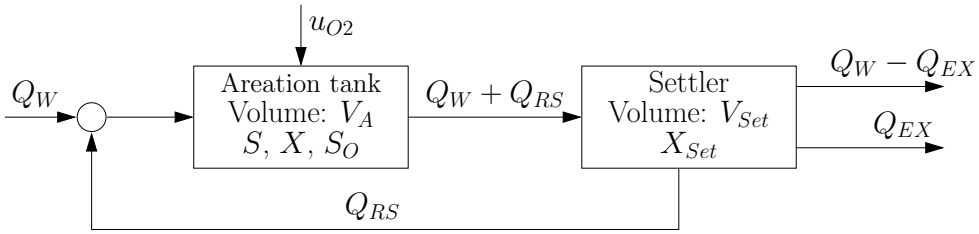


Figure 2.4: Block diagram of a biological waste water treatment plant.

Table 2.2: System state variables.

State variable	Physical meaning	unit
S	Substrate concentration	kg/m^3
X	Concentration of heterotrophic bacteria	kg/m^3
S_O	Concentration of dissolved oxygen (O_2)	kg/m^3
X_{Set}	Bacteria concentration in the settler	kg/m^3

Table 2.3: Nominal values of the BWTP system parameters.

Parameter	Physical meaning	Nominal value
V_A	volume of the aeration tank	8000 m^3
V_{Set}	volume of the settler	4545 m^3
Q_W	influent waste water flow rate	0.153 m^3/s
Q_{RS}	flow rate of return sludge	0.0916 m^3/s
Q_{EX}	flow rate of excess sludge	0.005 m^3/s
S_W	influent biodegradable substrate concentration	0.616 kg/m^3
S_{OW}	influent oxygen concentration in the wastewater	$0.5 \cdot 10^{-3}$ kg/m^3
$S_{O,Sat}$	saturation concentration of dissolved oxygen	$5.3 \cdot 10^{-3}$ kg/m^3
Y_H	yield coefficient of heterotrophic biomass	0.67 kg CSB/kg CSB
$\hat{\mu}_H$	max. specific growth rate of heterotrophic biomass	1/14400 1/s
b	specific decay rate of heterotrophic biomass	$7.176 \cdot 10^{-6}$ $1/s$
K_S	half saturation coefficient for heterotrophic biomass	0.02 kg/m^3
K_{OS}	oxygen half saturation coefficient	$2 \cdot 10^{-4}$ kg/m^3
u_{O2}	influent oxygen flow rate (here: constant)	1.487 m^3/s
ρ_{O2}	normal density of molecular oxygen	1.428 kg/m^3

2.5 Magnetic Levitation System

In Fig. 2.5 a magnetic levitation is shown [41]. This example is used for the verified robustness analysis in Chapter 6 and the verified state and parameter estimation in closed loop control in Chapter 7. It is a popular benchmark example in nonlinear control.
The system consists of a ferromagnetic sphere, which is located under an electromagnet. The

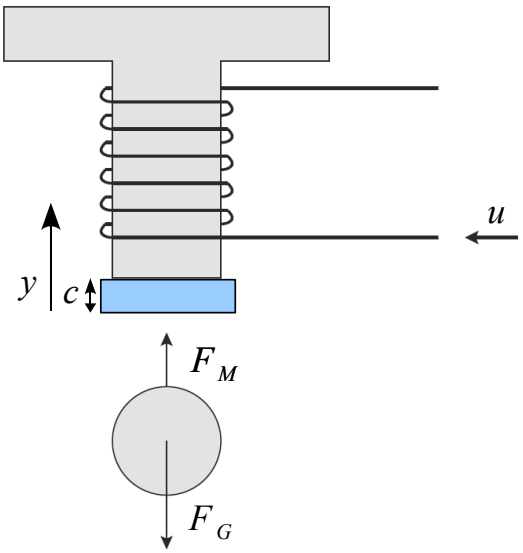


Figure 2.5: Magnetic levitation system.

electromagnet is fed by the coil current u. Because of a mechanical stopper, the distance of the surface of the sphere from the brace of the electromagnet is at least c. The sphere is affected by the magnetic force F_M in upward direction. The gravity force F_G affects the sphere in opposite direction. With

$$m\ddot{y} = F_M - F_G \tag{2.35}$$

the force balance equation is then given by

$$m\ddot{y} = \frac{ku^2}{(c-y)^2} - mg \ . \tag{2.36}$$

The parameter k describes the permeability of the air gap, the properties of the iron core and other influences affecting the system. The coil current u is the control variable. With $x_1 = y$ and $x_2 = \dot{y}$ the state space representation

$$\dot{x}_1 = x_2 \ , \tag{2.37}$$

$$\dot{x}_2 = \frac{k}{m}\frac{u^2}{(c-x_1)^2} - g \tag{2.38}$$

is obtained. Here, the position x_1 is measured and the velocity has to be estimated.

3 Interval Analysis and Taylor Models

First publications about interval arithmetic appeared in 1924 and 1931 [6, 74]. The development of modern interval arithmetic began in 1966 with the book of R. E. Moore [50] and it became an important tool to calculate verified enclosures of the solutions of numerical problems.

3.1 Interval Computation

In this section the basic rules and definitions for calculation with intervals are introduced [23, 27, 50, 51]. A real interval $[x]$ is a connected subset of $\mathbb{R}$

$$[x] = [\underline{x}; \overline{x}] \subset \mathbb{R} : \ \underline{x} \leq x \leq \overline{x}, \tag{3.1}$$

where $\underline{x}$ is called the *infimum* or *lower bound* and $\overline{x}$ is called the *supremum* or *upper bound* of the interval $[x]$. In addition also the notations

$$\inf([x]) := \underline{x} \quad \text{and} \quad \sup([x]) := \overline{x} \tag{3.2}$$

are used in this work.

The set of all intervals over $\mathbb{R}$ is denoted by $\mathbb{IR}$ where

$$\mathbb{IR} = [\underline{x}; \overline{x}] : \ \forall \, \underline{x}, \ \overline{x} \in \mathbb{R}, \ \underline{x} \leq \overline{x} \ . \tag{3.3}$$

Other important characteristic values of an interval are the *midpoint* of $[x]$,

$$\text{mid}([x]) = \frac{1}{2}(\underline{x} + \overline{x}) \ , \tag{3.4}$$

the *diameter* of $[x]$,

$$\text{diam}([x]) = \overline{x} - \underline{x} \ , \tag{3.5}$$

and the *radius* of $[x]$,

$$\text{rad}([x]) = \frac{1}{2}(\overline{x} - \underline{x}) \ . \tag{3.6}$$

The midpoint a and radius r of an interval can represent an interval as $\langle a, r \rangle$.
If an interval has zero diameter it is called a *point interval* and contains only a single point.

Infimum and supremum always have to be chosen such that they are floating point numbers enclosing the considered real value of $x \in \mathbb{R}$.

The *absolute* value of $[x]$ is defined by

$$|[x]| = \max\{|\underline{x}|\,;|\overline{x}|\} \quad . \tag{3.7}$$

All set-theoretic operations can be applied to intervals. The *intersection* of two non-empty intervals $[x]$ and $[y]$ is also an interval and defined as

$$[x] \cap [y] = \begin{cases} \left[\max\left\{\underline{x},\underline{y}\right\}; \min\left\{\overline{x},\overline{y}\right\}\right], & \text{if } \max\left\{\underline{x},\underline{y}\right\} \leq \min\left\{\overline{x},\overline{y}\right\} \quad , \\ \emptyset, & \text{otherwise} \quad . \end{cases} \tag{3.8}$$

The *union* operation is defined in a different way in interval arithmetic and is called the *interval hull* which is defined as

$$[x] \cup [y] = \left[\min\left\{\underline{x},\underline{y}\right\}; \max\left\{\overline{x},\overline{y}\right\}\right] \quad . \tag{3.9}$$

For instance, the interval hull of $[2;3] \cup [5;7]$ is the interval $[2;7]$, since the result has to be a connected subset of $\mathbb{R}$.

An interval $[x]$ is a *subset* of an interval $[y]$, $[x] \subseteq [y]$, if and only if $\underline{y} \leq \underline{x}$ and $\overline{y} \geq \overline{x}$.

If α is a real number and $[x]$ a non-empty interval, then multiplication of a real number and the interval gives

$$\alpha \cdot [x] = \begin{cases} [\alpha\underline{x}; \alpha\overline{x}] & \text{if} \quad \alpha \geq 0 \quad , \\ [\alpha\overline{x}; \alpha\underline{x}] & \text{if} \quad \alpha < 0 \quad . \end{cases} \tag{3.10}$$

The four classical operations of real arithmetic, namely addition $(+)$, subtraction $(-)$, multiplication $(\cdot)$, and division $(/)$ can be extended to intervals. Interval arithmetic operations are defined on $\mathbb{IR}$ such that the interval result encloses all possible real results. Given the two intervals $[x] = [\underline{x};\overline{x}]$ and $[y] = [\underline{y};\overline{y}]$ the four elementary operations are defined by

$$\begin{aligned} [x] + [y] &= \left[\underline{x} + \underline{y}\ ;\ \overline{x} + \overline{y}\right] \quad , \\ [x] - [y] &= \left[\underline{x} - \overline{y}\ ;\ \overline{x} - \underline{y}\right] \quad , \\ [x] \cdot [y] &= \left[\min\left\{\underline{x}\,\underline{y}, \underline{x}\,\overline{y}, \overline{x}\,\underline{y}; \overline{x}\,\overline{y}\right\}; \max\left\{\underline{x}\,\underline{y}, \underline{x}\,\overline{y}, \overline{x}\,\underline{y}, \overline{x}\,\overline{y}\right\}\right] \quad , \\ \frac{[x]}{[y]} &= [x] \cdot \left[\frac{1}{\overline{y}}\ ;\ \frac{1}{\underline{y}}\right] \quad \text{for} \quad 0 \notin y \quad . \end{aligned} \tag{3.11}$$

In definition (3.11) division by an interval containing zero is not defined. In *extended interval arithmetic* this operation is defined as

$$\frac{[x]}{[y]} = \begin{cases} \left[\overline{x}/\underline{y}; \infty\right] & \text{if } \overline{x} \leq 0 \text{ and } \overline{y} = 0 \quad , \\ [-\infty; \overline{x}/\overline{y}] \cup \left[\overline{x}/\underline{y}; \infty\right] & \text{if } \overline{x} \leq 0 \text{ and } \underline{y} < 0 < \overline{y} \quad , \\ [-\infty; \overline{x}/\overline{y}] & \text{if } \overline{x} \leq 0 \text{ and } \underline{y} = 0 \quad , \\ [-\infty; \infty] & \text{if } \underline{x} \leq 0 \leq \overline{x} \quad , \\ \left[-\infty; \underline{x}/\underline{y}\right] & \text{if } \underline{x} \geq 0 \text{ and } \overline{y} = 0 \quad , \\ \left[-\infty; \underline{x}/\underline{y}\right] \cup [\overline{x}/\overline{y}; \infty] & \text{if } \underline{x} \geq 0 \text{ and } \underline{y} \leq 0 \leq \overline{y} \quad , \\ [\underline{x}/\overline{y}; \infty] & \text{if } \underline{x} \geq 0 \text{ and } \underline{y} = 0 \quad . \end{cases} \tag{3.12}$$

Addition and subtraction of infinite or semi-infinite intervals are given by

$$\begin{aligned}
[\underline{x};\overline{x}] + [-\infty;\overline{y}] &= [-\infty;\overline{x}+\overline{y}] \quad , \\
[\underline{x};\overline{x}] + \left[\underline{y};\infty\right] &= \left[\underline{x}+\underline{y};\infty\right] \quad , \\
[\underline{x};\overline{x}] + [-\infty;\infty] &= [-\infty;\infty] \quad , \\
[\underline{x};\overline{x}] - [-\infty;\overline{y}] &= [\underline{x}-\overline{y};\infty] \quad , \\
[\underline{x};\overline{x}] - \left[\underline{y};\infty\right] &= \left[-\infty;\overline{x}-\underline{y}\right] \quad , \\
[\underline{x};\overline{x}] - [-\infty;\infty] &= [-\infty;\infty] \quad .
\end{aligned} \tag{3.13}$$

The properties of the basic operators for intervals differ from their properties in $\mathbb{R}$. For example $[x]-[x]$ is generally not equal to $[0;0]$. This is because $[x]-[x] = \{x-y \mid x \in [x],\ y \in [x]\}$ rather than $\{x-x \mid x \in [x]\}$. The subtraction of interval values thus does not take into account that both operands are identical in this example. This effect is called *dependency effect* or *dependency problem*.

Addition and multiplication remain associative and commutative. However, multiplication is no longer distributive with respect to addition. Instead

$$[x]\,([y]+[z]) \subseteq [x][y] + [x][z] \quad , \tag{3.14}$$

a property known as *subdistributivity* holds, which is a direct consequence of the dependency effect, as $[x]$ appears only once on the left-hand side but twice on the right-hand side. As a result, it is recommended to factorize expanded forms as much as possible.

Interval Vectors and Matrices

An *interval vector* $[\mathbf{x}]$ is a subset of $\mathbb{R}^n$ that can be defined as the Cartesian product of n intervals. The interval vector is simply called a *box* and written as

$$[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n], \quad \text{with } [x_i] = [\underline{x}_i;\overline{x}_i] \in \mathbb{IR}, \quad i = 1,\ldots n \quad . \tag{3.15}$$

Its i-th interval component $[x_i]$ is the projection of $[\mathbf{x}]$ onto the i-th axis. The set of all n-dimensional boxes is denoted by $\mathbb{IR}^n$. Non-empty boxes are n-dimensional axis-aligned parallelepipeds. Figure 3.1 illustrates the case $n = 2$, with $[\mathbf{x}] = [x_1] \times [x_2] \in \mathbb{IR}^2$.

Many of the previously introduced notations for intervals are directly applicable to boxes. The *lower bound* (*infimum*) of an interval vector $[\mathbf{x}]$ is a vector consisting of the lower bounds of all its interval components

$$\underline{\mathbf{x}} = [\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_n]^T \quad . \tag{3.16}$$

Similarly, the *upper bound* (*supremum*) of an interval vector $[\mathbf{x}]$ is the vector

$$\overline{\mathbf{x}} = [\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n]^T \quad . \tag{3.17}$$

The *midpoint* of $[\mathbf{x}]$ is

$$\mathrm{mid}([\mathbf{x}]) = [\mathrm{mid}([x_1]), \mathrm{mid}([x_2]), \ldots \mathrm{mid}([x_n])]^T \quad . \tag{3.18}$$

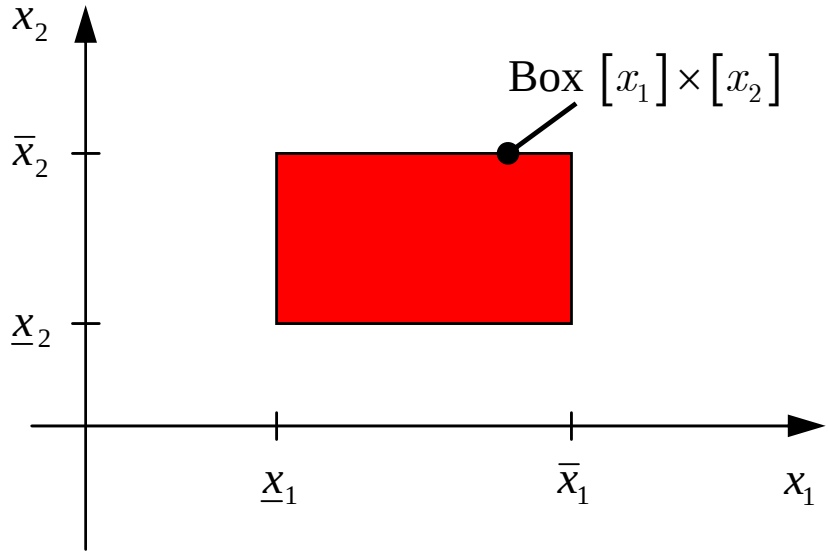


Figure 3.1: Interval vector for n=2.

The *diameter* of an interval vector $[\mathbf{x}]$ is the vector consisting of the diameters of all its components

$$\mathrm{diam}([\mathbf{x}]) = [\mathrm{diam}([x_1]), \mathrm{diam}([x_2]), \ldots \mathrm{diam}([x_n])]^T \quad . \tag{3.19}$$

Analogously, an $(m \times n)$-dimensional *interval matrix* is defined as

$$[\mathbf{A}] = \begin{bmatrix} [a_{11}] & \ldots & [a_{1n}] \\ \vdots & & \vdots \\ [a_{m1}] & \ldots & [a_{mn}] \end{bmatrix} \quad . \tag{3.20}$$

The space of all $m \times n$ matrices is denoted by $\mathbb{IR}^{m \times n}$.

The *lower bound* (*infimum*) of an interval matrix $[\mathbf{A}]$ is the matrix made up with the lower bounds of all its interval components

$$\underline{\mathbf{A}} = \begin{bmatrix} \underline{a}_{11} & \ldots & \underline{a}_{1n} \\ \vdots & & \vdots \\ \underline{a}_{m1} & \ldots & \underline{a}_{mn} \end{bmatrix} \tag{3.21}$$

and similarly, the *upper bound* (*supremum*) of $\mathbf{A}$ is

$$\overline{\mathbf{A}} = \begin{bmatrix} \overline{a}_{11} & \ldots & \overline{a}_{1n} \\ \vdots & & \vdots \\ \overline{a}_{m1} & \ldots & \overline{a}_{mn} \end{bmatrix} \quad . \tag{3.22}$$

The *midpoint* of the matrix $[\mathbf{A}]$ is given by

$$\operatorname{mid}([\mathbf{A}]) = \begin{bmatrix} \operatorname{mid}([a_{11}]) & \dots & \operatorname{mid}([a_{1n}]) \\ \vdots & & \vdots \\ \operatorname{mid}([a_{m1}]) & \dots & \operatorname{mid}([a_{mn}]) \end{bmatrix} . \tag{3.23}$$

The *diameter* of the matrix $[\mathbf{A}]$ is

$$\operatorname{diam}([\mathbf{A}]) = \begin{bmatrix} \operatorname{diam}([a_{11}]) & \dots & \operatorname{diam}([a_{1n}]) \\ \vdots & & \vdots \\ \operatorname{diam}([a_{m1}]) & \dots & \operatorname{diam}([a_{mn}]) \end{bmatrix} . \tag{3.24}$$

Arithmetic operations on interval vectors and matrices are carried out according to the operations on $\mathbb{IR}$ in the same way that real vector and matrix operations are carried out according to real operations. The intersection, interval hull, and comparison of interval vectors and matrices are carried out component wise.

In the following, the notation of scalars and vectors will be simplified. For example, a vector $\mathbf{x}$ is from now on denoted as x and its components with subscript index as x_i. Where there are exceptions, extra notations will be mentioned within the corresponding section.

Evaluation of Functions with Interval Arguments

The range of f over an interval vector $[x] = [[x_1], [x_2], \dots, [x_n]]^T$is defined by

$$\operatorname{Rg}(f; [x]) = \{f(x) | x \in [x]\} . \tag{3.25}$$

$\operatorname{Rg}(f; [x])$ is the exact mapping for the interval vector $[x]$ by the function f and will in the following be denoted by $f([x])$. A important problem in interval arithmetic is to compute an enclosure for $f([x])$. This enclosure should be as tight as possible.

The interval arithmetic evaluation of f on $[x]$ is obtained by replacing each occurrence of a real variable with a corresponding interval, by replacing the standard functions with enclosures of their ranges. The interval bound of this exact mapping $f([x])$ is given by $[f]([x])$. The natural inclusion function $[f_N]([x])$ is obtained by performing the previously introduced interval arithmetic operations instead of the real operations. This kind of evaluation is also called naive or natural interval evaluation.

Another important property of interval arithmetic is the *inclusion monotonicity*

$$\begin{aligned} [x_i] \subseteq [y_i] \;\; (i = 1, \dots, n) \\ \Rightarrow [f]([x_1], \dots, [x_n]) \subseteq [f]([y_1], \dots, [y_n]) . \end{aligned} \tag{3.26}$$

If an interval $[x]$ is split into subintervals $[x^{(l)}]$, $l = 1, \dots, L$ with

$$\bigcup_{l=1}^{L} [x^{(l)}] = [x], \text{with} \quad \operatorname{intr}\left\{[x^{(l_1)}]\right\} \cap \operatorname{intr}\left\{[x^{(l_2)}]\right\} = \emptyset, \quad (l_1 \neq l_2) . \tag{3.27}$$

inclusion monotonicity yields

$$\bigcup_{l=1}^{L}[f]([x^{(l)}]) \subseteq [f]([x]) \ . \tag{3.28}$$

The operator intr $\{\cdot\}$ denotes the interior of the corresponding interval.

In practice, $[f]([x])$ is not unique, because it depends on how f is evaluated in interval arithmetic. For example, expressions that are mathematically equivalent for scalars, such as $x(y+z)$ and $xy+xz$, may have different values if x, y, and z are intervals. No matter how $[f]([x])$ is evaluated, it follows from the inclusion monotonicity of the interval operations that

$$f([x]) \subseteq [f]([x]) \ . \tag{3.29}$$

Elementary functions such as exp, tan, sin, cos etc. extend to intervals [23,27,50,51]. Interval extensions of these functions are implemented in various function libraries [39, 45, 67, 68, 72].

3.2 Rounding in Interval Analysis

Digital computers basically have a limited computational accuracy. Therefore one has to take care when interval computation is performed on a computer. Consider the addition

$$[0.123; 0.456] + [0.0116; 0.0214] = [0.1346; 0.4774] \ . \tag{3.30}$$

Using a computational precision of 3 decimal digits the result is usually rounded to the next presentable number. In this case

$$[0.135; 0.477] \not\supseteq [0.1346; 0.4774] \tag{3.31}$$

is obtained. As it can be seen the exact interval is not contained in the rounded interval. To cope with this problem a so called *directed rounding* [24] has to be applied. For this example

$$[0.134; 0.478] \supset [0.1346; 0.4774] \tag{3.32}$$

is obtained.

3.3 Interval Newton Methods

Interval Newton methods [2, 22, 23, 55] are extensions of the Newton method for the solution of nonlinear systems of equations and have the following properties:

1. The existence of a solution can be guaranteed.

2. The non-existence of a solution can be guaranteed.

3. The method converges quadratically.

From the various interval Newton methods the Krawczyk method [55] is discussed in more detail. Consider a nonlinear function

$$f : \mathbb{R}^n \mapsto \mathbb{R}^n \tag{3.33}$$

and the system of equations

$$f(x) = 0 \quad \text{with} \quad x \in [x] \ . \tag{3.34}$$

The iteration rule for the Krawczyk-method is given by

$$[K]([x^{it}]) = x_m - Y f(x_m) + \left(\mathfrak{I} - Y \left[\frac{\partial f}{\partial x}\Big|_{x=[x^{it}]}\right]\right)([x^{it}] - x_m), \tag{3.35}$$

with $Y^{-1} \in \left[\frac{\partial f}{\partial x}\Big|_{x=[x^{it}]}\right]$, $x_m = \text{mid}([x^{it}])$ and it denotes the it-th iteration. The iteration is initialized by

$$[x^0] = [x]. \tag{3.36}$$

The iteration

$$[x^{it+1}] = [K]([x^{it}]) \cap [x^{it}] \tag{3.37}$$

is called Krawczyk iteration and leads to an iterative improvement of the enclosure of the solution set.

If

$$[K]([x^{it}]) \subset [x^{it}] \tag{3.38}$$

than it can be guaranteed, that a zero exists in the box $[x^{it}]$. On the other hand if

$$[K]([x^{it}]) \cap [x^{it}] = \emptyset \tag{3.39}$$

than it can be guaranteed, that no zero exists in $[x^{it}]$ and therefore also not in $[x^0]$.

For

$$[K]([x^{it}]) \not\subset [x^{it}] \quad \text{and} \quad [K]([x^{it}]) \cap [x^{it}] \neq \emptyset \tag{3.40}$$

the iteration is continued. However, if $[x^{it+1}] \approx [x^{it}]$ and hence no further tightening of the solution set is achieved the iteration is stopped.

3.4 Dependency Problem and Wrapping Effect

Interval arithmetic is sometimes affected by overestimation [23, 27, 50]. It often occurs in case of the previously described natural interval evaluation.

3.4.1 Dependency Problem

Overestimation is often caused by the *dependency problem* [23, 27, 50], which is the lack of interval arithmetic to identify different occurrences of the same variable. The probably simplest example, which has already been mentioned in Section 3.1, is the operation $x - x$, which is zero for all $x \in \mathbb{R}$. If $x - x$ is evaluated over the interval $[x] = [1; 2]$ by replacing the algebraic operation with their interval equivalents

$$[x] - [x] = [1; 2] - [1; 2] = [-1; 1] \tag{3.41}$$

is obtained. Another example is

$$f(x) = -x^2 + 4 \cdot x + 1 \ . \tag{3.42}$$

An evaluation with natural interval methods over the interval $[x] = [1; 3]$ yields

$$-[1; 3] \cdot [1; 3] + 4 \cdot [1; 3] + 1 = [-9; 1] + [4; 12] + 1 = [-9; -1] + [5; 13] = [-4; 12] \ . \tag{3.43}$$

Rewriting f results in

$$f(x) = -(x - 2)^2 + 5 \tag{3.44}$$

and for the enclosure

$$-([x] - 2)^2 + 5 = -(([1; 3] - 2) \cdot ([1; 3] - 2)) + 5 = [-1; 1] + 5 = [4; 6] \ , \tag{3.45}$$

is obtained, which is already significantly tighter. The application of the rule

$$x^n = \begin{cases} [1; 1] & \text{for } n = 0, \\ [\underline{x}^n; \overline{x}^n] & \text{for } \underline{x} \geq 0 \text{ or for } \underline{x} \leq 0 \leq \overline{x} \text{ and odd } n, \\ [\overline{x}^n; \underline{x}^n] & \text{for } \overline{x} \leq 0, \\ [0; \max(\underline{x}^n, \overline{x}^n)] & \text{for } \underline{x} \leq 0 \leq \overline{x} \text{ and even } n \end{cases} \tag{3.46}$$

on $-([x] - 2)^2 + 5$ determines the exact enclosure

$$-([x] - 2)^2 + 5 = -([1; 3] - 2)^2 + 5 = -([-1; 1])^2 + 5 = [4; 5] \ . \tag{3.47}$$

In general algebraic expressions should be simplified as much as possible before the evaluation with interval arguments to reduce the number of dependent variables, resulting in a reduction of the dependency problem.

3.4.2 Wrapping Effect

Another source of overestimation is the so called *wrapping effect*, which was first observed by Moore in 1966 [50]. It appears when intermediate results of a computation are enclosed into intervals, whereas the exact solution set is a complexly shaped region. This can be shown with the following example. Consider the discrete time system

$$x_{k+1} = A \cdot x_k \ . \tag{3.48}$$

with $A = \frac{1}{2}\sqrt{2}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$, perfoming a 45° rotation on x_k.

A natural interval evaluation over three time steps with an initial interval vector

$$[x_0] = \begin{bmatrix} [-1\ ;\ 1] \\ [-1\ ;\ 1] \end{bmatrix} \tag{3.49}$$

is applied. The results are depicted in Fig. 3.2. The exact solution at $k = 1$ is the rotated box, which is enclosed by the box resulting from the interval computation. At $k = 2$ the smaller axes parallel box is the exact solution obtained from the exact solution at $k = 1$. The rotated box is the exact solution resulting from the interval enclosure of the result at $k = 1$, which is again enclosed by an box. As it can be seen the box becomes larger in each time step. If the calculation is continued, the overestimation caused by the wrapping effect will increase exponentially in this example.

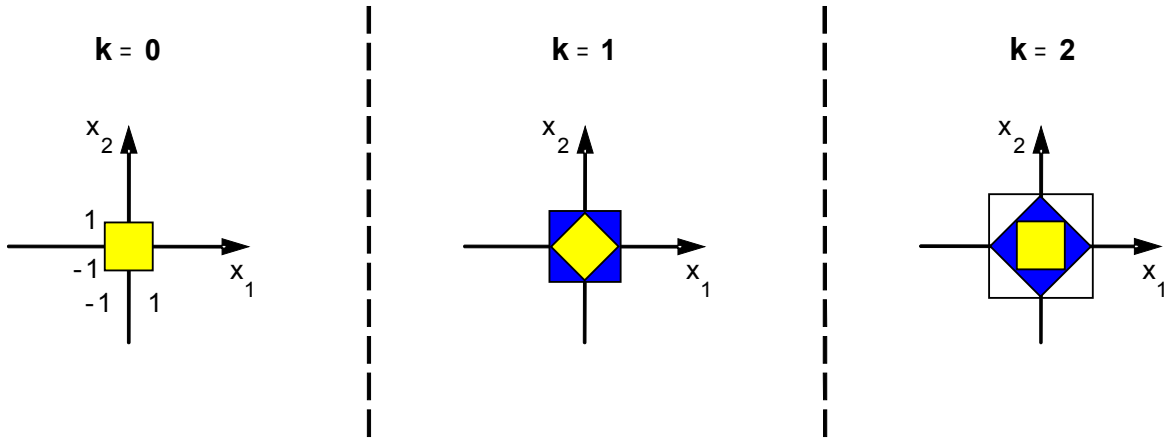


Figure 3.2: Illustration of the wrapping-effect.

The wrapping effect is one major problem in verified integrators of ordinary differential equations (ODEs) as it may lead to accumulation of overestimation over simulation time.

3.5 Optimized Interval Methods

In this section optimized interval methods for reduction of the dependency problem are summarized. Advanced techniques for reduction of the wrapping effect are discussed in Chapter 4.

3.5.1 Taylor Inclusion Functions

The Taylor expansion of order ρ of a function

$$f : \mathbb{R}^n \mapsto \mathbb{R} \tag{3.50}$$

around $\hat{x}$ is defined by

$$f(x) = \sum_{i=0}^{\rho} \left\{ \frac{1}{i!} \left[\sum_{l=1}^{n} (x_l - \hat{x}_l) \frac{\partial}{\partial \tilde{x}} \right]^i f(\tilde{x}) \right\}_{\tilde{x}=\hat{x}} + R(x, \hat{x}, \xi) \quad \xi \in [x, \hat{x}] \tag{3.51}$$

with

$$R(x, \hat{x}, \xi) = \frac{1}{(\rho+1)!} \left[\sum_{l=1}^{n} (x_l - \hat{x}_l) \frac{\partial}{\partial x_l} \right]^{\rho+1} f(\xi). \tag{3.52}$$

If x and $\hat{x}$ are in the interval $[x]$, then the remainder error R is enclosed by [45]

$$R(x, \hat{x}, \xi) \subset [R](x, \hat{x}, [x]) = \frac{1}{(\rho+1)!} \left[\sum_{l=1}^{n} ([x_l] - \hat{x}_l) \frac{\partial}{\partial x_l} \right]^{\rho+1} [f]([x]) \ . \tag{3.53}$$

If the function f is evaluated over an interval vector $[x]$, then

$$\begin{aligned} f([x]) \subseteq [f_T]([x]) = & \sum_{i=0}^{\rho} \left\{ \frac{1}{i!} \left[\sum_{l=1}^{n} ([x_l] - \hat{x}_l) \frac{\partial}{\partial \tilde{x}_l} \right]^i f(\tilde{x}) \right\}_{\tilde{x}=\hat{x}} \\ & + \left\{ \frac{1}{(\rho+1)!} \left[\sum_{l=1}^{n} ([x_l] - \hat{x}_l) \frac{\partial}{\partial x_l} \right]^{\rho+1} [f]([x]) \right\} \end{aligned} \tag{3.54}$$

holds. Where $[f_T]$ is the Taylor inclusion function.

Midpoint Inclusion Function

The so called midpoint-rule inclusion function [23, 27] for $f : \mathbb{R}^n \mapsto \mathbb{R}$ is given by

$$f([x]) \subseteq [f_M]([x]) = f(x_m) + \left[\frac{\partial f}{\partial x} \right]^T ([x]) \cdot ([x] - x_m) \ , \tag{3.55}$$

which is a zero-order Taylor series expansion of the function f at the midpoint $x_m = \text{mid}([x])$ with an interval evaluation of the first-order remainder term, where $\left[\frac{\partial f}{\partial x}\right]([x])$ is an inclusion function of the gradient of f. For vector valued functions $f : \mathbb{R}^n \mapsto \mathbb{R}^m$, in (3.55) the derivative is a Jacobian matrix.
When the width of $[x]$ is small, the effect of the pessimism possibly resulting from the interval evaluation of $\left[\frac{\partial f}{\partial x}\right]([x])$ is reduced by the scalar product with $([x] - x_m)$, which is a small interval centered on zero.
In general it is not a priori clear whether the midpoint-rule gives tighter enclosures than natural interval evaluation, especially if the interval vector $[x]$ has large components [27].
In [8] it is shown that overestimation for a decreasing width of the interval arguments decreases quadratically for the mid-point rule (quadratic approximation order), whereas it only decreases linearly for natural interval evaluation (linear approximation order). This means for sufficient small intervals the mid-point rule yields tighter enclosures. Therefore usually an intersection of both the result of the natural function evaluation and the result of the midpoint-rule is determined, if a function is evaluated for larger intervals. If the order of the expansion is $\rho = 1$, then a cubic approximation order is obtained if the polynomial part can be bounded without overestimation [24].

Examples

Higher order evaluation is often successful in the computation of ranges of expressions which have severe cancellation. Because of dependent intervals, the natural interval arithmetic or the midpoint-rule then suffer from much overestimation. In the following a few scalar examples are studied. The expansion is performed around the midpoint of the interval.

Consider the current-voltage-characteristic of a tunnel-diode which is given by the polynomial function [28]

$$f_1(x) = 17.76x - 103.79x^2 + 229.62x^3 - 226.31x^4 + 83.72x^5 \ . \tag{3.56}$$

This function which is depicted in Fig. 3.3 is evaluated with different orders ρ over different intervals.The expansion was performed in the midpoint of each interval and the obtained diameters of the interval enclosures $[f_{1,T}]([x])$ are listed in Tab. 3.1. The enclosure of the

Table 3.1: Interval diameters for $f_1(x)$.

	naiv	$\rho = 0$	$\rho = 1$	$\rho = 2$	$\rho = 3$	$\rho = 4$	$\rho = 5$
$[x] = [0;1]$	661.2000	$1.1252{\cdot}10^3$	721.7875	268.9688	31.8513	11.9844	11.9844
$[x] = [0;2]$	$8.5876{\cdot}10^3$	$1.8942{\cdot}10^4$	$1.3532{\cdot}10^3$	$7.2461{\cdot}10^3$	$1.2496{\cdot}10^3$	772.1000	772.1000
$[x] = [0;3]$	$4.5201{\cdot}10^4$	$7.8023{\cdot}10^4$	$3.7635{\cdot}10^4$	$1.9886{\cdot}10^4$	$1.0733{\cdot}10^4$	$1.0482{\cdot}10^4$	$1.0482{\cdot}10^4$

0-th order expansion is wider than the enclosure obtained by a naive evaluation. However, with increasing order the enclosure becomes tighter.

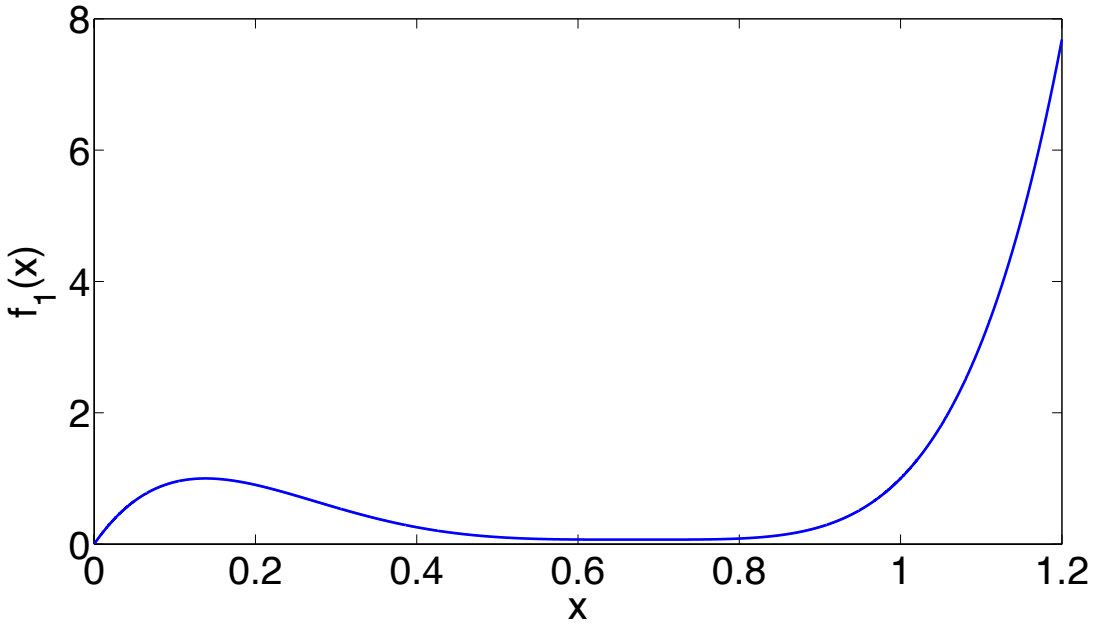


Figure 3.3: Current-Voltage characteristic of a tunnel diode.

Still, it cannot be guaranteed that increasing the order leads always to tighter results, especially if the intervals become wider [56]. This is illustrated with the function

$$f_2(x) = \frac{1}{1+x} + \frac{1}{1-x} - \frac{2}{1-x^2} \tag{3.57}$$

which is equal zero for all x. In Tab. 3.2 the bounds obtained for different orders for the expansion around the interval midpoint and different intervals are shown. For the interval $[x] = [2.9; 3.1]$ the diameter of the resulting enclosure becomes smaller if the order is increased. However, if f_2 is evaluated over a larger interval $[x] = [2; 4]$ increasing the order has the opposite effect.

Table 3.2: Interval diameters for $f_2(x)$.

	naiv	$\rho = 0$	$\rho = 1$	$\rho = 2$	$\rho = 3$	$\rho = 4$
$[x] = [2.9; 3.1]$	0.1003	0.01296	0.00133	$1.4931 \cdot 10^{-4}$	$1.4215 \cdot 10^{-5}$	$1.5731 \cdot 10^{-6}$
$[x] = [2; 4]$	1.3333	3.5556	5.9366	25.2745	39.5233	180.9808

Taylor forms lose their advantages when no cancellation is present, or when some higher order terms in Taylor expansion have a large coefficient [56].

For some arithmetic expressions, Taylor forms even generate dependence. This can be seen in examples such as

$$\frac{1}{1-x} = 1 + x + x^2 + \cdots + \frac{x^{2k}}{1-x} \ , \tag{3.58}$$

where the original expression has no dependence. In such cases, the interval evaluation gives exact results, while Taylor forms produce overestimation. In this example much of the dependence is of a particularly simple kind, namely additive except in the remainder term.

The function

$$f_3(x) = (1+x)^4 \tag{3.59}$$

is evaluated optimally by natural interval arithmetic with application of rule (3.46). In Fig. 3.4 the function has been plotted for values between -10 and 10. In Tab. 3.3 the resulting interval diameters for different intervals $[x]$ are shown. The widest enclosures are obatined for the midepoint-rule ($\rho = 0$). When increasing the order the enclosures get tighter, but never achieve the result of the naive evaluation.

Table 3.3: Interval diameters for $f_3(x)$.

	naiv	$\rho = 0$	$\rho = 1$	$\rho = 2$	$\rho = 3$	$\rho = 4$
$[x] = [-0.1; 0.1]$	0.808	1.0648	0.8726	0.8688	0.8681	0.8681
$[x] = [-1; 1]$	16	64	32	30	23	23
$[x] = [-10; 10]$	14641	106480	72680	88680	18680	18680

The effect of large Taylor coefficients can be seen from

$$f_4(x) = \frac{1}{1 + x + 100x^2}, \tag{3.60}$$

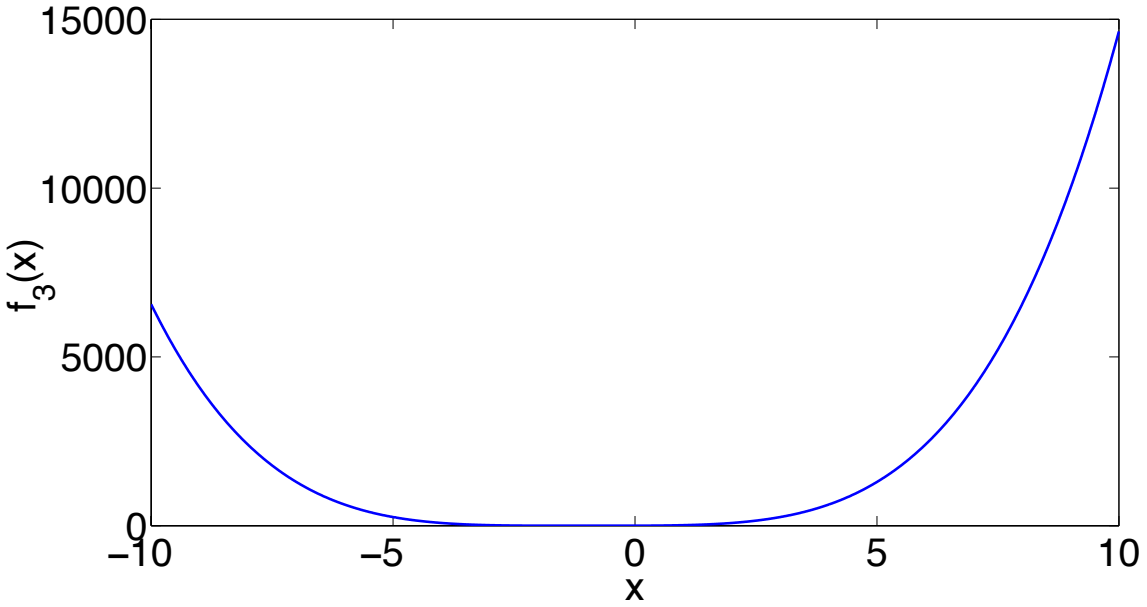


Figure 3.4: Example function $f_3(x)$.

which has on $x = [-0.1; 0.1]$ the range $[10/21; 400/399]$ (the maximum is attained at $x = -0.005$). Naive interval evaluation gives $[10/21; 10/9]$ with exact lower bound. A plot of $f_4(x)$ is presented in Fig. 3.5. With increasing expansion order the obtained enclosure becomes wider (see Tab. 3.4). The Taylor coefficients are shown in Tab. 3.5, their magnitude increases with the order.

Table 3.4: Interval diameters for $f_4(x)$.

	naiv	$\rho = 0$	$\rho = 1$	$\rho = 2$	$\rho = 3$	$\rho = 4$
$[x] = [-0.1; 0.1]$	0.6349	5.1852	7.2572	39.8457	55.9519	268.3878

Table 3.5: Taylor coefficients for $f_4(x)$.

	$\rho = 0$	$\rho = 1$	$\rho = 2$	$\rho = 3$	$\rho = 4$
$\hat{x} = 0$	1	-1	-99	199	9701

The examples in this subsection have illustrated, that increasing the expansion order of the Taylor expansion does not automatically lead to tighter enclosures. A more detailed discussion about this topic can be found in [56].

3.5.2 Monotonicity Test

Consider a nonlinear function

$$f : \mathbb{R}^n \mapsto \mathbb{R}^m \quad . \tag{3.61}$$

If an interval argument $[x]$ leads to monotonic behavior in the function $f(x)$, the function evaluation can be simplified by replacing the function interval vector arguments with point

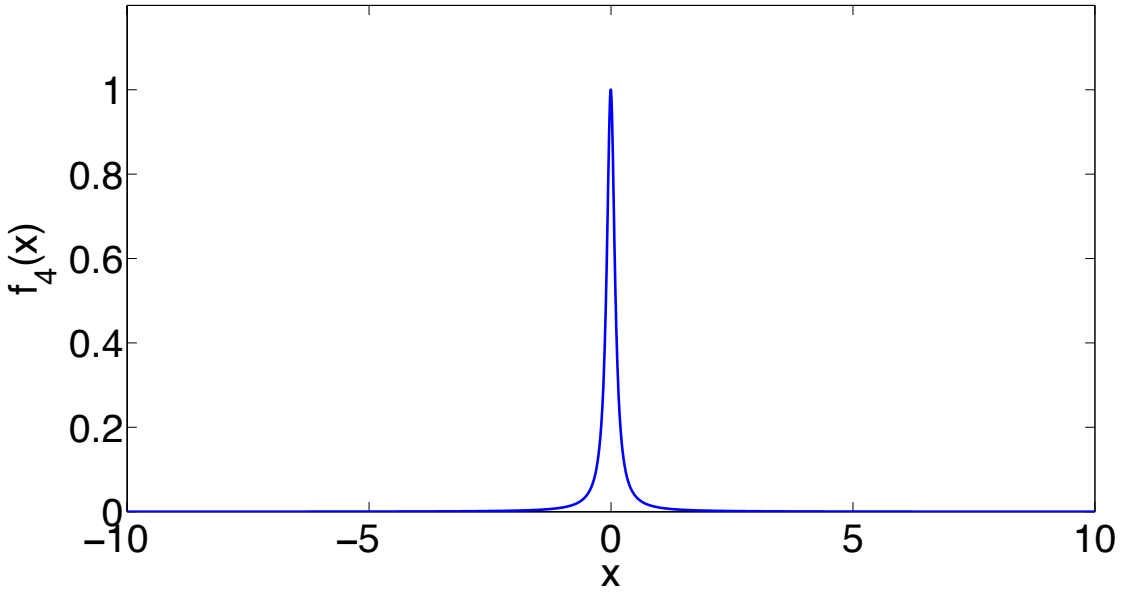


Figure 3.5: Example function $f_4(x)$.

vectors [23,24,59]. Namely, the infimum $\underline{f}$ and the supremum $\overline{f}$ can be evaluated by replacing $[x]$ in f with their lower and upper bounds $\underline{x}$, $\overline{x}$ in case of monotonicity. The overestimation of interval evaluation is reduced significantly, since now as many components of point intervals $\underline{x}$, $\overline{x}$ as possible are used in computation instead of intervals $[x]$ with non-zero diameters.

Monotonicity can easily be checked by interval evaluation of the Jacobian J, where

$$J_{i,j} = \frac{\partial f_i}{\partial x_j} \quad i = 1, \ldots, m, \; j = 1, \ldots, n, \tag{3.62}$$

are the derivatives of the i-th component f_i of f with respect to the j-th component of x. If the lower bound of the interval $[J_{i,j}]$ is strictly positive or if the upper bound is strictly negative, the intervals $[x]$ can be replaced by the values summarized in Tab. 3.6. The

Table 3.6: Replacement of state and parameter variables in case of monotonicity.

	$\underline{J}_{i,j} > 0$	$\overline{J}_{i,j} < 0$
$\inf \left\{ f_i(x)\vert_{x_j=\xi_j} \right\}$	$\xi_j = \underline{x}_j$	$\xi_j = \overline{x}_j$
$\sup \left\{ f_i(x)\vert_{x_j=\xi_j} \right\}$	$\xi_j = \overline{x}_j$	$\xi_j = \underline{x}_j$

enclosure of f over $[x]$ is then given by the interval which is specified by the infima and suprema corresponding to the rows of Tab. 3.6.

Interval evaluation with functions simplified after monotonicity test according to Tab. 3.6 improves the estimation of the true range of f. If replacement of the intervals by infimum or supremum is not possible due to non-monotonic behavior of the function $f(x)$, the enclosure of the intervals can be improved iteratively with the help of interval splitting as described in the following subsection.

3.5.3 Iterative Improvement of Infimum and Supremum

If the monotonicity test is not successful in at least one component $[x_i]$ of the interval argument $[x]$ of $f(x)$, the interval vector $[x]$ can be split into several subboxes which are tested for monotonicity again [23, 59]. Mapping each of these subboxes will lead to tighter approximations of the exact set. If only the upper and lower bounds of $f(x)$ are desired, the union of all resulting subboxes can be determined by searching for the smallest lower and largest upper bound of all subboxes obtained after the evaluation of $f(x)$.

This computation is done for each component f_i of f separately. If an interval vector is split into subboxes, first the splitting direction has to determined. The splitting direction denotes the component in which an interval vector is split.

The splitting can be done in the direction j_i^* determined by the component wise product of diameters of the gradient $\frac{\partial f_i}{\partial x}$ and of the components of the interval vector $[x]$ [9, 59], as follows

$$j_i^* = \underset{j=1\ldots n}{\arg\max} \left\{ \operatorname{diag} \left(\operatorname{diam} \left(\left[\frac{\partial f_i}{\partial x_j} \Big|_{x=[x]} \right] \right) \right) \cdot \operatorname{diam}([x]) \right\} \quad . \tag{3.63}$$

Alternative rules [27] are

$$j_i^* = \underset{j=1\ldots n}{\arg\max} \left\{ \operatorname{diag} \left(\frac{\partial f_i}{\partial x_j} \Big|_{x=\operatorname{mid}([x])} \right) \cdot \operatorname{diam}([x]) \right\} \quad , \tag{3.64}$$

or [9]

$$j_i^* = \underset{j=1\ldots n}{\arg\max} \left\{ \operatorname{diam} \left(\operatorname{diag} \left(\left[\frac{\partial f_i}{\partial x_j} \Big|_{x=[x]} \right] \right) \cdot ([x] - \operatorname{mid}([x])) \right) \right\} \quad . \tag{3.65}$$

3.6 Taylor Models

3.6.1 Definition of Taylor Models

The key idea of Taylor models is to describe the bulk of functional dependency through a Taylor polynomial and bound the deviation of the original function from the Taylor polynomial by an interval [45].

Taylor models are based on the Taylor expansion of order ρ of a function f (see also (3.51)). The Taylor expansion of a function

$$f : \mathbb{R}^n \mapsto \mathbb{R} \tag{3.66}$$

of order ρ around $\hat{x}$ is defined by

$$f(x) = \sum_{i=0}^{\rho} \left\{ \frac{1}{i!} \left[\sum_{l=1}^{n} (x_l - \hat{x}_l) \frac{\partial}{\partial \tilde{x}} \right]^i f(\tilde{x}) \right\}_{\tilde{x}=\hat{x}} + R(x, \hat{x}, \xi) \quad \xi \in [x, \hat{x}] \tag{3.67}$$

with

$$R(x,\hat{x},\xi) = \frac{1}{(\rho+1)!}\left[\sum_{l=1}^{n}(x_l-\hat{x}_l)\frac{\partial}{\partial x_l}\right]^{\rho+1} f(\xi). \tag{3.68}$$

If x and $\hat{x}$ are in the interval $[x]$, then

$$R(x,\hat{x},\xi) \subset R(x,\hat{x},[x]) = \frac{1}{(\rho+1)!}\left[\sum_{l=1}^{n}([x_l]-\hat{x}_l)\frac{\partial}{\partial x_l}\right]^{\rho+1} f([x]) \tag{3.69}$$

and

$$\begin{aligned} f(x) \in & \underbrace{\sum_{i=0}^{\rho}\left\{\frac{1}{i!}\left[\sum_{l=1}^{n}(x_l-\hat{x}_l)\frac{\partial}{\partial \tilde{x}_l}\right]^{i} f(\tilde{x})\right\}_{\tilde{x}=\hat{x}}}_{\text{Polynomial } P_{\rho,f}(x-\hat{x})} \\ & + \underbrace{\left\{\frac{1}{(\rho+1)!}\left[\sum_{l=1}^{n}([x_l]-\hat{x}_l)\frac{\partial}{\partial x_l}\right]^{\rho+1} f([x])\right\}}_{\text{Interval remainder } I_{\rho,f}} \quad \forall x \in [x] \end{aligned} \tag{3.70}$$

holds [45].

Definition of a Taylor Model [45]

Let $f : [x] \subset \mathbb{IR}^n \to \mathbb{IR}$ be a function that is $(\rho+1)$ times continuously partially differentiable on the n-dimensional domain interval vector $[x]$. Let $\hat{x}$ be a point in $[x]$. $P_{\rho,f}$ is the ρ-th order Taylor polynomial of f around $\hat{x}$. Let $I_{\rho,f}$ be an interval such that

$$f(x) \in P_{\rho,f}(x-\hat{x}) + I_{\rho,f}, \ \forall x \in [x]. \tag{3.71}$$

The pair $(P_{\rho,f}, I_{\rho,f})$ is an ρ-th order Taylor model of f around $\hat{x} \in [x]$ [45]. A pair $(P_{\rho,f}, I_{\rho,f})$ satisfying (3.71) is called a Taylor model of f and denoted by

$$T_{\rho,f} = (P_{\rho,f}, I_{\rho,f}) \tag{3.72}$$

or

$$T_{\rho,f} = P_{\rho,f} + I_{\rho,f} \ . \tag{3.73}$$

A *Taylor model vector* is a vector with Taylor model components. When no ambiguity arises, a Taylor model vector is simply called a Taylor model. Arithmetic operations for Taylor model vectors are defined component wise.

In interval arithmetic the basic data type are intervals in Taylor model arithmetic the basic data type are Taylor models. In computations that involve a Taylor model $T_{\rho,f}$, the polynomial part is propagated by symbolic calculations wherever possible. The interval remainder term is processed according to the rules of interval arithmetic. The Taylor coefficients are floating point numbers as all truncation and roundoff errors in intermediate operations are also enclosed into the remainder interval of the final result.

3.6.2 Taylor Model Arithmetic

Taylor models of complicated functions f can be determined by carrying Taylor model arithmetic through binary operations and intrinsic functions which compose the function f sequentially [45].

Addition of Taylor Models

Consider the Taylor models for f and g as $T_{\rho,f} = (P_{\rho,f}, I_{\rho,f})$ and $T_{\rho,g} = (P_{\rho,g}, I_{\rho,g})$, then Taylor models of the sum and difference of f and g can be obtained as

$$T_{\rho,f} \pm T_{\rho,g} = (P_{\rho,f} \pm P_{\rho,g}, I_{\rho,f} \pm I_{\rho,g}) \ . \tag{3.74}$$

Multiplication of Taylor Models

The Taylor models for the product of f and g can be obtained as

$$T_{\rho,f} \cdot T_{\rho,g} = (P_{\rho,f\cdot g}, I_{\rho,f\cdot g}) \ , \tag{3.75}$$

where $P_{\rho,f} \cdot P_{\rho,g} = P_{\rho,f\cdot g} + P_e$ with $P_{\rho,f\cdot g}$ being the ρ-th order polynomial of the result of the left hand side, and P_e the part of the product polynomial with order from $\rho + 1$ to 2ρ, and

$$I_{\rho,f\cdot g} = B(P_e) + B(P_{\rho,f}) \cdot I_{\rho,g} + B(P_{\rho,g}) \cdot I_{\rho,f} + I_{\rho,f} \cdot I_{\rho,g} \ , \tag{3.76}$$

where $B(\cdot)$ denotes the interval bounds obtained for the Taylor polynomials when they are evaluated over their corresponding domain interval vector. P_e is absorbed in the remainder bound to avoid an increase of the order.

Range of Taylor Models

Evaluating a Taylor model $T_\rho(x - \hat{x}) = P_\rho(x - \hat{x}) + I_\rho$ for all $x \in [x]$ the range

$$\mathrm{Rg}(T_\rho) := \{P(x - \hat{x}) + a | x \in [x], a \in I_\rho\} \tag{3.77}$$

of $T_\rho(x - \hat{x})$ is obtained.

Composition of Taylor Models

Two Taylor models T_1 and T_2 can be composed to the Taylor model T, namely

$$T(z) = (T_1 \circ T_2)(z) = (P_1(x) + I_1) \circ (P_2(z) + I_2) = (P_1(P_2(z) + I_2) + I_1) \ . \tag{3.78}$$

The composition of the two Taylor models is obtained by insertion of the Taylor model T_2 into the polynomial P_1 via Taylor model addition and multiplication, and subsequent addition of the remainder bound I_1. The i-th component of x_i is replaced by the i-th of T_2. However, it is required that the domain interval vector $[x]$ of T_1 contains the range of T_2 [54]:

$$\mathrm{Rg}(T_2) \subseteq [x] \ . \tag{3.79}$$

Intrinsic Functions of Taylor Models

In the following the computation of Taylor models for intrinsic functions is studied. It is assumed that a Taylor model $T_{\rho,f}$ for the function f is given, such that $f(x) \in P_{\rho,f}(x-\hat{x}) + I_{\rho,f}$. Let $c_f = f(\hat{x})$, and $\bar{f}$ be defined by $\bar{f}(x) = f(x) - c_f$. The Taylor model of $\bar{f}$ is given by $T_{\rho,\bar{f}} = P_{\rho,\bar{f}} + I_{\rho,\bar{f}}$, with $P_{\rho,\bar{f}}(x-\hat{x}) = P_{\rho,f}(x-\hat{x}) - c_f$ and $I_{\rho,\bar{f}} = I_{\rho,f}$.

For the intrinsic functions, here only the exponential, logarithmic, and, the reciprocal applied to a given function f from the Taylor model of f is discussed. Other intrinsic functions are explained in [45].

Exponential

Consider the exponential function evaluated for $f(x)$:

$$\exp(f(x)) \tag{3.80}$$

with $f : \mathbb{R}^n \mapsto \mathbb{R}$. This expression can be rewritten according to

$$\begin{aligned} \exp(f(x)) &= \exp(c_f + \overline{f}(x)) = \exp(c_f) \cdot \exp(\overline{f}(x)) \\ &= \exp(c_f) \cdot \left\{ 1 + \overline{f}(x) + \frac{1}{2!}(\overline{f}(x))^2 \ldots + \frac{1}{\tilde{\rho}!}(\overline{f}(x))^{\tilde{\rho}} \right. \\ &\quad \left. + \frac{1}{(\tilde{\rho}+1)!}(\overline{f}(x))^{\tilde{\rho}+1} \exp\left(\Theta \overline{f}(x)\right) \right\} \end{aligned} \tag{3.81}$$

where $0 < \Theta < 1$. Taking $\tilde{\rho} \geq \rho$, the part

$$\exp(c_f) \cdot \left\{ 1 + \bar{f}(x) + \frac{1}{2!}(\bar{f}(x))^2 + \cdots + \frac{1}{\rho!}(\bar{f}(x))^{\rho} \right\} \tag{3.82}$$

is only a polynomial of $\bar{f}$, of which the Taylor model is obtained by applying the rules for multiplication and addition.

The remainder part of $\exp(f(x))$, the expression

$$\exp(c_f) \cdot \left\{ \frac{1}{(\rho+1)!}(\bar{f}(x))^{\rho+1} + \cdots + \frac{1}{(\tilde{\rho}+1)!}(\bar{f}(x))^{\tilde{\rho}+1} \exp(\Theta \cdot \bar{f}(x)) \right\} \quad , \tag{3.83}$$

will be bounded by an interval. One first observes that since the Taylor polynomial of $\bar{f}$ does not have a constant part, the $(\rho+1)$-st through $(\tilde{\rho}+1)$-st powers of the Taylor model $(P_{\rho,\bar{f}}, I_{\rho,f})$ of $\bar{f}$ will have disappearing polynomial part, and thus the entire remainder part (3.83). The remainder bound interval for the Lagrange remainder term

$$\exp(c_f) \frac{1}{(\tilde{\rho}+1)!}(\bar{f}(x))^{\tilde{\rho}+1} \exp(\Theta \cdot \bar{f}(x)) \tag{3.84}$$

can be estimated because, for any $x \in [x]$, $P_{\rho,\bar{f}}(x-\hat{x}) \in B(P_{\rho,\bar{f}})$, and $0 < \Theta < 1$, and so

$$(\bar{f}(x))^{\tilde{\rho}+1} \exp\left(\Theta \cdot \bar{f}(x)\right) \in \left(B(P_{\rho,\bar{f}}) + I_{\rho,f}\right)^{\tilde{\rho}+1} \cdot \exp\left([0;1] \cdot \left(B(P_{\rho,\bar{f}}) + I_{\rho,\bar{f}}\right)\right) \quad . \tag{3.85}$$

For the estimation of the interval bounds of $\exp\left([0;1] \cdot \left(B(P_{\rho,\bar{f}}) + I_{\rho,\bar{f}}\right)\right)$ interval arithmetic is used. It takes consideration of the fact that the exponential function is monotonically increasing. Thus, it is sufficient to insert the lower and upper bounds of the argument in the exponential.
One may choose $\tilde{\rho} = \rho$ for simplicity [45], but it is not a priori clear which value of $\tilde{\rho}$ would yield the sharpest enclosures.

Logarithm

With the condition $\forall x \in [x]$, $P_{\rho,f}(x-\hat{x}) + I_{\rho,f} \subset [0,\infty]$, the logarithm equation can be written as

$$\begin{aligned} \log(f(x)) &= \log(c_f + \bar{f}(x)) = \log(\left\{c_f\right)\left(1 + \frac{\bar{f}(x)}{c_f}\right)\right\} \\ &= \log(c_f) + \frac{\bar{f}(x)}{c_f} - \frac{1}{2}\frac{(\bar{f}(x))^2}{c_f^2} - \ldots + (-1)^{\tilde{\rho}+1}\frac{1}{\tilde{\rho}}\frac{(\bar{f}(x))^{\tilde{\rho}}}{c_f^k} \\ &\quad +(-1)^{\tilde{\rho}+2}\frac{1}{\tilde{\rho}+1}\frac{(\bar{f}(x))^{\tilde{\rho}+1}}{c_f^{\tilde{\rho}+1}}\frac{1}{\left(1+\Theta\cdot\frac{\bar{f}(x)}{c_f}\right)^{\tilde{\rho}+1}} \quad . \end{aligned} \tag{3.86}$$

The polynomial part and interval remainder can be computed in a similar way to the previously described exponential function. With $\tilde{\rho} = \rho$, the second line is only Taylor addition and multiplication, and the third line gets an interval contribution, because the Taylor model $(\bar{P}_{\rho,f}, I_{\rho,f})$ of $\bar{f}$, when raised to the $(\tilde{\rho}+1)$-st power, disappears and produces no polynomial part.

Multiplicative Inverse

Under the condition $\forall x \in [x]$, $0 \notin P_{\rho,f}(x-\hat{x}) + I_{\rho,f}$, the multiplicative inverse equation is

$$\begin{aligned} \frac{1}{f(x)} &= \frac{1}{c_f}\left\{1 - \frac{\bar{f}(x)}{c_f} + \frac{(\bar{f}(x))^2}{c_f^2} - \ldots + (-1)^{\tilde{\rho}}\frac{(\bar{f}(x))^{\tilde{\rho}}}{c_f^k}\right\} \\ &\quad +(-1)^{\tilde{\rho}+1}\frac{(\bar{f}(x))^{\tilde{\rho}+1}}{c_f^{\tilde{\rho}+2}}\frac{1}{\left(1+\Theta\cdot\frac{\bar{f}(x)}{c_f}\right)^{\tilde{\rho}+2}} \end{aligned} \tag{3.87}$$

and when evaluated in Taylor model arithmetic for $\tilde{\rho} = \rho$, the second addition part only yields an interval contribution.

Integration of Taylor Models

The integration rule for of Taylor models is given by

$$\partial_i^{-1}(P_{\rho,f}, I_{\rho,f}) = (P_{\rho,\partial^{-1}f}, I_{\rho,\partial^{-1}f}) = \left(\int_0^{x_i} P_{\rho-1,f} dx_i,\ I_{\rho,\partial_i^{-1}f} \right), \quad \text{with} \quad x_i \in [x_i] \ , \quad (3.88)$$

where

$$I_{\rho,\partial_i^{-1}f} = (B(P_{\rho,f} - P_{\rho-1,f}) + I_{\rho,f}) \cdot [x_i] \ . \quad (3.89)$$

$B(P_{\rho,f} - P_{\rho-1,f})$ is the interval bound of the ρ-th order part of $P_{\rho,f}$. Thus, the order of the Taylor model with respect to x_i does not increase.

A definite integral over variable x_i from x_{il} to x_{iu}, which are both in the domain interval $[x_i]$ is determined by

$$\int_{x_{il}}^{x_{iu}} f(x)dx_i \in (P_{\rho,\partial^{-1}f}(x|_{x_i=x_{iu}-\hat{x}_i}) - P_{\rho,\partial^{-1}f}(x|_{x_i=x_{il}-\hat{x}_i}), I_{\rho,\partial^{-1}f}) \ . \quad (3.90)$$

The integration rule plays an important role in the verified integration of ordinary differential equations with Taylor models (see Section 4.4).

3.6.3 Range Bounding of Taylor Models

At the end of Section 3.5.1 univariate Taylor models of different order for various examples have been bounded over an interval, by performing natural interval evaluation of the Taylor models. It has been shown that increasing the order not always guarantees tighter enclosures. There are several possibilities to obtain the lower and upper bounds of a Taylor model T_ρ. And the way this is done determines the quality of the enclosure. The simplest approach is the application of natural interval arithmetic on T_ρ. The use of the Horner scheme may improve the enclosure quality [56]. A so called *linear dominated range bounder* [45] leads to tight enclosures if the linear part of the polynomial P_ρ of T_ρ is dominating, or in other words, if the nonlinear terms contribute less then the linear parts. A detailed description of this method can be found in [45, 56]. In [45] it is proposed to evaluate the exact range including the quadratics and to treat higher order terms by simple interval arithmetic, which however increases the computational effort significantly [56]. Optimized methods like the monotonicity test or iterative range computation as described in the Section 3.5 can be applied to obtain tighter bounds of a Taylor model.

4 Verified Simulation of Nonlinear Uncertain Systems

The nonlinear dynamical systems considered in this thesis are described by ordinary nonlinear differential equations (ODEs) according to

$$\dot{x}(t) = f_x(x(t), p(t), t), \tag{4.1}$$

where $x \in \mathbb{R}^{n_x}$ is the state vector and $p \in \mathbb{R}^{n_p}$ the parameter vector.The parameter vector p and the initial conditions $x(0)$ are assumed to be uncertain with $p \in [\underline{p}; \overline{p}]$ and $x(0) \in [\underline{x}(0); \overline{x}(0)]$.
If the parameters may vary over time this has also to be considered. In case no explicit expression of this time-variance is known, upper and lower bounds of the variation rates are sufficient for a verified integration of the ODE. The system equations are extended by the variation rates Δp according to

$$\begin{bmatrix} \dot{x}(t) \\ \dot{p}(t) \end{bmatrix} = \begin{bmatrix} f_x(x(t), p(t), t) \\ \Delta p \end{bmatrix} . \tag{4.2}$$

Introducing the extended state vector $z(t) = \left[x^T(t), \quad p^T(t)\right]^T$ results in

$$\dot{z}(t) = f(z(t), t) \quad \text{with} \quad f(z(t), t) = \begin{bmatrix} f_x(x(t), p(t), t) \\ \Delta p \end{bmatrix} \tag{4.3}$$

with $\Delta p \in [\underline{\Delta p}; \overline{\Delta p}]$, $f : \mathbb{R}^{n+1} \mapsto \mathbb{R}^n$, and $\mathbb{R}^n = \mathbb{R}^{n_x} \times \mathbb{R}^{n_p}$. Components p_i of the uncertain parameter vector which are time-invariant are described by $\Delta p_i = 0$.

In control engineering dynamical systems are often given in the form of

$$\dot{z}(t) = f(z(t), u(t), t) \quad \text{with} \quad z \in \mathbb{R}^n \tag{4.4}$$

in open loop control and

$$\dot{z}(t) = f(z(t), u(z(t), t), t) \tag{4.5}$$

in closed loop control respectively. The variable $u \in \mathbb{R}^{n_u}$ denotes the vector of control or input variables. Both expressions – (4.4) and (4.5) – can be rewritten to the form of (4.3).

In the following a grid $t_0 < t_1 < \cdots t_{k_{max}} < t_f$ is considered, which is not necessarily equally spaced. The step size from t_k to t_{k+1} is denoted by h_k with $h_k = t_{k+1} - t_k$. The step from t_k to t_{k+1} is referred to as the $(k+1)$-st step. The solution of (4.3) is denoted with an initial

condition $z(t_k)$ at t_k by $z(t; t_k, z(t_k))$. For an interval vector $[z(t_k)]$, the set of solutions is denoted by

$$z(t; t_k, [z(t_k)]) = \{z(t; t_k, z(t_k)) | z(t_k) \in [z(t_k)]\} \quad . \tag{4.6}$$

The goal is to compute interval vectors $[z(t_k)]$ $k = 1, 2, \cdots, k_{max}$, that are guaranteed to contain the solution of (4.3) at $t_1, t_2, \cdots, t_{k_{max}}$. That is

$$z(t_k; t_0, [z_0]) \subseteq [z(t_k)], \text{for } k = 1, 2, \cdots, k_{max}. \tag{4.7}$$

In this chapter different methods for a verified integration of nonlinear uncertain systems are discussed in detail. They are required for the prediction step in verified state and parameter estimation. The algorithms presented in Sections 4.1–4.3 are based on a Taylor series in time and the extended state vector is enclosed by intervals and/or parallelepipeds. The integration algorithm in Section 4.4 performs in addition to a Taylor expansion in time also an expansion in the initial conditions of the extended state vector. Here, the extended state vector is enclosed by Taylor models.

4.1 Verified Techniques Based on Interval Enclosures

4.1.1 Basic Algorithm

The most traditional approach for a verified integration of ODEs is based on a Taylor series expansion of the solution of the ODE in time [43, 53, 63], namely

$$z\left(t_{k+1}\right) = z\left(t_k\right) + \sum_{i=1}^{\nu} \frac{h_k^i}{i!} f^{(i-1)}\left(z\left(t_k\right), t_k\right) + e\left(z\left(\xi_k\right), \xi_k\right) \quad \text{with} \quad t_k \leq \xi_k \leq t_{k+1} \ , \tag{4.8}$$

where ν is the order and h_k is the integration step-size, i.e., $h_k = t_{k+1} - t_k$ and $e(\cdot)$ is the truncation error

$$e\left(z\left(\xi_k\right), \xi_k\right) = \frac{h_k^{\nu+1}}{(\nu+1)!} f^{(\nu)}\left(z\left(\xi_k\right), \xi_k\right) := e_k \ . \tag{4.9}$$

The approach consists of two stages. In stage one the existence and uniqueness of the solution is proven and an a priori enclosure of the solution is computed. In stage two a tight enclosure of the solution of the ODE is determined using the results from stage one. Both stages are summarized below.

Stage 1: Computing an a priori enclosure of the solution

In stage one an a priori enclosure $[B_k]$ of the solution is computed such that $z(t; t_k, z(t_k))$ is guaranteed to exist for all $t \in [t_k, t_{k+1}]$ and all $z(t_k) \in [z(t_k)]$; and the set of solutions $z(t; t_k, [z(t_k)])$ is a subset of $[B_k]$ for all $t \in [t_k, t_{k+1}]$. Banach's fixed-point theorem is applied to the Picard operator here [43, 53, 63], resulting into

$$\Phi([B_k]) = [z(t_k)] + [0, h_k] f([B_k], [t_k; t_{k+1}]) \ . \tag{4.10}$$

The interval vector $[z(t_k)]$ is a tight enclosure of the solution at $t = t_k$. Using the Picard operator, the a priori enclosure $[B_k]$ is determined iteratively. In (4.10), $[B_k]$ is initialized by $[z(t_k)]$. If $\Phi([B_k]) \not\subseteq [B_k]$, then $[B_k]$ is enlarged and the calculation has to be repeated until $\Phi([B_k]) \subseteq [B_k]$, then $[B_k]$ and $\Phi([B_k])$ are a priori enclosures, which enclose all possible values of the extended state vector z in the considered time interval. In order to tighten $[B_k]$, (4.10) is repeated recursively until the deviation between $\Phi([B_k])$ and $[B_k]$ is smaller than a desired value. In case that the algorithm does not converge the step-size has to be reduced.

Stage 2: Tightening the enclosure of the solution

In stage two the result of stage one is used to calculate a tight enclosure at t_{k+1} using the Taylor expansion of the solution of the ODE.

Applying interval methods it is possible to give guaranteed bounds for the time discretization or truncation error $e(\cdot)$. For the truncation error the following relation holds:

$$e\left(z\left(\xi_k\right), \xi_k\right) \subseteq \frac{h_k^{\nu+1}}{(\nu+1)!}[f^{(\nu)}]\left([B_k], [t_k; t_{k+1}]\right) := [e_k] \; . \tag{4.11}$$

In case that $[e_k]$ is too large, the step-size can be reduced. The tight interval enclosure $[z(t_{k+1})]$ of the flow of the ODE in the time interval $[t_k; t_{k+1}]$ is then given by

$$[z\left(t_{k+1}\right)] = [z\left(t_k\right)] + \sum_{i=1}^{\nu} \frac{h_k^i}{i!}[f^{(i-1)}]\left([z\left(t_k\right)]\right), t_k) + [e_k] \; . \tag{4.12}$$

This equation is summarized by

$$[z\left(t_{k+1}\right)] = [g]([z(t_k)], t_k) + [e_k] \; . \tag{4.13}$$

Analytical expressions of the higher derivatives $f^{(1)} = \ddot{x}$, $f^{(2)} = \dddot{x}$, ..., $f^{(\nu)} = x^{(\nu+1)}$ of the right hand side of the differential equation can be derived recursively with the following relation

$$\begin{aligned} f^{(0)}\left(z(t), t\right) &= f\left(z(t), t\right) \; , \\ f^{(\tau)}\left(z(t), t\right) &= \frac{\partial f^{(\tau-1)}\left(z(t), t\right)}{\partial t} + \frac{\partial f^{(\tau-1)}\left(z(t), t\right)}{\partial z} f^{(0)}\left(z(t), t\right) \; . \end{aligned} \tag{4.14}$$

In practice the higher order Taylor coefficients are obtained numerically by automatic differentiation [43, 52], because the analytical expression for the higher order coefficients usually become very large.

Remark to Stage 1:
The method used in step one is based on a first order expansion. It can be extended to a $\nu - th$ order expansion resulting into

$$\Phi([B_k]) = [z\left(t_k\right)] + \sum_{i=1}^{\nu} \frac{[0, h_k]^i}{i!}[f^{(i-1)}]\left([z\left(t_k\right)]\right), t_k) + \frac{[0, h_k]^{\nu+1}}{(\nu+1)!}[f^{(\nu)}]\left([B_k], [t_k; t_{k+1}]\right) \; . \tag{4.15}$$

Other methods for the calculation of a priori enclosures can be found in [64].

In order to simplify the notation, $z\left(t_{k+1}\right)$ is replaced by z_{k+1}. Then (4.13) becomes

$$[z_{k+1}] = [g]([z_k)], t_k) + [e_k] \; . \tag{4.16}$$

4.1.2 Mean value form

To reduce the dependency problem, when evaluating equation (4.12), the midpoint rule or mean value evaluation is applied to the Taylor expansion leading to the so called mean value form

$$
\begin{aligned}
[z_{k+1}] &= \hat{z}_k + \sum_{i=1}^{\nu} \frac{h_k^i}{i!} f^{(i-1)}\left(\hat{z}_k, t_k\right) + [e_k] && (4.17)\\
&\quad + \left(\mathfrak{I} + \sum_{i=1}^{\nu} \frac{h_k^i}{i!} [J]\left(f^{(i-1)}\left([z_k], t_k\right)\right) \right) \left([z_k] - \hat{z}_k\right) && (4.18)\\
&= g(\hat{z}, t_k) + \left[\left. \frac{\partial g}{\partial z_k} \right|_{z_k=[z_k]} \right] \cdot \left([z_k] - \hat{z}\right) + [e_k] && (4.19)\\
&= g(\hat{z}, t_k) + [\tilde{J}]([z_k]) \cdot \left([z_k] - \hat{z}\right) + [e_k] && (4.20)\\
&\quad \text{with} \quad \hat{z} = \mathrm{mid}([z_k]) \ , && (4.21)
\end{aligned}
$$

where $\mathfrak{I}$ is the $n \times n$ identity matrix, J denotes the Jacobian of the Taylor-coefficients and $\tilde{J}$ the Jacobian of the whole sum expression with respect to the extended state vector z_k. For even tighter bounds the intersection of the result of the naive evaluation of equation (4.12) and the mean value form is calculated.

4.1.3 Monotonicity Test and Iterative Range Computation

The interval enclosure of the Jacobian $[\tilde{J}]([z_k])$ in (4.17) obtained by the application of the mean-vale form can directly be used for a monotonicity test as described in Section 3.5.2 for the evaluation of g in the Taylor series expansion. When the Jacobian $\tilde{J}([z_k])$ has been calculated, the monotonicity properties for each component g_i of g are investigated. For each $g_i(\cdot)$ the corresponding modified interval arguments according to Table 3.6 are determined. In case of non-monotonic behavior for at least one component of the interval arguments in (4.12) of at least one component of $[z_k]$ iterative range computation as described in Section 3.5.3 can be applied. The corresponding interval vector can be split into subboxes and the monotonicity is repeated for the subboxes. In order to reduce the computational effort the monotonicity test and iterative range computation can be restricted to the lower order coefficients. The Taylor series expansion is split up into two parts:

$$
\begin{aligned}
[z_{k+1}] &= [z_k] + \sum_{i=1}^{\nu_1} \frac{h_k^i}{i!} [f^{(i-1)}]\left([z_k]\right), t_k)\\
&\quad + \sum_{i=\nu_1+1}^{\nu} \frac{h_k^i}{i!} [f^{(i-1)}]\left([z_k]\right), t_k) + [e_k]\\
&= [g^{(1)}]([z_k], t_k) + [g^{(2)}]([z_k], t_k) + [e_k]\\
&\quad \text{with} \quad t_k \le \xi_k \le t_{k+1} \ . && (4.22)
\end{aligned}
$$

Then the monotonicity test and iterative range computation respectively is only applied for $g^{(1)}$ and the interval enclosure of $g^{(2)}$ is added after wards. In this thesis the subboxes in the iterative range computation are evaluated with a combination of natural interval arithmetic, midpoint-rule and monotonicity test.

4.1.4 Implicit Methods

The Taylor expansion in (4.12) corresponds to an explicit integration method, an implicit integration method [35, 65] is obtained by

$$[z_k] = [z_{k+1}] + \sum_{i=1}^{\nu} \frac{(-h_k)^i}{i!} [f^{(i-1)}]([z_{k+1}]), t_{k+1}) + [e_k] \tag{4.23}$$

with

$$[e_k] = \frac{(-h)_k^{\nu+1}}{(\nu+1)!} [f^{(\nu)}]([B_k], [t_k \ ; \ t_{k+1}]) \ . \tag{4.24}$$

The implicit method often provides tighter solutions [65], however, the computational effort is bigger, since in order to determine $[z_{k+1}]$ an interval Newton method has to be applied to

$$0 = [z_{k+1}] + \sum_{i=1}^{\nu} \frac{(-h_k)^i}{i!} [f^{(i-1)}]([z_{k+1}]), t_{k+1}) + [e_k] - [z_k] \ . \tag{4.25}$$

Usually the explicit and implicit methods are used in combination. The solution obtained from the explicit method is used as the initial enclosure for the interval Newton method of the implicit method.

4.1.5 Coordinate Transformations

For the Reduction of the wrapping effect Lohner [43], Nedialkov [53] and others developed algorithms for the verified integration of ODEs which include a linear coordinate transformation in each time-step. This leads to a reduction of the wrapping effect because the enclosure of the solution at each time-step is represented as a linear transformation of a box and not by rectangular box. In the following the algorithm developed by Lohner [43, 53, 63] is described. Depending on the way the coordinate transformation is done either the *parallelepiped method* or the *QR method* is obtained. Both methods are summarized below, for a detailed discussion see [43, 53, 63].

For derivation of the *Lohner method* first the following definitions are made:

$$s_{k+1} = \text{mid}([e_k]) \ , \tag{4.26}$$

$$\bar{z}_{k+1} = \bar{z}_k + \sum_{i=1}^{\nu} \frac{h_k^i}{i!} f^{(i-1)}(\bar{z}_k, t_k) + s_{k+1} \ , \tag{4.27}$$

$$\begin{aligned} S_k &= \mathfrak{I} + \sum_{i=1}^{\nu-1} \frac{h_k^i}{i!} J(f^{(i-1)}(z_k, t_k)) \ , \\ &\in \mathfrak{I} + \sum_{i=1}^{\nu-1} \frac{h_k^i}{i!} [J](f^{(i-1)}([z_k], t_k)) =: [S_k] \ , \end{aligned} \tag{4.28}$$

$$\text{with} \quad \bar{z}_0 = \text{mid}([z_0]). \tag{4.29}$$

In view of (4.26) - (4.29)

$$\begin{aligned} z_1 &= \bar{z}_1 + S_0(z_0 - \bar{z}_0) + e_0 - s_1 \\ &\in \bar{z}_1 + ([S_0]A_0)[r_0] + [e_0] - s_1 =: [z_1^1] \end{aligned} \tag{4.30}$$

with $[r_0] = [z_0] - \text{mid}([z_0])$, $A_0 = \mathfrak{I}$ and

$$\begin{aligned} z_1 &= \bar{z}_1 + S_0(z_0 - \bar{z}_0) + e_0 - s_1 \\ &= \bar{z}_1 + A_1\{(A_1^{-1} S_0 A_0) r_0 + A_1^{-1}(e_0 - s_1)\} \\ &= \bar{z}_1 + A_1 r_1 \\ &\in \bar{z}_1 + A_1[r_1] =: [z_1^2] \quad , \end{aligned} \tag{4.31}$$

where $A_1 \in \mathrm{R}^{nxn}$ is regular, and

$$\begin{aligned} r_1 &= (A_1^{-1} S_0 A_0) r_0 + A_1^{-1}(e_0 - s_1) \\ &\in (A_1^{-1}([S_0]A_0))[r_0] + A_1^{-1}([e_0] - s_1) =: [r_1] \quad . \end{aligned} \tag{4.32}$$

holds. It is shown later how A_k is chosen. From (4.31) and (4.32) follows, that z_1 is contained in the intersection of $[z_1^1]$ and $[z_1^2]$, hence

$$z_1 \in [z_1^1] \cap [z_1^2] \quad =: [z_1] \quad . \tag{4.33}$$

The expression $z_1 \in \bar{z}_1 + A_1[r_1]$ is a parallelepiped enclosure for z_1.

Let $z_1 \in [z_1]$, then the calculation of $[z_2]$ is done in the same way:

$$\begin{aligned} z_2 &= \bar{z}_2 + S_1(z_1 - \bar{z}_1) + e_1 - s_2 \\ &= \bar{z}_2 + (S_1 A_1) r_1 + e_2 - s_2 \\ &\in \bar{z}_2 + ([S_1]A_1)[r_1] + [e_1] - s_2 =: [z_2^1] \quad , \end{aligned} \tag{4.34}$$

and

$$\begin{aligned} z_2 &= \bar{z}_2 + S_1(z_1 - \bar{z}_1) + e_1 - s_2 \\ &= \bar{z}_2 + A_2\{(A_2^{-1} S_1 A_1) r_1 + A_2^{-1}(e_1 - s_2)\} \\ &= \bar{z}_2 + A_2 r_2 \\ &\in \bar{z}_2 + A_2[r_2] =: [z_2^2] \quad , \end{aligned} \tag{4.35}$$

where z_1 is replaced by its parallelepiped enclosure $z_1 \in \bar{z}_1 + A_1[r_1]$, $A_2 \in \mathrm{R}^{nxn}$ is regular and

$$\begin{aligned} r_2 &= (A_2^{-1} S_1 A_1) r_1 + A_2^{-1}(e_1 - s_2) \\ &\in (A_2^{-1}([S_1]A_1))[r_1] + A_2^{-1}([e_1] - s_2) =: [r_2] \quad . \end{aligned} \tag{4.36}$$

holds. Like in (4.33),

$$[z_2] := [z_2^1] \cap [z_2^2] \tag{4.37}$$

holds.

Hence instead of propagating the interval enclosures of the extended state vector z_k, parallelepiped enclosures are propagated. The interval enclosures, however, are needed for the calculation of the interval enclosures of the Jacobians in each time-step.

The algorithm for the $k+1$-st step is then given by the following steps:

1. For $k = 0$: Initialization with given initial interval $[z_0]$:

$$\begin{aligned} \bar{z}_0 &= \text{mid}([z_0]), \\ [r_0] &= [z_0] - \bar{z}_0, \\ A_0 &= \mathfrak{I} \ . \end{aligned} \tag{4.38}$$

2. Calculations in time-step $k+1$ with known state variables at time-step k:

$$\begin{aligned} s_{k+1} &= \text{mid}([e_k]), \\ \bar{z}_{k+1} &= \bar{z}_k + \sum_{i=1}^{\nu} \frac{h_k^i}{i!} f^{(i-1)}\left(\bar{z}_k\right), t_k) + s_{k+1}, \\ [S_k] &= \mathfrak{I} + \sum_{i=1}^{\nu} [J] \left(\frac{h_k^i}{i!} f^{(i-1)}\left([z\left(t_k\right)]\right), t_k) \right), \\ [r_{k+1}] &= (A_{k+1}^{-1}([S_k]A_k))[r_k] + A_{k+1}^{-1}([e_k] - s_{k+1}), \\ [z_{k+1}^1] &= \bar{z}_{k+1} + ([S_k]A_k)[r_k] + [e_k] - s_{k+1}, \\ [z_{k+1}^2] &= \bar{z}_{k+1} + A_{k+1}[r_{k+1}], \end{aligned} \tag{4.39}$$

   where $[e_k]$ is calculated as described in Section 4.1.1 and $J(\cdot)$ is the Jacobian of the Taylor coefficients. How to determine A_{k+1} is described further below.

3. The state enclosure at time-step $k+1$ is given by:

$$[z_{k+1}] = [z_{k+1}^1] \cap [z_{k+1}^2]. \tag{4.40}$$

Parallelepiped Method

For

$$A_{k+1} = \text{mid}([S_k]A_K) \tag{4.41}$$

the *parallelepiped method* is obtained.

If $\text{diam}([J_k])$ is small and A_k is well conditioned, then

$$A_{k+1}^{-1}[J_k]A_k = \mathfrak{I} + [E_k] \ , \tag{4.42}$$

where $[E_k]$ is small [53]. Then the overestimation in the evaluation of $(A_{k+1}^{-1}([S_k]A_k))[r_k]$ is small. However, the choice of A_k does not guarantee the regularity of the matrix. And if the condition number $\text{cond}(A_k)$ is large, then large overestimation may occur.

QR Method

The parallelepiped method has the disadvantage that the matrices A_{k+1} may become ill-conditioned leading to large overestimation. To preserve good condition numbers in the matrices A_{k+1} , Lohner [43] developed the QR method, which stabilizes the iteration through orthogonalization. Each parallelepiped is wrapped by a rotated n-dimensional rectangle such that the longest edge of the rectangle coincides with the longest edge of the parallelepiped. Orthogonalization is then performed in the order of decreasing lengths of the edges of the parallelepiped [43]. When ill-conditioned matrices A_{k+1} arise in the parallelepiped method, then the excess area in the QR method is smaller than the excess area in acute spikes of the parallelepiped enclosure. Instead of inverting the matrix A_{k+1} only its transpose has to be determined. In the QR-method A_{k+1} is given by the orthogonal matrix Q_{k+1} obtained from a QR decomposition of the matrix $\tilde{A}_{k+1}$, which is obtained by sorting the columns of $\mathrm{mid}([S_k]A_K)$ by size in descending order:

$$\tilde{A}_{k+1} = Q_{k+1}R_{k+1} \ . \tag{4.43}$$

Q_{k+1} is an orthogonal matrix with $Q_{k+1} \cdot Q_{k+1}^T = \mathfrak{I}$. R_{k+1} is an upper triangular matrix

$$R_{k+1} = \begin{pmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_{n-1,n} \\ 0 & \cdots & 0 & r_{n,n} \end{pmatrix} \quad . \tag{4.44}$$

Such a decomposition always exists.

Intersection of Two Parallelepipeds

In [43] it is shown, how to intersect two parallelepipeds with different matrices $A_k$which is interesting when different methods are combined, e.g. the parallepiped method and the QR-method, to obtain tighter enclosures. In this thesis this is extended to the case when also $\bar{z}_k$ are different for both enclosures. Consider two parallelepipeds described by

$$z_{k+1}^a \in \bar{z}_{k+1}^a + A_{k+1}[r_{k+1}^a] \tag{4.45}$$

and

$$z_{k+1}^b \in \bar{z}_{k+1}^b + B_{k+1}[r_{k+1}^b] \ . \tag{4.46}$$

Both parallelepipeds can be rewritten according to

$$\begin{aligned} z_{k+1}^a &\in \bar{z}_{k+1}^a + A_{k+1}[r_{k+1}^a] \\ &= \bar{z}_{k+1}^a + \bar{z}_{k+1}^b - \bar{z}_{k+1}^b + A_{k+1}[r_{k+1}^a] \\ &= \bar{z}_{k+1}^b + B_{k+1}B_{k+1}^{-1}(\bar{z}_{k+1}^a - \bar{z}_{k+1}^b + A_{k+1}[r_{k+1}^a]) \\ &= \bar{z}_{k+1}^b + B_{k+1}(B_{k+1}^{-1}(\bar{z}_{k+1}^a - \bar{z}_{k+1}^b + A_{k+1}[r_{k+1}^a])) \end{aligned} \tag{4.47}$$

and

$$\begin{aligned} z^b_{k+1} &\in \bar{z}^b_{k+1} + B_{k+1}[r^b_{k+1}] \\ &= \bar{z}^b_{k+1} + \bar{z}^a_{k+1} - \bar{z}^a_{k+1} + B_{k+1}[r^b_{k+1}] \\ &= \bar{z}^a_{k+1} + A_{k+1}A^{-1}_{k+1}(\bar{z}^b_{k+1} - \bar{z}^a_{k+1} + B_{k+1}[r^b_{k+1}]) \\ &= \bar{z}^a_{k+1} + A_{k+1}(A^{-1}_{k+1}(\bar{z}^b_{k+1} - \bar{z}^a_{k+1} + B_{k+1}[r^b_{k+1}])) \ . \end{aligned} \tag{4.48}$$

In view of equations (4.45)–(4.48)

$$z^a_{k+1} \in \bar{z}^a_{k+1} + A_{k+1}(\underbrace{[r^a_{k+1}] \cap (A^{-1}_{k+1}(\bar{z}^b_{k+1} - \bar{z}^a_{k+1} + B_{k+1}[r^b_{k+1}]))}_{=[\bar{r}^a_{k+1}]}) \tag{4.49}$$

and

$$z^b_{k+1} \in \bar{z}^b_{k+1} + B_{k+1}(\underbrace{[r^b_{k+1}] \cap (B^{-1}_{k+1}(\bar{z}^a_{k+1} - \bar{z}^b_{k+1} + A_{k+1}[r^a_{k+1}]))}_{=[\bar{r}^b_{k+1}]}) \tag{4.50}$$

holds.

This intersection procedure can be repeated iteratively, by setting $[r^a_{k+1}] = [\bar{r}^a_{k+1}]$ and $[r^b_{k+1}] = [\bar{r}^b_{k+1}]$ and repeating (4.49) and (4.50) until no further significant decrease in size is achieved.

This procedure can be applied when several methods are combined, e.g. Lohner's method and the naive evaluation of the Taylor series, where the state vector is propagated in form of interval enclosures. And an interval enclosure for z_{k+1} can also be described as a parallelepiped with $\bar{z}_{k+1} = 0$, $[r_{k+1}] = [z_{k+1}]$ and $A_{k+1} = \mathfrak{I}$, where $\mathfrak{I}$ is the identity matrix.

Consider for example the parallelepiped

$$z^a_{k+1} \in \bar{z}^a_{k+1} + A^a_{k+1}[r^a_{k+1}] = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} [-1;1] \\ [-4;4] \end{pmatrix} \tag{4.51}$$

and the interval vector $[[-1;3],[-5;7]]^T$, described by the parallelepiped representation

$$z^b_{k+1} \in \bar{z}^b_{k+1} + A^b_{k+1}[r^b_{k+1}] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} [-1;3] \\ [-5;7] \end{pmatrix} \ . \tag{4.52}$$

Both z^a_{k+1} and z^b_{k+1}, are depicted in Fig. 4.1, the results for 1,2,5 and 10 iterations are shown in Fig. 4.2 and Fig. 4.3.

In order to provide that the computation steps in (4.39) are correct it is important that the vectors $\bar{z}^a_{k+1}$ and $\bar{z}^b_{k+1}$ are included in the interval vectors $[z^a_{k+1}]$ and $[z^b_{k+1}]$, respectively. An algorithm to ensure this is described in Section 4.1.7.

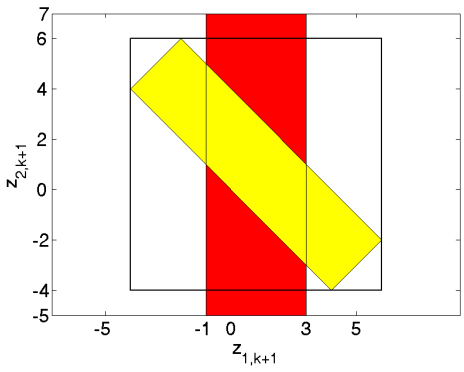
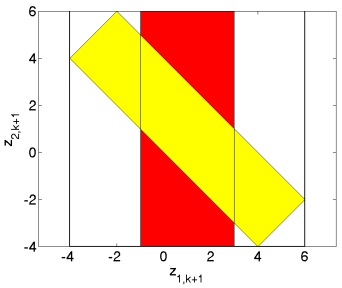
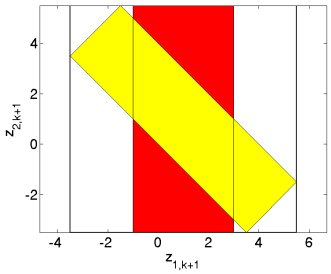


Figure 4.1: Intersection of two parallelepipeds: Initial enclosures.

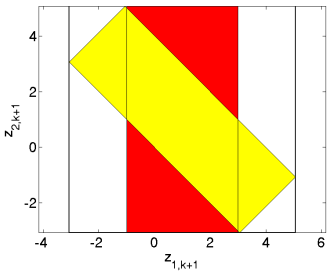


(a) After 1 Iteration.

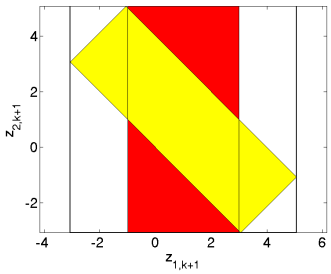


(b) After 2 Iterations.

Figure 4.2: Intersection of two parallelepipeds after 1 and after 2 iterations. .



(a) After 5 Iterations.



(b) After 10 Iterations.

Figure 4.3: Intersection of two parallelepipeds after 5 and after 10 iterations.

4.1.6 Interval Splitting

In nonlinear uncertain systems the solution sets are often given by complexly shaped non-convex regions. An enclosure of such sets by a single interval vector or a parallelepiped may lead to overestimation. However, if several interval vectors are used for enclosing the solution [26, 35, 36, 59, 62], the overestimation is reduced significantly as illustrated in Fig. 4.4. The extended state vector in time-step k is then enclosed by a list $\mathcal{Z}_k$ of interval vectors:

$$z_k \in \mathcal{Z}_k = \left\{ [z_k^{(1)}], [z_k^{(2)}], \ldots, [z_k^{(L_k)}] \right\} \quad \text{with} \quad L_k \leq L_{max}. \tag{4.53}$$

To avoid an exponential growth of the number of subboxes from time-step to time-step

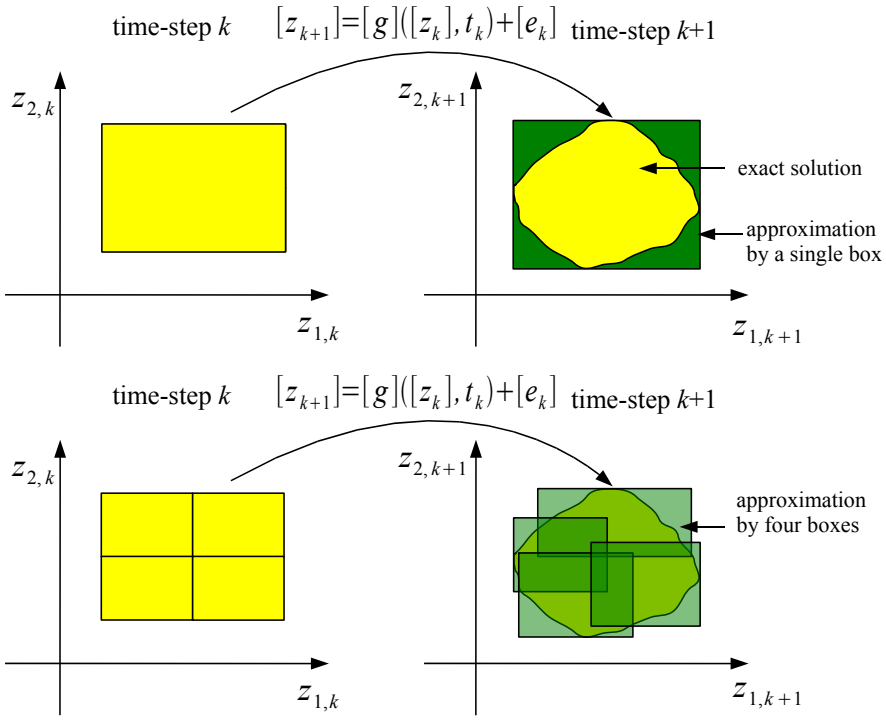


Figure 4.4: Enclosure of the solution set by a single interval vector and by several interval vectors.

efficient selection, splitting and merging strategies are required.

Selection Strategy

In every time-step a specified number of interval vectors is split into subboxes. The selection criterion is the pseudo volume of the interval vectors. The pseudo volume $Vol([z])$ of a

n-dimensonal interval vector $[z]$ is calculated by the multiplication of the interval diameters of all its components:

$$Vol([z]) = \prod_{i=1}^{n} \text{diam}([z_i]) \ . \tag{4.54}$$

In [40] a criterion which considers the stability properties of the interval vectors has been proposed. The interval vectors for which the system has potential instable behavior are selected for splitting. However, this criterion is not used in this work.

Splitting Strategy

If an interval vector is split into subboxes, first the splitting direction has to be determined [9,27,35,36,59,62]. Let $[z]$ be an interval vector which has to be split, the simplest splitting strategy is to split the interval vector in the direction μ with the largest diameter, according to

$$\mu = \underset{i=1...n}{\arg\max} \{\text{diam}\,([z_i])\} \ . \tag{4.55}$$

The bisection is usually performed in the midpoint of the selected component μ of the corresponding interval vector. However, this strategy requires usually a global scaling of the system, as each component of the interval vector may be related to a different physical unit. And such a global scaling is usually difficult.

A more efficient splitting strategy is described in the following. Consider a function $F(z) : \mathbb{R}^n \longmapsto \mathbb{R}^m$ which is evaluated over the interval vector $[z]$. The goal is now to find the component of z for which F is most sensitive. The 2-dimensional example in Fig. 4.5 illustrates that issue. Splitting an interval vector $[z]$ in the component $[z_1]$ before evaluation of F leads to two boxes with only little intersection in contrast to the case when the interval vector is split in the component $[z_2]$. Consequently the interval vector should be split in $[z_1]$ as the function F is more sensitive for this variable. The efficiency of the splitting can be calculated by

$$\eta = \frac{\text{Vol}([F]([z]))}{\text{Vol}([F]([z_\alpha])) + \text{Vol}([F]([z_\beta])) - \text{Vol}([F]([z_\alpha]) \cap [F]([z_\beta]))} \tag{4.56}$$

where $[z_\alpha]$ and $[z_\beta]$ are two subboxes obtained from the splitting operation. The splitting direction μ, which leads to the smallest η is chosen for splitting. This approach has the drawback that the function F has to be evaluated n times.

However, this criterion can be approximated by the following approach. First all

$$d_{i,j} = \text{diam}([z_i]) \cdot \left| \frac{\partial F_j}{\partial z_i} \right|_{z=\text{mid}([z])}, \quad i = 1...n, j = 1...m, \tag{4.57}$$

are determined. Alternatively the $d_{i,j}$ can be calculated by

$$d_{i,j} = \text{diam}([z_i]) \cdot \text{diam}\left(\left[\frac{\partial F_j}{\partial z_i} \Big|_{z=[z]} \right] \right), \quad i = 1...n, j = 1...m, \tag{4.58}$$

or

$$d_{i,j} = \operatorname{diam}\left(\left[\frac{\partial F_j}{\partial z_i}\Big|_{z=[z]}\right] \cdot ([z_i] - \operatorname{mid}([z_i]))\right), \quad i = 1...n, j = 1...m. \tag{4.59}$$

The box is split in the component μ for which

$$\mu = \underset{i=1...n}{\arg\max} \left\{ \sum_{j=1}^{m} d_{i,j} \right\} \tag{4.60}$$

holds. Here, the function F corresponds to equation (4.12) and $[z]$ corresponds to the interval

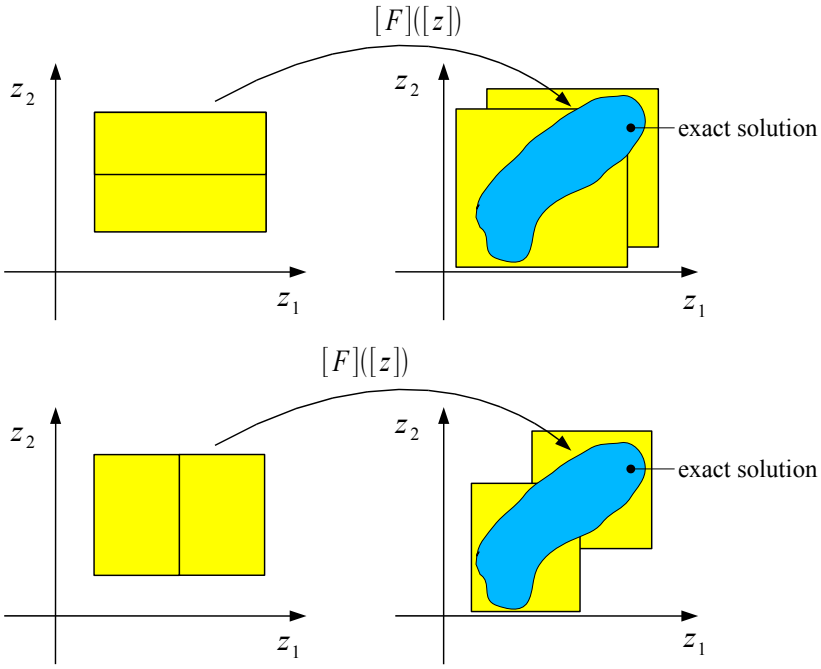


Figure 4.5: Inefficient and efficient splitting of an interval vector $[z]$.

vector $[z_k^{(l)}]$ from the list $\mathcal{Z}_k$ which has been selected for splitting.

Merging Strategy

In order to avoid an exponential growth of the number of interval vectors during simulation time, efficient merging strategies have to be applied [35, 36, 59, 62]. The merging of two intervals $[z_a]$ and $[z_b]$ means that they are replaced by the smallest possible interval vector around both. Two interval vectors are merged if they fulfill the following criterion:

The overestimation δ_{hull} in % in relation to the pseudo volume of the two interval vectors

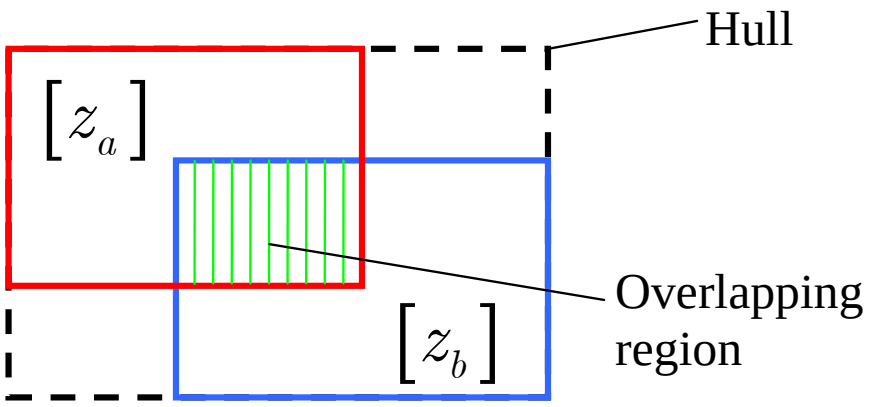


Figure 4.6: Merging of two interval vectors.

without the overlapping parts of the resulting overestimation has to be smaller then a specified value $\delta_{hull,limit}$, i.e.

$$\delta_{hull} \leq \delta_{hull,limit} \quad . \tag{4.61}$$

For the calculation of δ_{hull} the relation

$$\delta_{hull} = \frac{\mathrm{Vol}([z_a] \cup [z_b]) - \{\mathrm{Vol}([z_a]) + \mathrm{Vol}([z_b]) - \mathrm{Vol}([z_a] \cap [z_b])\}}{\mathrm{Vol}([z_a]) + \mathrm{Vol}([z_b]) - \mathrm{Vol}([z_a] \cap [z_b])} \cdot 100\% \tag{4.62}$$

holds.

For a second criterion the overlapping of the intersecting domain of $[z_a]$ and $[z_b]$ with the smaller interval vector of both

$$\delta_{intersect} = \frac{\mathrm{Vol}([z_a] \cap [z_b])}{\min\{\mathrm{Vol}([z_a]), \mathrm{Vol}([z_b])\}} \cdot 100\% \tag{4.63}$$

can be used. For interval merging

$$\delta_{intersect} \geq \delta_{intersect,limit} \tag{4.64}$$

has to be fulfilled, where $\delta_{intersect,limit}$ is a specified value. The choice of $\delta_{hull,limit}$ and $\delta_{intersect}$ is system dependent and relies on experience. The merging of two interval vectors is shown in Fig. 4.6

Reapproximation by Disjoint Interval Vectors

If a lot of interval vectors are overlapping splitting becomes inefficient. Since if one interval vector which is split is largely overlapped by another interval vector which is not split the splitting of the interval vector is actually useless. Thus, a reapproximation by disjoint interval

vectors [27, 62] becomes necessary. Starting with the hull of all subboxes, which is a single interval box, splitting into disjoint subboxes is performed. The split subboxes are compared with the original set and an intersection is performed. Subintervals, for which the intersection is an empty set, are deleted. The reapproximation procedure usually improves simulation quality after short-term widening of the bounds of the state variables, since splitting of disjoint intervals obtained after reapproximation is more efficient.

4.1.7 Combination of Interval Splitting and Coordinate Transformation

In the following it is shown how interval splitting can be combined with coordinate transformation in an efficient way. It is assumed that Lohners method, which uses parallelepipeds, is combined with a method employing interval vectors for the state enclosure, e.g. the naive evaluation or iterative evaluation of (4.12). The parallelepipeds are given by

$$z_{k+1}^p := \bar{z}_{k+1} + A_{k+1}[r_{k+1}] \tag{4.65}$$

and the interval enclosures by $[z_{k+1}^i]$. The extended state vector z_{k+1} is enclosed by

$$z_{k+1} \in \bar{z}_{k+1} + A_{k+1}[r_{k+1}] \cap [z_{k+1}^i] \ . \tag{4.66}$$

The first possibility is now to split the interval $[r_{k+1}]$ into subboxes $[\tilde{r}_{k+1}]$ leading to multiple parallelepipeds

$$\tilde{z}_{k+1}^p := \bar{z}_{k+1} + A_{k+1}[\tilde{r}_{k+1}] \ . \tag{4.67}$$

After the splitting each parallelepiped is intersected with the interval vector $[z_{k+1}^i]$ according to the algorithm described in 4.1.5, leading to a new subboxes $[\tilde{z}_{k+1}^i]$ for each subbox $[\tilde{r}_{k+1}]$ of $[r_{k+1}]$. This is depicted in Fig. 4.7 a) and Fig. 4.7 b). Fig. 4.7 a) shows two parallelepipeds obtained from the splitting of $[r_{k+1}]$ in two subboxes together with the interval vector $[z_{k+1}^i]$. Fig. 4.7 b) shows the final result after the intersection. The boxes $[\tilde{z}_{k+1}^i]$ are used for calculation of the Jacobian enclosure $[S_{k+1}]$ in the Lohner method in the next time-step. However, in order to provide that the computation steps in (4.39) are correct it is important that the vector $\bar{z}_{k+1}$ is included in the interval vector $[z_{k+1}^i]$ obtained after the intersection of $\tilde{z}_{k+1}^p \in \bar{z}_{k+1} + A_{k+1}[\tilde{r}_{k+1}]$ and $[z_{k+1}^i]$. To provide this first $\bar{z}_{k+1}$ and A_{k+1} are modified. Therefore the interval vector $[\tilde{r}_{k+1}]$ is expressed in terms of $[r_{k+1}]$:

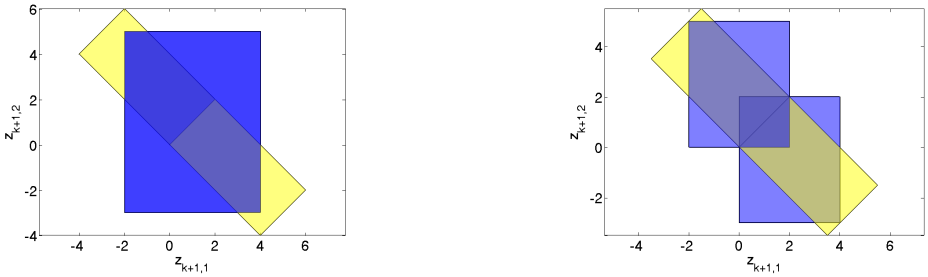
$$[\tilde{r}_{k+1}] = \inf([\tilde{r}_{k+1}]) - D \cdot \inf([r_{k+1}]) + D \cdot [r_{k+1}] \tag{4.68}$$

where D is a diagonal matrix with

$$d_{i,i} = \frac{\operatorname{diam}([\tilde{r}_{i,k+1}])}{\operatorname{diam}([r_{i,k+1}])} \ . \tag{4.69}$$

This results again in a modified representation of the same parallelepiped

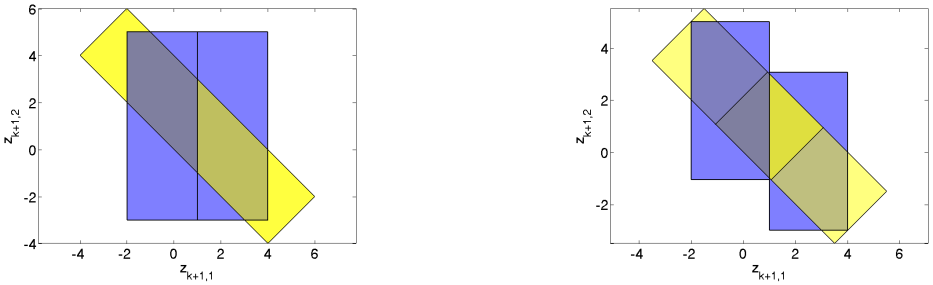
$$\begin{aligned} \tilde{z}_{k+1}^p &:= \bar{z}_{k+1} + A_{k+1}(\inf([\tilde{r}_{k+1}]) - D \cdot \inf([r_{k+1}])) + D \cdot [r_{k+1}]) \\ &= \bar{z}_{k+1} + A_{k+1}(\inf([\tilde{r}_{k+1}]) - D \cdot \inf([r_{k+1}]))) + A_{k+1}(D \cdot [r_{k+1}]) \\ &= \tilde{\bar{z}}_{k+1} + \tilde{A}_{k+1}[r_{k+1}] \ . \end{aligned} \tag{4.70}$$



(a) Initial Enclosure. (b) After 2 Iterations.

Figure 4.7: Intersection of two parallelepipeds with an interval vector.

Now, the parallelepiped is centered around $\tilde{\bar{z}}_{k+1}$. If $\tilde{\bar{z}}_{k+1}$ is still not included in $[\tilde{z}^i_{k+1}]$ the hull around $[\tilde{z}^i_{k+1}]$ and $\tilde{\bar{z}}_{k+1}$ has to be used the calculation of $[S_{k+1}]$ instead. This method has also to be applied when two parallelepipeds are intersected according to Section 4.1.5. If now instead of the parallelepipeds the interval vectors $[z^i_{k+1}]$ are split, then the subboxes $[\tilde{z}^i_{k+1}]$ should be intersected with the parallelepiped $\bar{z}_{k+1} + A_{k+1}[r_{k+1}]$ in order to improve the enclosure by the parallelepiped. This procedure is illustrated in Fig. 4.8. Fig. 4.8 a)



(a) Initial Enclosure. (b) After 2 Iterations.

Figure 4.8: Intersection of an interval vector with two parallelepipeds.

shows two subboxes obtained from the splitting of $[z^i_{k+1}]$ in two subboxes together with the parallelepiped enclosure $\bar{z}_{k+1} + A_{k+1}[r_{k+1}]$. Fig. 4.8 b) shows the final result after the intersection. The parallelepipeds are modified as described previously. If $\tilde{\bar{z}}_{i,k+1}$ is still not enclosed by the interval enclosure obtained after the intersection, the hull around $\tilde{\bar{z}}_{i,k+1}$ and the interval vector has been used when calculating $[S_{k+1}]$ in the next integration step of the Lohner method.

If two subboxes $[z^a_{k+1}]$ and $[z^b_{k+1}]$ are merged to the interval vector $[z_{k+1}]$ the transformation matrix A_{k+1}, the interval vector $[r_{k+1}]$ and $\bar{z}_{k+1}$ are reinitialized according to (4.38).

In summary, the extended state vector z_{k+1} at $t = t_{k+1}$ is enclosed by

$$\begin{aligned} z_{k+1} \in \mathcal{Z}_{k+1} = &\left\{[z^{(1)}_{k+1}], [z^{(2)}_{k+1}], \ldots, [z^{(L_{k+1})}_{k+1}]\right\} \\ &\cap \left\{\bar{z}^{(1)}_{k+1} + A^{(1)}_{k+1}[r^{(1)}_{k+1}], \bar{z}^{(2)}_{k+1} + A^{(2)}_{k+1}[r^{(2)}_{k+1}], \ldots, \bar{z}^{(L_{k+1})}_{k+1} + A^{(L_{k+1})}_{k+1}[r^{(L_{k+1})}_{k+1}]\right\} \\ &\text{with} \quad L_{k+1} \leq L_{max}. \end{aligned} \tag{4.71}$$

The list $\mathcal{Z}_{k+1}$ then contains in addition to the interval enclosures $[z_{k+1}^{(l)}]$ the information for the enclosures by the parallelepipeds namely $\bar{z}_{k+1}^{(l)}$, $A_{k+1}^{(l)}$ and $[r_{k+1}^{(l)}]$.

Illustrative Example: Lotka-Volterra Equations

The Lotka–Volterra equations are given by:

$$\begin{aligned} \dot{z}_1 &= z_1(a - bz_2) \ , \\ \dot{z}_2 &= z_2(cz_1 - d) \ . \end{aligned} \tag{4.72}$$

For $a = 1$, $b = 0.01$, $c = 0.02$,and $d = 1$ the equations become

$$\begin{aligned} \dot{z}_1 &= z_1(1 - 0.01z_2) \ , \\ \dot{z}_2 &= z_2(0.02z_1 - 1) \ . \end{aligned} \tag{4.73}$$

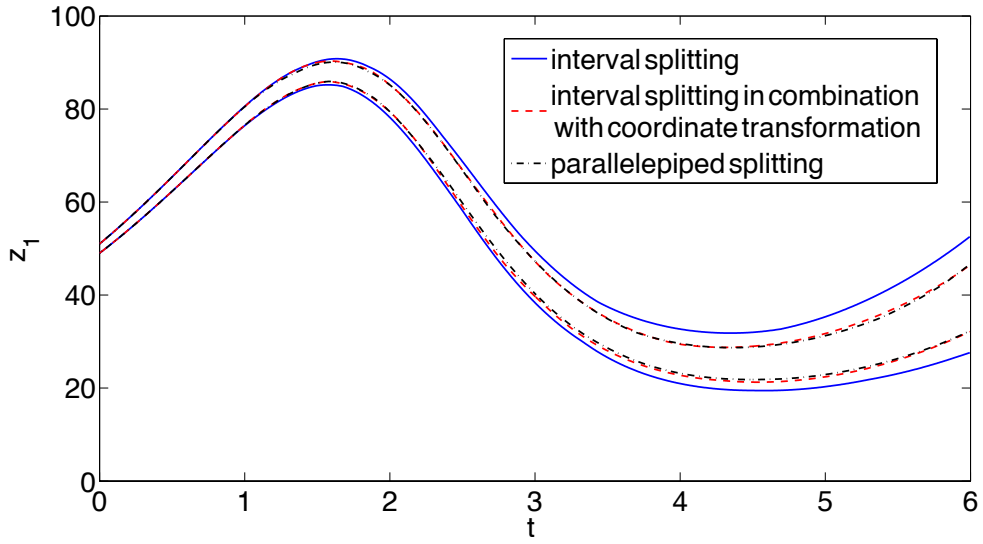
Three simulations were carried out over 1200 time-steps. The initial state variables were $z_1(0), z_2(0) \in [49; 51]$. In all simulations, the order was $\nu = 1$ and the step size was $h = 0.005$. In the first interval splitting and merging techniques without coordinate transformation was applied. The computation time was 263 s. The second employed additionally a coordinate transformation, and the computation time was 273 s. In the third simulation also a coordinate transformation was applied but the parallelepiped enclosures have been split. The computation time was 281 s. The maximum number of splittings and interval vectors was 20. Splitting and merging was performed every 20 time-steps. The widest bounds are obtained in the first case. The results of the second simulation are much tighter with only a slight increase in computation time. The third simulation lead to the best enclosures, however, they are only slightly better than those of the second case and the computation time was 8 s longer.

4.2 Consistency Techniques for Reduction of Overestimation

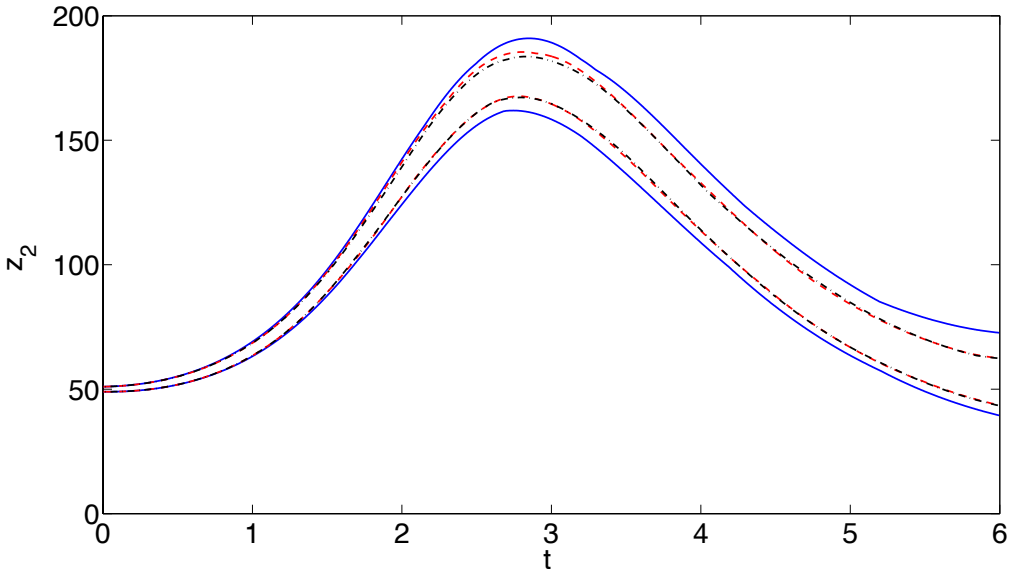
The simulation algorithms described so far rely on a forward integration of the system equation. The consistency tests [11,35,38] described in this section are based on the backward integration of the system equation according to

$$[z_k] = [z_{k+1}] + \sum_{i=1}^{\nu} \frac{(-h_k)^i}{i!} f^{(i-1)}([z_{k+1}], t_{k+1}) + [e_{k+1}]. \tag{4.74}$$

for subboxes and a comparison of the results with the solution obtained from the forward integration. The result gives information, if the subbox belongs to the solution of the ODE at time-step $k + 1$ or if it is a result from overestimation.



(a) Interval enclosures for z_1.



(b) Interval enclosures for z_2.

Figure 4.9: Comparison of the interval enclosures resulting from the verified integration of the Lotka-Volterra Equations.

4.2.1 One Step Consistency Tests

In the following a one-step consistency test [35,38] is described which consists of an alternating application of forward an backward steps. This illustrated in Fig. 4.10. First a forward step from time-step k to $k+1$ is performed resulting into $[z_{k+1}]$. The exact set is given by $\check{\mathscr{Z}}_{k+1}$. Then subboxes $[\tilde{z}_{k+1}]$ of the obtained solution are backward integrated resulting into $[\tilde{z}_k]$. Now three different cases have to be distinguished:

a) $[\tilde{z}_k] \cap [z_k] = \emptyset$
The backward integrated interval vector lies completely outside of $[z_k]$. Therefore, $[\tilde{z}_{k+1}]$ is *inconsistent* with respect to the enclosure at time-step k and resulted from overestimation in the forward integration. It belongs not to the solution at $k+1$ with respect to $[z_k]$ and can be deleted.

b) $[\tilde{z}_k] \subset [z_k]$
In this case, $[\tilde{z}_k]$ lies completely inside of $[z_k]$. That means $[\tilde{z}_{k+1}]$ is consistent and belongs to the exact solution of time step $k+1$ with respect to $[z_k]$.

c) $[\tilde{z}_k] \cap [z_k] \neq \emptyset$ and $[\tilde{z}_k] \not\subset [z_k]$
In this case, no conclusion can be made because the interval vector $[\tilde{z}_k]$ is only included partially in $[z_k]$ and further splitting of $[\tilde{z}_{k+1}]$ is required.

The repeated application of splitting and backward calculation yields an improved approximation of the actual solution in time step $k+1$.

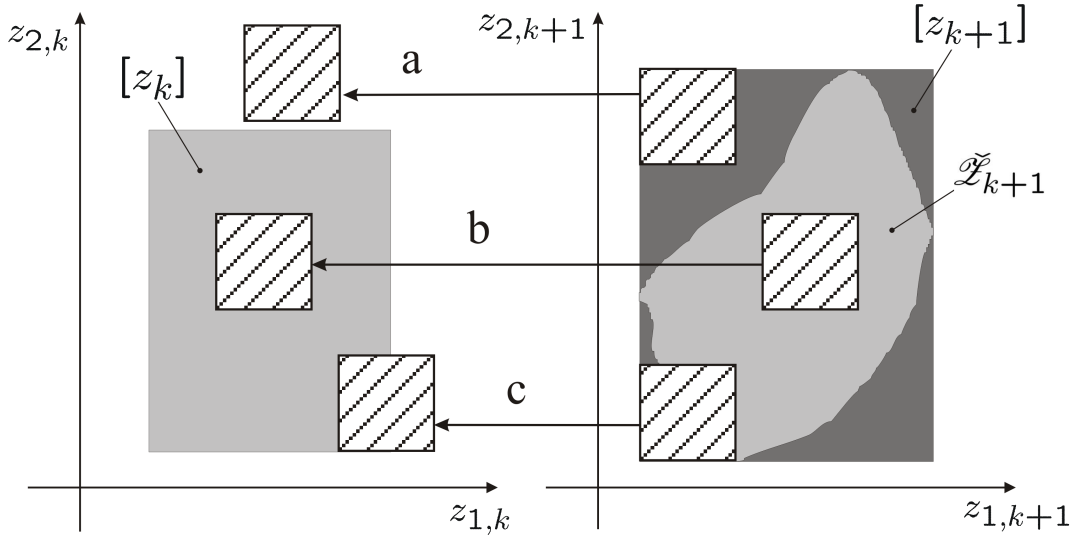


Figure 4.10: Distinction between three different cases: In case *a*) the box is inconsistent and can be deleted; the box in case *b*) is consistent and belongs to the solution set; in case *c*) further splitting is required.

Selection Strategy

If after the application of the consistency test no conclusion about the consistency can be made, the corresponding interval vector has to be split further. During the consistency test, for many subboxes no conclusion can be made and the corresponding intervals have to be split further. In order to limit the computational effort the number of allowed splitting operations is limited. Therefore, efficient selection strategies are required. In case *c*) after the consistency test $[\tilde{z}_k]$ and $[z_k]$ have an intersecting set given by the interval

$$[\chi_k] = [\tilde{z}_k] \cap [z_k]. \tag{4.75}$$

The ratio between the volume of $[\chi_k]$ and the volume of the subbox $[\tilde{z}_k]$ in time-step k is given by

$$\gamma_k = \frac{\mathrm{Vol}([\chi_k])}{\mathrm{Vol}([\tilde{z}_k])} \quad . \tag{4.76}$$

The larger γ_k, the larger is the intersection of $[\tilde{z}_k]$ with $[z_k]$ in relation to the total volume of $[\tilde{z}_k]$. It is unlikely that, if an interval with a large value of γ_k is split into two subboxes, the consistency test will be successful and lead to case *a*) or *b*). If however γ_k is small, then also the intersection of $[\tilde{z}_k]$ with $[z_k]$ in relation to the total volume of $[\tilde{z}_k]$ is small and a splitting of $[\tilde{z}_{k+1}]$ increases the probability of a successful consistency test after splitting. This is also depicted in Fig. 4.11.

In order to prevent that already very small intervals are split further, γ_k is weighted with the inverse of the pseudo volume of $[\tilde{z}_{k+1}]$ according to

$$\gamma_k^* = \frac{\gamma_k}{\mathrm{Vol}([\tilde{z}_{k+1}])} \quad . \tag{4.77}$$

A specified number of intervals with small γ_k^* are selected for the consistency test.

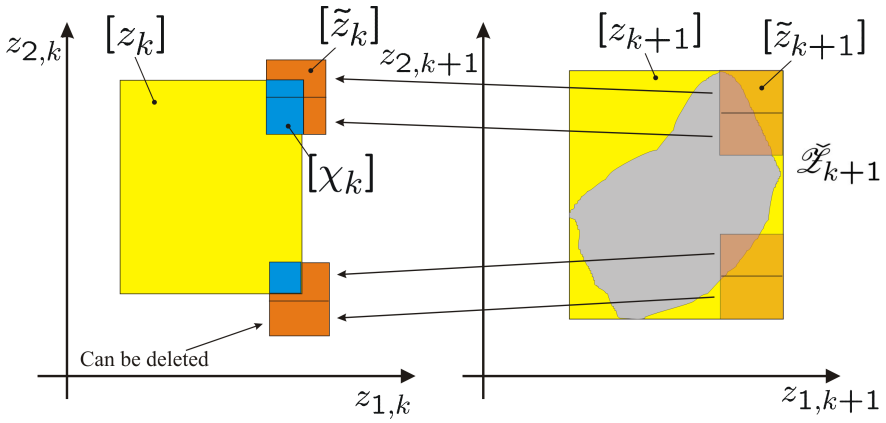


Figure 4.11: Comparison of two interval vectors with different overlapping regions.

Splitting Strategy

An efficient strategy to determine the splitting direction enables a faster deletion of inconsistent subbox. To determine the splitting direction the criterion described in Section 4.1.6 is applied. The sensitivity analysis is performed to equation (4.74).

Flow Diagram

Fig. 4.12 summarizes the algorithm. At time-step $k+1$ the state vector z_{k+1} is enclosed by a list of interval vectors $\mathcal{Z}_{k+1}$:

$$z_{k+1} \in \mathcal{Z}_{k+1} = \left\{ [z_{k+1}^{(1)}], [z_{k+1}^{(2)}], \ldots, [z_{k+1}^{(L_{k+1})}] \right\} \quad \text{with} \quad L_{k+1} \leq L_{max}. \tag{4.78}$$

After the forward step first the values γ^* for each interval $[\tilde{z}_{k+1}^{(l)})]$ is set to $\frac{1}{\mathrm{Vol}([\tilde{z}_{k+1}^{(l)}])}$. Then a given number of interval vectors is selected for the consistency test. After the evaluation of the splitting criterion, the subboxes are split and the backward step is applied. Inconsistent boxes are deleted; consistent boxes are stored in a different list and are not split further. For the remaining interval vectors, the values for γ_k^* according to equation (4.77) are calculated. Next, a user specified number of new intervals with the smallest value of γ^* are selected for the consistency test. This procedure is repeated until a given maximum number of interval vectors or splitting operations is reached.

The consistency test can be combined with the previously described reapproximation routine. Then the algorithm starts with a single box enclosing all interval vectors. The resulting enclosure then consists of disjoint interval vectors.

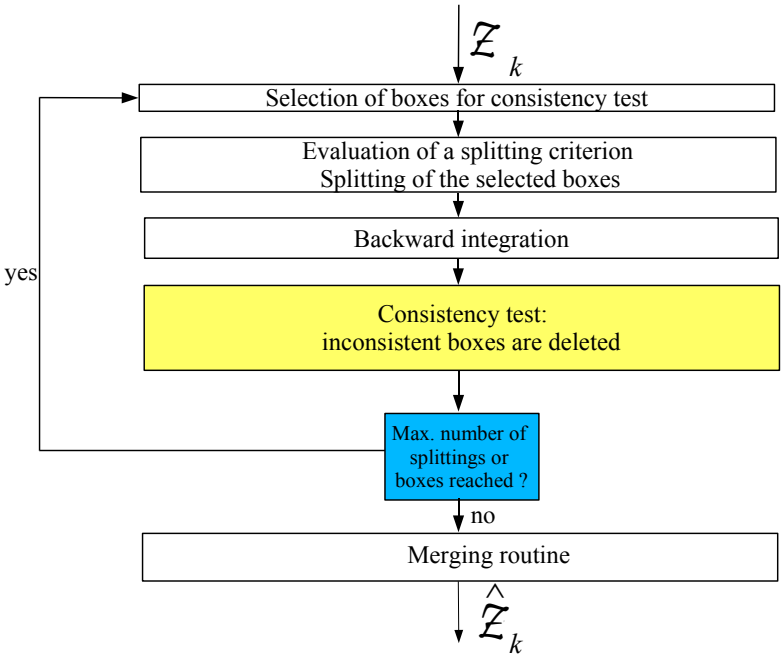
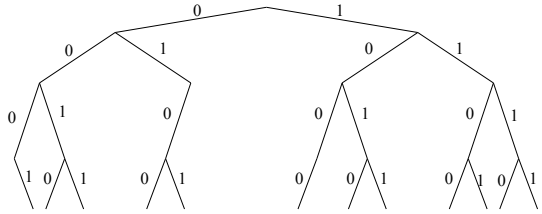


Figure 4.12: Flow diagram of the consistency test.

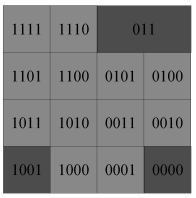
4.2.2 Merging Strategy

The consistency test and the associated splitting leads inevitably to an increasing number of interval vectors. To avoid an exponential growth of the interval vector number from time-step to time-step also efficient merging strategies have to be applied. Two interval vectors

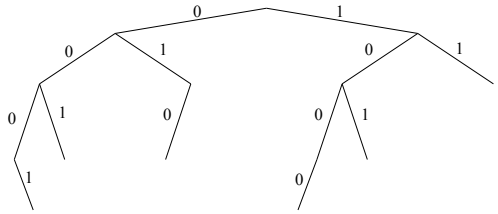
can be merged if the smallest interval around both leads to no or only little overestimation. The merging consists of several steps. During the splitting of an interval vector, each subbox



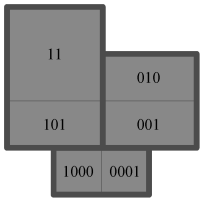
(a) Tree after splitting and deleting of inconsistent subboxes.



(b) Intervals after splitting and deleting of inconsistent subboxes.



(c) Tree after merging.



(d) Intervals after merging.

Figure 4.13: Binary tree after splitting and after merging.

gets a binary number that is stored in a tree structure. If an interval vector is split into two subboxes each subbox gets a new binary number consisting of the binary number of the original interval vector and an additional 0 or 1 respectively at the end leading to a binary tree with two new branches after each splitting. If an interval vector is deleted by the consistency test, no new branches arise from this interval vector. After the splitting, the tree gives information about interval vectors, which can be merged without further overestimation. It just has to be checked if two subboxes descend from the same original box. This is the case if the corresponding binary number differs only at the last digit. After the merging with the help of the tree there may be still some boxes left which could be merged. Therefore the merging routine described in Section 4.1.6 is used. The merging routine is illustrated in Fig. 4.13. The dark gray boxes (Fig. 4.13 (b)) have been deleted exemplarily as well as the corresponding branches in binary tree (Fig. 4.13 (a)). The gray boxes around the boxes in Fig. 4.13 (d) mark boxes, which are merged by the second merging routine.

4.2.3 Multi-Step Consistency Techniques

In this work, it is shown how the previous described consistency tests can be extended to a so-called *multi-step consistency test*. In a *multi-step consistency test* the backward

integration is done over several time-steps. This has the advantage that also overestimation with respect to previous time-steps can be reduced. For the backward step again (4.74) is used and is evaluated over several time-steps. In each backward step, the obtained result is compared with the result from the forward propagation and can be therefore checked for consistency. Fig. 4.14 illustrates the basic principle of the *multi-step consistency test*. A forward calculation of the box $[z_k]$ has been done over two time-steps. The exact solution at $k+1$ is given by the set $\mathscr{Z}_{k+1}$ and the interval enclosure of z_{k+1} is denoted by $[z_{k+1}]$. In time $k+2$, the exact solution set with respect to $\mathscr{Z}_{k+1}$ is the set $\mathscr{Z}_{k+2}$, whereas the exact solution with respect to $[z_{k+1}]$ is given by the set $\check{\mathscr{Z}}_{k+2}$, which is enclosed by the interval vector $[z_{k+2}]$. In this illustrative example the overestimation of $[z_{k+2}]$ compared to the real solution set $\mathscr{Z}_{k+2}$ is already large. The consistency test is now applied to four different subboxes α, β, Υ, and X of $[z_{k+2}]$. The subbox α lies completely outside of $\mathscr{Z}_{k+2}$ and $[z_{k+2}]$. After the backward step the exact solution set in time-step $k+1$ is already completely outside of $[z_{k+1}]$, but the interval enclosure is intersecting $[z_{k+1}]$. The interval that forms the intersection is now backward propagated to time step k. The resulting exact solution and the corresponding interval enclosure are completely outside of $[z_k]$. Therefore, the inconsistency of α has been prove and α is deleted. Interval β lies completely inside of $\check{\mathscr{Z}}_{k+2}$ but completely outside of $\mathscr{Z}_{k+2}$. It is therefore consistent with $[\mathscr{Z}_{k+1}]$ but inconsistent with respect to $\mathscr{Z}_{k+1}$ and therefore also inconsistent with respect to $[z_k]$. This has to be proven by the backward calculation. It is first backward calculated to time-step $k+1$ and then further backward calculated. The exact solution is now already completely outside of $[z_{k+1}]$, however the interval enclosure is intersecting this box. After the second backward step the resulting interval vector is lying completely outside of $[z_k]$. The third interval Υ is completely inside $\mathscr{Z}_{k+2}$ after two backward steps the resulting box is also completely inside $[z_k]$. Therefore, it has been proved that Υ belongs to the exact solution set $\mathscr{Z}_{k+2}$ with respect to $[z_k]$. The forth interval X lies partially inside and partially outside of $\mathscr{Z}_{k+2}$. The resulting interval at k after two backward steps lies also partially inside and partially outside of z_k. This interval has to be split further and the backward computation has to be repeated for the corresponding subboxes.

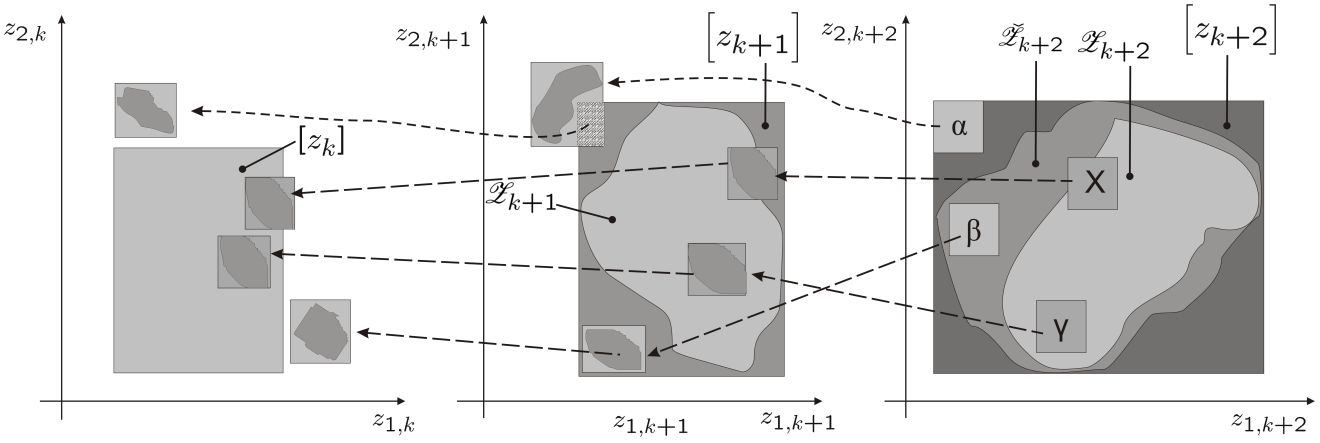


Figure 4.14: Backward calculation over three time-steps.

This illustrative example employed only a single box in the forward calculation. If multiple interval boxes are used, then the result of the backward calculation has to be compared with

all interval vectors in each backward step. Around the sets defining the intersection with each subbox an interval hull has to be determined to continue the backward computation. This is depicted in Fig. 4.15.

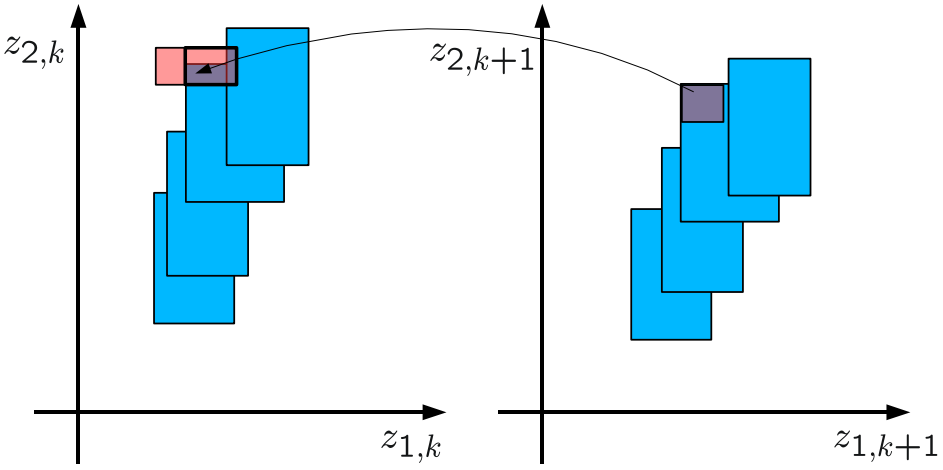


Figure 4.15: Intersection with several interval vectors in the backward calculation and enclosure of the intersecting set by an interval vector.

Selection Strategy

In order to keep the computational effort as low as possible it is important to apply efficient strategies for the selection of intervals for the consistency test. The following three strategies are proposed:

1. One strategy is to choose the interval vector with the largest pseudo-volume.

2. If in the forward step interval splitting had been applied, it is also possible to select boxes with maximum distance to the balance point of the original set, where for the balance point an approximate value is sufficient. However, before this value is calculated, a normalization of the original set has to be done.

3. The selection criterion described in Section 4.1.6 can also be used for the *multi-step consistency test*. If it has been carried out from time-step $k+1$ over N time-steps ($N < k+1$), then the sum over the values of γ_k in each time is calculated, namely

$$\gamma_{kN} = \sum_{l=0}^{N} \gamma_{k-l} \ . \tag{4.79}$$

Additionally a weighting with the inverse of the pseudo-volume of the corresponding interval $[z_{k+1}]$ at time-step $k+1$ is done in accordance to (4.77), which results in

$$\gamma^*_{kN} = \frac{\gamma_{kN}}{\mathrm{Vol}([z_{k+1}])} \ . \tag{4.80}$$

The interval vectors with the smallest value of γ^*_{kN} are selected for the consistency test.

Terminating Condition

The γ-values can be used to judge if it makes sense to continue the backward integration. For that purpose the γ-values are compared if the value of γ remains constant or is even increasing. The integration is stopped if over a particular number of M backward steps

$$\gamma_{\tilde{k}} \leq \gamma_{\tilde{k}-1} \leq \gamma_{\tilde{k}-2} \cdots \leq \gamma_{\tilde{k}-M} \ . \tag{4.81}$$

holds. Then new intervals are selected for the consistency test, since it becomes improbable that the interval can be deleted after further backward calculation. If $k+1$ is the starting point of the backward integration, then $\tilde{k} \leq k$ holds. This terminating condition leads to a reduction of computational effort. If γ is decreasing in between the counting of the backward steps is set to zero again.

Splitting Strategy

The splitting strategy described in Subsection 4.2.1 can be extended to the *multi-step consistency test*. Therefore, the influence of the components of the state variable z is analyzed over several time-steps $\tilde{N} \leq N$, where N is the maximum allowed number of backward steps. For the calculation of the state vector over $\tilde{N}$ time-steps

$$\begin{aligned}
[z_k] &= [g^-_{k+1}]([z_{k+1}]) + [e_{k+1}]) \ , \\
[z_{k-1}] &= [g^-_k]([z_k]) + [e_k] = [g^-_k](g^-_{k+1}([z_{k+1}]) + [e_{k+1}]) + [e_k] \ , \\
&\vdots \\
[z_{k-\tilde{N}}] &= [g^-_{k-\tilde{N}-1}](\ldots g^-_k(g^-_{k+1}([z_{k+1}]) + [e_{k+1}]) + [e_k], \ldots) + [e_{k-\tilde{N}-1}]
\end{aligned} \tag{4.82}$$

holds. The $-$ in g^-_k indicates the backward integration and the index k denotes g in time-step k, hence

$$g^-_k(z_k) := g^-(z_k, t_k) \ . \tag{4.83}$$

Applying the chain rule and neglecting the error terms, the gradient results in

$$\begin{aligned}
\frac{\partial z_k}{\partial z_{k+1}} &= \frac{\partial g_{k+1}^-}{\partial z_{k+1}} \ , \\
\frac{\partial z_{k-1}}{\partial z_{k+1}} &= \frac{\partial g_k^-}{\partial g_{k+1}^-} \cdot \frac{\partial g_{k+1}^-}{\partial z_{k+1}} = \frac{\partial g_k^-}{\partial z_k} \cdot \frac{\partial g_{k+1}^-}{\partial z_{k+1}} \ , \\
&\vdots \\
\frac{\partial z_{k-\tilde{N}}}{\partial z_{k+1}} &= \frac{\partial g_{k-\tilde{N}+1}^-}{\partial g_{k-\tilde{N}+2}^-} \cdot \ldots \cdot \frac{\partial g_k^-}{\partial g_{k+1}^-} \frac{\partial g_{k+1}^-}{\partial z_{k+1}} \ , \\
&= \frac{\partial g_{k-\tilde{N}+1}^-}{\partial z_{k-\tilde{N}+1}} \cdot \ldots \cdot \frac{\partial g_k^-}{\partial z_k} \frac{\partial g_{k+1}^-}{\partial z_{k+1}} \ .
\end{aligned} \tag{4.84}$$

Next, all values

$$\begin{aligned}
d_{i,j} &= \operatorname{diam}([z_{i,k}]) \sum_{l=0}^{N} \left| \frac{\partial z_{j,k-l}}{\partial z_{i,k-l+1}}\Big|_{z_{k-l+1}=\operatorname{mid}([z_{k-l+1}])} \right| \\
i,j &= 1,\ldots,n
\end{aligned} \tag{4.85}$$

are calculated. Alternatively the $d_{i,j}$ can be calculated by

$$d_{i,j} = \operatorname{diam}([z_i]) \cdot \operatorname{diam}\left(\sum_{l=0}^{N} \left[\frac{\partial z_{j,k-l}}{\partial z_{i,k-l+1}}\Big|_{z_k=[z_k]} \right] \right), \quad i = 1...n, j = 1...m, \tag{4.86}$$

or

$$d_{i,j} = \operatorname{diam}\left(\sum_{l=0}^{N} \left[\frac{\partial z_{j,k-l}}{\partial z_{i,k-l+1}}\Big|_{z_k=[z_k]} \right] \cdot ([x] - \operatorname{mid}([x])) \right), \quad i = 1...n, j = 1...m. \tag{4.87}$$

The splitting direction μ is determined according to

$$\mu = \arg \max_{j=1...n} \left\{ \sum_{i=1}^{n} d_{i,j} \right\} \ . \tag{4.88}$$

Flow Diagram

A block diagram of the *multi-step consistency test* is given in Fig. 4.16. If in the forward integration multiple interval vectors have been used it can be advantageous to replace these interval vectors by a single box. Especially if there are a lot of overlapping boxes. This assures that after the consistency test the interval vectors are disjoint. Then this single box is backward integrated and the splitting direction is determined. Next, the box is split into two subboxes. These subboxes are backward integrated again. This is repeated until the number of interval vectors is above a particular number $\tilde{L}$. Then only $\tilde{L}$ interval vectors are selected for the next application of the consistency test according to the chosen selection strategy. The consistency test is repeated IT times. If this number has been reached, the remaining interval vectors are merged and the forward integration can be continued.

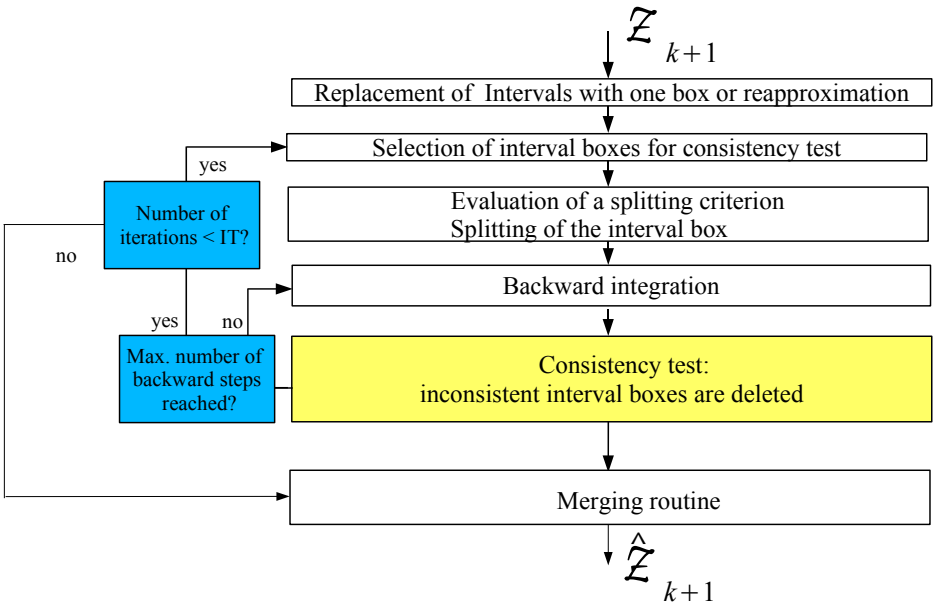


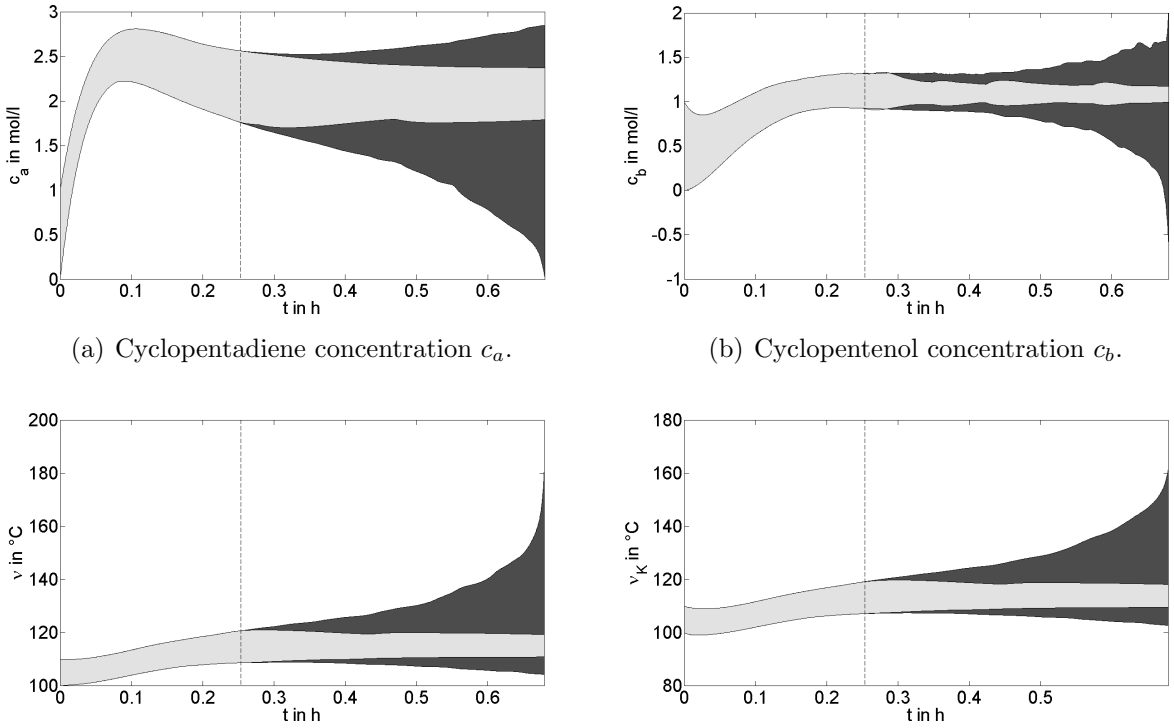
Figure 4.16: Flow diagram of the multi-step consistency test.

4.2.4 Application: Non-Isothermal Stirred Tank Reactor

In this section, the simulation results of a non-isothermal stirred tank reactor show the performance and the significant improvement of the result when the proposed algorithm is applied. The parameter that denotes the maximum number of backward steps is N. M is the number used for the termination criterion. The initial conditions were given by $c_a(0) \in [0;1]\text{mol/l}$, $c_b(0) \in [0;1]\text{mol/l}$, $\nu(0) \in [100;110]°C$, $\nu_K(0) \in [100;110]°C$. ΔH_{rab} was assumed to be uncertain with $\Delta H_{rab} \in [4.2 - 2.36; 4.2 + 2.36]\frac{\text{kJ}}{\text{kgK}}$ [13]. The variation rate of ΔH_{rab} was chosen to $\Delta \dot{H}_{rab} \in [-0.1; 0.1]\frac{\text{kJ}}{\text{kgKh}}$. The extended state vector $z(t)$ is defined as $z(t) = [c_a, c_b, v, v_k, \Delta H_{rab}]^T$. The step-size was chosen as $T = 0.0005h$ and the Taylor series was truncated after order $\nu = 1$. A maximum of $L_{max} = 500$ interval vectors was allowed during the forward propagation. Iterative calculation of infimum and supremum in combination with naive interval evaluation, mean-value evaluation, and monotonicity test was applied in the forward propagation. In each time-step, the maximum number of splittings during the iterative computation was 10 for each box.

The simulation results are shown in Fig. 4.17. The dark gray curves show simulation results where only the forward step was applied. It can be observed that after $t = 0.25h$ the results blow up dramatically. The bright gray results show the results, where a consistency test was performed at $t = 0.25h$, with $IT = 50$, $\tilde{L} = 100$, $M = 10$ and $N = 20$ was used. In the backward integration only naive interval analysis was applied. Optimized techniques could be used to improve the results further at the cost of more computation time. The selection criterion which uses the distance to the approximate balance-point of the original set was used (selection strategy 2). The splitting direction was determined by calculating

the sensitivity only with respect to the first backward-step. After the consistency test, the simulation continued with the forward propagation. It can be seen clearly, that the results have been improved significantly by the consistency test.



(a) Cyclopentadiene concentration c_a.

(b) Cyclopentenol concentration c_b.

(c) Reactor temperature v.

(d) Jacket temperature v_k.

Figure 4.17: Simulation results without and with consistency test.

Fig. 4.18 shows the result for fixed values of IT, M and $\tilde{L}$ and different values of M in the $[v, v_k]$-plane. Fig. 4.19 depicts the results for fixed values of IT, N and $\tilde{L}$ and different values of N.

The results show that it is crucial, not to stop the backward propagation too early. In the first case the results for $M = 20$ are much better than for $M = 5$. For variable values of N the results also differ a lot between $N = 20$ and $N = 100$.

In Fig. 4.20 simulation results for fixed values of $\tilde{L}$ and M and for different values of N and IT are shown. The different values of N and IT were chosen in a way that their product is 1000. Increasing the number of backward steps leads to better results with fewer splitting operations, if the overestimation between two backward steps is not too large. This means also that after the consistency test fewer boxes have to be forward calculated. Usually the number of maximum number of intervals for the forward propagation is also restricted. If after the consistency test too interval vectors are present no interval splitting is allowed until the merging routine has reduced this number below its maximum allowed value. Nevertheless, if the number of Iterations IT becomes too low while the maximum allowed number of backward steps N is increased at the same time, the quality of the simulation results also decreases.

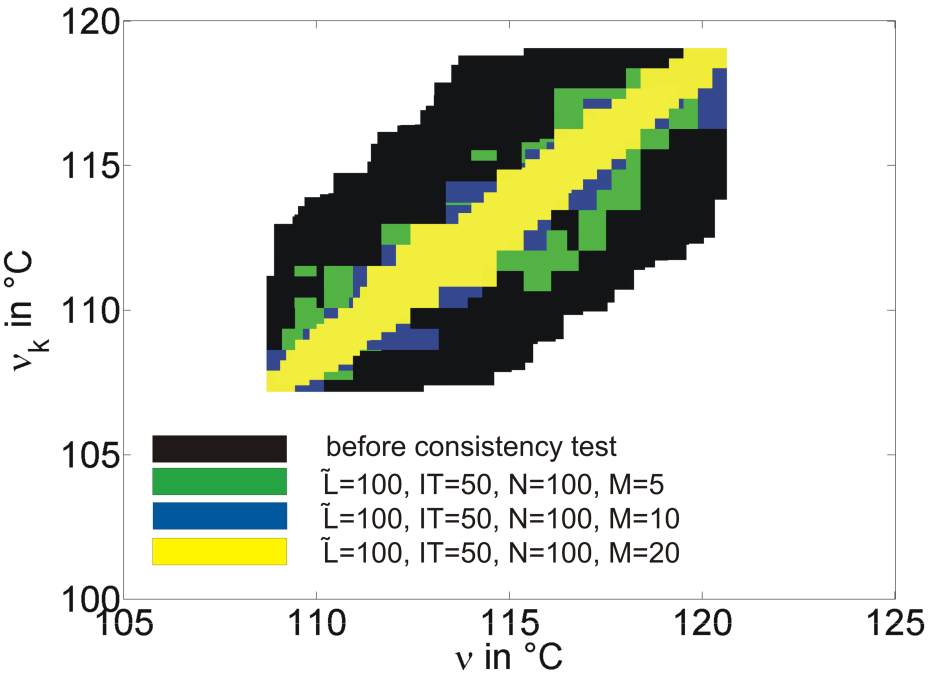


Figure 4.18: Projection in the $[v, v_k]$-plane for $N = 100$, $\tilde{L} = 100$, $IT = 50$ and different values of M.

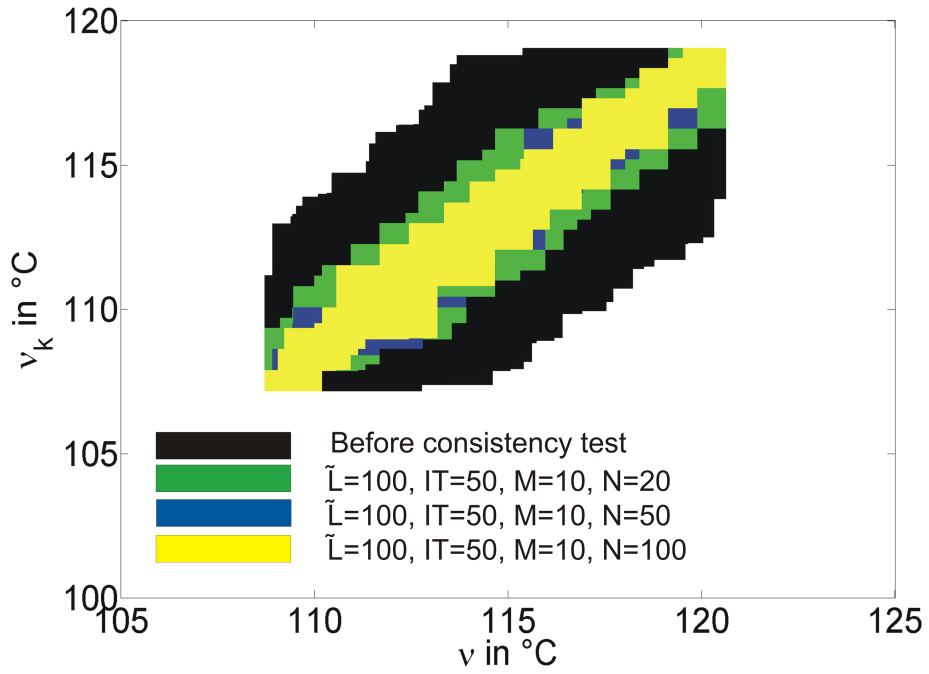


Figure 4.19: Projection in the $[v, v_k]$-plane for $M = 10$, $\tilde{L} = 100$, $IT = 50$ and different values of N.

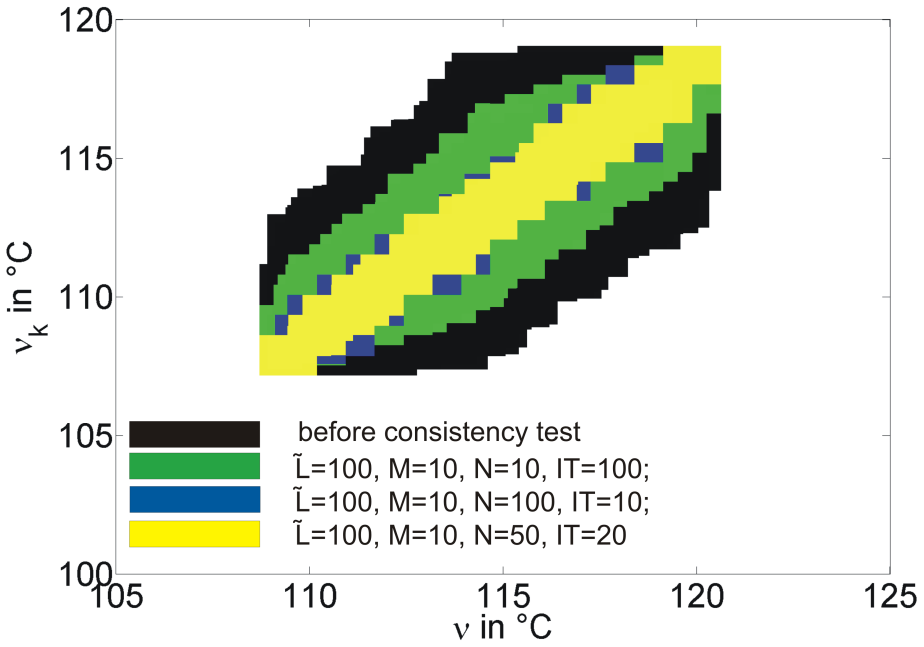


Figure 4.20: Projection in the $[v, v_k]$-plane for $M = 10$, $\tilde{L} = 100$ and different values of N and IT.

In Fig. 4.21 the results from Fig. 4.17 are compared with results if selection strategy 3 is used. The dark enclosures represent the result for selection strategy 3. The bright enclosures the results for selection strategy 2. The results when applying selection strategy 2 are slightly better; however the computational effort for evaluation of this criterion is larger.

4.3 Verified Integration of Uncertain Systems with State-Dependant Switching Characteristics

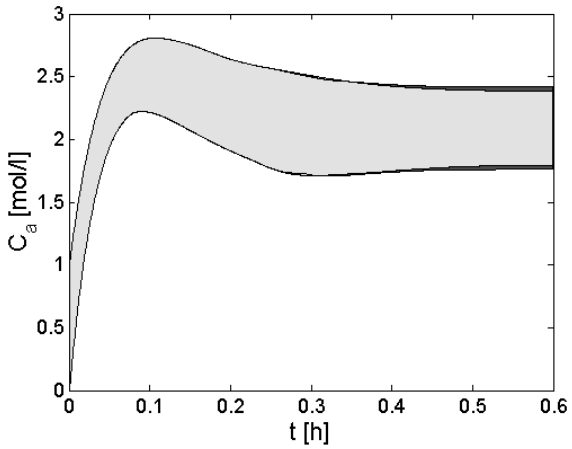
For many technical systems, the influence of friction or hysteresis effects [7,49,69] cannot be neglected. For the simulation of such systems the switching between different models has to be taken into account [61], e.g. the switching between static friction and sliding friction. A system with switching characteristics can be described by l different models

$$\mathcal{S} = \{S_1, S_2, ..., S_l\} \quad . \tag{4.89}$$

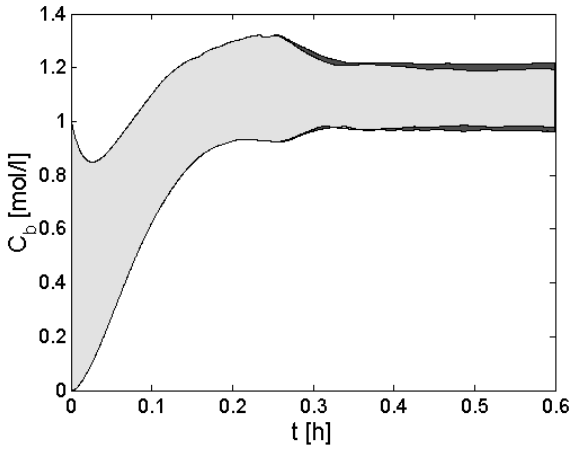
Each of these models is described by a its state space representation

$$\dot{z} = f_{S_i}(z(t), t), \quad i = 1, ..., l \quad . \tag{4.90}$$

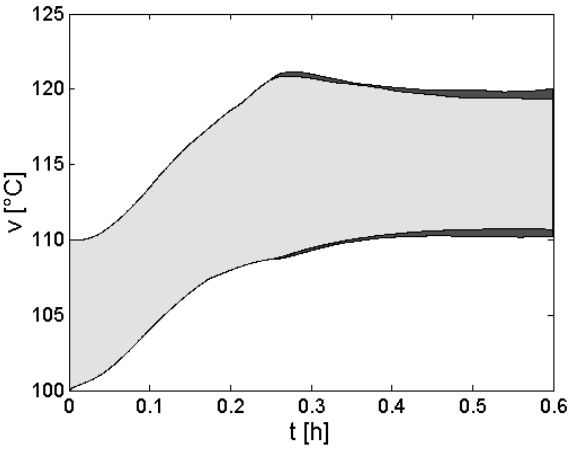
A transition from model S_i to another model S_j with $i, j = 1, ..., l$ occurs, if the necessary transition conditions $T_i^j(z, t)$ are fulfilled [61]. If the transition conditions are not fulfilled, the current model remains. In the case of systems with interval uncertainty, the transition conditions also become intervals. Therefore, the possibility has to be considered that more



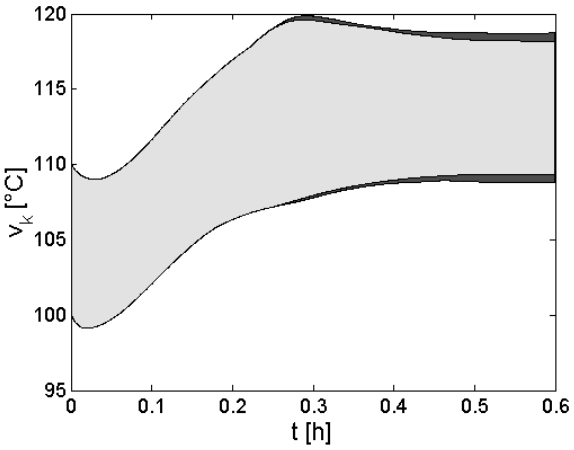
(a) Cyclopentadiene concentration c_a.



(b) Cyclopentenol concentration c_b.



(c) Reactor temperature v.



(d) Jacket temperature v_k.

Figure 4.21: Comparison of selection strategies.

than one of the model S_i can be active simultaneously. Note, that for the real system still only one state is active at each point of time. The fact that more than one model can be active at the same point of time in the simulation model reflects the dependency of the switching between the different models upon the parameters of the friction characteristic. This is taken into account by the fact that in the case of interval uncertainty more than one transition condition starting from the same model can be active simultaneously. The true system behavior is always included in the verified simulation results, which is guaranteed by a conservative choice of the parameters of the friction characteristic.

4.3.1 Basic Algorithm

The algorithm for the simulation of systems with model switching includes the following major points in each time step:

- a simulative detection of all possible points of time when a transition between two models may occur,
- the simulation of the dynamical system model, which includes all active partial models and,
- a check, which models can be deactivated in the next time-step.

Too obtain guaranteed enclosures of all reachable states in case of state dependent transitions between multiple continuous-time system models, the simulation algorithm described in Section 4.1 is extended, such that all possible points of time, were a transition condition T_i^j for the transition from model S_i to S_j is active, are included. The deactivation of models has to be done as soon as possible to obtain tight enclosures of the reachable states. The basic algorithm consists of the following four steps [61]:

- Step 1: **Calculation of a coarse enclosure** $[B_{a,k}]$ for all reachable states in the time interval $[t_k; t_{k+1}]$.
  For that purpose, the Picard Iteration is applied for the state equation $f_a(z(t), t)$, which is a union of all models that are active at $t = t_k$, i.e.

$$f_a([z_k], t_k) \supseteq \bigcup_{i \in \mathcal{I}_a} f_{S_i}([z_k], t_k) \ , \tag{4.91}$$

  where

$$\mathcal{I}_a = \left\{ i \;\middle|\; S_i = \texttt{true} \right\} \quad \text{for} \quad t = t_k \ . \tag{4.92}$$

- Step 2: **Test, if additional transition conditions for one of the active models S_i, $i \in \mathcal{I}_a$ becomes active.**
  The coarse enclosure $[B_{a,k}]$ from Step 1 has to be recalculated under consideration of all additional active models, if $\tilde{\mathcal{I}}_a \neq \mathcal{I}_a$, where

$$\tilde{\mathcal{I}}_a = \mathcal{I}_a \cup \{j \mid (T_i^j([B_{a,k}], [t_k; t_{k+1}]) = \texttt{true} \cap (i \in \mathcal{I}_a)\} \ . \tag{4.93}$$

  After the assignment of $\tilde{\mathcal{I}}_a$ by $\mathcal{I}_a$, the modified system equation $\tilde{f}_a(z(t), u(t), t)$ is now enclosed such that

$$\tilde{f}_a([B_{a,k}], [t_k; t_{k+1}]) \supseteq \bigcup_{i \in \tilde{f}\mathcal{I}_a} f_{S_i}([B_{a,k}], [t_k; t_{k+1}]) \ . \tag{4.94}$$

  If $\tilde{\mathcal{I}}_a = \mathcal{I}_a$, the substitutions $f_a = \tilde{f}_a$ and $\mathcal{I}_a = \tilde{\mathcal{I}}_a$ are carried out and the evaluation is continued in Step 3.

- Step 3: **Calculation of the interval enclosure** of the state $[z_{k+1}]$ at t_{k+1} by evaluation of equation (4.12).
  Hereby the state equation $f(z(t), t)$ has to be replaced by $f_a(z(t), t)$.
  The bounding box $[B_k]$ corresponds to $[B_{a,k}]$.

- Step 4: **Deactivation of models**, which can no longer be active at t_{k+1}.
  All states that correspond to the indices $\mathcal{I}_a$ have to be checked for their admissibility. Deleting of an index is possible, if there exists a contradiction between the definition of a model and the interval enclosure $[z_{k+1}]$.

4.3.2 Optimized Algorithm

In this thesis, the basic algorithm described above is extended to the simulation with interval splitting similar to Section 4.1.6. Therefore, the list of interval vectors $\mathcal{Z}_k$ is divided into l different Lists (see Fig. 4.22):

$\Rightarrow$ The first list L_1 contains all interval vectors with one active model,
$\Rightarrow$ the second list L_2 contains all interval vectors with two active models
$\vdots$
$\Rightarrow$ the l-th list L_l contains all interval vectors with l active models.

The interval vectors in each list are sorted according to their pseudo-volume, where the first interval vector in each list is the interval vector with the largest pseudo-volume. All l lists are assembled in one list, where the interval vectors of the list L_l are at the beginning of the entire list, then the interval vectors of the list L_{l-1} and at the end the interval vectors of list L_1. The following steps are identical to the algorithm described in Section 4.1.6.

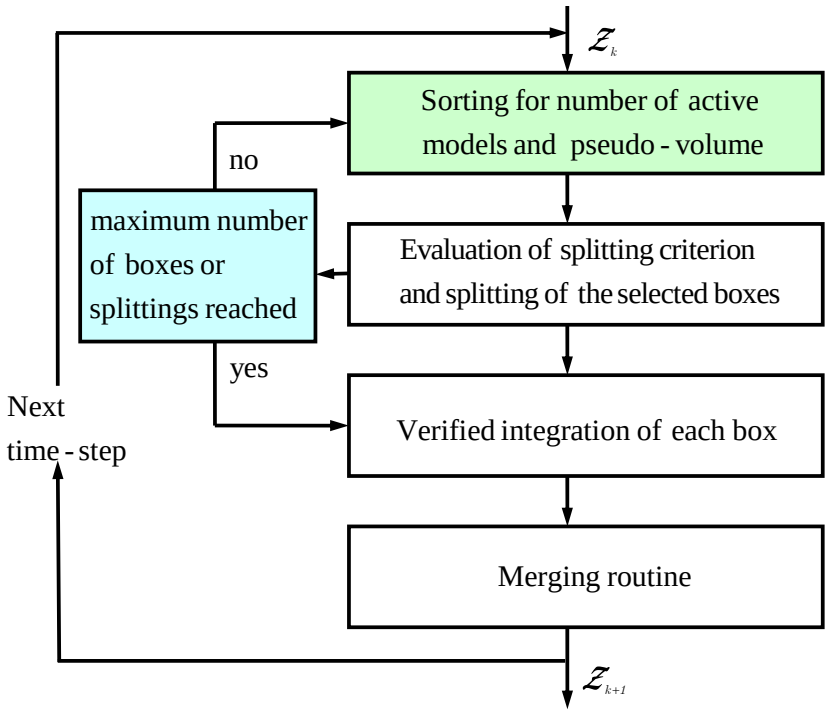


Figure 4.22: Flow diagram of the simulation algorithm.

4.3.3 Application: Mechanical Positioning System

In this section, simulation results for the mechanical positioning system described in Section 2.2 are presented. The friction characteristic $F_f(x_2)$ is described by three different models $\mathcal{S} = \{S_1, S_2, S_3\}$:

- model S_1: sliding friction for motion in negative (backward) direction ($x_2 < 0$),
- model S_2: static friction ($x_2 = 0$),
- model S_3: sliding friction for motion in positive (forward) direction ($x_2 > 0$) .

The model transition diagram in Fig. 4.23 shows all possible transitions for *nominal* parameter values, i.e. the static friction and the sliding friction coefficients are point intervals. In this case, only one model can be active at the same time.

Friction characteristic with parameter uncertainties

The influence of an uncertain static friction $F_s \in [\underline{F_s}; \overline{F_s}]$ and an uncertain sliding friction coefficient $\mu \in [\underline{\mu}; \overline{\mu}]$ on the friction characteristic is illustrated in Fig. 4.24. The friction

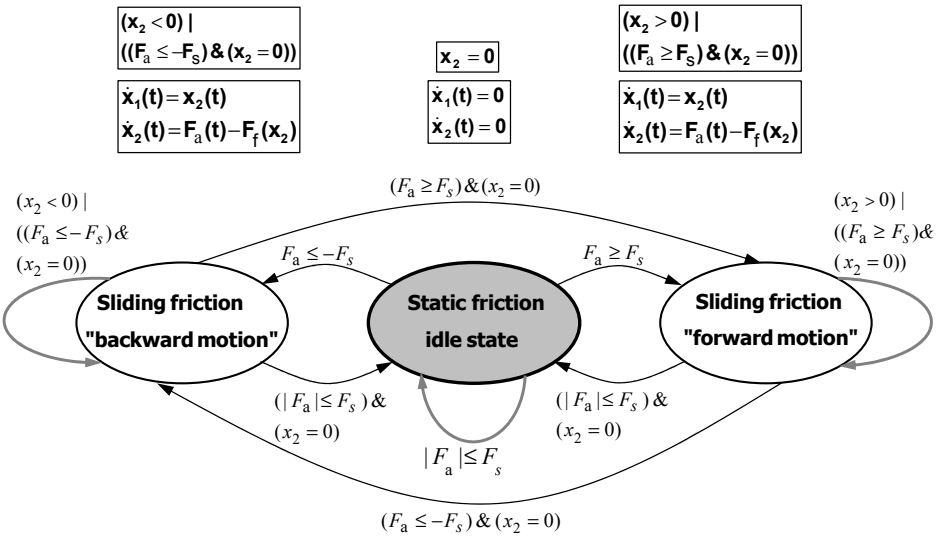


Figure 4.23: Switching between different models for *nominal* parameters.

characteristic is assumed to be symmetric. The dashed lines are the characteristics obtained for $\overline{\mu}$; the solid lines are the characteristics for $\underline{\mu}$. If these parameters are uncertain also the position x_1 and the velocity x_2 become intervals. Therefore it is possible that more than one model is active at the same point of time. The transition diagram for *uncertain* parameters is depicted in Fig. 4.25. If the system is in motion, the resulting sliding friction force is

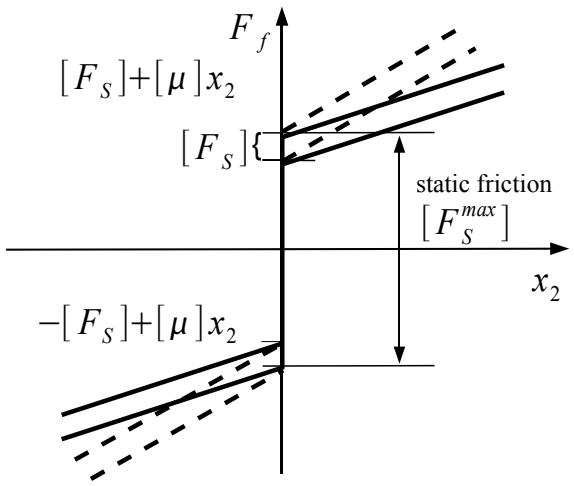


Figure 4.24: Friction characteristic with uncertain parameters.

given by

$$F_f(x_2) = \begin{cases} -[F_s] + [\mu] \cdot x_2 & \text{for } S_1 = active \ , \\ +[F_s] + [\mu] \cdot x_2 & \text{for } S_3 = active \ . \end{cases} \tag{4.95}$$

The static friction force in the idle state is given by

$$F_f(x_2) \in [F_s^{max}] = [-F_s; F_s], \quad S_2 = active \ . \tag{4.96}$$

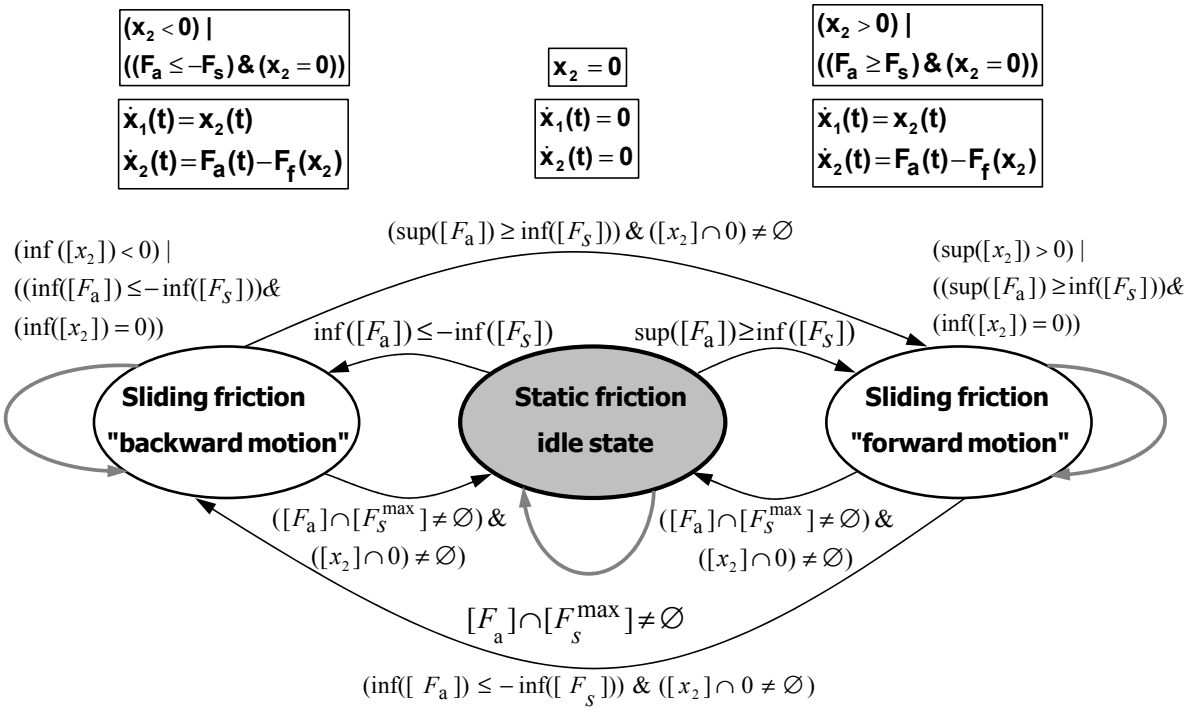


Figure 4.25: Switching between different models for *uncertain* parameters.

During the simulation, the active models are stored in a mode vector *mode* according to

- *mode*=[1 0 0] only the model for sliding friction for motion in negative (backward) direction ($x_2 < 0$) is active,
- *mode*=[0 1 0] only the model for static friction is active,
- *mode*=[0 0 1] only the model for sliding friction for motion in positive (forward) direction ($x_2 < 0$) is active,
- *mode*=[1 1 0] the model for sliding friction for motion in negative (backward) direction ($x_2 < 0$) and for static friction are active,
- *mode*=[0 1 1] the model for sliding friction for motion in positive (forward) direction ($x_2 < 0$) and for static friction are active,
- *mode*=[1 1 1] all models are active,
- *mode*=[1 0 1] cannot occur.

The transition from one model S_i to another model S_j takes only place if the corresponding transition condition is fulfilled. Model transitions are described in so-called transition matrices. If for example the system is in the idle state – hence only the model for static friction is active– and the driving force is negative ($F_{\mathbf{a}} < 0$), then the following transition conditions hold as long $F_{\mathbf{a}} \in -[F_s]$:

$$T_2^1 = \inf([F_{\mathbf{a}}]) \leq -\inf([F_s]) \quad \text{and} \quad T_2^2 = [F_{\mathbf{a}}] \cap [F_s^{max}] \neq \emptyset \quad . \tag{4.97}$$

Analogously for a positive driving force($F_{\mathbf{a}} > 0$) as long $F_{\mathbf{a}} \in [F_s]$:

$$T_2^3 = \sup[F_{\mathbf{a}}]) \geq \inf([F_s]) \quad \text{and} \quad T_2^2 = [F_{\mathbf{a}}] \cap [F_s^{max}] \neq \emptyset \quad . \tag{4.98}$$

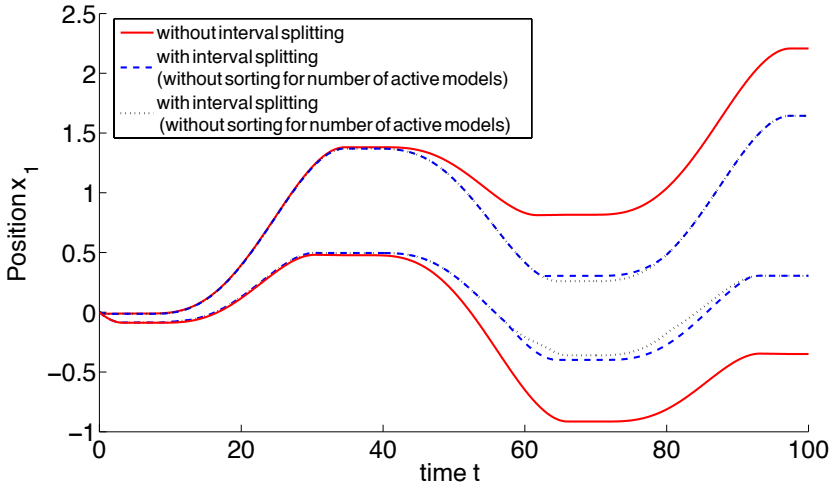
As mentioned above deleting of an index is possible, if there exists a contradiction between the definition of a state and the interval enclosure $[z_{k+1}]$. Depending upon the considered application, it is possible to check further conditions allowing guaranteed statements whether one or more of the discrete states can be deactivated.

In the considered scenario, the following additional constraints are applicable:

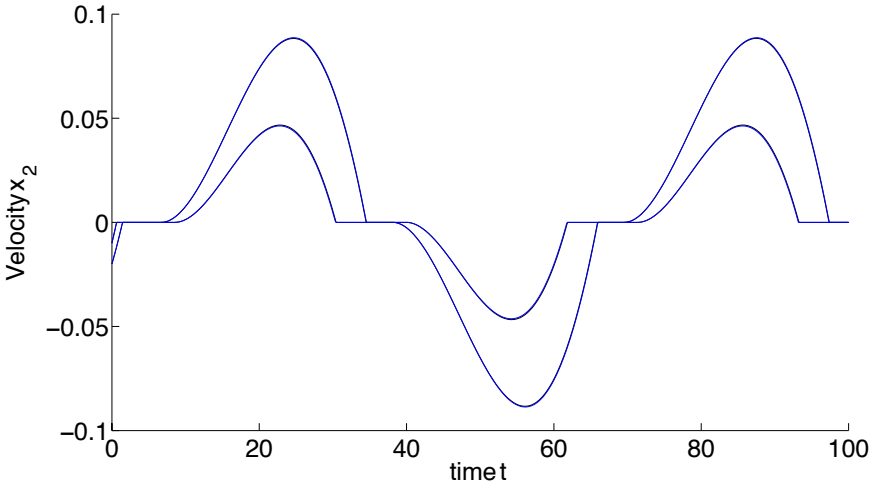
$$\text{If } (S_1\,(t_k) = \texttt{true}) \cap (\sup\,(F_a\,([t_k\,;t_{k+1}])) \geq 0)\ (\text{or } (S_3\,(t_k) = \texttt{true}) \cap (\inf\,(F_a\,([t_k\,;t_{k+1}])) \leq 0)),$$

S_1 (S_3) can be deactivated if simulation by the Steps 1 to 3 of the largest possible subinterval of $[z_k]$ which is compatible with S_1 (S_3) leads to a result which is no longer compatible with S_1 (S_3). A similar statement can additionally be provided for state S_2. After wards the simulation is continued with Step 1 for the next time interval $[t_{k+1}\ ;\ t_{k+2}]$.

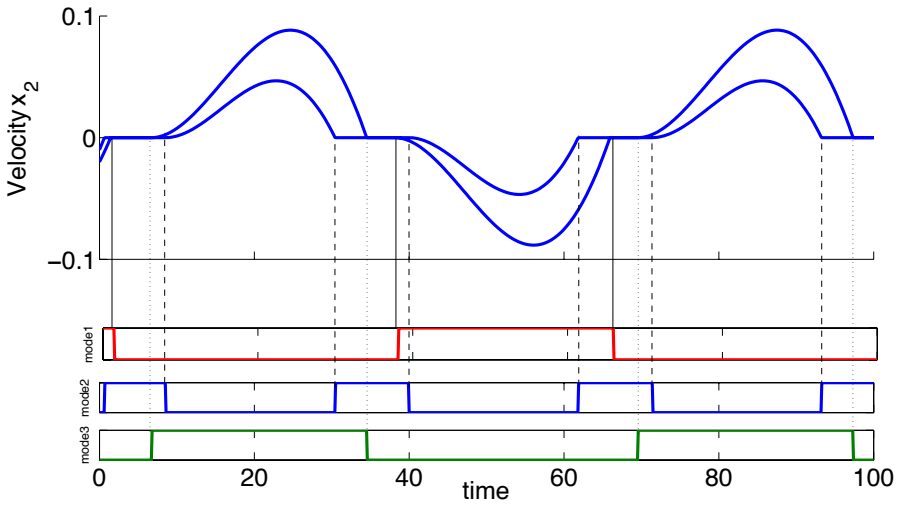
The following simulation results are based on normalized parameter values. The mass was given by $m = 1$. F_s and μ were uncertain with $F_s \in [0.0125; 0.015]$, $\mu \in [0.001; 0.0015]$. Both uncertain parameters were assumed to be time-invariant. The initial conditions were given by $x_1(0) \in [0; 0]$ and $x_2(0) \in [-0.05; -0.02]$. The step-size was constant with $h = 0.1$. The simulation was carried out for $t \in [0; 100]$ with $F_a(t) = 0.020 \cdot \sin(0.1t)$. Fig. 4.26 (a) and Fig. 4.26 (b) show the the resulting interval enclosures for x_1 and x_2, respectively. The solid lines depict the resulting enclosures without interval splitting and merging. The dashed lines represent the results of a simulation with the application of interval splitting and merging, however, without the list sorting algorithm described in Section 4.3.2. And the dotted lines depict results with additional application of the list sorting according to Section 4.3.2. Without interval splitting the results for x_1 the are quite pessimistic. Interval splitting leads to a drastic improvement. The enclosure of x_1 is tightened further when the interval vectors are sorted according to the number of activated models. For the velocity x_2 no difference between the three results can be seen. In contrast to the simulation with nominal system parameters F_s and μ, more than one model $\mathcal{S} = \{S_1, S_2, S_3\}$ can be active at the same time if the system parameters are described by interval uncertainties. This is depicted in Fig. 4.26 (c). Here only the results for the strategy with sorting for the number of active models is shown.



(a) Interval enclosure of the position.



(b) Interval enclosure of the velocity.



(c) Interval enclosure of the velocity.

Figure 4.26: Interval enclosures of position and velocity.

4.4 Verified Techniques Based on Taylor Models

The verified integration methods described so far employ only a Taylor expansion in time. Validated techniques based on Taylor models as developed by Berz and Makino [5, 45–47] and implemented in COSY-VI perform also an expansion in the initial state vector, which is in the following denoted by $\mathfrak{z}$. The domain interval vector for $\mathfrak{z}$ is given by $[\mathfrak{z}]$. The expansion point for the expansion in the initial state vector $\mathfrak{z}$ is given by $\hat{\mathfrak{z}}$ with $\hat{\mathfrak{z}} \in [\mathfrak{z}]$. The expansion point for the expansion in time is t_k. The flow of the differential equation in a given time interval $[t_k; t_{k+1}]$ is enclosed by a n-dimensional Taylor model

$$\begin{aligned} T_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) &:= P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}, \\ \text{with} \quad \mathfrak{z} \in [\mathfrak{z}] \quad &\text{and} \quad t \in [t_k; t_k + 1] \ , \end{aligned} \tag{4.99}$$

where $P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ is the multivariate polynomial part of order ρ and $I_{\rho,k+1}$ the remainder interval vector. Components i of $T_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ are denoted by $T_{\rho,i}(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$. The Taylor model at $t = t_{k+1}$ is

$$T_{\rho,k+1}(\mathfrak{z} - \hat{\mathfrak{z}}) := P_{\rho,k+1}(\mathfrak{z} - \hat{\mathfrak{z}}) + I_{\rho,k+1} \ . \tag{4.100}$$

Components i of $T_{\rho,k+1}(\mathfrak{z} - \hat{\mathfrak{z}})$ are given by $T_{\rho,i,k+1}(\mathfrak{z} - \hat{\mathfrak{z}})$.

The flow representation by Taylor models makes it possible to obtain tight enclosures of non-convex sets and leads to a reduction of both the dependency problem and the wrapping effect.

4.4.1 Basic Principle

In the following the key elements of verified integration based on Taylor model methods are explained. First the differential equation

$$\dot{z}(t) = f(z(t), t) \tag{4.101}$$

is rewritten into integral form:

$$z(t) = z(t_0) + \int_{t_0}^{t} f(z(t'), t')dt', \tag{4.102}$$

where $z(t_0)$ denotes the solution of the ODE at $t = t_0$. The initial value problem has a (unique) solution if and only if the corresponding integral equation has a (unique) solution. Now the operator

$$\mathcal{O} : \vec{C}^0[t_0, t_1] \to \vec{C}^0[t_0, t_1] \tag{4.103}$$

on the space of continuous functions from $[t_0, t_1]$ to $\mathbb{R}^n$ via

$$\mathcal{O}\,(z)\,(t) := z(t_0) + \int_{t_0}^{t} f(z(t'), t')dt' \tag{4.104}$$

is introduced. The problem of finding a solution to the differential equation is reduced to a fixed-point problem

$$z = \mathcal{O}(z) \ . \tag{4.105}$$

For the verified in integration of the flow of the ODE in a given time-interval $[t_k, t_{k+1}]$ the goal is now to find a Taylor model $T_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) = P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}$ such that

$$\begin{aligned} &\mathcal{O}(P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}) \subset P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1} \\ &\forall \mathfrak{z} \in [\underline{\mathfrak{z}}; \overline{\mathfrak{z}}] \quad \text{and} \quad \forall t \in [t_k; t_{k+1}]. \end{aligned} \tag{4.106}$$

holds. This Taylor model provides a verified enclosure of the solution of the flow of the ODE in the time interval $[t_k; t_k + 1]$.

In [5, 45] Schauder's fixpoint theorem is applied to show that a Taylor model with the property (4.106) proves the existence of a solution of the fixpoint equation – but not the uniqueness of the solution. In [12] Banach's fix-point theorem is established to show that a Taylor model according to (4.106) prove the existence and uniqueness of the solution of the fix-point equation.

Applying the Operator $\mathcal{O}$ to a Taylor model for the integration in the time-interval $[t_k; t_{k+1}]$ yields

$$\mathcal{O}(P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}) = z(t_k) + \int_{t_k}^{t} f(P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t' - t_k) + I_{\rho,k+1}) dt' \ , \tag{4.107}$$

where $z(t_k)$ is represented by its corresponding Taylor model enclosure at $t = t_k$

$$T_{\rho,k} = P_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}}) + I_{\rho,k} \ . \tag{4.108}$$

This leads to

$$\mathcal{O}(P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}) = P_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}}) + I_{\rho,k} + \int_{t_k}^{t} f(P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1}) dt' \ . \tag{4.109}$$

The task is to find a suitable choice for $P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ and $I_{\rho,k+1}$ which fulfill (4.106), and furthermore it is desirable to have $I_{\rho,k+1}$ as tight as possible.

The basic algorithm consists of two stages:

Stage 1: Computing a polynomial part

For a sufficient order ρ the polynomial part $P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ is already close to the solution. The polynomial $P_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ is obtained by iteratively apply the fix point equation according to

$$P^{(j+1)}(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) = \mathcal{O}(P^{(j)}(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)) \ . \tag{4.110}$$

This iteration is also called Picard iteration. $P^{(0)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)$ is given by the polynomial $P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})$ part of the Taylor model $T_{\rho,k}$ of $z(t_k)$, hence

$$P^{(0)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)=P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}}) \tag{4.111}$$

and

$$\begin{aligned}P^{(j+1)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)&=\mathcal{O}(P^{(j)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k))=P^{(0)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\int_{t_k}^{t}f(P^{(j)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k))dt'\\&=P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+\int_{t_k}^{t}f(P^{(j)}(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k))dt'.\end{aligned} \tag{4.112}$$

Note, that this computation is a pure symbolic integration with respect to the time variable t. No accumulation of remainder errors takes place in this stage. After each iteration $j+1$, terms higher than order $j+1$ are truncated. After ρ steps the ρ-th order expansion

$$P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)=P^{(\rho)}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k) \tag{4.113}$$

is obtained and terms higher than ρ are omitted.

Stage 2: Determination of the interval remainder

Now an interval remainder bound $I_{\rho,k+1}$ has to be found such that the inclusion requirement

$$\begin{aligned}&\mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1})\subset P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}\\&\forall\ \mathfrak{z}\in[\underline{\mathfrak{z}};\overline{\mathfrak{z}}]\quad\text{and}\quad\forall\quad t\in[t_k;t_k+1]\end{aligned} \tag{4.114}$$

is fulfilled. Here the integration rule for Taylor models from equation (3.88) has to be applied, leading to

$$\begin{aligned}&\mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1})=P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+I_{\rho,k}+\int_{t_k}^{t}\underbrace{f(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+I_{\rho,k+1})}_{\hat{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+\hat{I}_{\rho,k+1}}dt'\\&=P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+I_{\rho,k}+\int_{t_k}^{t}\hat{P}_{\rho-1}(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)dt'\\&+\{B(\hat{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)-\hat{P}_{\rho-1}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k))+\hat{I}_{\rho,k+1})\}\cdot\underbrace{([t_k;t_{k+1}]-t_k)}_{=[0;h_k]}\\&=P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I^*_{\rho,k+1}\\&\forall\ \mathfrak{z}\in[\underline{\mathfrak{z}};\overline{\mathfrak{z}}]\quad\text{and}\quad\forall\quad t\in[t_k;t_k+1]\ .\end{aligned} \tag{4.115}$$

Note that the polynomial part is reproduced in the integration [5, 12, 45], only the interval remainder changes. Thus, the inclusion property (4.106) is fulfilled for

$$I^*_{\rho,k+1}\subset I_{\rho,k+1}\ . \tag{4.116}$$

The suitable choice for $I_{\rho,k+1}$ requires some trial and error. First an initial estimate $I_{\rho,k+1}^{(0)}$ has to be determined. This can be done by

$$\begin{aligned} P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(0)} &= \mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+[0;0]^n) \\ &= P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+I_{\rho,k}+\int_{t_k}^{t} f(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+[0;0]^n)dt' \ . \end{aligned} \tag{4.117}$$

However, in most applications $I_{\rho,k+1}^{(0)}$ is obtained iteratively by

$$\begin{aligned} P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(0,j+1)}) &= \mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(0,j)}) \\ &= P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+I_{\rho,k}+\int_{t_k}^{t} f(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+I_{\rho,k+1}^{(0,j)})dt' \ , \end{aligned} \tag{4.118}$$

with

$$I_{\rho,k+1}^{(0,0)} = [0;0]^n \ . \tag{4.119}$$

If the iteration is stopped after j^* iterations, then

$$I_{\rho,k+1}^{(0)} = I_{\rho,k+1}^{(0,j^*+1)} \tag{4.120}$$

holds. In most applications the iteration is repeated two times.

The interval remainder is now iteratively increased by $I_{\rho,k+1}^{(j)} = q^j \cdot I_{\rho,k+1}^{(0)}$ with $q > 1$, i.e.

$$\begin{aligned} P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{*(j)}) &= \mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(j)}) \\ &= P_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}})+I_{\rho,k}+\int_{t_k}^{t} f(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+I_{\rho,k+1}^{(j)})dt' \ . \end{aligned} \tag{4.121}$$

until a computational inclusion

$$\begin{aligned} &P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{*(j)} = \mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(j)}) \subset P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+I_{\rho,k+1}^{(j)} \\ &\forall\ \mathfrak{z}\in[\underline{\mathfrak{z}};\overline{\mathfrak{z}}] \quad \text{and} \ \forall\ t\in[t_k;t_k+1] \end{aligned} \tag{4.122}$$

is found.

In practice, a computational inclusion can be found in a few iterations with q between 1 and 2. If this is not the case, it can be forced by slightly reducing the integration step size, which reduces the main contribution of the remainder bound of the mapped set.

Let $I_{\rho,k+1}^{*(j^*)}$ be the desired interval remainder fulfilling (4.106), which has bee obtained after j^* iterations. The width of the remainder interval can be decreased now by iteratively applying the fixed point operator. Denoting $\tilde{I}_{\rho,k+1}^{(0)} = I_{\rho,k+1}^{*(j^*)}$, the sequence of Taylor models

$$\begin{aligned} P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\tilde{I}_{\rho,k+1}^{(j+1)} &= \mathcal{O}(P_\rho(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\tilde{I}_{\rho,k+1}^{(j)}) \\ &\text{with} \quad \tilde{I}_{\rho,k+1}^{(j+1)} \subset \tilde{I}_{\rho,k}^{(j)} \end{aligned} \tag{4.123}$$

is applied iteratively until no further significant decrease in size of the interval remainder is achieved.

As a result after j^* refinement iterations,

$$T_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) = P_{\rho,k+1}(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + I_{\rho,k+1} \text{ with } I_{\rho,k+1} = \tilde{I}^{(j^*)}_{\rho,k+1} \tag{4.124}$$

is the desired sharp inclusion of the flow of the ODE in the considered time-interval. The Taylor model obtained at $t = t_{k+1}$ of the current integration step is given by

$$T_{\rho,k+1} = P_{\rho,k+1}(\mathfrak{z} - \hat{\mathfrak{z}}) + I_{\rho,k+1}. \tag{4.125}$$

To integrate over time, the procedure is applied at each time step. Automatic step-size controllers can be used to assure that if the solution at a time step is not favorable, the step-size is decreased [5,45–47]. On the other hand, if a time step can proceed without much growth of errors, the next step size is increased.

For numerical and implementation reasons it is advantageous to have the unit box $[-1;1]^n$ as domain interval vector and 0 in each component of the expansion point for the initial state vector in each integration step [46,47]. If at $t = 0$ the initial vector is given by an interval box with $z(0) \in [z_0]$, this interval box is expressed as a Taylor model

$$z(0) \in T_{rho,0} = c + D\mathfrak{z} \text{ with } \mathfrak{z}_i \in [-1;1] \ , i = 1 \ldots, n \ , \tag{4.126}$$

where c is the midpoint of $[z_0]$ and D is a diagonal matrix with $d_{i,i} = \text{rad}([z_0])$.

Because the solution set is described as a Taylor model describing the dependence on initial conditions, the dependency problem based on the repeated use of the solution set in subsequent operations is reduced. And because the sets are represented by Taylor models also non-convex sets can be enclosed much more efficiently than by a single interval vector or by a parallelepiped. Thus, also the wrapping effect is reduced.
However, the interval remainder part the integration is still affected by overestimation. The polynomial parts of the Taylor models are independent of the initial domain intervals and independent of the step size, but the interval remainder bounds are not.

Transversal Weighting

COSY supports sparsity, i.e. coefficients which are below a pre-specified accuracy threshold are included in the interval remainder. Thus, only coefficients which are larger than this threshold contribute to computational effort. In cases where the interval enclosures of some components $z_i(0)$ of the initial state vector $z(0)$ are small, the expansion order in the corresponding normalized initial state variables $\mathfrak{z}_i$ (with $\mathfrak{z}_i \in [-1;1]$, $i = 1, 2, \ldots, n$) , can be kept lower than for the components with larger interval enclosures. This is done by a weighting of these coefficients by some suitable odd integer power w [48]. In this way, throughout the computation, only powers of $\mathfrak{z}_i$ that are multiples of w appear, which limits the expansion in initial state variables to the largest $\tilde{\rho}$ that satisfies $\rho \leq w \cdot \tilde{\rho}$. Combined with sparsity methods, this can drastically reduce computational expense and storage requirements. Without

sparsity support, the number of possible coefficients N_c of order ρ in n initial conditions with weighting factor w for one component of a n-dimensional Taylor model enclosing z is given by [48]

$$N_c(\rho, n, w) = \sum_{j=0}^{\mathrm{int}(\rho/w)} \frac{(j+n-1)!}{j! \cdot (n-1)!}(\rho - w \cdot j + 1) \ . \tag{4.127}$$

Here $\mathrm{int}(\rho/w)$ denotes the smallest integer not exceeding ρ/w. The number N_c also considers the expansion in time in addition to the expansion in the n initial state variables. To obtain the number of coefficients for the complete n-dimensional Taylor model vector,N_c has to be multiplied by n. If $w = 1$, then this number is given by

$$N_{c,tot}(\rho, n, w = 1) = n \cdot \frac{((n+1)+\rho)!}{\rho! \cdot (n+1)!} \ . \tag{4.128}$$

Example for Solving Linear Ordinary Differential Equations by Taylor Model Methods

In order to illustrate verified integration of Taylor model methods and in order to obtain a better understanding of the previously described algorithm, a simple linear example [44] is considered below. A nonlinear example is given in A.1. The differential equation is given by:

$$\begin{aligned} \dot{z}_1 &= -z_2 \ , \\ \dot{z}_2 &= z_1 \ . \end{aligned} \tag{4.129}$$

The following initial conditions are considered,

$$\begin{aligned} z_1(0) &\in [1; 3] \ , \\ z_2(0) &\in [-1; 1] \ , \end{aligned} \tag{4.130}$$

which leads to the initial Taylor model

$$\begin{aligned} T_{\rho,1,0} &= 2 + \mathfrak{z}_1 \ , \\ T_{\rho,2,0} &= 0 + \mathfrak{z}_2 \quad \text{with} \quad \mathfrak{z}_1, \mathfrak{z}_2 \in [-1; 1] \ . \end{aligned} \tag{4.131}$$

The following calculation is intended to show the procedures of the algorithms.

The First Time Step

The fixed point equations are

$$\begin{aligned} z_1(t) &= z_1(t_0) + \int_{t_0}^{t} (-z_2(t'))dt' = \mathcal{O}_1(z(t)) \ , \\ z_2(t) &= z_2(t_0) + \int_{t_0}^{t} (z_1(t'))dt' = \mathcal{O}_2(z(t)) \ . \end{aligned} \tag{4.132}$$

with $t_0 = 0$. First the polynomial part is determined, and then Taylor models which satisfy the inclusion requirement have to be found. Finally the Taylor models are refined. The step size is constant with $h = \frac{\pi}{6}$ and the order is set to $\rho = 5$ in time and initial state varibales.

Calculation of the Polynomial Part

The initial polynomial $P^{(0)}$ is given by $T_{\rho,1,0}$ and $T_{\rho,2,0}$.

Fixed point iteration: Step 1

$$\begin{aligned} P_1^{(1)}(\mathfrak{z},t) &= 2+\mathfrak{z}_1+\int_0^t[-\mathfrak{z}_2]dt' = 2+\mathfrak{z}_1-\mathfrak{z}_2 t \ , \\ P_2^{(1)}(\mathfrak{z},t) &= \mathfrak{z}_2+\int_0^t[2+\mathfrak{z}_1]dt' = \mathfrak{z}_2+(2+\mathfrak{z}_1)t \ . \end{aligned} \tag{4.133}$$

Fixed point iteration: Step 2

$$\begin{aligned} P_1^{(2)}(\mathfrak{z},t) &= 2+\mathfrak{z}_1+\int_0^t[-\mathfrak{z}_2-(2+\mathfrak{z}_1)t']dt' = 2+\mathfrak{z}_1-\mathfrak{z}_2 t-(2+\mathfrak{z}_1)\frac{t^2}{2} \ , \\ P_2^{(2)}(\mathfrak{z},t) &= \mathfrak{z}_2+\int_0^t[2+\mathfrak{z}_1-\mathfrak{z}_2 t']dt' = \mathfrak{z}_2+(2+\mathfrak{z}_1)t-\mathfrak{z}_2\frac{t^2}{2} \ . \end{aligned} \tag{4.134}$$

This is repeated until Step 5.

$$\begin{aligned} P_1^{(5)}(\mathfrak{z},t) &= 2+\mathfrak{z}_1-\mathfrak{z}_2 t-(2+\mathfrak{z}_1)\frac{t^2}{2}+\mathfrak{z}_2\frac{t^3}{3!}+(2+\mathfrak{z}_1)\frac{t^4}{4!}-\mathfrak{z}_2\frac{t^5}{5!} \ , \\ P_2^{(5)}(\mathfrak{z},t) &= \mathfrak{z}_2+(2+\mathfrak{z}_1)t-\mathfrak{z}_2\frac{t^2}{2}-(2+\mathfrak{z}_1)\frac{t^3}{3!}+\mathfrak{z}_2\frac{t^4}{4!}+(2+\mathfrak{z}_1)\frac{t^5}{5!} \ . \end{aligned} \tag{4.135}$$

So for the order $\rho = 5$ depending on time t and the initial state vector $\mathfrak{z}$, the polynomial is given by

$$\begin{aligned} P_{\rho,1}(\mathfrak{z},t) &= 2+\mathfrak{z}_1-\mathfrak{z}_2 t-(2+\mathfrak{z}_1)\frac{t^2}{2}+\mathfrak{z}_2\frac{t^3}{3!}+(2+\mathfrak{z}_1)\frac{t^4}{4!} \ , \\ P_{\rho,2}(\mathfrak{z},t) &= \mathfrak{z}_2+(2+\mathfrak{z}_1)t-\mathfrak{z}_2\frac{t^2}{2}-(2+\mathfrak{z}_1)\frac{t^3}{3!}+\mathfrak{z}_2\frac{t^4}{4!}+2\cdot\frac{t^5}{5!} \ . \end{aligned} \tag{4.136}$$

Calculation of the Interval Remainder

Using the polynomial solution part (4.136), the initial estimate $I_{\rho,1}^{(0)}$ is determined by

$$\begin{aligned} P_{\rho,1}(\mathfrak{z},t)+I_{\rho,1,1}^{(0)} &= 2+\mathfrak{z}_1+\int_0^t[-z_2(t)]dt \\ &= P_{\rho,1}(\mathfrak{z},t)+\left\{B\left(-\mathfrak{z}_2\frac{t^4}{4!}-2\cdot\frac{t^5}{5!}\right)+[0;0]\right\}\cdot[0;h] \\ &= P_{\rho,1}(\mathfrak{z},t)+I_{\rho,1,1}^{(0)} \ , \\ P_{\rho,2}(\mathfrak{z},t)+I_{\rho,2,1}^{(0)} &= P_{\rho,2}(\mathfrak{z},t)+\left\{B\left(\mathfrak{z}_1\frac{t^4}{4!}\right)+[0;0]\right\}\cdot[0;h] \\ &= P_{\rho,2}(\mathfrak{z},t)+I_{\rho,2,1}^{(0)} \ . \end{aligned} \tag{4.137}$$

And so

$$\begin{aligned} I^{(0)}_{\rho,1,1} &= [-1.99 \times 10^{-3}; 1.64 \times 10^{-3}] \ , \\ I^{(0)}_{\rho,2,1} &= [-1.64 \times 10^{-3}; 1.64 \times 10^{-3}] \end{aligned} \tag{4.138}$$

holds. Note, that here only a single iteration has been carried out for determining the initial estimate of the interval remainder. As mentioned previously in many application this iteration is repeated.

The remainder bound is inflated repeatedly by 2 until it satisfies the self inclusion condition

$$\begin{aligned} I^{(1)}_{\rho,1,1} &= 2 \cdot I^{(0)}_{\rho,1,1} = [-3.97 \times 10^{-3}; 3.28 \times 10^{-3}] \ , \\ I^{(1)}_{\rho,2,1} &= 2 \cdot I^{(0)}_{\rho,2,1} = [-1.64 \times 10^{-3}; 1.64 \times 10^{-3}] \ . \end{aligned} \tag{4.139}$$

Applying the Picard operation iteratively, then yields

$$\begin{aligned} I^{*(1)}_{\rho,1,1} &= [-3.71 \times 10^{-3}; 3.36 \times 10^{-3}] \ , \\ I^{*(1)}_{\rho,2,1} &= [-3.72 \times 10^{-3}; 3.36 \times 10^{-3}] \ , \end{aligned} \tag{4.140}$$

$$\begin{aligned} I^{(2)}_{\rho,1,1} &= 2^2 \cdot I^{(0)}_{\rho,1,1} = [-7.94 \times 10^{-3}; 6.56 \times 10^{-3}] \ , \\ I^{(2)}_{\rho,2,1} &= 2^2 \cdot I^{(0)}_{\rho,2,1} = [-6.56 \times 10^{-3}; 6.36 \times 10^{-3}] \ , \end{aligned} \tag{4.141}$$

$$\begin{aligned} I^{*(2)}_{\rho,1,1} &= [-5.42 \times 10^{-3}; 5.08 \times 10^{-3}] \ , \\ I^{*(2)}_{\rho,2,1} &= [-5.80 \times 10^{-3}; 5.08 \times 10^{-3}] \ . \end{aligned} \tag{4.142}$$

Thus, a self including solution $P_\rho(\mathfrak{z}, t) + I^{*(2)}_\rho$ has been found.

Now, the Picard operation is applied repeatedly until the desired sharpness of enclosure is achieved. The refinement is initialzed by $\tilde{I}^{(0)}_{\rho,1} = I^{*(2)}_{\rho,1}$. The first refinement iteration yields

$$\begin{aligned} P_\rho(\mathfrak{z}, t) + \tilde{I}^{(1)}_{\rho,1} &= \mathcal{O}(P_\rho(\mathfrak{z}, t) + \tilde{I}^{(0)}_{\rho,1}) = P_\rho(\mathfrak{z}, t) + \begin{pmatrix} [-4.64 \times 10^{-3}; 4.68 \times 10^{-3}] \\ [-4.48 \times 10^{-3}; 4.30 \times 10^{-3}] \end{pmatrix} \ , \\ P_\rho(\mathfrak{z}, t) + \tilde{I}^{(2)}_{\rho,1} &= \mathcal{O}(P_\rho(\mathfrak{z}, t) + \tilde{I}^{(1)}_{\rho,1}) = P_\rho(\mathfrak{z}, t) + \begin{pmatrix} [-4.24 \times 10^{-3}; 3.99 \times 10^{-3}] \\ [-4.07 \times 10^{-3}; 4.09 \times 10^{-3}] \end{pmatrix} \ . \end{aligned} \tag{4.143}$$

Continuing until the relative tolerance of 1% is met, results in

$$P_\rho(\mathfrak{z}, t) + \tilde{I}^{(7)}_{\rho,1} = \mathcal{O}(P_\rho(\mathfrak{z}, t) + \tilde{I}^{(6)}_{\rho,1}) = P_\rho(\mathfrak{z}, t) + \begin{pmatrix} [-3.84 \times 10^{-3}; 3.57 \times 10^{-3}] \\ [-3.66 \times 10^{-3}; 3.52 \times 10^{-3}] \end{pmatrix} \ . \tag{4.144}$$

The desired interval remainder $I_{\rho,1}$ is then given by $I_{\rho,1} = \tilde{I}^{(7)}_{\rho,1}$.

Taylor Model Solution at $t = \dfrac{\pi}{6}$

$$\begin{aligned} z_1(t = \frac{\pi}{6}) \in T_{\rho,1,1}(\mathfrak{z}) &= P_{\rho,1}(\mathfrak{z}, t = \frac{\pi}{6}) + I_{\rho,1,1} = P_{\rho,1,1}(\mathfrak{z}) + I_{\rho,1,1} \\ &= 1.732 + 0.866\mathfrak{z}_1 - 0.500\mathfrak{z}_2 + [-3.84 \times 10^{-3}; 3.57 \times 10^{-3}] \ , \\ z_2(t = \frac{\pi}{6}) \in T_{\rho,2,1}(\mathfrak{z}) &= P_{\rho,2}(\mathfrak{z}, t = \frac{\pi}{6}) + I_{\rho,2,1} = P_{\rho,2,1}(\mathfrak{z}) + I_{\rho,2,1} \\ &= 1.000 + 0.500\mathfrak{z}_1 + 0.866\mathfrak{z}_2 + [-3.66 \times 10^{-3}; 3.52 \times 10^{-3}] \ . \end{aligned} \tag{4.145}$$

In Fig. 4.27 and Tab. 4.1 the result is illustrated. The region between outer rotated square

Table 4.1: Initial position and mapped position

Initial position $(\mathfrak{z}_1, \mathfrak{z}_2)$ at $t = 0$	Mapped position $(P_{\rho,1,1}, P_{\rho,2,1})$ at $t = \frac{\pi}{6}$
$(0, 0)$	$(1.732, 1.000)$
$(1, 1)$	$(2.098, 2.366)$
$(-1, 1)$	$(0.366, 1.366)$
$(-1, -1)$	$(1.366, -0.366)$
$(1, -1)$	$(3.098, 0.634)$

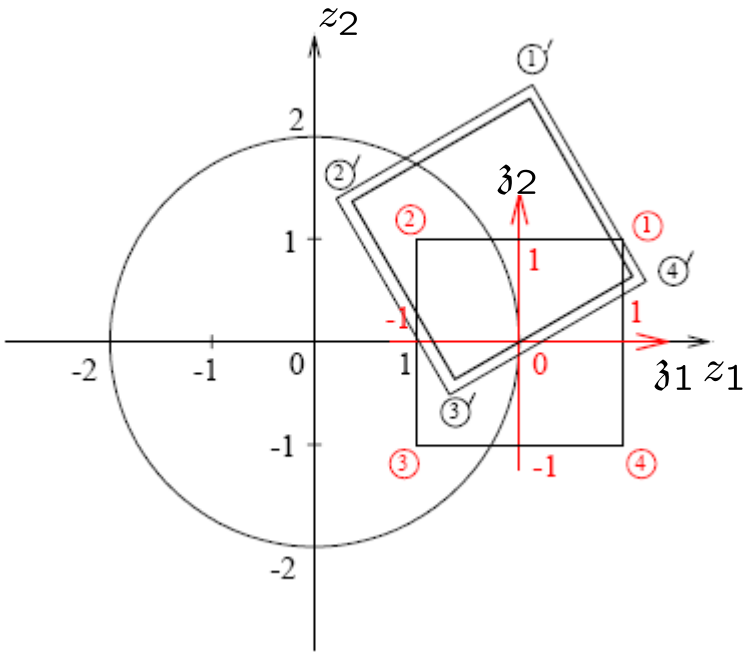


Figure 4.27: Initial position and mapped position.

and inner rotated square in Fig. 4.27 denotes the remainder bounds.

Taylor Model Solution at the Second Time Step $(t = 2 \cdot \frac{\pi}{6} = \frac{\pi}{3})$

$$\begin{aligned} z_1(t = \frac{\pi}{3}) \in T_{\rho,1,2}(\mathfrak{z}) &= P_{\rho,1}(\mathfrak{z}, t = \frac{\pi}{3}) + I_{\rho,1,2} = P_{\rho,1,2}(\mathfrak{z}) + I_{\rho,1,2} \\ &= 1.000 + 0.500\mathfrak{z}_1 - 0.866\mathfrak{z}_2 + [-3.84 \times 10^{-3}; 3.57 \times 10^{-3}] \ , \\ z_2(t = \frac{\pi}{3}) \in T_{\rho,1,2}(\mathfrak{z}) &= P_{\rho,2}(\mathfrak{z}, t = \frac{\pi}{3}) + I_{\rho,2,2} = P_{\rho,2,2}(\mathfrak{z}) + I_{\rho,2,2} \\ &= 1.732 + 0.866\mathfrak{z}_1 + 0.500\mathfrak{z}_2 + [-3.66 \times 10^{-3}; 3.52 \times 10^{-3}] \ . \end{aligned} \tag{4.146}$$

The initial Taylor model for the integration from $\frac{\pi}{6}$ to $\frac{\pi}{3}$ is the resulting Taylor model from $t = \frac{\pi}{6}$.

Taylor Model Solution at the Third Time Step ($t = 3 \cdot \frac{\pi}{6} = \frac{\pi}{2}$)

$$\begin{aligned} z_1(t = \frac{\pi}{2}) \in T_{\rho,1,3}(\mathfrak{z}) &= P_{\rho,1,3}(\mathfrak{z}_1) + I_{\rho,1,3} \\ &= -1.000\mathfrak{z}_2 + [-1.29 \times 10^{-2}; 1.26 \times 10^{-2}] \ , \\ z_2(t = \frac{\pi}{2}) \in T_{\rho,2,3}(\mathfrak{z}) &= P_{\rho,2,3}(\mathfrak{z}) + I_{\rho,2,3} \\ &= 2.000 + 1.000\mathfrak{z}_1 + [-1.24 \times 10^{-2}; 1.28 \times 10^{-2}] \ . \end{aligned} \tag{4.147}$$

4.4.2 Reduction of the Wrapping Effect in Taylor Model based Verified Integrators

In the k-th integration step, the solution of the ODE is enclosed by the region occupied by the Taylor model

$$T_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}}) = P_{\rho,k}(\mathfrak{z} - \hat{\mathfrak{z}}) + I_{\rho,k} \ . \tag{4.148}$$

This Taylor model presents the initial state enclosure for the next integration step as discussed in the previous section. Assuming $\mathfrak{z}_i \in [-1; 1], i = 1, 2..n$ and $\hat{\mathfrak{z}}_i = 0, i = 1, 2..n$ the Taylor model results in

$$T_{\rho,k}(\mathfrak{z}) = P_{\rho,k}(\mathfrak{z}) + I_{\rho,k} \ . \tag{4.149}$$

As already mentioned, the expansion in initial conditions reduces the dependency problem and the wrapping effect. The interval remainder part however is affected by the dependency problem and the wrapping effect like the integration methods in Section 4.1. In order to limit the long-term growth of the truncation error and to further reduce overestimation the following strategies can be applied [37, 46, 47].

Shrink Wrapping

Shrink wrapping [47] is a method to control the long term growth of integration errors. The idea is to include remainder error in the range of the polynomial part of Taylor model.

This problem can be reduced to only linear algebra while in the linear case, but the situation is more complicated in the nonlinear case [47].

After the application of shrink wrapping a Taylor model $T^{SW}_{\rho,k}$ very similar to the original Taylor model $T_{\rho,k}$, with

$$T_{\rho,k}(\mathfrak{z}) \subset T^{SW}_{\rho,k}(\mathfrak{z}) = P^{SW}_{\rho,k}(\mathfrak{z}) + I^{SW}_{\rho,k} \ , \tag{4.150}$$

is obtained. $P_{\rho,k}^{SW}$ is a slightly modified polynomial, and $I_{\rho,k}^{SW}$ is a significantly reduced interval of the size of machine precision.
The Taylor model is rewritten according to

$$T_{\rho,k}(\mathfrak{z}) = c_k + \tilde{P}_{\rho,k}(\mathfrak{z}) = c_k + C_k \cdot \mathfrak{z} + \mathcal{N}_{\rho,k}(\mathfrak{z}) + I_{\rho,k}, \tag{4.151}$$

where c_k is the constant part, $C_k \cdot \mathfrak{z}$ the linear part and $\mathcal{N}_{\rho,k}(\mathfrak{z})$ the nonlinear part of $P_{\rho,k}$.

Shrink wrapping consists of the following steps.

1. First the constant part is removed resulting into

$$\tilde{T}_{\rho,k}(\mathfrak{z}) = \tilde{P}_{\rho,k}(\mathfrak{z}) + I_{\rho,k} = C_k \cdot \mathfrak{z} + \mathcal{N}_{\rho,k}(\mathfrak{z}) + I_{\rho,k} \tag{4.152}$$

2. Next, the obtained Taylor model $\tilde{T}_{\rho,k}$ is multiplied by C_k^{-1}, resulting into

$$\hat{T}_{\rho,k}(\mathfrak{z}) = \mathfrak{z} + \hat{\mathcal{N}}_{\rho,k}(\mathfrak{z}) + \hat{I}_{\rho,k} \ . \tag{4.153}$$

   The inversion of C_k will be successful if it is sufficiently well-conditioned. If not, additional steps may be necessary [46].
   Because of rounding errors in the inversion, there are some tiny linear corrections.

3. $\hat{I}_{\rho,k}$ is included into the interval box $d \cdot [-1;1]^n$, where d is a small number.

4. Next, the nonlinearity $\hat{\mathcal{N}}_{\rho,k}(\mathfrak{z})$ and its Jacobian $\frac{\partial \hat{\mathcal{N}}_{\rho,k}}{\partial \mathfrak{z}}$ are estimated by the number s and t according to

$$\begin{aligned} s &\geq \left| \hat{\mathcal{N}}_{\rho,i,k}(\mathfrak{z}) \right|, \forall \mathfrak{z} \in [-1;1]^n, 1 \leq i \leq n, \\ t &\geq \left| \frac{\partial \hat{\mathcal{N}}_{\rho,i,k}}{\partial \mathfrak{z}_j} \right|, \forall \mathfrak{z} \in [-1;1]^n, 1 \leq i,j \leq n \ . \end{aligned} \tag{4.154}$$

5. Then the so called *shrink wrap factor* is defined as follows [47]:

$$q = 1 + d \cdot \frac{1}{(1-(n-1)t)(1-s)} \ . \tag{4.155}$$

   The bounds s and t for the polynomials $\hat{\mathcal{N}}_{\rho,i,k}$ and $\frac{\partial \hat{\mathcal{N}}_i}{\partial \mathfrak{z}_j}$ can be computed by interval evaluation.

6. Now the Polynomial part of $\tilde{T}_{\rho,k}$ is multiplied by q and the constant c_k is added resulting into

$$T_{\rho,k}^{SW}(\mathfrak{z}) = c_k + (q \cdot \tilde{P}_{\rho,k}(\mathfrak{z})) + I_{\rho,k}^{SW} = P_{\rho,k}^{SW}(\mathfrak{z}) + I_{\rho,k}^{SW} \tag{4.156}$$

   with an interval remainder $I_{\rho,k}^{SW}$, which is close to machine precision, and the property

$$T_{\rho,k}(\mathfrak{z}) \subset T_{\rho,k}^{SW}(\mathfrak{z}) \ . \tag{4.157}$$

   The interval remainder of $T_{\rho,k}$ is absorbed by the polynomial part of $T_{\rho,k}^{SW}$ and the integration is continued with $T_{\rho,k}^{SW}$.

On the first glance it seems to be a drawback that the enclosure of the state vector is blown up due to the absorption of the interval remainder in polynomial part. However, because the new interval remainder term is close to zero the dependency problem is drastically reduced in the following integration step. Also the wrapping effect is decreased. However, there are also limitations for shrink wrapping. The measures of nonlinearities s, and t must not become too large. Besides, the application of the inverse of the linear part should not lead to large increases in the size of remainder bounds. If the matrix C_k is not invertible, a QR decomposition as described in Section 4.1.5 can be applied, which however may increase s and t too much. In [47] a more sensitive method called *blunting* is proposed. *Blunting* provides an upper bound for the condition number of the modified matrix, by an appropriate increase of the angel between the column vectors.

Example for Shrink Wrapping [54]

The following Taylor model vector is considered:

$$\begin{aligned} T_{\rho,1}(\mathfrak{z}) &= 2 + 4\mathfrak{z}_1 + \frac{1}{2}\mathfrak{z}_1^2 + [-0.2; 0.2], \\ T_{\rho,2}(\mathfrak{z}) &= 1 + 3\mathfrak{z}_2 + \mathfrak{z}_1\mathfrak{z}_2 + [-0.1; 0.1] \\ &\text{with} \quad \mathfrak{z}_1, \mathfrak{z}_2 \in [-1; 1] \ . \end{aligned} \tag{4.158}$$

First the constant part c_k of the polynomial is removed and the following equation is obtained:

$$\begin{aligned} \tilde{T}_{\rho,1}(\mathfrak{z}) &= 4\mathfrak{z}_1 + \frac{1}{2}\mathfrak{z}_1^2 + [-0.2; 0.2], \\ \tilde{T}_{\rho,2}(\mathfrak{z}) &= 3\mathfrak{z}_2 + \mathfrak{z}_1\mathfrak{z}_2 + [-0.1; 0.1]. \end{aligned} \tag{4.159}$$

The matrix associate with the linear part of the Taylor model (4.158) is

$$C := \begin{pmatrix} 4 & 0 \\ 0 & 3 \end{pmatrix} \ . \tag{4.160}$$

Multiplying (4.159) with C^{-1}, then gives

$$\begin{aligned} \hat{T}_{\rho,1}(\mathfrak{z}) &= \mathfrak{z}_1 + \frac{1}{8}\mathfrak{z}_1^2 + [-0.05; 0.05], \\ \hat{T}_{\rho,2}(\mathfrak{z}) &= \mathfrak{z}_2 + \frac{1}{3}\mathfrak{z}_1\mathfrak{z}_2 + [-0.034; 0.034]. \end{aligned} \tag{4.161}$$

Estimating the nonlinear part and the interval terms as described above, numbers s, t, and d satisfying

$$\begin{aligned} & s \geq \left|\frac{1}{8}{\mathfrak{z}_1}^2\right|, \ s \geq \left|\frac{1}{3}\mathfrak{z}_1\mathfrak{z}_2\right| \ \forall \ \mathfrak{z}_1, \ \mathfrak{z}_2 \in [-1; 1], \\ & t \geq \left|\frac{1}{4}\mathfrak{z}_1\right|, \ t \geq \left|\frac{1}{3}\mathfrak{z}_2\right|, \ t \geq \left|\frac{1}{3}\mathfrak{z}_1\right| \ \forall \ \mathfrak{z}_1, \ \mathfrak{z}_2 \in [-1; 1], \\ & d \geq 0.05, \ d \geq 0.034 \end{aligned} \tag{4.162}$$

are computed.

These conditions are fulfilled for $s = t = \frac{1}{3}$ and $d = 0.05$, from which the shrink wrap factor is deduced (4.155)

$$q = 1 + d \cdot \frac{1}{(1-(m-1)t)(1-s)} = \frac{89}{80}. \tag{4.163}$$

The final Taylor model after shrink wrapping is

$$\begin{aligned} T^{sw}_{\rho,1}(\mathfrak{z}) &= c_{1,1} + q\tilde{P}_1(\mathfrak{z}) = 2 + \frac{89}{20}\mathfrak{z}_1 + \frac{89}{160}{\mathfrak{z}_1}^2, \\ T^{sw}_{\rho,2}(\mathfrak{z}) &= c_{2,1} + q\tilde{P}_2(\mathfrak{z}) = 1 + \frac{287}{80}\mathfrak{z}_2 + \frac{89}{80}\mathfrak{z}_1\mathfrak{z}_2. \end{aligned} \tag{4.164}$$

The sets of the Taylor models before (4.158) and after shrink wrapping (4.164) are shown in

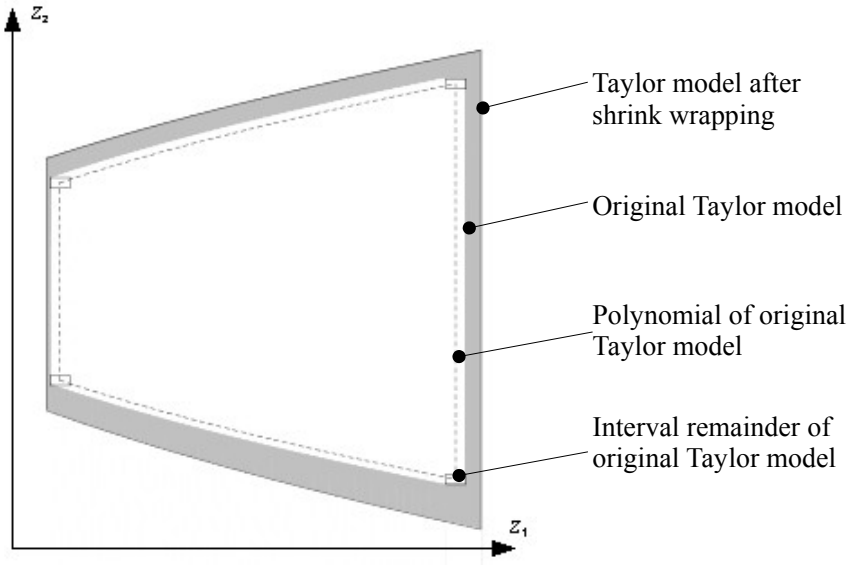


Figure 4.28: Sets of the Taylor models before and after shrink wrapping.

Fig. 4.28. The dotted line is the boundary of the set that is described by the polynomial of the original Taylor model. The white area is the set described by the original Taylor model, including the interval remainder. The excess area introduced by shrink wrapping is shaded in gray.

Preconditioning

Another method for reduction of overestimation in Taylor model based verified integration is based on preconditioned Taylor models [46, 54]. In this method the flow of the ODE is represented by a composition of a left and a right Taylor model

$$T_l \circ T_r = (P_l + I_l) \circ (P_r + I_r). \tag{4.165}$$

The composition

$$T_\rho(\mathfrak{z}) := (P_l(\bar{\mathfrak{z}}) + I_l) \circ (P_r(\mathfrak{z}) + I_r) \tag{4.166}$$

of the two Taylor models

$$T_l(\bar{\mathfrak{z}}) = P_l(\bar{\mathfrak{z}}) + I_l \ \ , \bar{\mathfrak{z}} \in [\bar{\mathfrak{z}}] \tag{4.167}$$

and

$$T_r(\mathfrak{z}) = P_r(\mathfrak{z}) + I_r \ \ , \mathfrak{z} \in [\mathfrak{z}] \tag{4.168}$$

is called *preconditioned* Taylor model if

$$\mathrm{Rg}(T_r) \subseteq \bar{\mathfrak{z}} \ . \tag{4.169}$$

T_l is the *preconditioner* and T_r the *conditioned* Taylor model. $P_l + I_l$ is chosen in a way that I_l is zero up to roundoff, and the operations on I_r are minimized. For practical purposes $(P_r + I_r)$ is normalized in the factorization [46], such that each of its components has a range in $[-1; 1]$.The left Taylor model T_l can be seen as a specific coordinate system in which the flow of the ODE is studied.

If the flow of an ODE is enclosed by the composition $T_l \circ T_r$, then in order to compute the enclosure of flow in the next integration step it is sufficient to integrate the ODE for the set enclosed by $\mathrm{Rg}(T_l)$, and to compose the integrated Taylor model with T_r. Higher order terms appearing in the composition process are included in the interval remainder of the result.
This has the effect that during the integration between two time-steps only the left Taylor model is affected the right Taylor model T_r remains unchanged [46, 54]. In practice the interval bound in the Picard iteration is not determined for the $\bar{\mathfrak{z}} \in \mathrm{Rg}(T_r)$ but for the larger set $\bar{\mathfrak{z}} \in [\bar{\mathfrak{z}}]$. This is a potential source of overestimation. However, the dependency problem is reduced since the remainder error I_l is close to zero. The advantage of preconditioning becomes apparent after several integration steps.

In the following the computations for preconditioning in each integration step are outlined. Let

$$T_{\rho,k}(\mathfrak{z}) = T_{l,k} \circ T_{r,k} = (P_{l,k} + I_{l,k}) \circ (P_{r,k} + I_{r,k}) \tag{4.170}$$

be a factored Taylor model that encloses the flow of the ODE at time t_k. Let

$$\check{T}_{l,k+1} = \check{P}_{l,k+1} + \check{I}_{l,k+1} \tag{4.171}$$

be the result of integrating $T_{l,k}(\mathfrak{z})$ from t_k to t_{k+1}. Then

$$\check{T}_{\rho,k+1} = (\check{P}_{l,k+1} + \check{I}_{l,k+1}) \circ (P_{r,k} + I_{r,k}) \tag{4.172}$$

is a factorization of the flow at time t_{k+1} [46].

In the beginning at $t = 0$ of the integration, the flow of the initial condition box can be represented as the composition of two identity Taylor models

$$\bar{\mathfrak{z}} \circ \mathfrak{z} \ . \tag{4.173}$$

The beneficial use of the method, and in particular its use in reducing the growth of remainder terms, is the moving terms between the left and right factors [46]. This is described in the following.

To get the factorization, the following steps are necessary. The inclusion of the flow in the $k+1$-st integration step is given by the Taylor model $(\check{P}_{l,k+1} + \check{I}_{l,k+1}) \circ (P_{r,k} + I_{r,k})$. Let $\check{c}_{k+1}$ be the constant part of $\check{P}_{l,k+1}$, $\check{C}_{k+1}$ the linear part of $\check{P}_{l,k+1}$, and $\check{N}_{k+1}$ includes the nonlinear part and the remainder interval. So $\check{T}_{l,k+1} = \check{c}_{k+1} + \check{C}_{k+1} + \check{N}_{k+1}$. Set $c_{l,k+1} = \check{c}_{k+1}$ and assume that $C_{l,k+1}$ is the desired linear part of the left factor. Later it will be shown how to choose $C_{l,k+1}$. The identity transformation $(C_{l,k+1} \circ C_{l,k+1}^{-1})$, is inserted and the inclusion of the flow results in,

$$\begin{aligned}
&(\check{c}_{k+1} + \check{C}_{k+1} + \check{N}_{k+1}) \circ (P_{r,k} + I_{r,k}) \\
&= \check{c}_{k+1} + (\check{C}_{k+1} + \check{N}_{k+1}) \circ (P_{r,k} + I_{r,k}) \\
&= c_{l,k+1} + (C_{l,k+1} + [0,0]) \circ \left(C_{l,k+1}^{-1} \circ (\check{C}_{k+1} + \check{N}_{k+1}) \circ (P_{r,k} + I_{r,k})\right) \\
&= (c_{l,k+1} + C_{l,k+1} + [0,0]) \circ \left\{(C_{l,k+1}^{-1} \circ \check{C}_{k+1} + C_{l,k+1}^{-1} \circ \check{N}_{k+1}) \circ (P_{r,k} + I_{r,k})\right\}.
\end{aligned} \tag{4.174}$$

The expression in the curly brackets is denoted by $(\bar{P}_{r,k+1} + \bar{I}_{r,k+1})$ and its component interval bounds are calculated. In order to continue the integration with the left Taylor model with the unit box as domain interval, a scaling matrix S_{k+1} is determined such that the new right Taylor model is bounded by $[-1;1]^n$. To this effect the identity transformation $(S_{k+1} \circ S_{k+1}^{-1})$ is introduced in (4.174) according to

$$(c_{l,k+1} + C_{l,k+1} + [0,0]) \circ (S_{k+1} \circ S_{k+1}^{-1}) \circ (\bar{P}_{r,k+1} + \bar{I}_{r,k+1}) \ . \tag{4.175}$$

Denoting $(P_{r,k+1} + I_{r,k+1}) = S_{k+1}^{-1} \circ (\bar{P}_{r,k+1} + \bar{I}_{r,k+1})$, an enclosure of the flow at t_{k+1} is given by the composition

$$T_{\rho,k+1} = \underbrace{(c_{l,k+1} + C_{l,k+1} + [0,0]) \circ S_{k+1}}_{T_{l,k+1}} \circ \underbrace{(P_{r,k+1} + I_{r,k+1})}_{T_{r,k+1}} = T_{l,k+1} \circ T_{r,k+1} \tag{4.176}$$

As mentioned above, in the next integration step only $T_{l,k+1}$ is affected by the integration. There are several points to analyze the effects of this procedure [46, 54].

1. This method is successful, if the amount of overestimation arising by the wrapping of $\check{T}_{\rho,k+1} = \check{T}_{l,k+1} \circ T_{\rho,r,k}$ in $T_{\rho,k+1} = T_{l,k+1} \circ T_{r,k+1}$ is sufficiently small and if the $\mathrm{Rg}(T_{l,k+1})$ is more appropriate for continuing the integration than $\check{T}_{l,k+1}$.

2. The polynomial part of $C_{l,k+1}^{-1} \circ \check{N}_{k+1}$ is nonlinear, so its action on $(P_{r,k} + I_{r,k})$ via composition will introduce only a small contribution to the remainder bound, if $I_{r,k}$ is still small.

3. The remainder part of $C_{l,k+1}^{-1} \circ \check{N}_{k+1}$, which contains as one important contribution the action of $C_{l,k+1}^{-1}$ on the remainder interval of $\check{N}_{k+1}$, will be added to $I_{r,k}$. The magnification of the remainder bound of $\check{N}_{k+1}$ by the action of $C_{l,k+1}^{-1}$ is proportional to the condition number of $C_{l,k+1}$.

4. Contributions of a similar magnitude as $I_{r,k}$ come from application of the linear term $C_{l,k+1}^{-1} \circ \check{C}_{k+1}$ to $I_{r,k}$. If this term is not chosen properly, exponential growth of the remainder bound over time could happen.

There are several choices for the determination of $C_{l,k+1}$. Three different types are implemented in COSY VI, which are QR, curvilinear and blunting.

The QR preconditioning is to choose $C_{l,k+1}$ as the matrix Q of the matrix of the QR factorization of the matrix, which is obtained by sorting the columns of $\check{C}_{k+1}$ by size in descending order. The matrix is determined in the same way as described in Section 4.1.5. The QR preconditioning leads to a coordinate system that is orthogonal, and thus the transformation in and out of this system is well conditioned [46].

Further preconditioning techniques are curvilinear preconditioning and blunted parallelepiped preconditioning. The curvilinear preconditioning results also in an orthogonal coordinate system.The blunted parallelepiped preconditioning is to choose $C_{l,k+1}$ to be the q-blunting of $\check{C}_{k+1}$, where q is a suitable blunting factor. The q-blunting provides an upper bound for the condition number of the matrix $C_{l,k+1}$, and thus a strict upper limit to the overestimation. A detailed description for both is given in [46]. In this thesis only QR-preconditioning is used, as it is the most robust method for all studied applications.

An example for the application of preconditioning can be found in Appendix A.2.

Splitting of the Domain Interval Vector

In the following, splitting of the domain interval vector into subboxes [37] is proposed as a tool for reduction of overestimation. The state vector z_{k+1} is then enclosed by a list $\mathcal{T}_{k+1}$ of Taylor models:

$$\begin{aligned} z_{k+1} \in \mathcal{T}_{k+1} = \left\{ T_{\rho,k+1}^{(1)}(\mathfrak{z}), T_{\rho,k+1}^{(2)}(\mathfrak{z}), \ldots T_{\rho,k+1}^{(L_{k+1})}(\mathfrak{z}) \right\} \\ \text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1,2,\ldots,n \quad \text{and} \quad L_{k+1} \leq L_{max} \ . \end{aligned} \tag{4.177}$$

Consider a Taylor model $T_{\rho,k+1}(\mathfrak{z})$ with the domain interval vector $[\mathfrak{z}]$, hence $\mathfrak{z} \in [\mathfrak{z}]$. The domain interval vector of this Taylor model is split into subboxes $[\tilde{\mathfrak{z}}]$.

For numerical and implementation reasons it is advantageous to have the unit box $[-1;1]^n$ as a domain interval vector in each integration step [47]. To obtain again the unit box as a domain interval vector, $[\tilde{\mathfrak{z}}]$ is expressed as a Taylor model according to

$$\begin{aligned} [\tilde{\mathfrak{z}}] = \tilde{T}(\mathfrak{z}) = \tilde{c} + \tilde{D}\,\mathfrak{z} \\ \text{with} \quad \mathfrak{z}_i \in [-1;1] \ , i = 1\ldots,n \ , \end{aligned} \tag{4.178}$$

where $\tilde{c}$ is the midpoint of $[\tilde{\mathfrak{z}}]$ and $\tilde{D}$ is a diagonal matrix with $\tilde{d}_{i,i} = \cdot\mathrm{rad}([\tilde{\mathfrak{z}}_i])$. The components of the vector of the original initial state vector $\mathfrak{z}$ of $T_{\rho,k+1}(\mathfrak{z})$ are replaced by the components of $\tilde{T}(\mathfrak{z})$ by substituting $\tilde{T}_i(\mathfrak{z})$ for $\mathfrak{z}_i$, which results in a modified Taylor model

$$\tilde{T}_{\rho,k+1}(\mathfrak{z}) = T_{\rho,k+1}(\tilde{T}(\mathfrak{z})) \tag{4.179}$$

for each subbox $[\tilde{\mathfrak{z}}]$.

To determine the component in which the domain interval vector has to be split, a sensitivity analysis is performed. The component μ of the domain interval vector $\mathfrak{z}$ is chosen for splitting in which the Taylor model $T_{\rho,k+1}(\mathfrak{z})$ is most sensitive. For that purpose, all $w_{i,j}$

$$\begin{aligned} w_{i,j} &= \mathrm{diam}([\mathfrak{z}_i]) \cdot \left| \frac{\partial T_{\rho,k+1,j}(\mathfrak{z})}{\partial \mathfrak{z}_i}\Big|_{\mathfrak{z}=\mathrm{mid}([\mathfrak{z}])} \right| \\ & i = 1,\ldots,n,\ \ j = 1,\ldots,n \end{aligned} \tag{4.180}$$

have to be calculated and the component μ is determined by

$$\mu = \underset{i=1\ldots n}{\arg\max} \left(\sum_{j=1}^{n} w_{i,j} \right) \quad . \tag{4.181}$$

Alternatively the $w_{i,j}$ can be computed by

$$\begin{aligned} w_{i,j} &= \mathrm{diam}([\mathfrak{z}_i]) \cdot \mathrm{diam}\left(\left[\frac{\partial T_{\rho,k+1,j}(\mathfrak{z})}{\partial \mathfrak{z}_i}\Big|_{\mathfrak{z}=[\mathfrak{z}]} \right] \right) \\ & i = 1,\ldots,n,\ \ j = 1,\ldots,n \end{aligned} \tag{4.182}$$

or

$$\begin{aligned} w_{i,j} &= \mathrm{diam}\left(\left[\frac{\partial T_{\rho,k+1,j}(\mathfrak{z})}{\partial \mathfrak{z}_i}\Big|_{\mathfrak{z}=[\mathfrak{z}]} \right] \cdot ([\mathfrak{z}_i] - \mathrm{mid}([\mathfrak{z}_i])) \right) \\ & i = 1,\ldots,n,\ \ j = 1,\ldots,n. \end{aligned} \tag{4.183}$$

In (4.180), the results are the coefficients of the linear part of the Taylor model $T_{\rho,k+1}(\mathfrak{z})$ multiplied by the factor 2, because $\mathrm{mid}([\mathfrak{z}_i]) = 0$ and $\mathrm{diam}([\mathfrak{z}_i]) = 2$ for $\mathfrak{z}_i \in [-1;1]$ $, i = 1,\ldots,n$.

If several Taylor models are already present, the most appropriate Taylor model for the splitting of the domain interval vector has to be selected. This is done by calculating the interval enclosure of each Taylor model and the corresponding pseudo volume of the resulting interval vector. The Taylor model with the largest pseudo volume is selected for splitting. Alternatively Taylor models with the largest interval remainder can be selected for splitting.

Example

Consider the following two dimensional Taylor model (for simplicity the remainder interval is set to zero) enclosing the state vector $z = [z_1, z_2]^T$:

$$\begin{aligned} z_1 \subseteq T_1 &= 1 + 0.5\mathfrak{z}_1 + 2\mathfrak{z}_2 - \mathfrak{z}_1^2 - \mathfrak{z}_2^2 \ , \\ z_2 \subseteq T_2 &= 1 + 0.5\mathfrak{z}_1 + \mathfrak{z}_2^2 \\ & \qquad \text{with } \mathfrak{z}_1, \mathfrak{z}_2 \in [-1;1] \ . \end{aligned} \tag{4.184}$$

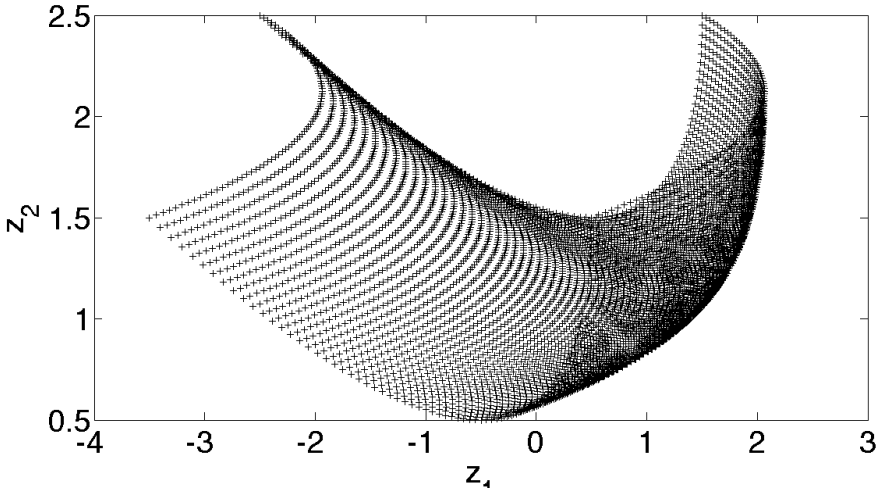


Figure 4.29: Before Splitting.

The domain interval vector is split in the second component; hence $[\mathfrak{z}_2]$ is split in $[\mathfrak{z}_2^a] = [0;1]$ and $[\mathfrak{z}_2^b] = [-1;0]$. The obtained subintervals $[\mathfrak{z}_2^a]$ and $[\mathfrak{z}_2^b]$ are expressed as Taylor models with domain interval $[-1;1]$:

$$\begin{aligned} \mathfrak{z}_2^a \in & T_a = 0.5 + 0.5\mathfrak{z}_2 \ , \\ \mathfrak{z}_2^b \in & T_b = -0.5 + 0.5\mathfrak{z}_2 \\ & \text{with } \mathfrak{z}_2 \in [-1;1] \ . \end{aligned} \tag{4.185}$$

T^a and T^b are substituted for $\mathfrak{z}_2$ in T leading to two Taylor models:

$$\begin{aligned} T_1^a &= 1.75 + 0.5\mathfrak{z}_1 + 0.5\mathfrak{z}_2 - \mathfrak{z}_1^2 - 0.25\mathfrak{z}_2^2 \ , \\ T_2^a &= 1.25 + 0.5\mathfrak{z}_1 + 0.5\mathfrak{z}_2 + 0.25\mathfrak{z}_2^2 \\ & \text{with } \mathfrak{z}_1 \text{ and } \mathfrak{z}_2 \in [-1;1] \end{aligned} \tag{4.186}$$

and

$$\begin{aligned} T_1^b &= -0.25 + 0.5\mathfrak{z}_1 + 1.5\mathfrak{z}_1 - 0.25\mathfrak{z}_2^2 - \mathfrak{z}_1^2 \ , \\ T_2^b &= 1.25 + 0.5\mathfrak{z}_1 - 0.5\mathfrak{z}_2 + 0.25\mathfrak{z}_2^2 \\ & \text{with } \mathfrak{z}_1 \text{ and } \mathfrak{z}_2 \in [-1;1] \ . \end{aligned} \tag{4.187}$$

In Figs. 4.29 and 4.30, the Taylor models before and after splitting are depicted in terms of points obtained by evaluation of the Taylor models with multiple points of the domain interval vectors.

Splitting in Combination with Preconditioning

If preconditioning techniques are applied the flow of the ODE in the $k+1$-st integration step is enclosed by the composition

$$T_{\rho,k+1}(\mathfrak{z}) := (P_{l,k+1}(\bar{\mathfrak{z}}) + I_{l,k+1}) \circ (P_{r,k+1}(\mathfrak{z}) + I_{r,k+1}) \ . \tag{4.188}$$

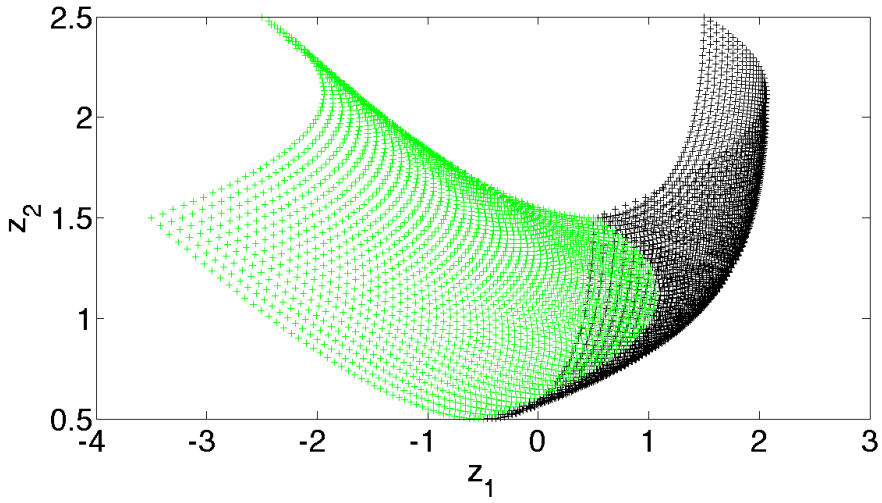


Figure 4.30: After Splitting.

The initial state vector is given by $\mathfrak{z}$. Thus, if the domain interval vector of a component $\mathfrak{z}_i$ is split the right Taylor model is affected. The splitting is done before the application of the scaling to the unit box. Before the scaling the Taylor model

$$(c_{l,k+1} + C_{l,k+1} + [0,0]) \circ (\bar{P}_{r,k+1} + \bar{I}_{r,k+1}) \tag{4.189}$$

is given.

Since when the domain interval of a component $\mathfrak{z}_i$ is split, each subinterval is again defined by a Taylor model with the unit box as domain interval a constant part $\tilde{\bar{c}}_r$ appears in the right Taylor models. Each Taylor model $\tilde{T}_{\rho,k+1}(\mathfrak{z})$ after splitting is then given by

$$\tilde{T}_{\rho,k+1}(\mathfrak{z}) = (c_{l,k+1} + C_{l,k+1} + [0,0]) \circ (\tilde{\bar{c}}_r + \tilde{\bar{C}}_{r,k+1} + \tilde{\bar{N}}_{r,k+1} + \bar{I}_{r,k+1}) \quad , \tag{4.190}$$

where $\tilde{\bar{c}}_{r,k+1}$ is the constant part, $\tilde{\bar{C}}_{r,k+1}$ the linear part and $\tilde{\bar{N}}_{r,k+1}$ the nonlinear part of the new right Taylor model, after splitting. The remainder term is not effected, thus it is the same for each new Taylor model. The constant part is extracted leading to

$$(\underbrace{c_{l,k+1} + C_{l,k+1} \circ \tilde{\bar{c}}_r}_{\tilde{c}_{l,k+1}} + C_{l,k+1} + [0,0]) \circ (\tilde{\bar{C}}_{r,k+1} + \tilde{\bar{N}}_{r,k+1} + \bar{I}_{r,k+1}) \quad . \tag{4.191}$$

Now the scaling can be applied to (4.191):

$$(\tilde{c}_{l,k+1} + C_{l,k+1} + [0,0]) \circ (S_{k+1} \circ S_{k+1}^{-1}) \circ (\tilde{\bar{C}}_{r,k+1} + \tilde{\bar{N}}_{r,k+1} + \bar{I}_{r,k+1}) \ . \tag{4.192}$$

Denoting $(\tilde{P}_{r,k+1} + I_{r,k+1}) = S_{k+1}^{-1} \circ (\tilde{\bar{C}}_{r,k+1} + \tilde{\bar{N}}_{r,k+1} + \bar{I}_{r,k+1})$. The l-th Taylor model after splitting is given by

$$T_{\rho,k+1}^{(l)} = \underbrace{(\tilde{c}_{l,k+1} + C_{l,k+1} + [0,0]) \circ S_{k+1}}_{\tilde{T}_{l,k+1}^{(l)}} \circ \underbrace{(\tilde{P}_{r,k+1} + I_{r,k+1})}_{\tilde{T}_{r,k+1}^{(l)}} = \tilde{T}_{l,k+1}^{(l)} \circ \tilde{T}_{r,k+1}^{(l)} \ . \tag{4.193}$$

4.5 Summary

In this Chapter algorithms for the integration of nonlinear uncertain systems have been presented. The methods and Section 4.1 are based on a interval evaluation of the Taylor series expansion of the extended state vector in time. After the basic computation method, advanced methods like the mean-value evaluation, monotonicity tests and iterative range computation, and implicit integration have been explained in detail. These concepts are especially suited to fight the dependency problem.

Algorithms for reduction of the wrapping effect have also been developed in this work. Methods based on coordinate transformation like Lohner's methods, which has been described in this chapter, employ parallelepipeds for the state vector enclosure. In this thesis an efficient algorithm to calculate the intersection of different parallelepipeds, and therefore also the intersection of a parallelepiped with an interval vector, has been introduced.In order to obtain tight enclosures for non-convex several interval vectors are used. Therefore, efficient splitting and merging routines have to be applied. A combination of coordinate transformation and interval splitting can lead to even tighter enclosures as the illustrative example has shown. Coordinate transformation and interval splitting are important methods reducing of the wrapping effect.

The consistency tests through backward integration developed in Section 4.2 are efficient to reduce overestimation, which has been accumulated over one or several integration steps. This method reduces already occurred overestimation, whereas the methods from Section 4.1 try to avoid the occurrence overestimation in the first place. The combination of an efficient forward evaluation and the consistency test as applied here is a very powerful method for obtaining tight enclosures of the solution sets.

Interval methods also allow integrating systems with state depended switching characteristics and can therefore also be used for models including friction effects. This has been illustrated here by a system with static and sliding friction.

Section 4.4 has been devoted to verified integration based on Taylor models. In addition to an expansion in time, also an expansion in the initial states is performed. The expansion in the initial states reduces the dependency problem and the wrapping effect. The interval remainder part of the Taylor model, however, is affected by the dependency problem and the wrapping effect like the integration methods in Section 4.1. Methods to limit the long-term growth of the truncation error and to further reduce overestimation are shrink wrapping and preconditioned Taylor models. The first method is only suitable if the non-linearities are not strong, the later method is much more efficient and robust. However, in order to handle large uncertainty, splitting of the domain interval vector becomes unavoidable. The extended state vector is then enclosed by a list of Taylor models.

5 Verified State and Parameter Estimators

5.1 Problem Formulation

Control strategies in engineering require [13,70] information about the state variables. However, in many systems not all state variables can be measured or the measurement is too costly. Provided that the system is observable, a state observer performs a model based reconstruction of the non-measured state variables by taking into account the system dynamics described by the system model. In addition to an estimation of the state variables often also an estimation of uncertain system parameters is required.

Observers are used in open and closed loop control systems. A general observer concept for open loop control is depicted in Fig. 5.1. The control variable of the plant is also fed into the observer. With the help of the system dynamics and the measured values the state and parameters are estimated by the observer. In closed loop control the estimated state variables and parameters are fed back into the controller (see Fig. 5.2). Classical observer concepts are, e.g., the Luenberger observer [17] or the nonlinear tracking observer [16]. In this thesis an introduction to the design of a nonlinear tracking observer is given in Section 6.2.4. In case of interval uncertainty a classical observer concept is not able to give worst case estimations of the state variables and parameters, i.e., to determine guaranteed bounds of all possible values of the the systems states and parameters.
A so called *verified state and parameter estimator* (VSPE) takes into account the knowledge about the guaranteed state and parameter enclosures during the reconstruction of the non-measured state variables and parameters.

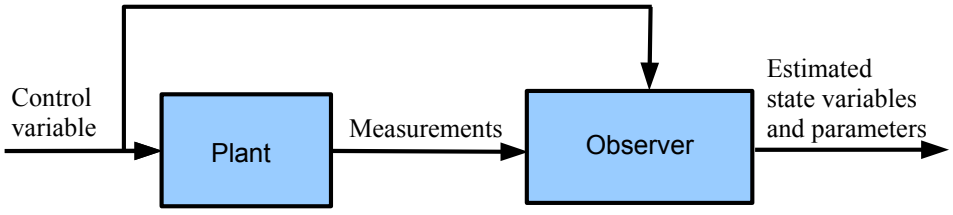


Figure 5.1: Observer in open loop control.

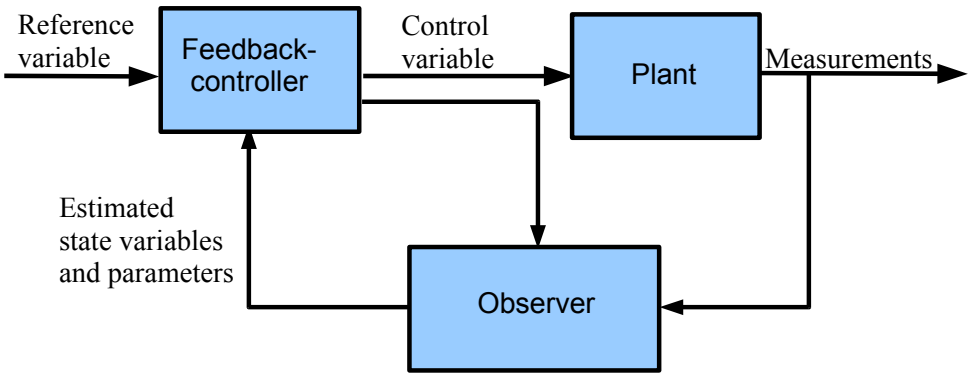


Figure 5.2: Observer in closed loop control.

In this chapter only open loop estimation is considered. In Chapter 7 it is shown how a VSPE is applied in closed loop control.

A VSPE requires in addition to the system model and the measured values a measurement model model describing the sensor characteristics. The ideal measurement equation depending on the extended state vector $z(t)$ and time t is given by

$$\check{y}(t) = \check{h}\left(z\left(t\right), t\right) \tag{5.1}$$

with $\check{h} : D \mapsto \mathbb{R}^m \times \mathbb{R}$, $D \subset \mathbb{R}^n$. The vector $\check{y}(t)$ denotes the ideal measurement vector. In the following it is assumed that the measurements are affected by an additive measurement noise $\delta(t)$ and the vector y of measured values is given by

$$y(t) = \check{y}(t) + \delta(t) = \check{h}\left(z\left(t\right), t\right) + \delta\left(t\right) \quad . \tag{5.2}$$

The additive measurement noise is assumed to be bounded by an upper bound $\overline{\delta}$ for the absolute value of the measurement error, hence $\delta(t) \in \mathbb{R}^m$ is bounded by $\delta(t) \in [-\overline{\delta}; \overline{\delta}]$ for all t. And the ideal measurement vector is contained in the interval vector given by addition of the measured values and the interval vector of the measurement error, namely,

$$\check{y}(t) \in [y(t)] = [y(t) - \overline{\delta}, y(t) + \overline{\delta}] \quad . \tag{5.3}$$

If additionally uncertain parameters q in the sensor model are considered, the measurement equation is given by

$$y(t) = \check{h}\left(z\left(t\right), q\left(t\right), t\right) + \delta\left(t\right) = h\left(z\left(t\right), q\left(t\right), \delta\left(t\right), t\right) \quad . \tag{5.4}$$

with $h : D \mapsto \mathbb{R}^m$, $D \subset \mathbb{R}^n \times \mathbb{R}^{n_q} \times \mathbb{R}^m \times \mathbb{R}$. The parameter vector $q(t) \in \mathbb{R}^{n_q}$ of the measurement equation containing also non-additive uncertainties is bounded by $q(t) \in [\underline{q}; \overline{q}]$ for all t. In general also uncertain parameters $q(t)$ can be estimated by the VSPE. If measured data is available at discrete points of time the measurement equation becomes.

$$y(t_{k+1}) = \check{h}\left(z\left(t_{k+1}\right), q\left(t_{k+1}\right), t_{k+1}\right) + \delta\left(t_{k+1}\right) = h\left(z\left(t_{k+1}\right), q\left(t_{k+1}\right), \delta\left(t_{k+1}\right), t_{k+1}\right) \quad . \tag{5.5}$$

Here $y(t_{k+1})$ denotes the measurements at the discrete points of time t_{k+1}. To simplify the notation (5.5) is rewritten according to

$$y_{k+1} = \check{h}\left(z_{k+1}, q_{k+1}, t_{k+1}\right) + \delta_{k+1} = h\left(z_{k+1}, q_{k+1}, \delta_{k+1}, t_{k+1}\right) \quad . \tag{5.6}$$

The estimation consists of two steps. *Step 1* is the so called *prediction step* and it performs a prediction of the extended state vector and parameters between two points of time at which measured data is available. The time-span between two measurements is denoted by ΔT_m. If the system is described by a set of ODEs which is assumed in this thesis, the prediction corresponds to a verified integration of the system under consideration of all uncertainties. As a result a worst case enclosure of all reachable state variables and parameters is obtained. *Step 2* is called *correction step* in which subsets of the predicted set are eliminated, which are inconsistent with the measurement model, the measured data and all uncertainties. If measured values are available after each integration step of the ODE, then ΔT_m is equivalent to the step-size h_k of the verified integration.

As a result, verified enclosures of state variables and parameters are obtained that comply with the system and measurement model and all uncertainties. Due to the fact that the system dynamics are considered the measurement error can be reduced, as the enclosure for the measured state variables estimated by the VSPE may be tighter than the vector of measured values plus the interval vector for the measurement uncertainty.

In contrast to stochastic state estimation approaches [10, 57], the verified estimation leads to guaranteed bounds. It also does not require any information about any distribution of the uncertainty. It also allows for the consideration of nonlinearities without any previous linearization like in the case of the Extended Kalman-filter [10, 57].

The main advantages of both set valued estimation approaches and stochastic estimation approaches are:

- The reconstruction of non-measured system state variables and parameters, provided that the system is observable.
- In case of stochastic estimators (stochastic uncertainties) and VSPEs (interval uncertainties), the estimation directly delivers statements about the accuracy of the estimates. For stochastic uncertainties this is expressed in terms of covariances and correlations. For set valued uncertainties such as intervals the accuracy is expressed in terms of the width of the interval bounds enclosing the estimates.
- Measurement data collected from different sensors is combined in a fusion process for reduction of the uncertainty of the estimated state variables and parameters.
- It is not required that the complete history of measured data is stored, since the estimates of all desired values are computed only in terms of the current measured value and previously in the prediction step calculated estimates. This allows for improvement of the current estimate as soon as new measured values are available.

The basic concept of the VSPE is illustrated in Fig. 5.3. At a given point of time t_k the set enclosing the extended state vector is predicted by a verified integration of the system model to t_{k+1}. If at t_{k+1} measured values are available the state variables and parameters are reconstructed with the help of the measurement model and sub domains of the predicted set which are inconsistent with the measured data and the measurement equation are deleted, which results in an improved enclosure which is used for the next prediction step. This procedure is repeated recursively. The set enclosing the extended state vector is predicted until the next measured values are available, then the next correction step is performed. The way the prediction and correction steps are performed depends on the underlying type of VSPE.

An important aspect of verified state and parameter estimation is the observability of the considered system. This is discussed in Section 5.2. In the Sections 5.3–5.5 two different types of VSPEs are introduced and compared. In Section 5.3 a verified state and parameter estimation concept based on interval splitting and merging routines in combination with preconditioning techniques is presented — the so called interval observer. The Taylor model observer in Section 5.4 uses Taylor models in order to enclose the extended state vector.

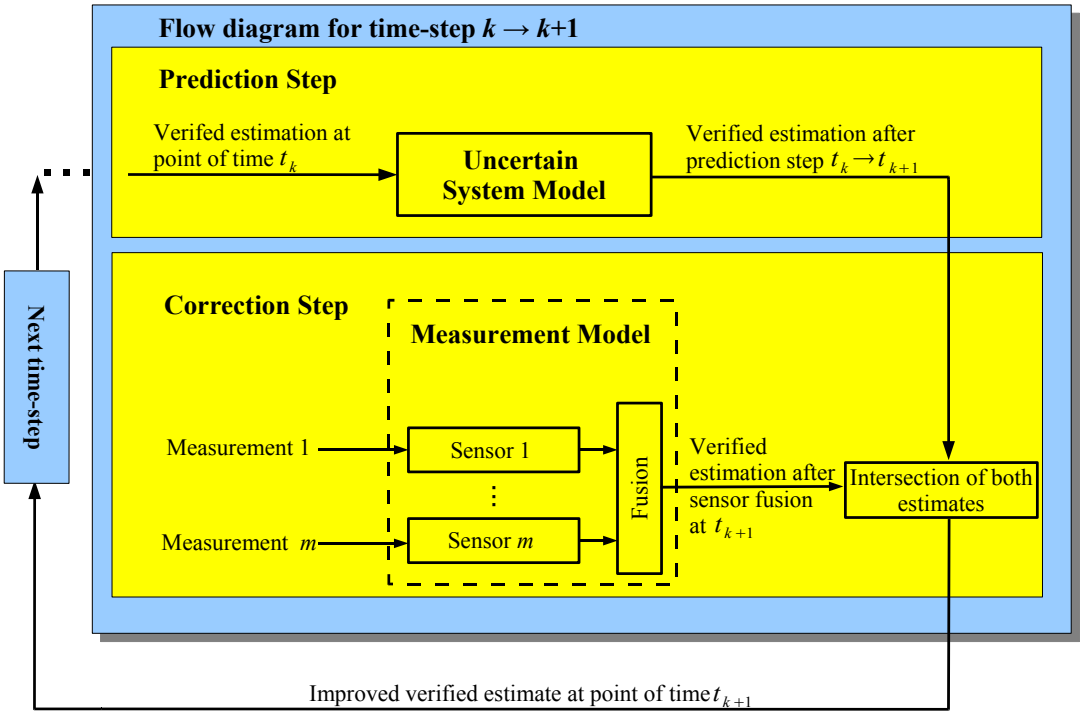


Figure 5.3: Interval observer concept.

5.2 Observability Analysis

5.2.1 Observability in Linear Systems

Consider a liner system of the form

$$\begin{aligned} \dot{z}(t) &= A \cdot z(t) + B \cdot u(t) \ , \\ y &= C \cdot z(t) + D \cdot u(t) \end{aligned} \tag{5.7}$$

with $z(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^{n_u}$, $y(t) \in \mathbb{R}^m$, $A \in \mathbb{R}^n \times \in \mathbb{R}^n$, $B \in \mathbb{R}^n \times \in \mathbb{R}^{n_u}$, $C \in \mathbb{R}^m \times \in \mathbb{R}^n$, and $D \in \mathbb{R}^m \times \in \mathbb{R}^{n_u}$.

A system is observable if the *Kalman observability matrix*

$$Q = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{5.8}$$

has the full column rank [17], hence

$$\text{rank}(Q) = n \ . \tag{5.9}$$

5.2.2 Observability in Nonlinear Systems

Consider the following nonlinear system model

$$\dot{z}(t) = f(z(t)) \tag{5.10}$$

and a measurement model

$$y(t) = h(z(t)) \ . \tag{5.11}$$

Since non-autonomous dynamical models can easily be rewritten as autonomous ones, discussion is restricted to the autonomous case. If the linearized system is observable in a given operating point z_s, then the nonlinear system is also is observable. For the linearized system

$$\Delta \dot{z} = A \Delta z \tag{5.12}$$

and

$$\Delta y = C \Delta z \tag{5.13}$$

with

$$\Delta z = (z - z_s), \quad A = \frac{\partial f}{\partial z}\Big|_{z=z_s}, \quad \text{and} \quad C = \frac{\partial h}{\partial z}\Big|_{z=z_s} \tag{5.14}$$

holds. Therefore it has to checked if the *Kalman observability matrix*

$$Q = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{5.15}$$

has the full column rank [13], hence

$$\text{rank}(Q) = n \ . \tag{5.16}$$

However, if the linearized system is not observable, it is still possible that the corresponding nonlinear system is observable. Consider the following scalar example:

$$\begin{aligned} \dot{z_1} &= f(z_1) \ , \\ y &= z_1^3 \ . \end{aligned} \tag{5.17}$$

Linearization in $z_s = 0$ leads to the linearized measurement equation $\Delta y = 0$. Whereas the original measurement equation can be solved uniquely for z_1 with $z_1 = \sqrt[3]{y}$.

One approach of nonlinear observability analysis is based on the so called observability map [13]

$$q(z) = \begin{bmatrix} q_1(z) \\ q_2(z) \\ \vdots \\ q_m(z) \end{bmatrix} \text{ with } q_i(z) = \begin{bmatrix} y_i \\ \dot{y}_i \\ \vdots \\ y_i^{(\mu_i - 1)} \end{bmatrix} = \begin{bmatrix} L_f^0 h_i(z) \\ L_f h_i(z) \\ \vdots \\ L_f^{\mu_i - 1} h_i(z) \end{bmatrix}$$

with $\mu = \sum_{i=1}^{m} \mu_i$ and $\mu \geq n$, where for the Lie-derivatives

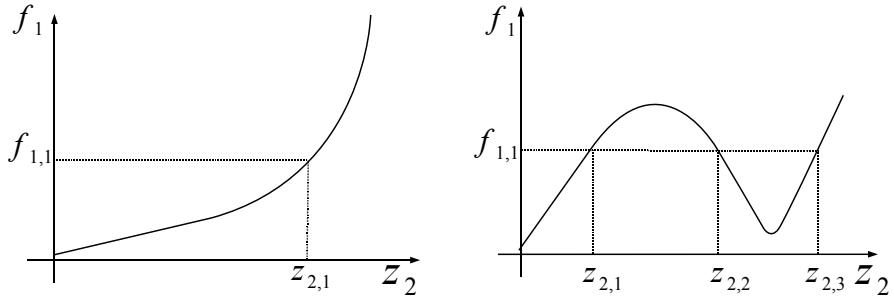
$$L_f^r h_i(z) = \frac{\partial (L_f^{r-1} h_i(z))}{\partial z} f(z) \text{ with } L_f^0 h_i(z) = h_i(z) \tag{5.18}$$

holds.

The observability map states a relation between the state variables and the time-derivatives of the output variables $y_i(t)$, $i = 1, \ldots, m$. The time-derivatives of the functions $h_i(z)$ along the system trajectory $z(t)$ are obtained by recursive application of the Lie-derivative (5.18). A difference to linear systems is the non fixed number of time derivatives $\mu_i, i = 1, \ldots, m$.

In a so called *selection map* q_a at least n arbitrary equations of the observability map are collected. If q_a is invertible, then also q is invertible and the system is observable. If for an observability map q more than one q_a exists, the measurement equation is called *redundant*, because there are several possibilities to reconstruct the state vector z from the measurements y.

In nonlinear systems it has to be distinguished between *global* and *local* observability. If the system observability matrix is globally uniquely invertible, the corresponding system is



(a) Global observability. (b) Local observability.

Figure 5.4: Comparison between *global* and *local* observability.

globally observable. However, if the system observability matrix is locally uniquely invertible the corresponding system is locally observable.

This shall be illustrated by the following example. Consider the system

$$\begin{aligned} \dot{z_1} =& f_1(z_1, z_2) \ , \\ \dot{z_2} =& f_2(z_1, z_2) \end{aligned} \tag{5.19}$$

with the measurement equation

$$y = z_1 \ . \tag{5.20}$$

The observability map is

$$\begin{aligned} y =& z_1 \ , \\ \dot{y} =& f_1(z_1, z_2) \ . \end{aligned} \tag{5.21}$$

As z_1 is known, only the second equation of (5.21) has to be solved for z_2. The function $f_1(z_1, z_2)$ in Fig. 5.4a) can be solved globally unique for z_2, therefore the system would be globally observable. The function $f_1(z_1, z_2)$ in Fig. 5.4b) can be solved only locally unique for z_2, therefore the system would just be locally observable.

For systems of the form

$$\dot{z} = \begin{bmatrix} f_1(z_1, z_2, \ldots, z_{m+1}) \\ f_2(z_1, z_2, \ldots, z_{m+2}) \\ \vdots \\ f_{n-m}(z_1, z_2, \ldots, z_n) \\ \vdots \\ f_n(z_1, z_2, \ldots, z_n) \end{bmatrix} \quad , \quad y = \begin{bmatrix} h_1(z_1) \\ h_2(z_1, z_2) \\ \vdots \\ h_m(z_1, z_2, \ldots, z_m) \end{bmatrix} \tag{5.22}$$

and

$$\frac{\partial h_i}{\partial z_i} \neq 0 \ \ i = 1, \ldots, m, \quad \frac{\partial f_i}{\partial z_{m+i}} \neq 0 \ \ i = 1, \ldots, n-m \ \ \forall z \in \mathbb{R}^n \tag{5.23}$$

a global observability can be guaranteed [13]. Condition (5.23) ensures, that the functions f_i and h_i are monotonic with respect to the last variable. Because of the global invertibility

of these functions, the first m state variables $z_1, z_2, \ldots, z_m$ can be determined from the measurement equation of (5.22). The remaining state variables $z_{m+1}, z_{m+2}, \ldots, z_n$ can be derived from the $n - m$ differential equations. In general it is very difficult to proof the global observability as it has to be proven that the observability map is invertible for all z.

It is much easier to investigate the local observability of a system and the local invertibility of $q(z)$ respectively in the neighborhood of a point z_s, by checking the rank of the Jacobian

$$Q(z) = \frac{\partial q}{\partial z} \ . \tag{5.24}$$

A nonlinear system is locally observable in a given point $z = z_s$, if

$$\operatorname{rank}(Q(z_s)) = n \tag{5.25}$$

holds [13]. However this criterion is only sufficient as the previous example (5.17) has shown.

5.2.3 Verified Observability Analysis

The purpose of a verified observability analysis is to calculate a guaranteed enclosure of the set of observable state vectors for a given system. The main tool for this purpose is the investigation of the rank of $Q(z)$. Let $[z]$ be an interval vector for which a system is checked for observability. The system is observable over $[z]$ if $Q(z)$ has full rank for all $z \in [z]$. Therefore the interval matrix $[Q]([z])$ is checked for full rank. In practice it is often necessary to split the considered interval $[z]$ into subboxes $[\tilde{z}]$ and check the full rank for each $[\tilde{Q}]$ corresponding to each subinterval $[\tilde{z}]$. For those subboxes $[\tilde{z}]$ where a full rank is obtained, it has been proven that the system is observable in the corresponding subinterval. Those subboxes $[\tilde{z}]$ for which a full rank cannot be guaranteed have to be split further and the test has to be repeated.
In the following an algorithm to check the rank of an interval matrix $[Q]$ is briefly described [66].

First $[Q]$ is expressed as

$$[Q] = [Q_m - \Delta, Q_m + \Delta] \ , \ \text{with} \ Q_m = \operatorname{mid}([Q]) \ \text{and} \ \Delta = \operatorname{rad}([Q]) \ . \tag{5.26}$$

A sufficient condition for a full rank is

$$\sigma(|(Q_m^T Q_m)^{-1} Q_m^T| \Delta) < 1 \ , \tag{5.27}$$

where $\sigma(\cdot)$ is the spectral radius. For square systems this is equivalent to

$$(\mathfrak{I} - |(Q_m^T Q_m)^{-1} Q_m^T| \Delta)^{-1} \geq 0 \ . \tag{5.28}$$

Illustrative example:

Consider the nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 \ , \\ \dot{x}_2 &= \frac{1}{3} k_1 x_1^3 - k_2 x_1 x_2 \ . \end{aligned} \tag{5.29}$$

The measurement equation is given by

$$y = x_2 \ . \tag{5.30}$$

It is assumed that the parameters k_1 and k_2 are uncertain with $k_1 \in [1;2]$ and $k_2 \in [2;3]$. The observability map is

$$q(x) = \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{1}{3} k_1 x_1^3 - k_2 x_1 x_2 \end{bmatrix} \ . \tag{5.31}$$

The Jacobian of $Q(x)$ with respect to the state vector is given by:

$$\begin{aligned} \frac{\partial q}{\partial x} &= \begin{bmatrix} \frac{\partial q_1}{\partial x_1} & \frac{\partial q_1}{\partial x_2} \\ \frac{\partial q_2}{\partial x_1} & \frac{\partial q_2}{\partial x_2} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 \\ k_1 x_1^2 - k_2 x_2 & -k_2 x_2 \end{bmatrix} \ . \end{aligned} \tag{5.32}$$

The observability analysis is performed for $x_1 \in [-10;10]$ and $x_2 \in [-2;10]$. The Jacobian is invertible – and thus the system observable – for $0 \notin k_1 x_1^2 - k_2 x_2$. The initial box of the state variables, is subdivided into subboxes and the rank condition test is performed each subbox. If the test fails, the corresponding box is split further. The result is depicted in Fig. (5.5). The light gray boxes are boxes, where the Jacobian Q has rank n. For the dark gray boxes the rank condition (5.28) is not fulfilled. The black curves $x_2 = \frac{k_1 x_1^2}{k_2}$ and $x_2 = \frac{\overline{k_1} x_1^2}{\underline{k_2}}$ bound the region where the Jacobian Q does not fulfill the rank condition. Additionally to the interval vector $[x]$ enclosing the system state vector x also the intervals for the parameters k_1 and k_2 can be split, leading to an observability analysis of the extended state vector $z = [x_1, x_2, k_1, k_2]^T$.

In this example it was quite easy to perform an observability analysis for the whole domain. The observability analysis for large boxes can be performed offline, while smaller boxes can be checked for observability online. The computation time can be reduced by the application of automatic differentiation for the generation of the observability map.

Remark:
It is important to point out that even if not all state variables and parameters are observable, it is still possible to calculate guaranteed enclosure of those state variables by integration of the system model.

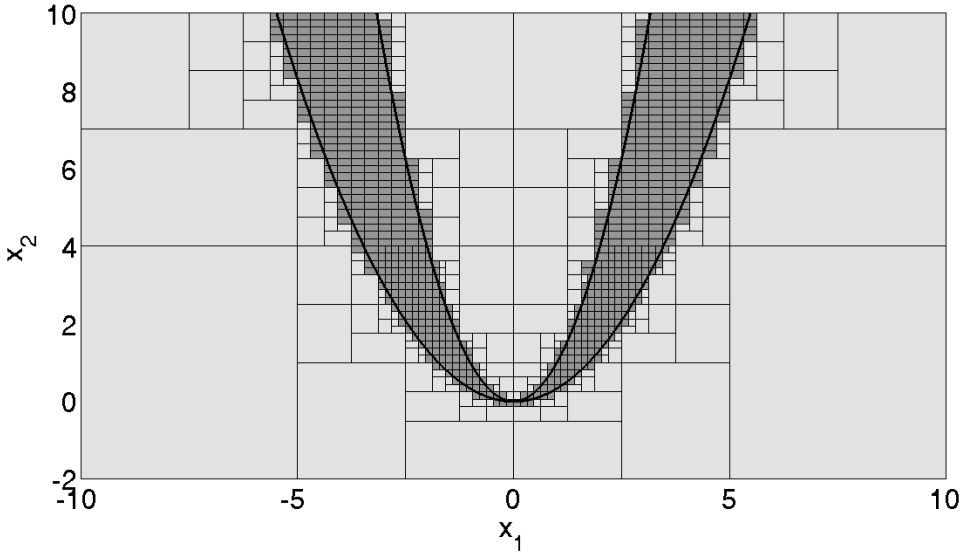


Figure 5.5: Result of the observability analysis.

5.3 Interval Observers

In the interval observer concept [26, 36] the extended state vector is enclosed by a list $\mathcal{Z}$ of interval vectors. At time-step k

$$z_k \in \mathcal{Z}_k = \left\{ [z_k^{(1)}], [z_k^{(2)}], \ldots, [z_k^{(L_k)}] \right\} \quad \text{with} \quad L_k \leq L_{max} \tag{5.33}$$

holds.

5.3.1 Prediction Step

In the prediction step a verified integration between two points of time when measurements are available is carried out. The prediction step from time-step k to time-step $k+1$ for the interval observer concept consists of the evaluation of (4.12).

The prediction step is illustrated in Fig. 5.6. At time step k the set is described by a list of interval vectors $\mathcal{Z}_k$. First the largest box (the box with the largest pseudo-volume) of the list $\mathcal{Z}_k$ is selected for splitting. Next a splitting criterion is evaluated and the selected box is split. This is repeated recursively until a maximum number of splittings or maximum number of interval vectors L_{max} is reached. Then for each box (4.12) is evaluated, with a combination of naive evaluation, midpoint-rule evaluation, monotonicity test, and iterative range computation (see Section 4.1).

If additional a coordinate transformation is applied (see Fig. 5.7), in parallel each box is evaluated by the corresponding method for the coordinate transformation. Here the Lohner-method described in Section 4.1.5 is used. The results are then intersected by the algorithm derived in Section 4.1.5. The state vector z_k is enclosed by

$$\begin{aligned} z_k \in \mathcal{Z}_k = &\left\{[z_k^{(1)}], [z_k^{(2)}], \ldots, [z_k^{(L_k)}]\right\} \\ &\cap \left\{\bar{z}_k^{(1)} + A_k^{(1)}[r_k^{(1)}], \bar{z}_k^{(2)} + A_k^{(2)}[r_k^{(2)}], \ldots, \bar{z}_k^{(L_k)} + A_k^{(L_k)}[r_k^{(L_k)}]\right\} \\ &\text{with} \quad L_k \leq L_{max}. \end{aligned} \tag{5.34}$$

The list $\mathcal{Z}_k$ then contains in addition to the interval vector enclosures $[z_k^{(l)}]$ the information for the enclosures by the parallelepipeds namely $\bar{z}_k^{(l)}$, $A_k^{(l)}$ and $[r_k^{(l)}]$. In Fig. 5.8 an interval $[z_k^{(l)}]$ and the corresponding parallelepipeds $\bar{z}_k^{(l)} + A_k^{(l)}[r_k^{(l)}]$ are shown. Instead of splitting the box $[z_k^{(l)}]$ also the parallelepipeds $\bar{z}_k^{(l)} + A_k^{(l)}[r_k^{(l)}]$ can be split by splitting of $[r_k^{(l)}]$ as described in Section 4.1.7.

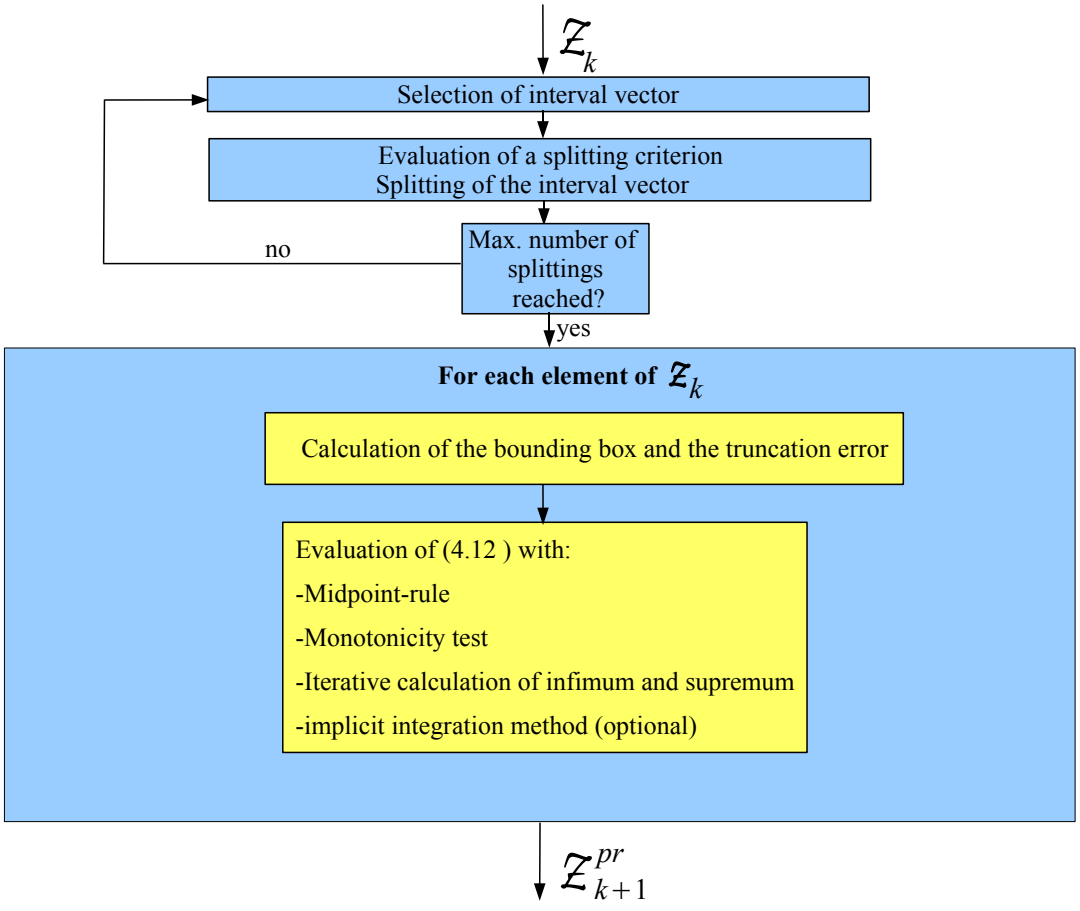


Figure 5.6: Flow diagram.

At the end of the prediction step the predicted set $\mathcal{Z}_{k+1}^{pr}$ is obtained:

$$\mathcal{Z}_{k+1}^{pr} = \left\{[z_{k+1}^{(1)}], [z_{k+1}^{(2)}], \ldots, [z_{k+1}^{(L_{k+1}^{pr})}]\right\} \quad \text{with} \quad L_{k+1}^{pr} \leq L_{max} \ , \tag{5.35}$$

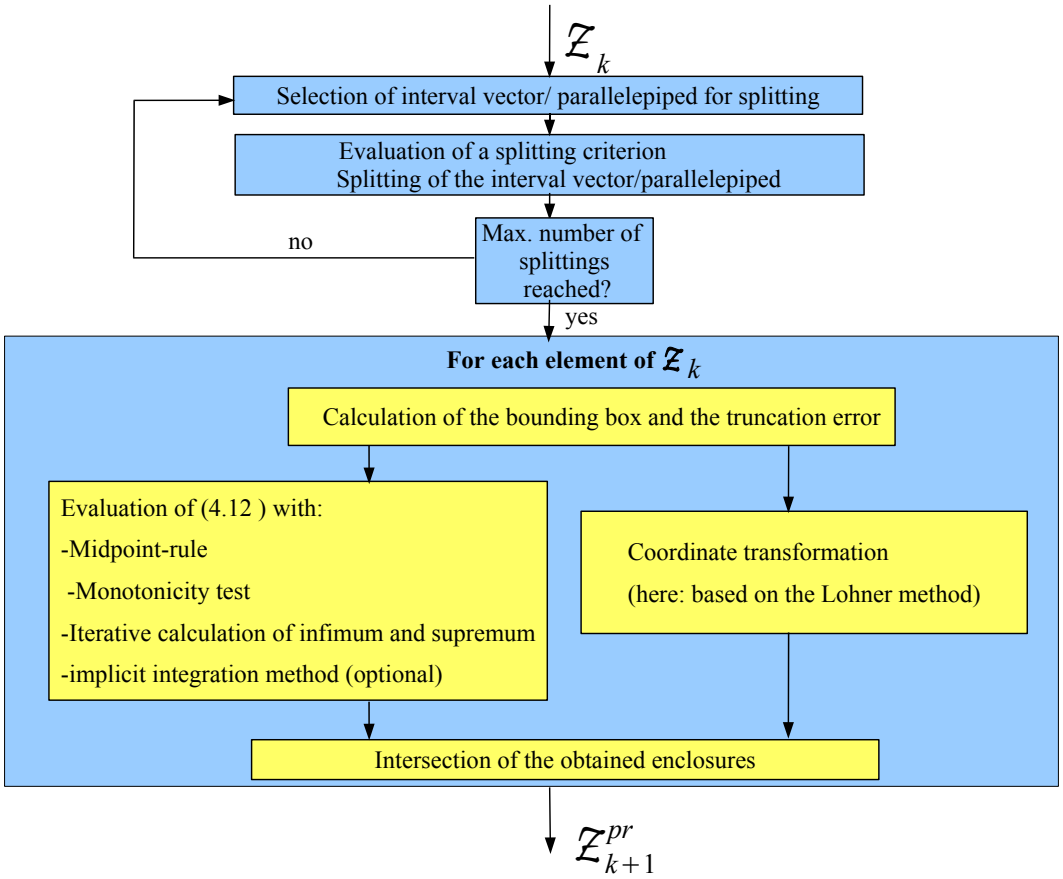


Figure 5.7: Flow diagram with coordinate transformation.

if no coordinate transformation is employed and

$$\begin{aligned}\mathcal{Z}^{pr}_{k+1} = &\left\{[z^{(pr,1)}_{k+1}], [z^{(pr,2)}_{k+1}], \ldots, [z^{(pr,L^{pr}_{k+1})}_{k+1}]\right\} \\ &\cap \left\{\bar{z}^{(pr,1)}_{k+1} + A^{(pr,1)}_{k+1}[r^{(pr,1)}_{k+1}], \bar{z}^{(pr,2)}_{k+1} + A^{(pr,2)}_{k+1}[r^{(pr,2)}_{k+1}], \ldots, \right. \\ &\left. \bar{z}^{(pr,L^{pr}_{k+1})}_{k+1} + A^{(pr,L^{pr}_{k+1})}_{k+1}[r^{(pr,L^{pr}_{k+1})}_{k+1}]\right\} \\ &\text{with} \quad L^{pr}_{k+1} \leq L_{max} \ ,\end{aligned} \tag{5.36}$$

if additionally a coordination transformation is applied.

5.3.2 Correction Step

The correction step eliminates regions which are inconsistent with the measured data and the measurement equation. The initial enclosure for the correction step is the predicted set.

First for each $[z^{(pr,l)}_{k+1}]$ of $\mathcal{Z}^{pr}_{k+1}$ the expression $\check{h}(\cdot)$ in (5.6) is evaluated leading to

$$[\check{y}^{(l)}_{k+1}] = [\check{h}]([z^{(pr,l)}_{k+1}], [q_{k+1}]) \ . \tag{5.37}$$

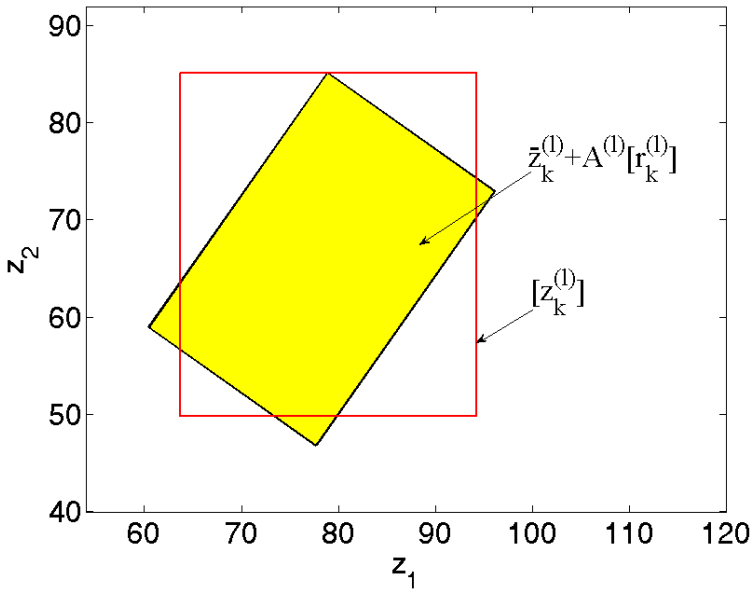


Figure 5.8: Enclosure of the extended state vector z by an interval and a parallelepiped.

Now the interval vectors $[z_{k+1}^{(pr,l)}]$ can be deleted if

$$[\check{y}_{k+1}^{(l)}] \cap y_{k+1} - [\delta_{k+1}] = \emptyset \tag{5.38}$$

holds.

If

$$[\check{y}_{k+1}^{(l)}] \subset y_{k+1} - [\delta_{k+1}] \ , \tag{5.39}$$

the predicted set is consistent with the measurement model and the measured values and no further calculations are done.

For the remaining interval vectors the measurement equation (5.6) is inverted for each interval $[z_{k+1}^{(pr,l)}]$ according to

$$z_{k+1} = h^{-1}(y_{k+1}, q_{k+1}, \delta_{k+1}) \quad \text{with} \quad z_{k+1} \in [z_{k+1}^{(pr,l)}], \quad q_{k+1} \in [q_{k+1}], \quad \delta_{k+1} \in [\delta_{k+1}] \tag{5.40}$$

in order to reconstruct the extended state variables. In the case of a nonlinear system or a nonlinear measurement equation this is done by application of an interval Newton method to

$$0 = h(z_{k+1}, q_{k+1}, \delta_{k+1}) - y_{k+1} \quad \text{with} \quad z_{k+1} \in [z_{k+1}^{(pr,l)}], \quad q_{k+1} \in [q_{k+1}], \quad \delta_{k+1} \in [\delta_{k+1}] \tag{5.41}$$

for each remaining box $[z_{k+1}^{(pr,l)}]$. In this work, the Krawczyk method has been used. The resulting box $[z_{k+1}^{(c,l)}]$ encloses the complete range of zeros of (5.41).

For $m < n$ the inversion of the measurement equation (5.6) is under determined. Thus, direct inversion with the considered interval Newton method is not possible. One solution

is to solve the equation only for m state variables and considering the remaining $n-m$ variables as constant intervals [36]. And repeat this iteratively. If the measurement equation is for example

$$y_{k+1} = h(z_{1,k+1}, z_{2,k+1}, \delta_{k+1}) \tag{5.42}$$

with $m = 1$ and $n = 2$, then it can be solved for instance for $z_{1,k+1}$ whereas $z_{2,k+1}$ is considered as a constant interval value. Then the equation is solved for $z_{2,k+1}$ with the new $z_{1,k+1}$ as constant interval value.

For illustration, consider the measurement equation

$$y_{k+1} = z_{1,k+1} + z_{2,k+1} + \delta_{k+1} \ . \tag{5.43}$$

Let the interval vector $[z^{pr}_{k+1}] = [[z^{pr}_{1,k+1}], [z^{pr}_{2,k+1}]]^T$ with $[z^{pr}_{1,k+1}] = [0.5; 2.5]$, and $[z^{pr}_{2,k+1}] = [-1; -3]$ be the predicted set, and $y_{k+1} = 1$ the measured value, and $[\delta_{k+1}] = [-0.5; 0.5]$ the measurement uncertainty. Now, equation (5.42) is first solved for $z_{1,k+1}$ resulting into

$$[z^c_{1,k+1}] = y_k - [\delta_{k+1}] - [z^{pr}_{2,k+1}] = [0.5; 1.5] - [-1; -3] = [1.5; 4.5] \ . \tag{5.44}$$

As the example is linear no interval Newton method is required. Now this result has to be intersected with $[z^{pr}_{1,k+1}]$ resulting into

$$[z_{1,k+1}] = [z^c_{1,k+1}] \cap [z^{pr}_{1,k+1}] = [1.5; 4.5] \cap [0.5; 2.5] = [1.5; 2.5] \ . \tag{5.45}$$

Next, the (5.42) is solved for $z_{2,k+1}$ with the new $[z_{1,k+1}]$:

$$[z^c_{2,k+1}] = y_k - [\delta_{k+1}] - [z_{1,k+1}] = [0.5; 1.5] - [1.5; 2.5] = [-2; 0] \tag{5.46}$$

and intersection with $[z^{pr}_{2,k+1}]$ yields in

$$[z_{2,k+1}] = [z^c_{2,k+1}] \cap [z^{pr}_{2,k+1}] = [-2; 0] \cap [-3; -1] = [-2; -1] \ . \tag{5.47}$$

This is also illustrated in Fig. (5.9).

The second possibility is to obtain the missing equations from former measurements [36]. The right hand sides of the measurement equations are expressed in terms of z_{k+1}. This is done by backward integration of the system from time-step $k+1$ to $k-N$

$$\begin{aligned}
y_k &= h(g^-_k(z_{k+1}) + e^-_{k+1}, q_k, \delta_k) \\
y_{k-1} &= h(g^-_{k-1}(g^-_k(z_{k+1}) + e^-_{k+1}) + e^-_k, q_{k-1}, \delta_{k-1}) \\
y_{k-2} &= h(g^-_{k-2}(g^-_{k-1}(g^-_k(z_{k+1}) + e^-_{k+1}) + e^-_k) + e^-_{k-1}, q_{k-2}, \delta_{k-2}) \\
&\vdots \\
y_{k-N} &= h(g^-_{k-N}(\dots(g^-_{k-2}(g^-_{k-1}(g^-_k(z_{k+1}) + e^-_{k+1}) + e^-_k) \\
&\quad + e^-_{k-1})\dots) + e^-_{k-N}, q_{k-N}, \delta_{k-N}) \ ,
\end{aligned} \tag{5.48}$$

where the $^-$ in g^- indicates the direction of the integration. Note, that the bounding boxes $[B_{k+1}] \dots [B_{k-N}]$ do not have to be recomputed, because they are known from the forward integration steps. For $m = 1$ and $n = 2$, one backward integration has to be performed to obtain the missing second equation. The obtained system of equations can be solved

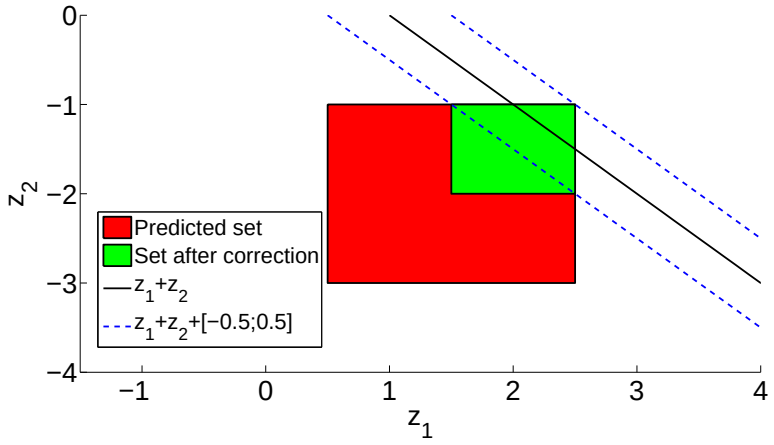


Figure 5.9: Example for correction step.

for both state variables. The quality of the correction step depends on the quality of the prediction as well as on the measurement error and the time span between two consecutive measurements. If the truncation error is very large, the additional information obtained from backward integration step may be poor, however, the result is still guaranteed.

Note, that if the measurement equation only depends on $\tilde{n} < n$ variables, which is often the case in practice, it will also only be solved for these variables.Then even if $m < n$, the measurement equation can be directly be solved, if $m \geq \tilde{n}$.

The initial extended state for the interval Newton method is given by the prediction step. Thereby the intersection is performed already within the inversion. If (5.41) yields no solution, the corresponding interval $[z_{k+1}^{(pr,l)}]$ is deleted.

The combination of prediction and correction step yields

$$[z_{k+1}^{(l)}] = [z_{k+1}^{(pr,l)}] \cap [z_{k+1}^{(c,l)}] \ . \tag{5.49}$$

The list of interval vectors and the extended state vector enclosure after the correction step is given by:

$$z_{k+1} \in \mathcal{Z}_{k+1} = \left\{ [z_{k+1}^{(1)}], [z_{k+1}^{(2)}], \ldots, [z_{k+1}^{(L_{k+1})}] \right\} \quad \text{with} \quad L_{k+1} \leq L_{k+1}^{pr} \leq L_{max}. \tag{5.50}$$

or

$$\begin{aligned} \mathcal{Z}_{k+1} = & \left\{ [z_{k+1}^{(1)}], [z_{k+1}^{(2)}], \ldots, [z_{k+1}^{(L_{k+1})}] \right\} \\ & \cap \left\{ \bar{z}_{k+1}^{(1)} + A_{k+1}^{(1)}[r_{k+1}^{(1)}], \bar{z}_{k+1}^{(2)} + A_{k+1}^{(2)}[r_{k+1}^{(2)}], \ldots, \right. \\ & \left. \bar{z}_{k+1}^{(L_{k+1})} + A_{k+1}^{(L_{k+1})}[r_{k+1}^{(L_{k+1})}] \right\} \\ & \text{with} \quad L_{k+1} \leq L_{max} \ . \end{aligned} \tag{5.51}$$

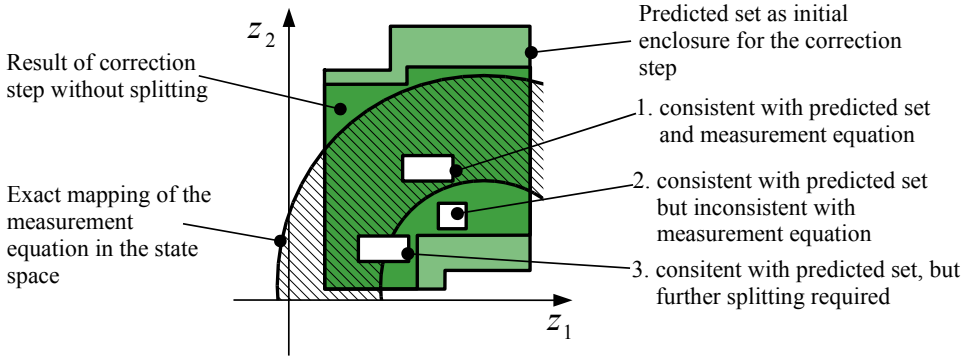


Figure 5.10: Consistency test.

For further improvement of the correction step consistency tests [26, 31, 32, 36] are applied for each $[z_{k+1}^{(pr,l)}]$. Therefore $\check{h}(\cdot)$ in (5.6) is evaluated for subboxes $[\tilde{z}_{k+1}^{(pr,l)}]$ of $[z_{k+1}^{(pr,l)}]$, resulting in

$$[\tilde{\tilde{y}}_{k+1}] = \left[\check{h}\right]\left([\tilde{z}_{k+1}^{(pr,l)}], [q_{k+1}]\right) . \tag{5.52}$$

The resulting interval vectors $[\tilde{\tilde{y}}_{k+1}]$ are checked for consistency with the actual measurements y_{k+1} and the measurement error.

Three cases have to be distinguished:

- interval vectors for which $[\tilde{\tilde{y}}_{k+1}] \subseteq y_{k+1} - [\delta_{k+1}]$ holds, are consistent and do not have to be split further.
- interval vectors for which $[\tilde{\tilde{y}}_{k+1}] \cap y_{k+1} - [\delta_{k+1}] = \emptyset$ holds, can be deleted because they are inconsistent with the measurements y_{k+1} and the measurement error $[\delta_{k+1}]$.
- Remaining interval vectors are split further and (5.52) is applied. These two steps are repeated recursively.

The three cases are also illustrated in Fig. 5.10.

The consistency test can be combined with the evaluation of (5.41) by the interval Newton method. In that case, before the consistency test, first the interval Newton method is applied. During the consistency test for each remaining interval vector (5.41) can be solved additionally for further improvement of the enclosures.

After the correction steps a merging routine is applied to reduce the number of interval vectors. Interval vectors which have been split during the correction step are stored in a tree structure to simplify the merging (see also Section 4.2). Then, the merging routine described in Section 4.1.6 is applied for further reduction of the number of interval vectors. Reapproximation by disjoint interval vectors is performed between a user defined number of time-steps.

Illustrative Example: Localization of a parachutist

A parachutist of mass m jumps out of an aircraft and has to be localized by a verified state estimation. For illustrative purposes, the localization problem is simplified to the localization in a plane. The equation of motion in y-direction is given by

$$m\ddot{y}(t) = F_G - F_R = mg - k\dot{y}^2 \ , \tag{5.53}$$

F_G denotes the gravity force and F_R the air resistance. The friction constant is given by k and $g = 9.81\text{m/s}^2$ is the gravitation constant. The equation is rewritten according to

$$\ddot{y}(t) = g - \frac{k}{m}\dot{y}^2 = g - b\dot{y}^2 \ , \tag{5.54}$$

where b is the uncertain ballistic coefficient with $b \in [0.0029; 0.0031]$. The equation of motion in x-direction is given by

$$\dot{x} = v_x \ , \tag{5.55}$$

with $v_x \in [100; 105]\text{m/s}$. The initial conditions for the position in y and x were $y(0) \in [4990; 5010]\text{m}$ and $x(0) \in [545; 550]\text{m}$. The initial speed in y direction was $\dot{y} = [-10; 10]\text{m/s}$. These two equations are rewritten in state-space representation according to

$$\begin{aligned} \dot{z}_1(t) &= -z_2(t) \ , \\ \dot{z}_2(t) &= g - z_4 z_2^2 \ , \\ \dot{z}_3(t) &= v_x \ , \\ \dot{z}_4(t) &= 0 \end{aligned} \tag{5.56}$$

with $z_1 = y$, $z_2 = \dot{y}$, $z_3 = x$ and $z_4 = b$. For the prediction step, a Taylor series expansion of order $\nu = 4$ was employed. The parachute is opened after a free fall of $t = 70s$. It is assumed that the ballistic coefficient b changes to $b \in [0.29, 0.31]$ instantaneously. No interval splitting was performed in the prediction and correction steps. First two radar stations are available and two measurement equations are obtained. It is assumed that the distance of the parachutist from the radar stations

$$\begin{aligned} r_{1,k+1} &= \sqrt{(z_{1,k+1} - l_1)^2 + (z_{3,k+1} - h_1)^2} + \delta_{1,k+1} \ , \\ r_{2,k+1} &= \sqrt{(z_{1,k+1} - l_2)^2 + (z_{3,k+1} - h_2)^2} + \delta_{2,k+1} \end{aligned} \tag{5.57}$$

is measured, where $l_1 = 5,000\text{m}$, $l_2 = 50,000\text{m}$ and $h_1 = 1,000\text{m}$, $h_2 = 1,000\text{m}$ define the positions of the radar stations in the $z_3 = x$ and $z_1 = y$ direction respectively. The measurement uncertainties are assumed to be $\delta_1 \in [-3; 3]\text{m}$ and $\delta_2 \in [-3; 3]\text{m}$. At time-step k first a prediction step is calculated. Then the result is used as initial box for the inversion of the measurement equations by an interval Newton method. The results are depicted in Fig. 5.11. The integration step-size is $T = 0.5$ s for $t < 80$ s and $T = 0.01$ s for $t \geq 80$ s. It is assumed that measured values are available at every 0.5 s. If only one radar station at $l_1 = 5,000\text{m}$ and $h_1 = 1,000\text{m}$ is available for $t < 80$ s the measurements from the previous time-step are taken into account and express the state at time-step $k-1$ by the state at time-step k through backward calculation. For $t \geq 80$ s the measurement equation is first solved only for $z_1 = y$ and then for $z_3 = x$ without considering previous measurements as the

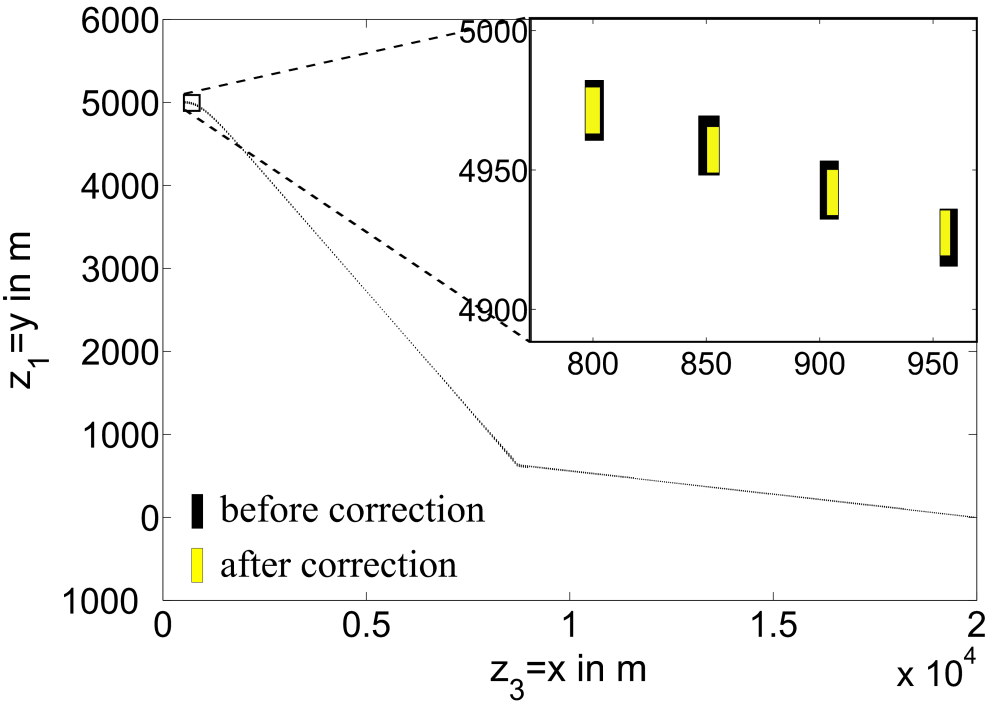


Figure 5.11: Results for 2 radar stations.

step-size is then reduced to 0.01 s and therefore 50 backward steps would be required. The result is shown in Fig. 5.12. It can be seen that the measurements are only able to improve the predictions in two regions. The first region is when the parachutist is directly above the radar station, the second when radar station and parachutist are approximately on the same height. In the first case, the box is tightened in z_1 direction in the second case it is tightened in z_3-direction. The reason is that only in these two cases two subsequent measurements are different enough.

5.3.3 Applications

In this Section results of a verified state and parameter estimation for four applications from Chapter 2 are presented. For all applications iterative calculation of infimum and supremum in combination with naive interval evaluation, mean-value evaluation, and monotonicity test was applied in the prediction step (see Sections 4.1.1–4.1.3). However, the monotonicity test and iterative range computation was limited only up to the order 1 of the Taylor series expansion. In each time-step, the maximum number of splittings during the iterative computation was 10 for each box of the list $\mathcal{Z}$. The hull limit was $\delta_{hull,limit} = 2.5\%$ for all applications. The measured values were generated by a verified integration of the corresponding systems with nominal values for the initial state variables and uncertain parameters and a subsequent addition of an uniformly distributed noise vector bounded by the interval vector of the measurement uncertainty. The nominal values of the initial states were if not stated otherwise given by the midpoints of the corresponding interval uncertainty.

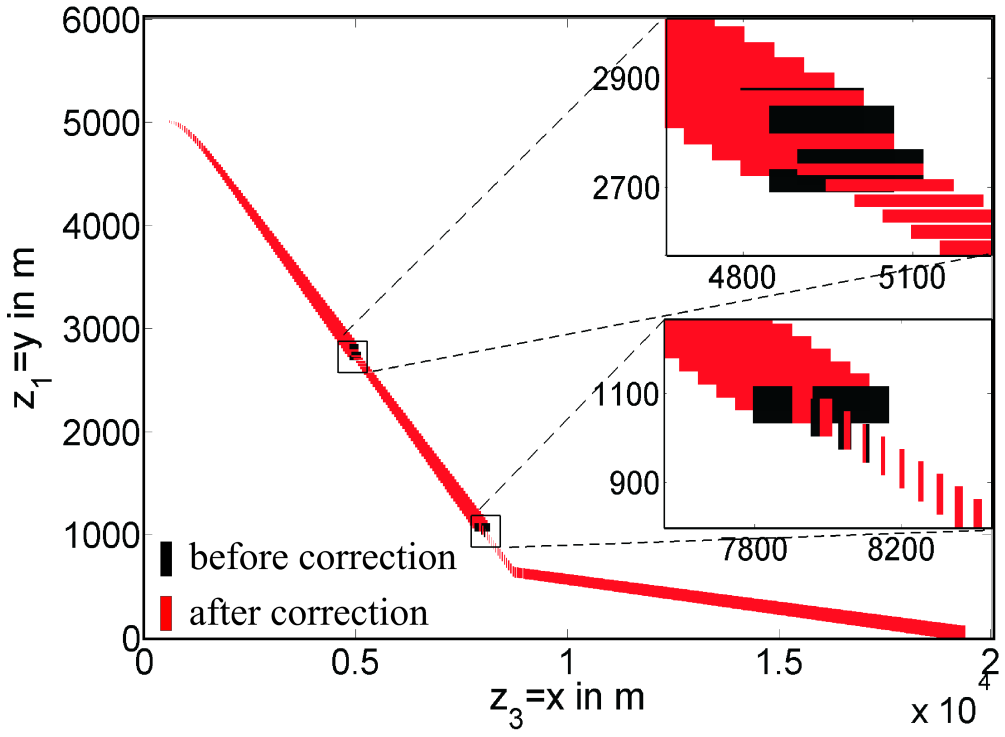


Figure 5.12: Results for 1 radar station.

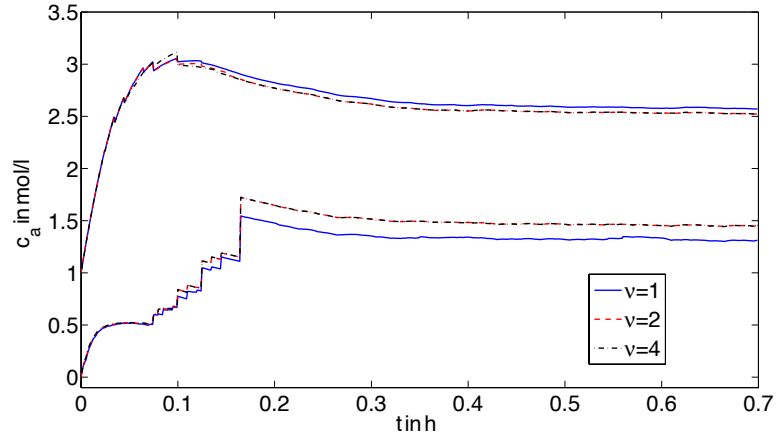
Non-Isothermal Stirred Tank Reactor

Consider the NISTR from Section 2.1. The initial conditions for the estimation were $c_a(0) \in [0;1]$mol/l, $c_b(0) \in [0;1]$mol/l, $v(0) \in [100;110]°$C, $v_K(0) \in [100;110]°$C. The parameters E_1 and E_3 were assumed to be uncertain and time-invariant, with $E_1 \in [0.9;1.1] \cdot 9758.3$ K and $E_3 \in [0.9;1.1] \cdot 8560$ K. The measurement equation is given by [13]

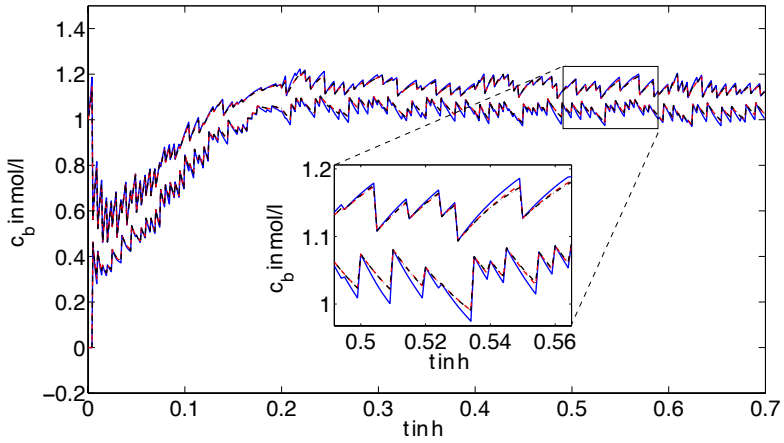
$$\begin{aligned} y_{1,k+1} &= c_{b,k+1} + \delta_{1,k+1} \ , \\ y_{2,k+1} &= v_{k+1} + \delta_{2,k+1} \ . \end{aligned} \tag{5.58}$$

The step size was $h = 0.001$h and the measurements were available every 0.005h, hence $\Delta T_m = 0.005$h. The measurement errors were given by $\delta_{1,k+1} \in [\delta_1] = [-0.05;0.05]$mol/l and $\delta_{2,k+1} \in [\delta_2] = [-0.5;0.5]°$C. The measured values were generated on the basis of the midpoints of the corresponding uncertainty. First three estimations with different orders ν of the Taylor expansion were carried out. The maximum number of interval vectors was chosen to be 200 and the maximum number of splittings was 100. Splitting and merging was carried out every 5 time-steps. Reapproximation by 200 disjoint interval vectors was employed every 200 time-steps. The results for the orders $\nu = 1$, $\nu = 2$, $\nu = 4$ are depicted in Figs. 5.13 and 5.14; the computation times are listed in Tab. 5.1. Increasing the order from $\nu = 1$ to $\nu = 2$ leads to improved enclosures especially for c_a, E_1, and E_3. If the order is further increased to $\nu = 4$ the improvement is negligible since the computation time is much larger. In Figs. 5.13(b), 5.13(c), and 5.14(a) some time intervals are depicted additionally in enlarged form. In Fig. 5.13(c) $\nu = 4$ yields the tightest results for the left enlarged time interval. For the right enlarged time interval, however, the results for $\nu = 2$ are better than

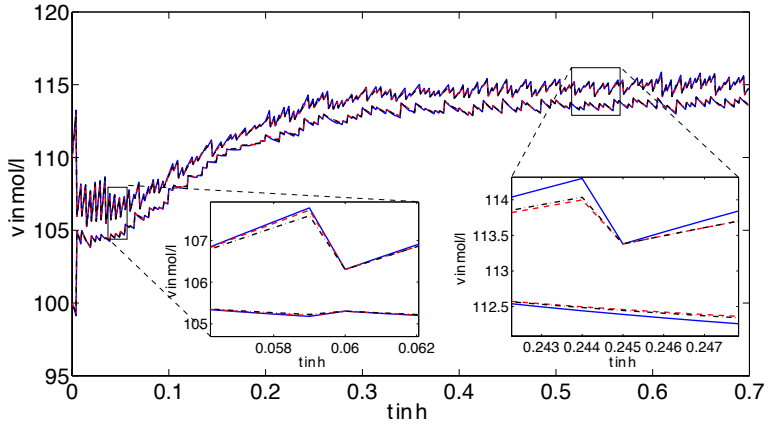
for $\nu = 4$. The main goal of increasing the order is to reduce the size of the remainder error. But with increasing order also more expressions have to be evaluated with interval methods. This may lead to more overestimation especially when the evaluation has to be done for large intervals. At least it has the effect that in the case of the NISTR an increase above order $\nu = 2$ is not improving the enclosures. In Figs. 5.15 and 5.16 results for different numbers of splitting operations and different numbers of maximum allowed interval vectors L_{max} are shown. The order was $\nu = 1$ in all cases. As expected, increasing both numbers leads to tighter results at cost of more computational effort (see Tab. 5.2). The diameters of the enclosures decrease drastically when increasing $L_{max} = 200$ to $L_{max} = 500$. However, the improvement obtained when raising $L_{max} = 500$ to $L_{max} = 1000$ is not so strong.



(a) Cyclopentadien concentration c_a.

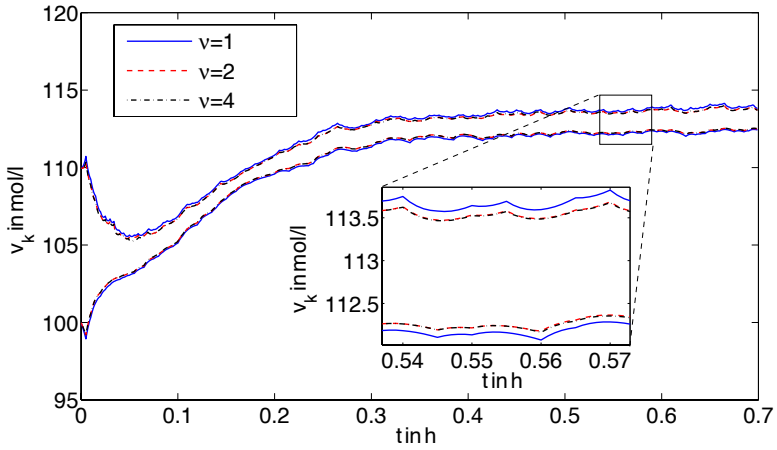


(b) Cyclopentenol concentration c_b.

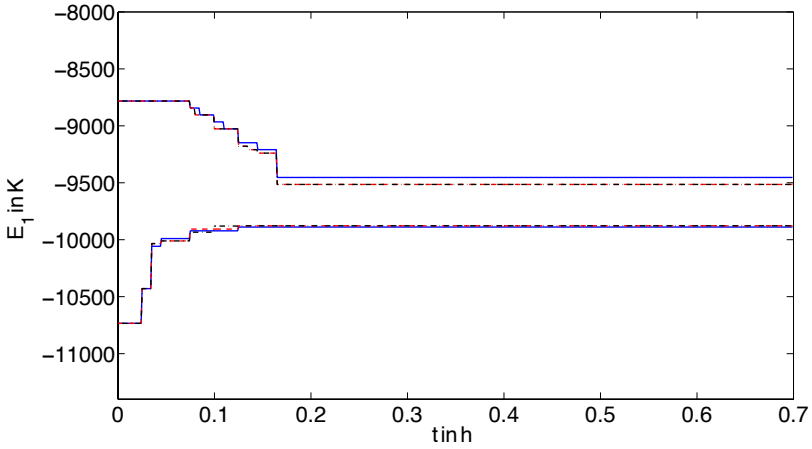


(c) Reactor temperature v

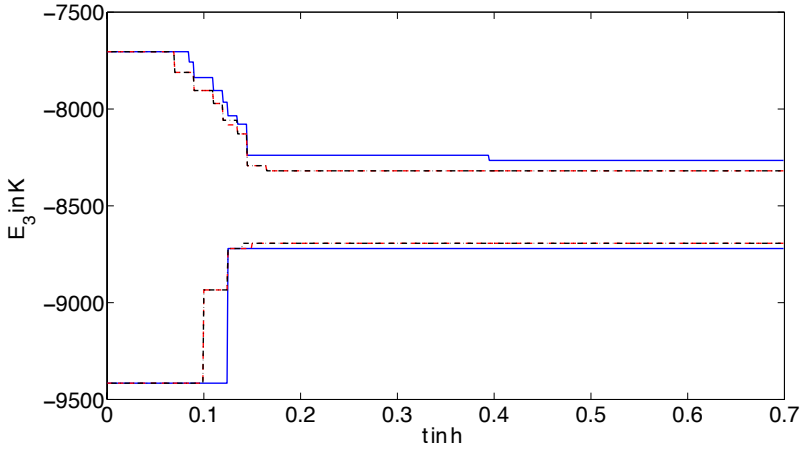
Figure 5.13: NISTR: State enclosures of c_a, c_b, and v for different orders ν of the Taylor series expansion.



(a) Jacket temperature v.



(b) Uncertain parameter E_1.



(c) Uncertain parameter E_3.

Figure 5.14: NISTR: State enclosures of v_k, E_1, and E_3 for different orders ν of the Taylor series expansion.

Table 5.1: NISTR: Comparison of the computation time for different orders ν.

Order ν of the Taylor sereis expansion	Computation time in s
1	1100
2	1781
4	2265

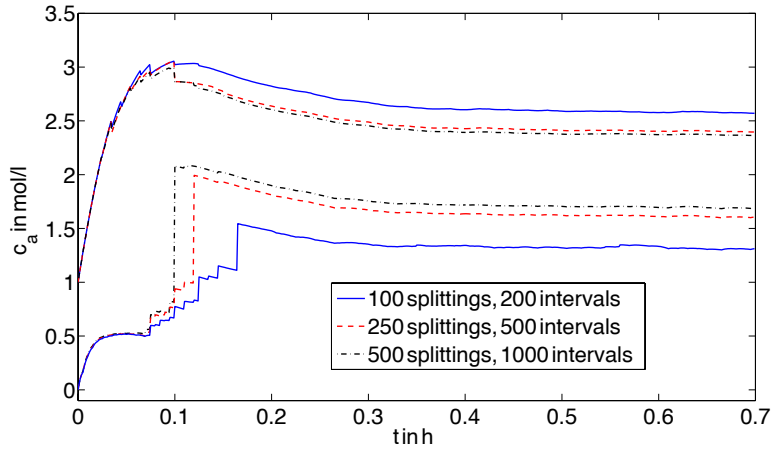
Table 5.2: NISTR: Estimation for different numbers of splittings and interval vectors.

Number of max. splittings	Max. number L_{max} of interval vectors	Computation time in s
100	200	1100
250	500	2863
500	1000	6304

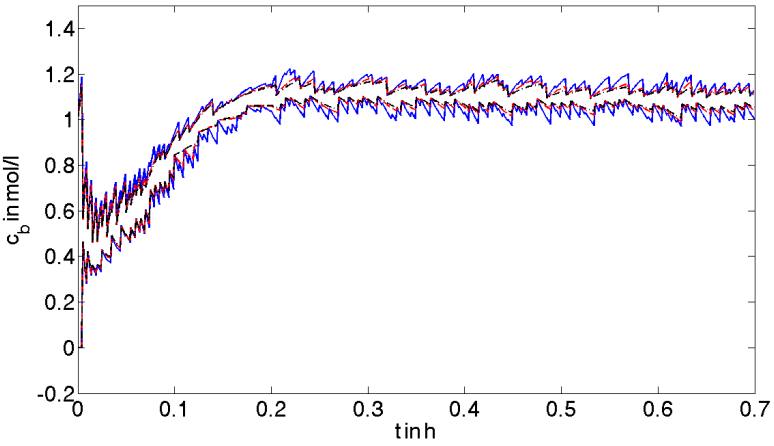
Fig. 5.17 and Fig. 5.18 compare between an estimation with interval splitting and without coordinate transformation, an estimation with interval splitting and with coordinate transformation (Lohner's method with QR preconditioning), and an estimation with coordinate transformation and splitting of the parallelepipeds instead of the interval vectors. The order of the Taylor expansion in time was $\nu = 2$ for all estimations. The maximum number of splittings and interval vectors was 100 and 200, respectively. In comparison to the estimation without coordinate transformation, a combination with the coordinate transformation according to the algorithms described in Section 4.1.5 and 4.1.7 leads to improved enclosure but the computation time was longer (see Tab.5.3). Except of the first 0.1 h of $c_a(t)$ the enclosures for splitting of the parallelepipeds are in average even tighter. Though an estimation with 250 splittings and 500 interval vectors which is not depicted here showed no significant difference between the results.
The algorithm for the coordinate transformation is based on the mean-value evaluation and requires an interval evaluation of the Jacobian. Therefore the interval enclosure obtained by an iterative range computation of 4.12 is often better than the parallelepiped enclosure obtained from the coordinate transformation. In this case the intersection of both results does not improve the enclosure. In future work this fact has to be considered by an improved evaluation of the Jacobian.

Table 5.3: NISTR: Comparison of the computation time for different orders ν.

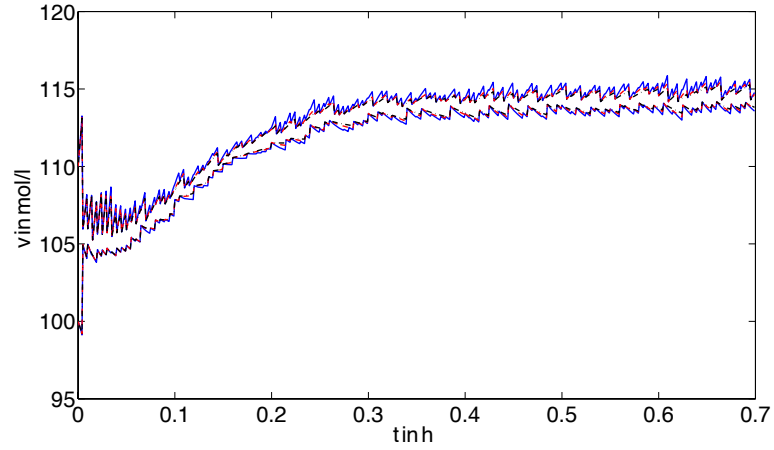
Enclosure type and splitting method.	
Interval splitting	1781
Interval splitting in combination with coordinate transformation	2325
Parallelepiped splitting	2201



(a) Cyclopentadiene concentration c_a.

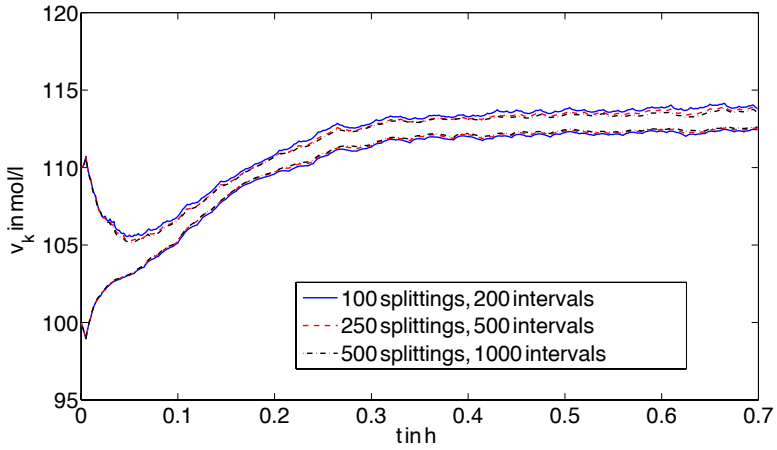


(b) Cyclopentenol concentration c_b.

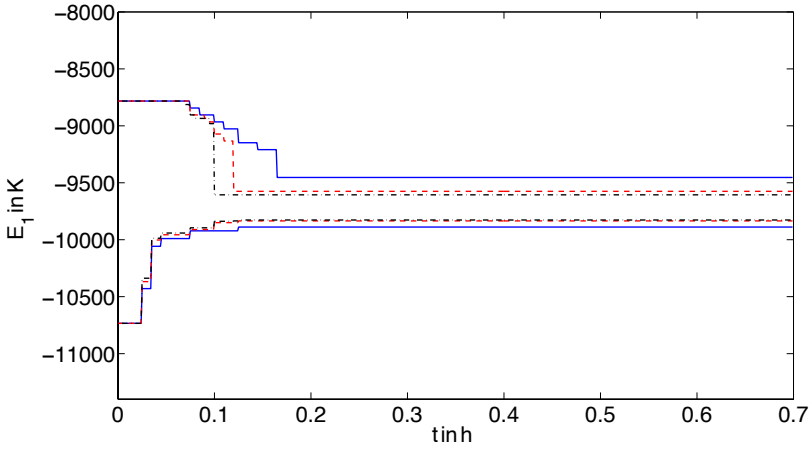


(c) Reactor temperature v

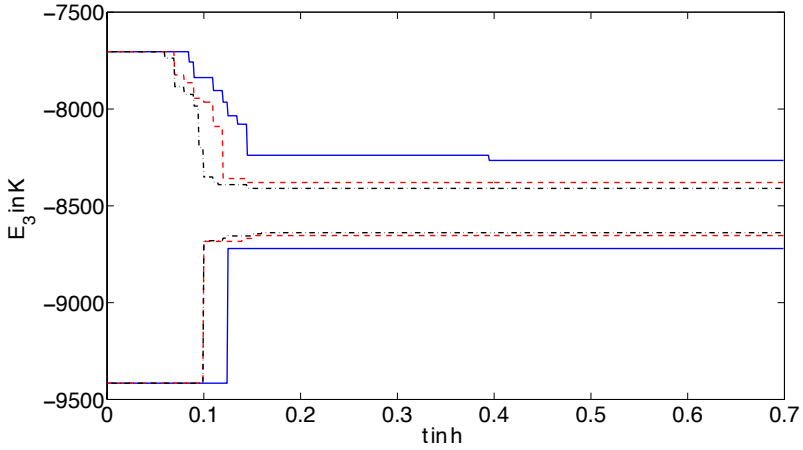
Figure 5.15: NISTR: State enclosures of c_a, c_b, and v for different numbers of splittings and interval vectors.



(a) Jacket temperature v.

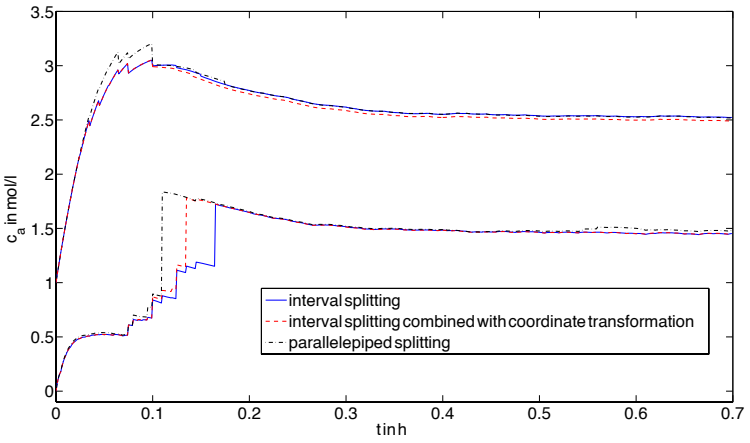


(b) Uncertain parameter E_1.

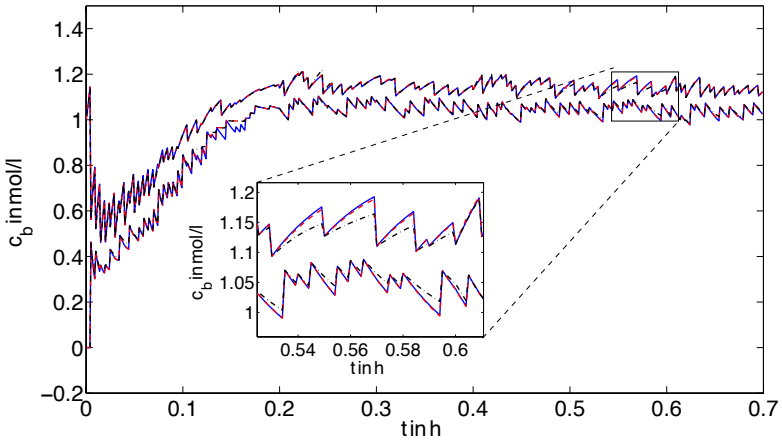


(c) Uncertain parameter E_3.

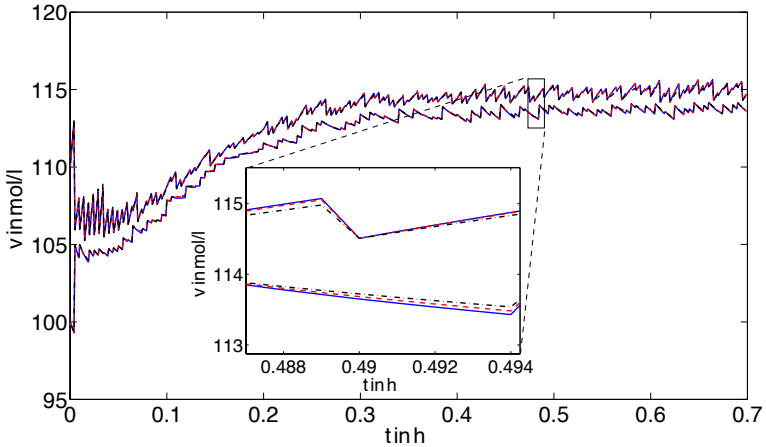
Figure 5.16: NISTR: State enclosures of v_k, E_1, and E_3 for different numbers of splittings and interval vectors.



(a) Cyclopentadiene concentration c_a.

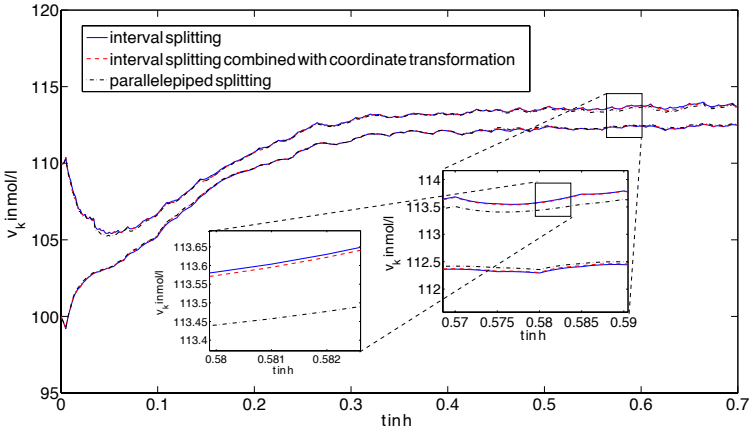


(b) Cyclopentenol concentration c_b.

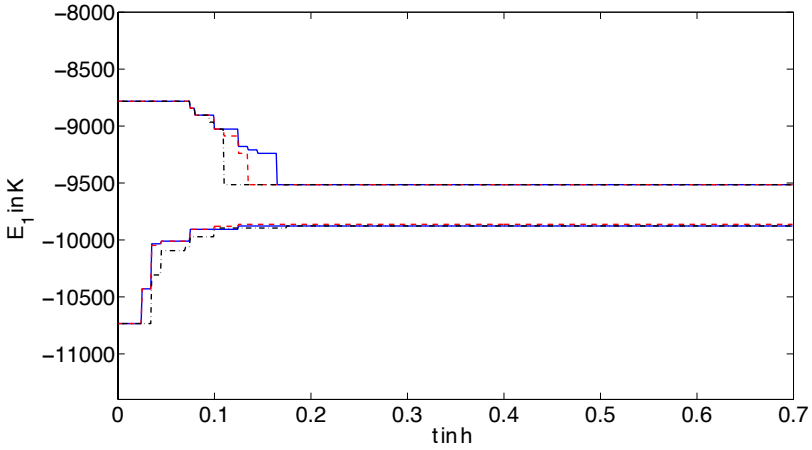


(c) Reactor temperature v

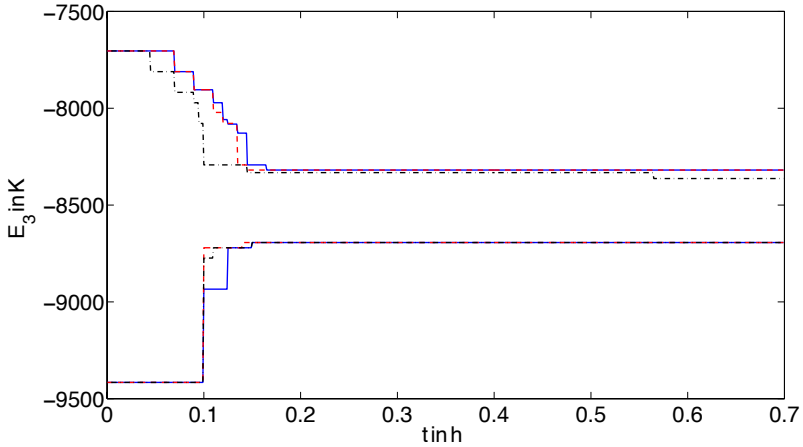
Figure 5.17: NISTR: Comparison of the state enclosures of c_a, c_b, and v with and without coordinate transformation.



(a) Jacket temperature v.



(b) Uncertain parameter E_1.



(c) Uncertain parameter E_3.

Figure 5.18: NISTR: Comparison of the state enclosures of v_K, E_1, and E_3 with and without coordinate transformation.

Double Pendulum

Consider the double pendulum, which is described in Section 2.3. According to (2.26) the measurement equations including the measurement errors are given by

$$\begin{aligned} r_{1,k+1} =& \sqrt{(l_1 \sin(\theta_{1,k+1}) - x_0)^2 + (-l_1 \cos(\theta_{1,k+1}) - y_0)^2} + \delta_{1,k+1} \ , \\ r_{2,k+1} =& \sqrt{(l_1 \sin(\theta_{1,k+1}) + l_2 \sin(\theta_{2,k+1}) - x_0)^2 + (-l_1 \cos(\theta_{1,k+1}) - l_2 \cos(\theta_{2,k+1}) - y_0)^2} + \delta_{2,k+1} \ . \end{aligned} \tag{5.59}$$

For the estimation the system parameters were given by $l_1 = l_2 = 1\text{m}, m_1 = m_2 = 1\text{kg}, x_0 = 3\text{m}, y_0 = 0.2\text{m}$. The initial conditions were assumed to be $\theta_1(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]\text{rad}$, $\theta_2(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]\text{rad}$, $\dot{\theta}_1(0) \in [-0.01; 0.01]\text{rad/s}$, $\dot{\theta}_2(0) \in [-0.01; 0.01]\text{rad/s}$. For the prediction step a maximum number of 50 splittings and 100 interval vectors were allowed, the order was set to $\nu = 1$. The assumed measurement errors were $[\delta_1] = [\delta_2] = [-0.01; 0.01]\text{m}$. The time between two measurements was $\Delta T_m = 0.005\text{s}$. The measured values were generated on the basis of the midpoints of the corresponding uncertainty. In the correction step only the interval Newton method was applied, since a estimation with additional 500 splittings in the correction step did not improve the results significantly. Splitting and merging was carried out every 5 time-steps. Reapproximation by 100 disjoint interval vectors was employed every 200 time-steps. The results are shown in Fig. 5.19 and 5.20. Estimations with constant step sizes $h = 0.001\text{s}$, $h = 0.0025\text{s}$, and $h = 0.005\text{s}$ have been carried out. With increasing step size the enclosures become wider; especially if the step size is increased to $h = 0.005\text{s}$. However, the computation time is decreased significantly (see Tab. 5.4).

Table 5.4: Double pendulum: Comparison of the computation time for different step-size.

Step-size h in s	Computation time in s
0.001	3014
0.0025	1215
0.005	661

Biological Waste Water Treatment Plant

The third application is the BWTP, described in Section 2.4. The initial state variables were given by $S(0) \in [0.9; 1.1] \cdot 0.616\ \text{kg/m}^3$, $X(0) \in [0.97; 1.03] \cdot 0.1\ \text{kg/m}^3$, $S_O(0) \in [0.99; 1.01] \cdot S_{O,sat}/2 \cdot 10^{-4}\ \text{kg/m}^3$ and $X_{Set}(0) \in [0.9; 1.1] \cdot 0.001\ \text{kg/m}^3$. The maximum specific growth rate $\hat{\mu}_H$ was assumed to be uncertain and time-invariant with $\hat{\mu}_H \in [0.8; 1.2] \cdot 1/14400\ 1/\text{s}$. For the measurement equation

$$\begin{aligned} y_{1,k+1} =& S_{k+1} + \delta_{1,k+1} \ , \\ y_{2,k+1} =& S_{O,k+1} + \delta_{2,k+1} \end{aligned} \tag{5.60}$$

holds, with $\delta_{1,k+1} \in [-0.01; 0.01]\ \text{kg/m}^3$ and $\delta_{2,k+1} \in [-0.01; 0.01] \cdot S_{O,sat}\ \text{kg/m}^3$. The measured values were again generated on the basis of the midpoints of the corresponding

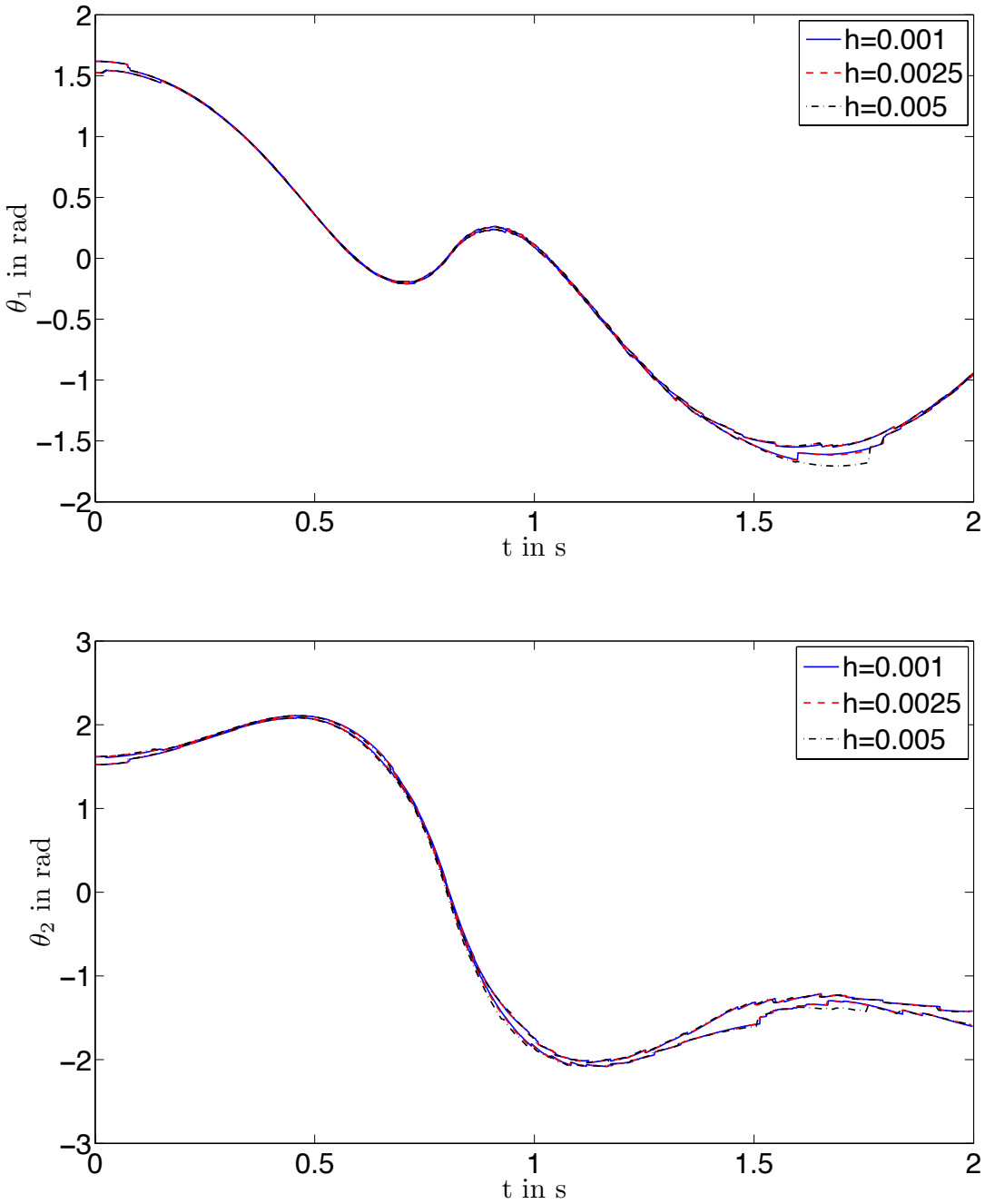


Figure 5.19: Double pendulum: Interval enclosures for θ_1 and θ_2.

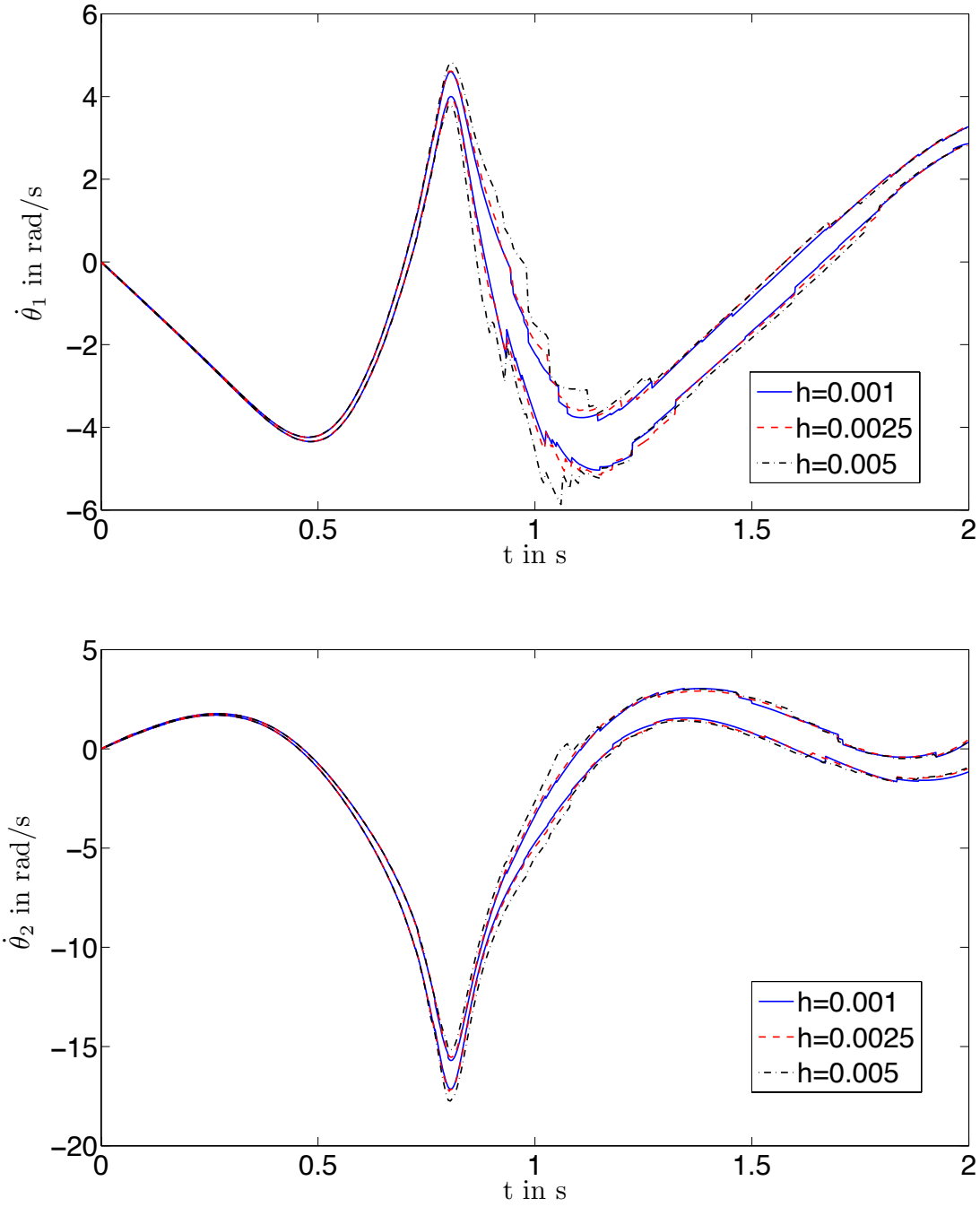


Figure 5.20: Double pendulum: Interval enclosures for $\dot{\theta}_1$ and $\dot{\theta}_2$.

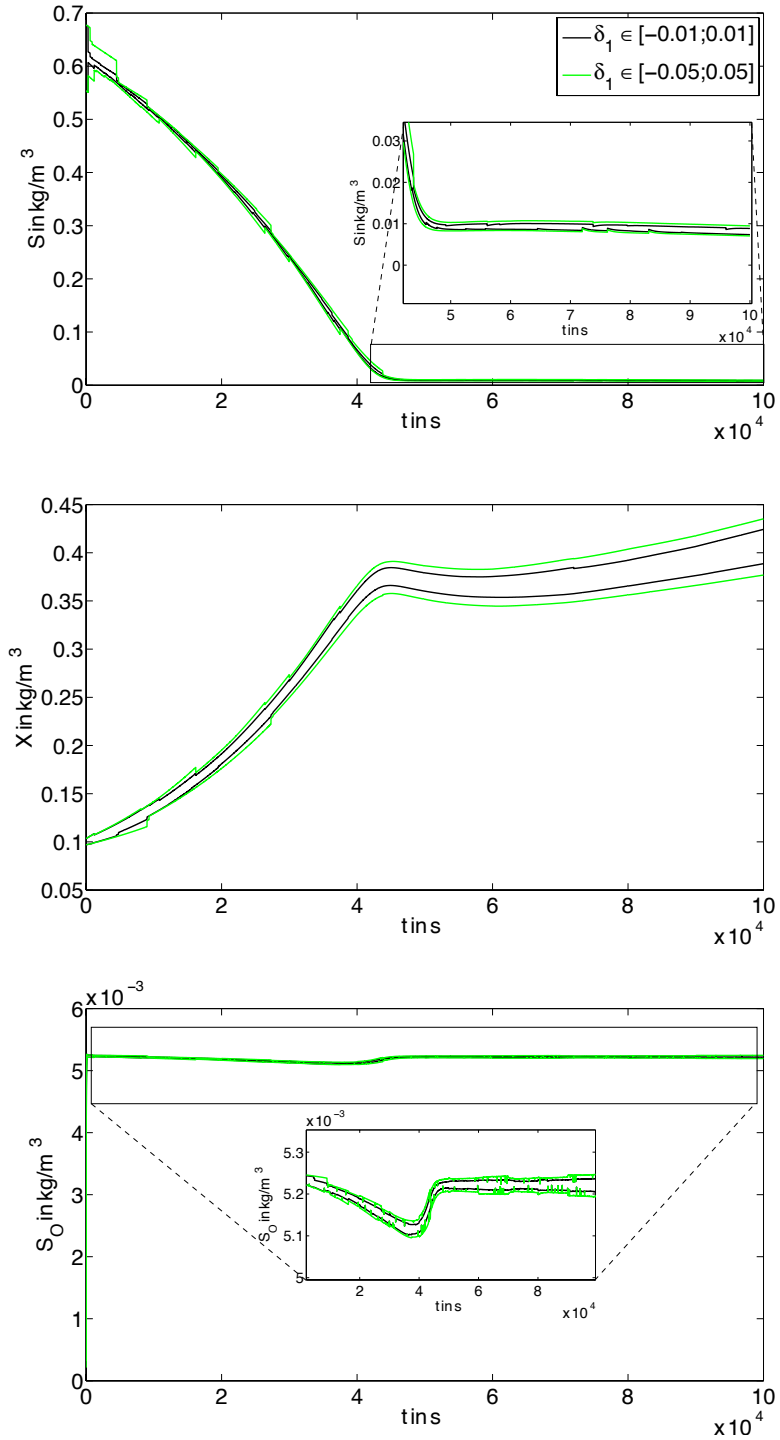


Figure 5.21: BWTP: State enclosures for $S(t)$, $X(t)$ and $S_O(t)$ for different $[\delta_1]$.

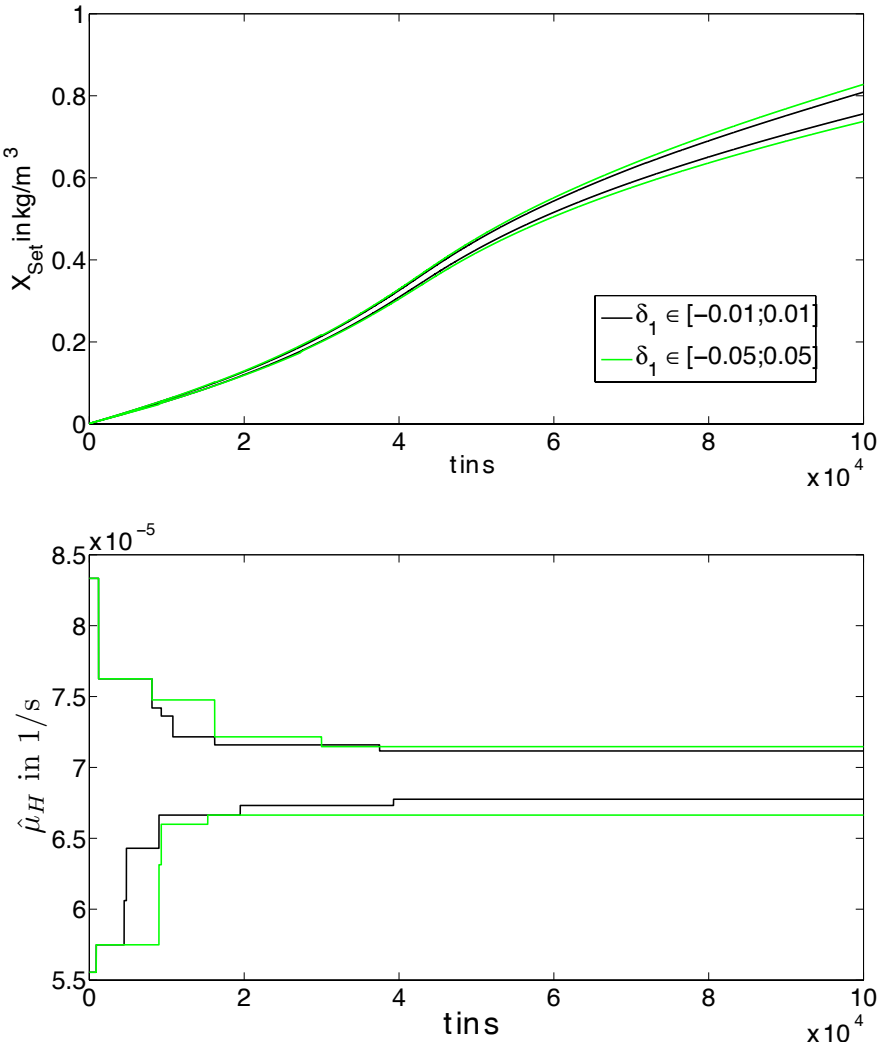


Figure 5.22: BWTP: State enclosures for $X_{Set}(t)$ and $\hat{\mu}_H(t)$ for different $[\delta_1]$.

uncertainty. The order of the Taylor series was $\nu = 1$, the maximum number of splittings was 100 and the maximum number of interval vectors was given by $L_{max} = 200$, and the step size was $h = 10$s. Measurements were performed every 300 s, hence $\Delta T_m = 300$s. Splitting and merging was carried out every 5 time-steps. Reapproximation by 200 disjoint interval vectors was employed every 1000 time-steps. A second estimation with $\delta_{1,k+1} \in [-0.05; 0.05]$ kg/m^3 was also carried out. The state enclosures are presented in Figs. 5.21 and 5.22.
Figs. 5.23 and 5.24 depict results for $\Delta T_m = 300$s and $\Delta T_m = 600$s with
$\delta_{1,k+1} \in [-0.01; 0.01]$ kg/m^3 for both values of ΔT_m. The accuracy of the measurements and

Table 5.5: BWTP: Comparison of the computation time for different $[\delta_1]$.

$[\delta_1]$ in kg/m^3	Computation time in s
[-0.01;0.01]	33546
[-0.05;0.05]	35215

Table 5.6: BWTP: Comparison of the computation time for different ΔT_m.

ΔT_m	Computation time in s
300 s	33546
600 s	32740

the time between two measurements are crucial to the estimation quality. If in practice the desired accuracy of the estimation is high, then more accurate sensors have to be employed and/or measurements have to be taken more frequently. If however more conservative bounds can be tolerated, less accurate and cheaper sensors can be used.

Mechanical Positioning System

In Section 4.3 simulation results for a mechanical positioning system are presented. In the following it is assumed that the position x_1 can be measured every 50 time-steps. The mass was given by $m = 1$. F_s and μ were uncertain with $F_s \in [0.0125; 0.015]$, $\mu \in [0.001; 0.0015]$. Both uncertain parameters were assumed to be time-invariant. The initial conditions were given by $x_1(0) \in [0; 0]$ and $x_2(0) \in [-0.05; -0.02]$. The step-size was constant with $h = 0.1$. The results for three different values of the measurement uncertainties $\delta_1 \in [-0.2; 0.2]$, $\delta_1 \in [-0.1; 0.1]$ and $\delta_1 \in [-0.01; 0.01]$ are depicted in Fig. 5.25 and Fig. 5.26 . The maximum number of interval vectors was 400. The measured values were generated with the values $x_1(0) = 0$, $x_2(0) = -0.2$, $F_S = 0.13$, and $\mu = 0.0012$. The measurement of the position improves also the enclosures for x_2 and the uncertain parameter F_s, since inconsistent interval vectors were deleted in the correction step. The enclosure of the uncertain parameter μ could not be tightened. The best enclosures were obtained for $\delta_1 \in [-0.01; 0.01]$. The enclosures for $\delta_1 \in [-0.1; 0.1]$ are slightly wider. However, if the uncertainty is increased to $\delta_1 \in [-0.2; 0.2]$, the estimation quality decreases significantly.

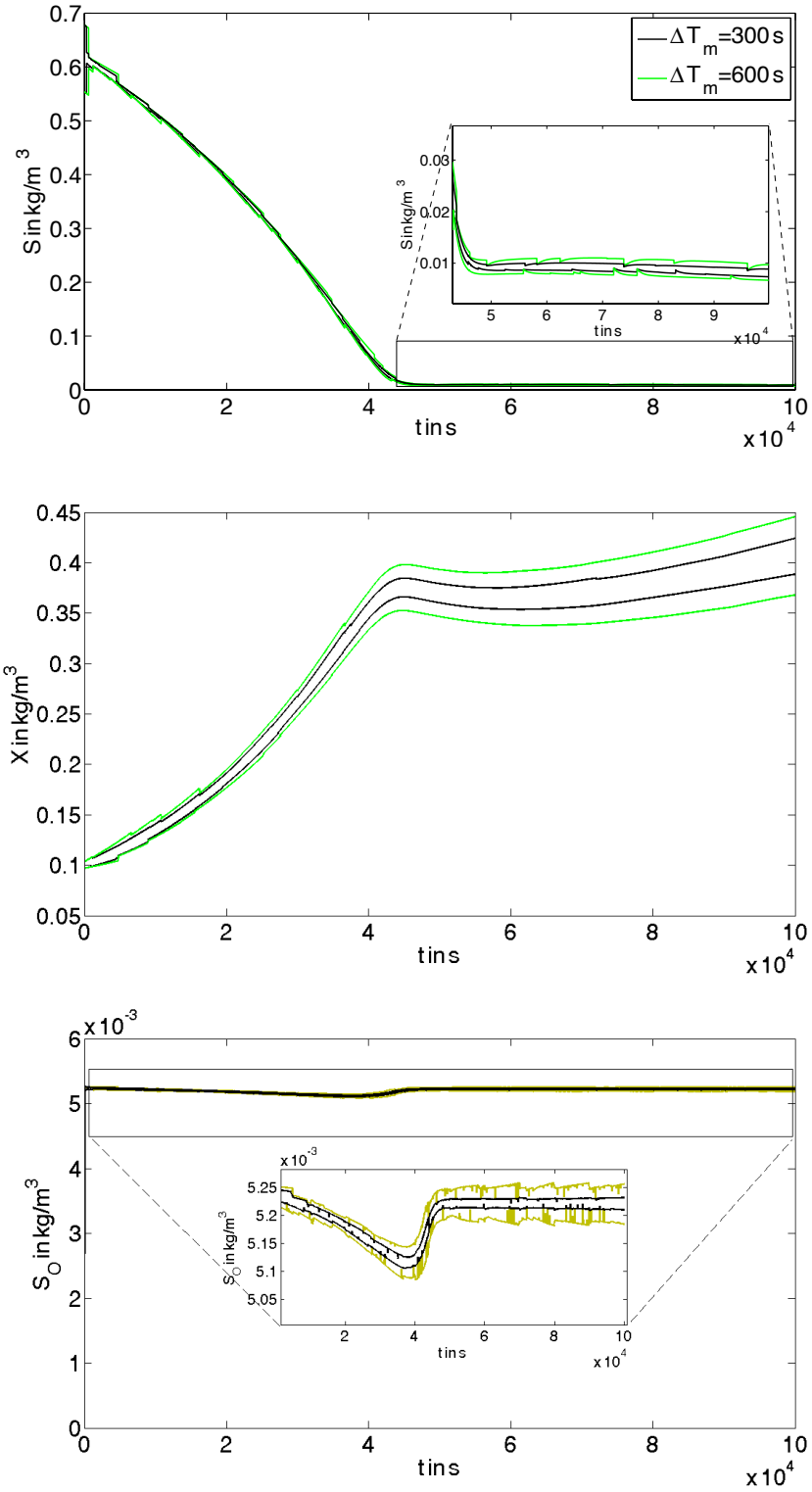


Figure 5.23: BWTP: State enclosures for $S(t)$, $X(t)$ and $S_O(t)$ for different ΔT_m.

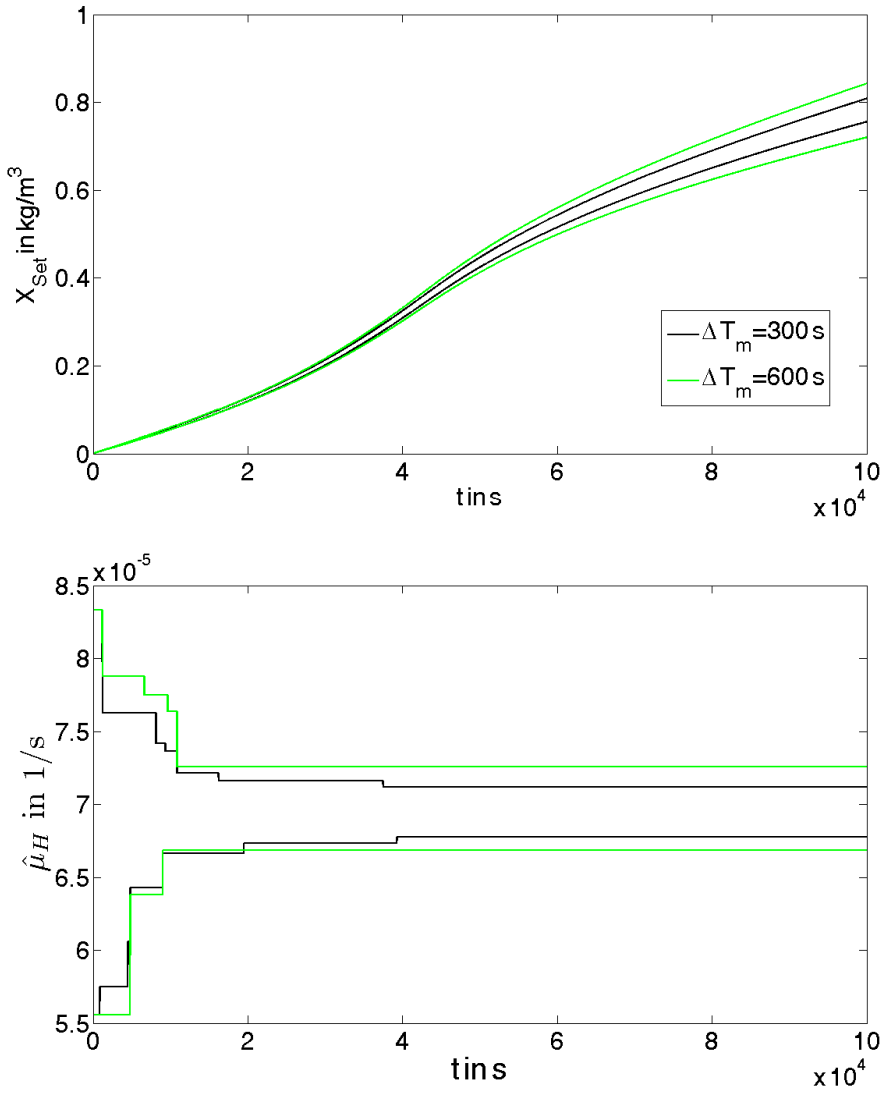
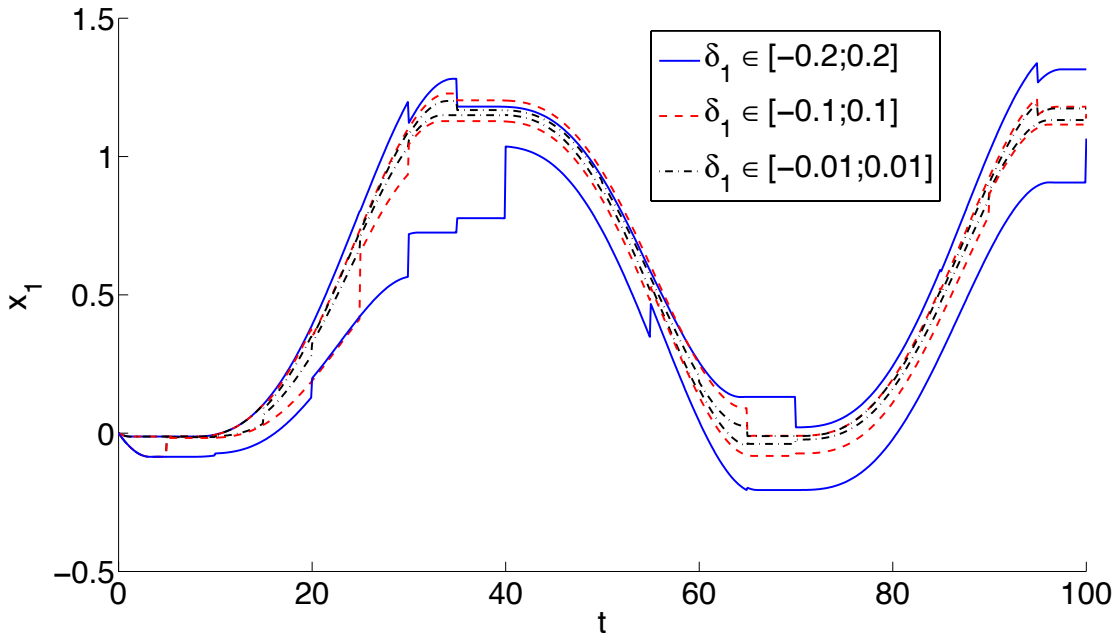
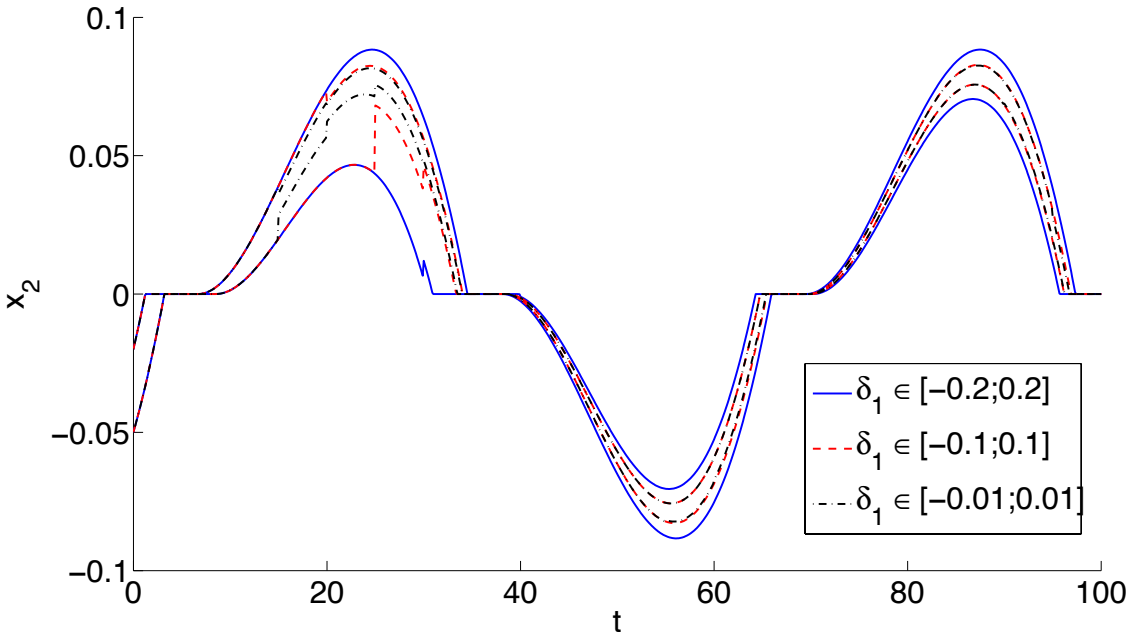


Figure 5.24: BWTP: State enclosures for $X_{Set}(t)$ and $\hat{\mu}_H(t)$ for different ΔT_m.

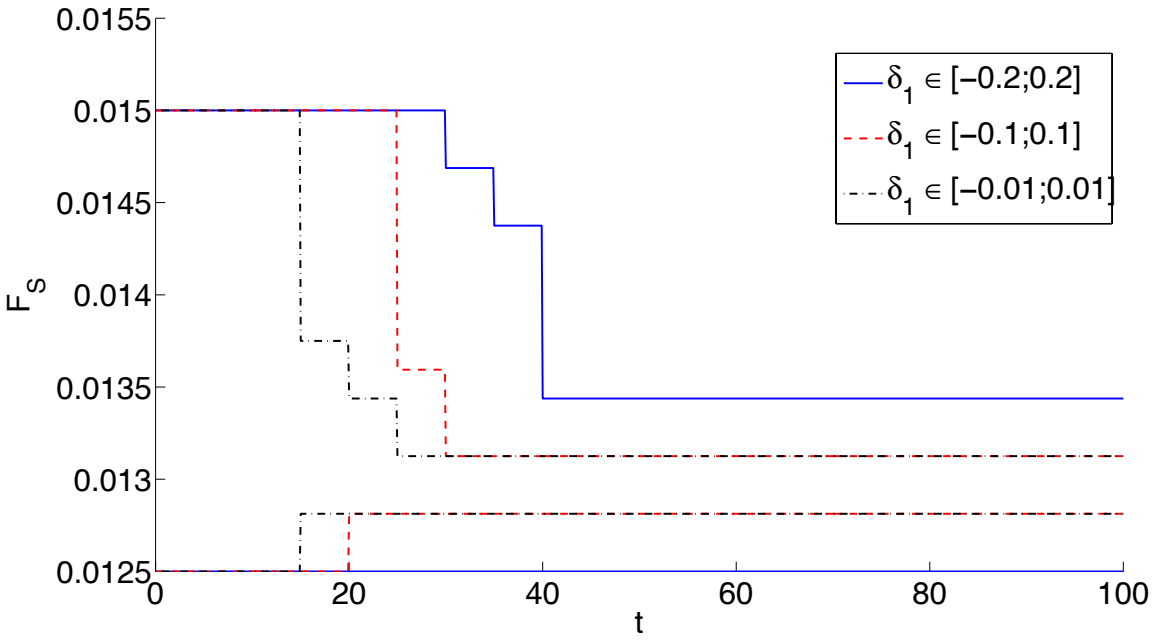


(a) Interval enclosure of the position x_1.

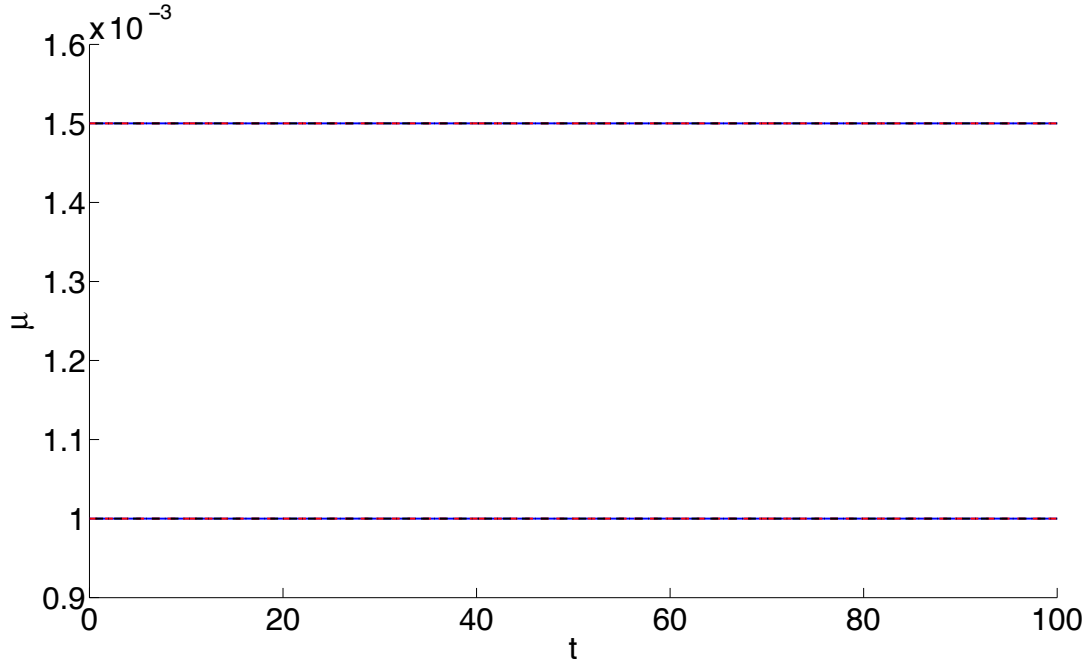


(b) Interval enclosure of the velocity x_2.

Figure 5.25: State enclosures for x_1 and x_2 for different values of $[\delta_1]$.



(a) Interval enclosure of F_s.



(b) Interval enclosure of μ.

Figure 5.26: Enclosures for the estimated parameters F_s and μ for different values of $[\delta_1]$.

5.4 Taylor Model Observer

The second VSPE concept is based on Taylor models [37]. At time-step k the extended state vector is enclosed by a list $\mathcal{T}_k$ of Taylor models:

$$\begin{aligned} z_k \in & \mathcal{T}_k = \left\{ T_{\rho,k}^{(1)}(\mathfrak{z}), T_{\rho,k}^{(2)}(\mathfrak{z}), \dots T_{\rho,k}^{(L_k)}(\mathfrak{z}) \right\} \\ & \text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1,2,\dots,n \quad \text{and} \quad L_k \leq L_{max} \ . \end{aligned} \tag{5.61}$$

5.4.1 Prediction Step

The prediction step is illustrated in Fig. 5.27. At time step k the set is described by a list of Taylor models $\mathcal{T}_k$. First a Taylor model is selected for splitting, next a splitting criterion is evaluated and the Taylor model is split by splitting the domain interval vector in subboxes. This is done according to Section 4.4.2. Taylor models are split until a pre-specified number of Taylor models or number of splittings is reached. Next, for each Taylor model a verified integration is performed.
The resulting enclosure of the extended state vector at time-step $k+1$ after the prediction step is then given by

$$\begin{aligned} & \mathcal{T}_{k+1}^{pr} = \left\{ T_{\rho,k+1}^{(pr,1)}(\mathfrak{z}), T_{\rho,k+1}^{(pr,2)}(\mathfrak{z}), \dots T_{\rho,k+1}^{(pr,L_{k+1}^{pr})}(\mathfrak{z}) \right\} \\ & \text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1,2,\dots,n \quad \text{and} \quad L_{k+1}^{pr} \leq L_{max} \ . \end{aligned} \tag{5.62}$$

The prediction step is repeated until measured values are available. The result of the prediction step is used as an initial enclosure of the below described correction step.

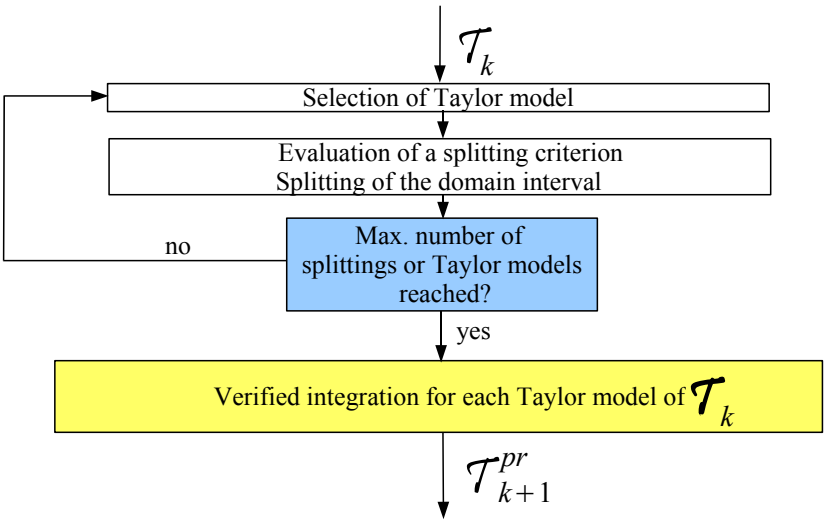


Figure 5.27: Flow diagram.

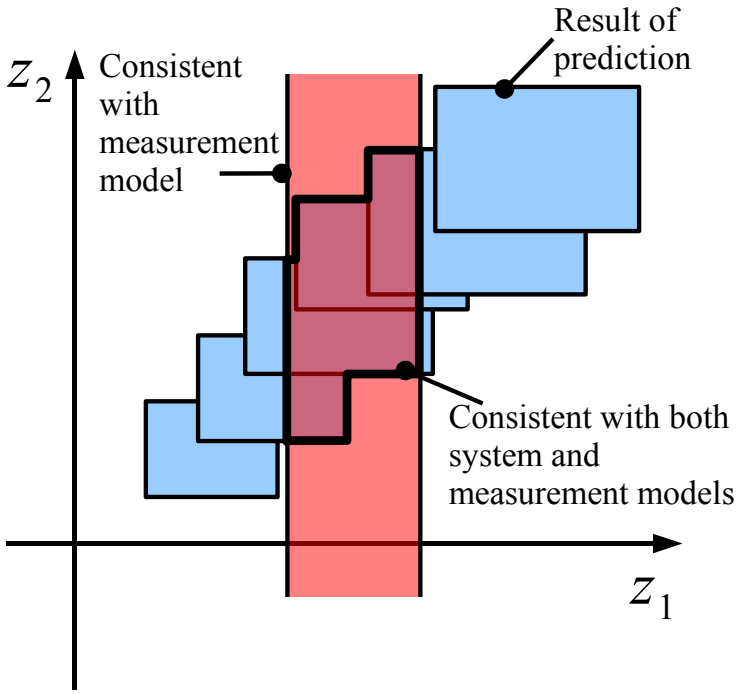


Figure 5.28: Intersection of the predicted set and the set consistent with the measurement in case of the interval observer.

5.4.2 Correction Step

In the correction step [37], guaranteed enclosures of the extended state variables are reconstructed from the measurements. Domains obtained from the prediction step which are inconsistent with the measurement equation and the measured values are deleted. If a state can be measured directly, the implementation of the correction step is fairly easy, if an interval observer as described in Section 5.3 is used. Then only, an intersection of the predicted result with the sum of the measured value and the measurement uncertainty is required. This is illustrated in Fig. 5.28. However, if a Taylor model observer is considered, even in this case, the correction step is not trivial (see Fig. 5.29).

Basic Correction Step

The measurement equation (5.6) is again rewritten according to

$$h\left(z_{k+1}, [q_{k+1}], [\delta_{k+1}]\right) - y_{k+1} = 0 \ . \tag{5.63}$$

In order to perform the correction step, each Taylor model $T^{(pr,l)}_{\rho,k+1}(\mathfrak{z})$ of $\mathcal{T}^{(pr)}_{k+1}$ is now considered separately and is substituted for z_{k+1} in (5.63):

$$h\left(T^{(pr,l)}_{\rho,k+1}(\mathfrak{z}), [q_{k+1}], [\delta_{k+1}]\right) - y_{k+1} = 0 \ . \tag{5.64}$$

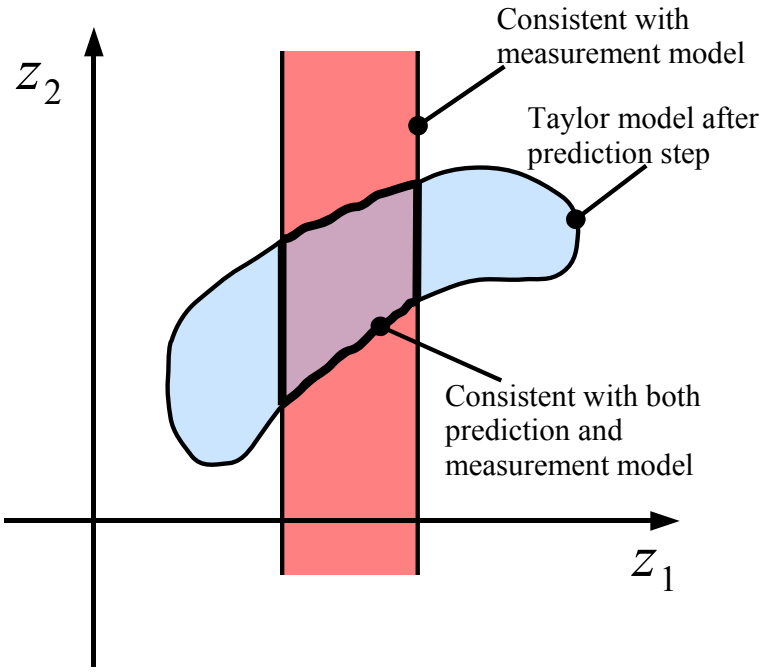


Figure 5.29: Intersection of the predicted set and the set consistent with the measurement in case of the Taylor model observer.

Equation (5.64) is solved for $\mathfrak{z}$ with an interval Newton method [37], which leads to a tightened domain interval $[\tilde{\mathfrak{z}}]$, hence $\mathfrak{z} \in [\tilde{\mathfrak{z}}]$ (see also Fig. 5.30). Here, the Krawczyk method is used. For $n > m$, (5.64) is under determined and cannot be inverted directly. One solution approach is to solve the (5.64) only for m variables (components of $\mathfrak{z}$) and considering the remaining $n - m$ variables as constant intervals. Another possibility is to consider a sufficient number of previous measurements $y(t \leq t_k)$ and the corresponding Taylor models of the right hand side of (5.64) to obtain the missing $n - m$ equations.

If no solution in $[\mathfrak{z}]$ can be found the corresponding Taylor model is inconsistent with the the measured values and the measurement model and can be deleted.

The components of the vector of the original initial conditions $\mathfrak{z}$ are replaced by the Taylor model of the resulting interval $[\tilde{\mathfrak{z}}]$ with $\mathfrak{z}_i \in [-1; 1]$ as domain interval. This Taylor model is given by

$$\begin{aligned} [\tilde{\mathfrak{z}}] \in & \tilde{T}(\mathfrak{z}) = \tilde{c} + \tilde{D}\,\mathfrak{z} \\ & \text{with } \mathfrak{z}_i \in [-1; 1] \quad , i = 1 \ldots, n \quad , \end{aligned} \tag{5.65}$$

where $\tilde{c}$ is the midpoint of $[\tilde{\mathfrak{z}}]$ and $\tilde{D}$ is a diagonal matrix with $\tilde{d}_{i,i} = \mathrm{rad}([\tilde{\mathfrak{z}}_i])$. The Taylor model $T_{k+1}^{(c,l)}$ after the correction step is then given

$$T_{\rho,k+1}^{(c,l)}(\mathfrak{z}) = T_{\rho,k+1}^{(pr,l)}(\tilde{T}(\mathfrak{z})) \tag{5.66}$$

which defines the l-th Taylor model for the next prediction step:

$$T_{\rho,k+1}^{(l)}(\mathfrak{z}) = T_{\rho,k+1}^{(c,l)}(\mathfrak{z}) \quad . \tag{5.67}$$

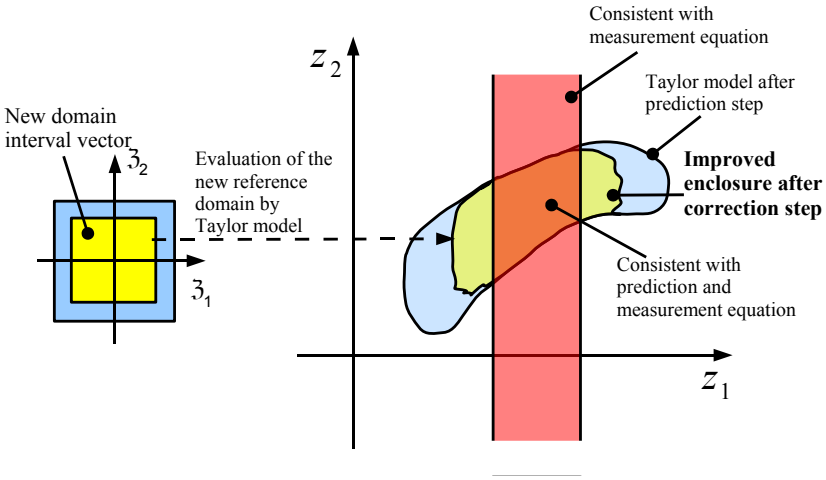


Figure 5.30: Basic correction step.

And the List of Taylor models at $k+1$ is then given by:

$$z_{k+1} \in \mathcal{T}_{\rho,k+1} = \left\{ T^{(1)}_{\rho,k+1}(\mathfrak{z}), T^{(2)}_{\rho,k+1}(\mathfrak{z}), \ldots T^{(L_{k+1})}_{\rho,k+1}(\mathfrak{z}) \right\} \\ \text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1,2,\ldots,n \quad \text{with} \quad L_{k+1} \leq L^{pr}_{k+1} \leq L_{max} \ . \tag{5.68}$$

This approach can also be used for consideration of time-varying interval bounded parameters with interval bounded variation rates, as described in Section 4.1. The upper and lower bounds of the parameters can be considered as bounds for a measurement $y_p(t) \in [\underline{p}; \overline{p}]$ for all t. If these parameter bounds are exceeded during the estimation, i.e. $\inf(p(t)) < \underline{p}$ or $\sup(p(t)) > \overline{p}$ for any t, they can be limited by the same approach that is implemented for the correction step of the observer.

Consider the two Taylor models

$$T^A = 1 + \mathfrak{z}_1 \tag{5.69}$$

and

$$T^B = 1 + 0.5\mathfrak{z}_1 + 0.3\mathfrak{z}_2 \ . \tag{5.70}$$

With $\mathfrak{z}_1$, $\mathfrak{z}_2 \in [-1;1]$ the interval bound of T^A is $B(T^A) = [0;2]$, the interval bound of T^B is $B(T^B) = [0.2;1.8]$. Both Taylor models are now intersected with the interval $[y] = [0.9;1.3]$. First the modified interval domain $[\tilde{\mathfrak{z}}]$ for T^A is calculated:

$$[\tilde{\mathfrak{z}}] = ([y] - 1) \cap [\mathfrak{z}_1] = [-0.1;0.3], \tag{5.71}$$

leading to

$$T^{A,c} = 1 + \mathfrak{z}_1 \quad \text{with} \quad \mathfrak{z}_1 \in [-0.1;0.3] \tag{5.72}$$

or

$$T^{A,c} = 1 + (0.1 + 0.2\mathfrak{z}_1) \quad \text{with} \quad \mathfrak{z}_1 \in [-1;1] \tag{5.73}$$

and $B(T^{A,c}) = [0.9;1.3]$.

Next, the modified interval domain $[\tilde{\mathfrak{z}}]$ for T^B is determined:

$$[\tilde{\mathfrak{z}}] = \frac{[y] - 1 - 0.3[\mathfrak{z}_2]}{0.5} \cap [\mathfrak{z}_1] = [-0.8; 1]. \tag{5.74}$$

The width of the interval domain of $\mathfrak{z}_2$ cannot be decreased in this example. This results in

$$T^{B,c} = 1 + 0.5\mathfrak{z}_1 + 0.3\mathfrak{z}_2 \quad \text{with} \quad \mathfrak{z}_1 \in [-0.8; 1], \mathfrak{z}_2 \in [-1; 1] \tag{5.75}$$

or

$$T^{B,c} = 1 + (0.1 + 0.9\mathfrak{z}_1) + 0.3\mathfrak{z}_2 \quad \text{with} \quad \mathfrak{z}_1, \mathfrak{z}_2 \in [-1; 1] \tag{5.76}$$

and $B(T^{B,c}) = [0.3; 1.8]$.

Thus, despite the fact that the bounds of T^B are tighter than for T^A, the bounds after the correction step are tighter for $T^{A,c}$, since T^B depends on two independent variables whose coefficients have similar size. This example illustrates that the efficientcy of the correction step highly depends on the relation between the coefficients belonging to different independent variables.

Consistency Tests

To improve the correction step, additionally consistency tests can be performed. The domain interval vector $[\mathfrak{z}]$ of a Taylor model $T^{(c,l)}_{\rho,k+1}(\mathfrak{z})$ after the basic correction step is split into subboxes $[\tilde{\mathfrak{z}}]$ leading to a modified Taylor models $\tilde{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z})$ and the measurement equation (5.6) is rewritten according to

$$\check{h}\left(z_{k+1}, [q_{k+1}]\right) = y_{k+1} - [\delta_{k+1}] \ . \tag{5.77}$$

Next, consistency tests are performed by evaluation of (5.77) for all Taylor models $\tilde{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z})$. Now, three different cases have to be distinguished (see also Fig. 5.4.2):

1. If $B\left(\check{h}\left(\tilde{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}), [q_{k+1}]\right)\right) \subseteq \left(y_{k+1} - [\delta_{k+1}]\right)$ holds, then $[\tilde{\mathfrak{z}}]$ is consistent.
2. If $B\left(\check{h}\left(\tilde{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}), [q_{k+1}]\right)\right) \cap \left(y_{k+1} - [\delta_{k+1}]\right) = \emptyset$ holds, then $[\tilde{\mathfrak{z}}]$ is inconsistent and can be deleted.
3. All remaining subboxes $[\tilde{\mathfrak{z}}]$ have to be split further.

Note that the bounds $B(\cdot)$ are calculated for $\mathfrak{z} \in [\tilde{\mathfrak{z}}]$. The consistency test can be combined with the interval Newton method like in the basic correction step.

The obtained subset of the domain interval vector after the consistency test consists of several subboxes and for each subbox Taylor models are obtained. In order to avoid an exponential growth of the number of Taylor models during the estimation process, efficient merging strategies have to be applied. Subboxes can be merged in case of small overestimation of the union of the merged subboxes. Another possibility is to replace the result by a single interval vector which encloses all subboxes. This is a special case of the merging routine described in Section 4.1.6. In this work a further approach is proposed. The goal is to enclose the obtained set by a rotated box.

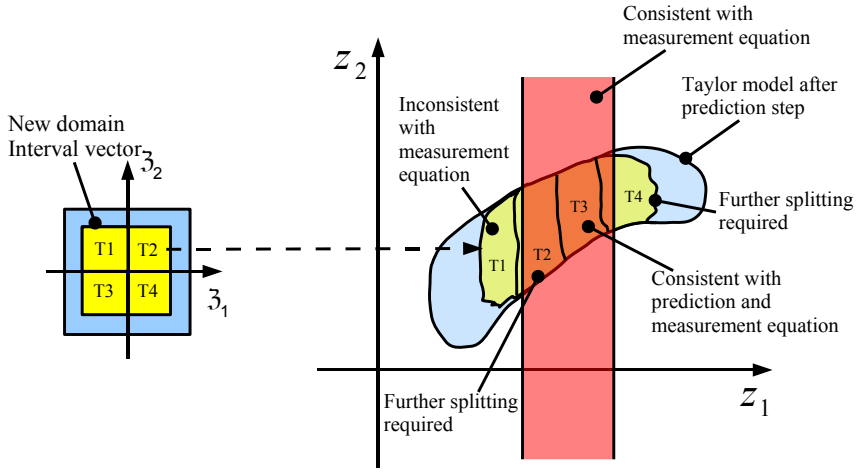


Figure 5.31: Consistency test.

The algorithm consists of the following steps:

1. Let $\hat{Z} = \{\hat{z}^{(1)}, \hat{z}^{(2)}, \ldots, \hat{z}^{(L)}\}$ be a list including all interval vertices and midpoints of the subboxes remaining after the consistency test.

2. Calculate the balance point c and the covariance matrix C of the distribution of $\hat{Z}$ with

$$c = \frac{1}{L} \sum_{l=1}^{L} \hat{z}^{(l)} \tag{5.78}$$

and

$$C = \frac{1}{L} \sum_{l=1}^{L} (\hat{z}^{(l)} - c) \cdot (\hat{z}^{(l)} - c)^T \quad . \tag{5.79}$$

3. Determine the eigenvectors of the covariance matrix C.

4. The initial enclosure is determined by the balance point and the eigenvectors of the covariance matrix and can be expressed as a Taylor model

$$c + V\,\mathfrak{z} \text{ with } \mathfrak{z}_i \in [-1;1] \quad , i = 1 \ldots, n \quad , \tag{5.80}$$

where c corresponds to the balance point. V is a matrix containing the eigenvectors normalized to length 1.

5. Check whether all subboxes are included in the initial enclosure. This is done by transforming all subboxes $[\tilde{\mathfrak{z}}]$ to $[\tilde{\mathfrak{z}}']$ by a subtraction of c and multiplication with V^{-1}, i.e.

$$[\tilde{\mathfrak{z}}'] = V^{-1}([\tilde{\mathfrak{z}}] - c) \quad . \tag{5.81}$$

If not all $[\tilde{\mathfrak{z}}']$ are contained in the unit interval vector $[-1, 1]^n$, the initial enclosure does not contain all subboxes and has to be inflated by increasing the length of the eigenvectors in V until all subboxes are contained.

6. If all subboxes are contained, i.e.,

$$V^{-1}([\tilde{\mathfrak{z}}] - c) \subseteq [-1, 1]^n \text{ for all } [\tilde{\mathfrak{z}}] \ , \tag{5.82}$$

a contraction is performed by decreasing the the length of the eigenvectors in V until any further significant improvement cannot be achieved. The new matrix formed by the resulting eigenvectors is $\hat{V}$. A modification of the balance point c leading to a new point $\hat{c}$ can result in a further improvement of the enclosure. This modification is done by shifting c slightly in and/or against the direction of the vectors in $\hat{V}$.

7. This enclosure or rotated box can be again expressed as a Taylor model according to

$$\hat{T}(\mathfrak{z}) = \hat{c} + \hat{V}\,\mathfrak{z} \text{ with } \mathfrak{z}_i \in [-1; 1] \ , i = 1 \ldots, n \ . \tag{5.83}$$

8. The components $\mathfrak{z}_i$ of the vector of the original initial conditions $\mathfrak{z}$ of $T^{(c,l)}_{\rho,k+1}(\mathfrak{z})$ are replaced with the components of $\hat{T}(\mathfrak{z})$ by substituting $\hat{T}_i(\mathfrak{z})$ for $\mathfrak{z}_i$ which results in a modified Taylor model $\hat{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}) = T^{(c,l)}_{\rho,k+1}(\hat{T}(\mathfrak{z}))$.

The l-th Taylor model for the next prediction step is then defined by:

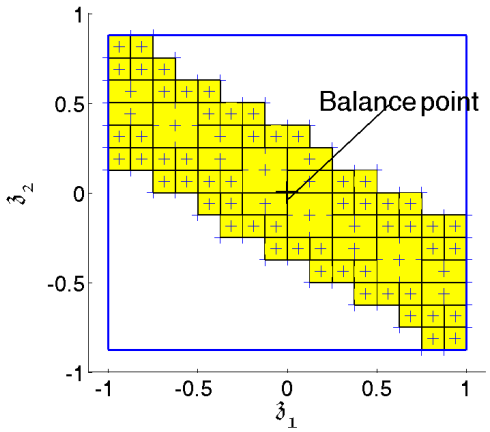
$$T^{(l)}_{\rho,k+1}(\mathfrak{z}) = \hat{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}) \ . \tag{5.84}$$

Fig. 5.32(a) shows the remaining subboxes of the domain interval vector after the consistency test together with the interval vertices, midpoints, and the balance point. Fig. 5.32(b) depicts the corresponding initial enclosure and Fig. 5.32(c) the final enclosure, which is used as a new domain interval vector replacing the original domain interval vector $\mathfrak{z}$ by the corresponding Taylor model of the rotated box.

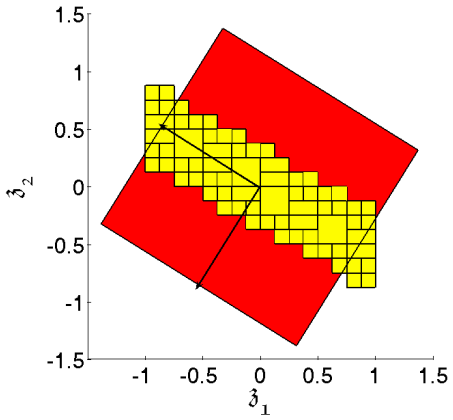
Note, that the corner points of the rotated box are in general outside of the interval hull around all subboxes. Thus, it is important that the pseudo-volume of the enclosure by the rotated box is sufficiently smaller than the enclosure by a single interval vector. If the set of subboxes describing the reference domain after the consistency test is highly non-convex, it has to be approximated by several rotated boxes. This will be considered in future research.

Remark: If several Taylor models are used, in the prediction step, (5.77) is evaluated for each Taylor model before the correction step and only Taylor models for which case 3 of the consistency test holds are further investigated in the correction step. The Taylor model belonging to case 1 are consistent with the measured values and the measurement model, the Taylor models to case 2 are deleted since they are inconsistent.

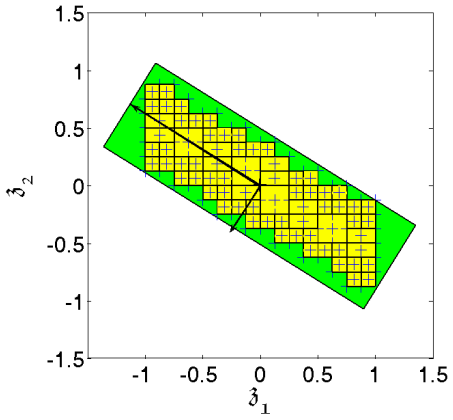
After the correction step the reference domain of some of the remaining Taylor models are split up to a specified number. And the prediction step is performed until the next measured values are available. Merging routines in the case of the interval observer concept enable to reduce the number of interval vectors. There exists however no efficient merging algorithm which is able to bound two similar Taylor models by a new Taylor model with little overestimation. If the maximum number of Taylor models is reached, splitting of the reference domain can only be done, if Taylor models are deleted in the correction step.



(a) Result of the consistency tests.



(b) Initial enclosure.



(c) Final enclosure.

Figure 5.32: Inclusion by rotated box.

The Correction Step in Preconditioned Taylor Models

If preconditioning is applied in the prediction step, which is necessary in most nonlinear systems, the correction step can be either be applied to the composition of the left and right Taylor model

$$\check{T}(\mathfrak{z})_{\rho,k+1} := (\check{P}_{l,k+1}(\bar{\mathfrak{z}}) + \check{I}_{l,k+1}) \circ (P_{r,k}(\mathfrak{z}) + I_{r,k}) \tag{5.85}$$

after the corresponding integration step or only to the left part$(\check{P}_{l,k+1}(\bar{\mathfrak{z}}) + \check{I}_{l,k+1})$. In the first case the initial state vector $\mathfrak{z}$ is affected leading to modified right Taylor models, here the correction step is performed before the scaling like in the case of splitting of the domain interval vector (see Section 4.4.2). In the second case the independent variables $\bar{\mathfrak{z}}$ of the left Taylor model are affected which enclose the range of the right Taylor models.

If the correction step is applied to the left Taylor model leading to modified $\bar{\mathfrak{z}}$, then either the right Taylor models have to be reinitialized by the identity Taylor model, or the components of the right Taylor model which is related to the modified components of $\bar{\mathfrak{z}}$ have to be replaced with a Taylor model with zero polynomial part an interval remainder given by the interval [-1;1]. Therefore overestimation is introduced.

However, in some cases, when the correction step applied to the composed Taylor model is not successful but a significant tightening of the left Taylor model by the correction step is possible, it can be advantageous to employ the correction step to the left Taylor model. The tightening of the left Taylor model by the correction step is often easier, since the polynomials in the left Taylor model are less complex than the polynomials of the composed Taylor model, because only linear contributions occur. This effect is similar to the example above, where the correction step was applied to two Taylor models T^A and T^B.

Illustrative Example: Volterra Equations

Consider again the Volterra equations for $a = b = c = d = 1$:

$$\begin{aligned} \dot{z}_1 &= z_1(1 - z_2) \ , \\ \dot{z}_2 &= z_2(z_1 - 1) \ . \end{aligned} \tag{5.86}$$

Two different estimations have been carried out. One with the basic correction step and consistency tests, but the remaining subboxes have been enclosed by an interval vector. In the second estimation the remaining subboxes are enclosed by a parallelepiped in case the pseudo volume of the parallelepiped was two times smaller than the pseudo volume of the interval enclosure, in order to ensure that the algorithm described above provided a significant tight parallelepiped enclosure. The order was $\rho = 5$ in time and initial state variables. The maximum number of splitting operations in the consistency test was 200. No splitting of the domain interval vector was performed in the prediction step. Thus, only one Taylor model was used for the enclosure of the state vector in each time-step. The enclosure for z_1 is tighter for $t = 0$ to approximately $t = 2.5$, after that both algorithms yield comparable results.

As seen before, the success of the correction step depends on the coefficients of the Taylor model obtained when evaluating the measurement equation. If the coefficients belonging to one initial state variable $\mathfrak{z}_i$ are dominating compared to the others, it is more likely that the correction step is successful in tightening the domain interval vector of this initial state variable than when the coefficients of the other initial state variables have comparable size. Therefore, even if the outer bounds of one Taylor model before the correction step is tighter than the outer bound of another Taylor model, it cannot be guaranteed that the Taylor model after the correction step is also tighter

This is what happens for the second state variable at $t = 2.5$. The enclosure resulting from the second estimation is tighter before the correction step, whereas after the correction step the enclosure of the first estimation becomes tighter.

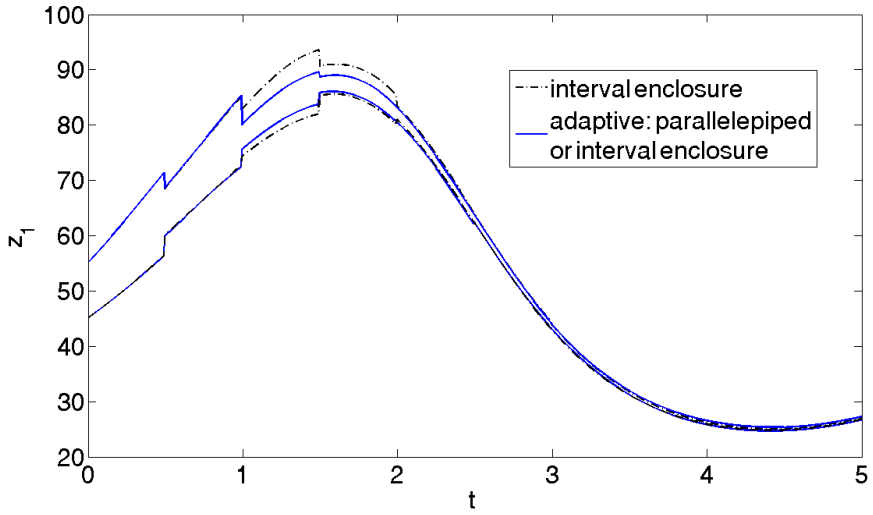


Figure 5.33: Time-dependent enclosures of z_1.

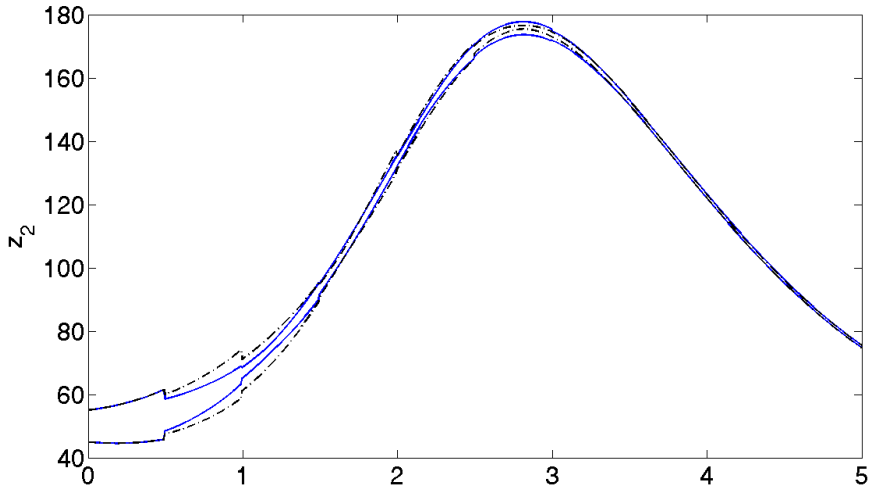


Figure 5.34: Time-dependent enclosures of z_2.

5.4.3 Applications

In this Section, estimation results for the NISTR, for the double pendulum and for the BWTP are presented. The measured values for each application were generated on the basis of the midpoints of the corresponding intervals of the parameters and initial values.

Non-Isothermal Stirred Tank Reactor

First the NISTR is considered. The initial conditions were give by $c_a(0) \in [0.25; 0.75]$mol/l, $c_b(0) \in [0.25; 0.75]$mol/l, $v(0) \in [100; 110]°C$, $v_K(0) \in [100; 110]°C$. The time-invariant parameter E_1 was assumed to be uncertain with $E_1 \in [-1.01; -0.99] \cdot 9758.3$K. The measurement equation for the temperature in the reactor is given by

$$\begin{aligned} y_{1,k+1} &= c_{b,k+1} + \delta_{1,k+1} \ , \\ y_{2,k+1} &= \nu_{k+1} + \delta_{2,k+1} \ . \end{aligned} \tag{5.87}$$

The measurement uncertainties were assumed to be $\delta_1(t) \in [-0.05; 0.05]$mol/l and $\delta_2(t) \in [-0.5; 0.5]°$C for all t and it was further assumed that measurements are available every 0.005h. The estimation was carried out until $0.7h$. Only the basic correction step was employed. The estimation results for three different orders ρ and three different numbers of Taylor models L_{max} are shown. For all three estimations QR preconditioning was chosen. For a successful execution of the estimation, the algorithm required a smaller step size at the beginning. The maximum step-size was $h_{max} = 0.001$. In Figs. 5.35 (a) – 5.35 (c) and in Figs. 5.36 (a) and 5.36 (b) the estimation results are compared. The result of a simulation without inclusion of measurement information is also included for comparison.

All estimations are of similar quality, where the tightest results are obtained for 2 Taylor models with order $\rho = 4$. The computation time was also the shortest here.

It can be seen that c_a is very sensitive with respect to E_1 as it tightens significantly as soon as a large range of E_1 is excluded by the Taylor model observer. The results indicate that

Table 5.7: NISTR: Comparison of the computation time for different ρ and different L_{max}.

Order ρ	Number of Taylor models L_{max}	Computation time in s
3	4	192
4	2	166
5	1	178

only increasing the order does not always lead to the tightest enclosures. A combination with splitting of the domain interval vector is more efficient concerning both estimation quality and computation time.

Estimation results with two uncertain parameters (E_1 and E_3) are presented in the following. The initial conditions, uncertain parameters and measurement equation were assumed to be the same like in Section 5.3.3, namely: $c_a(0) \in [0; 1]$mol/l, $c_b(0) \in [0; 1]$mol/l, $v(0) \in$

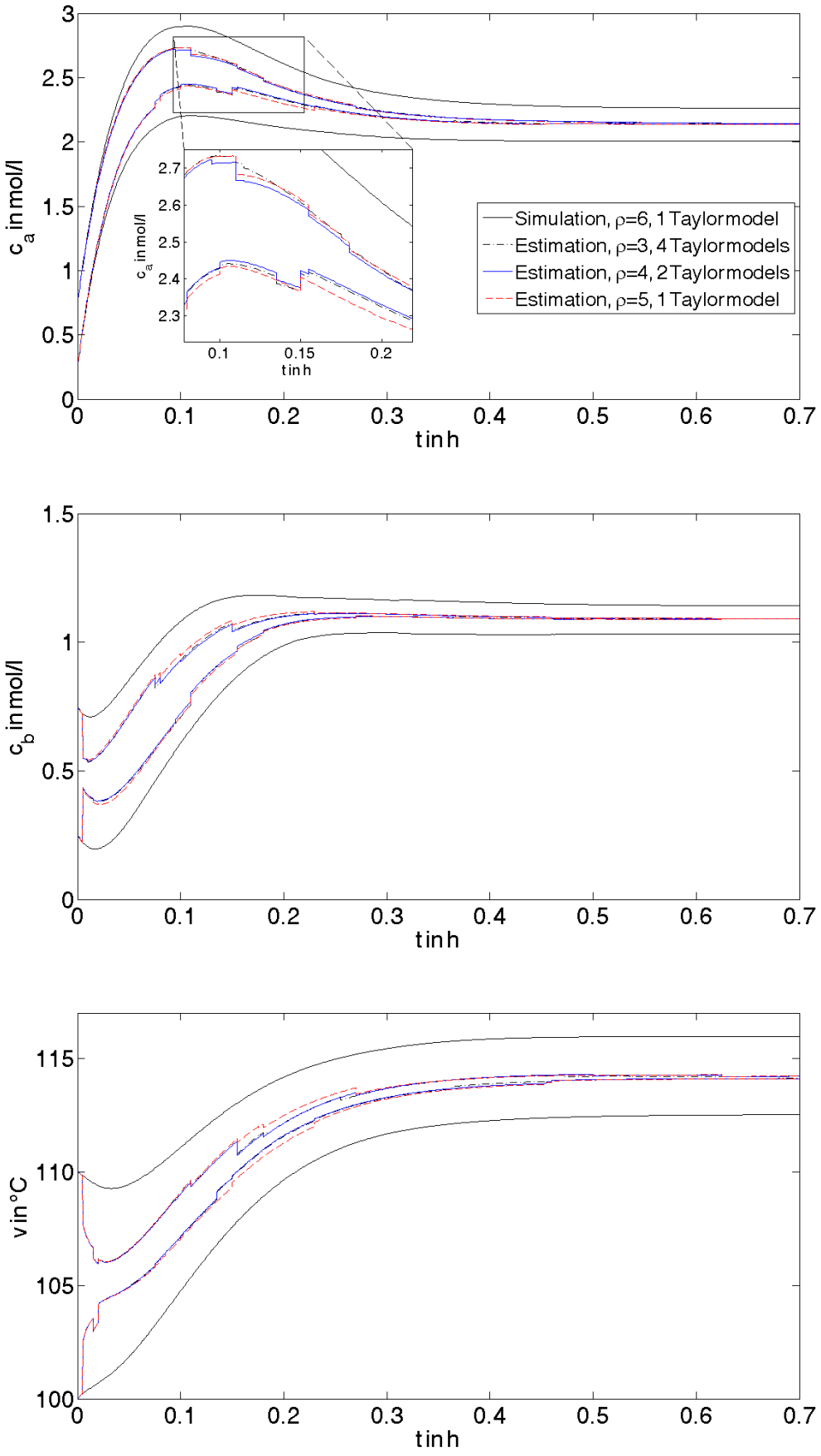


Figure 5.35: NISTR: Interval enclosures for cyclopentadiene concentration c_a and cyclopentenol concentration c_b, and reactor temperature v.

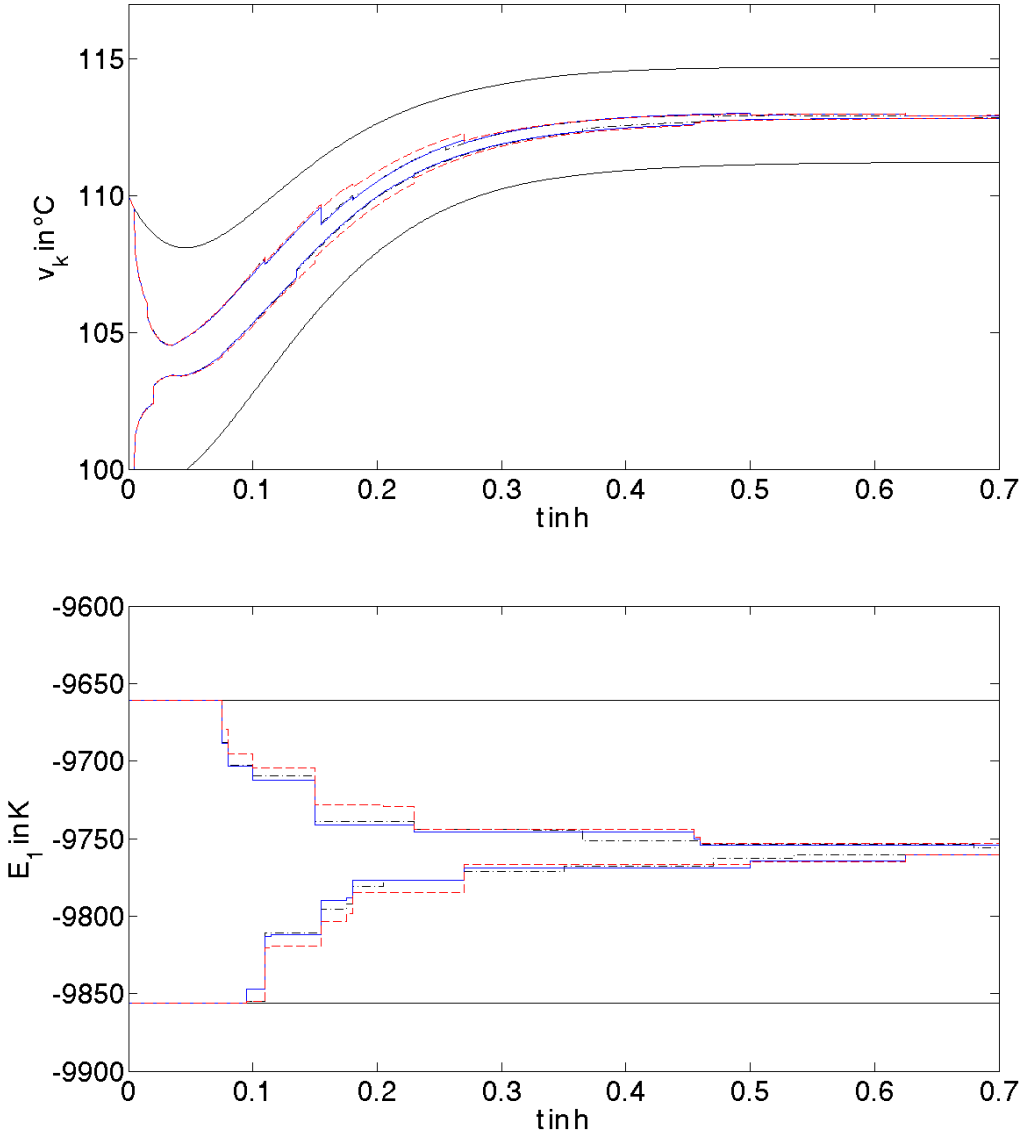


Figure 5.36: NISTR: Interval enclosures for v_k and E_1.

$[100; 110]$°C, $v_K(0) \in [100; 110]$°C, $E_1 \in [0.9; 1.1] \cdot 9758.3$ K, and $E_3 \in [0.9; 1.1] \cdot 8560$K. For the state and parameter estimation of the NISTR again three estimations were carried out: the first with 200 Taylor models of order $\rho = 4$ in time and initial state variables, the second with 100 Taylor models of order $\rho = 5$ in time and initial state variables, and the third 50 Taylor models of order $\rho = 6$ in time and initial state variables. Additionally QR-Preconditioning was applied. The computation times are listed in Tab. 5.8. The results are presented in Figs. 5.37 and 5.38. Differences can be seen mainly at the beginning of the estimation. The first and second estimations provided similar enclosures with similar computation time. The results of the third estimation with 50 Taylor models of order $\rho = 6$ were wider and the computation time was at the same time much longer. Here it becomes even more clear that increasing only the order does not lead always to better results when the computation time has to be kept in a similar magnitude. Using more Taylor models with a slightly lower order is more efficient.

Table 5.8: NISTR: Comparison of the computation time for different ρ and different L_{max}.

Order ρ	Number of Taylor models	Computation time in s
4	200	31318
5	100	34020
6	50	55968

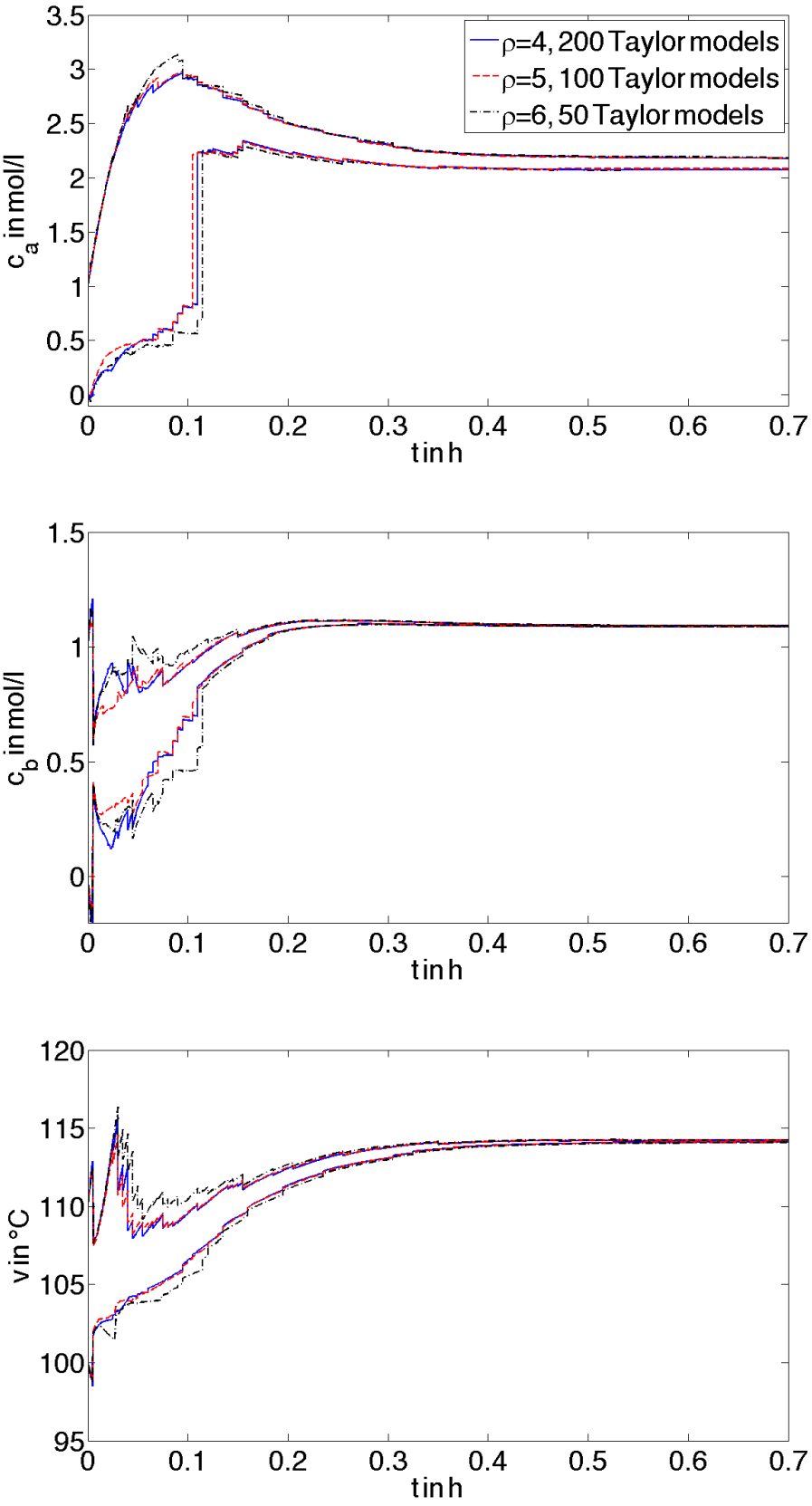


Figure 5.37: NISTR: Comparison of the interval enclosures for cyclopentadiene concentration c_a and cyclopentenol concentration c_b, and reactor temperature v.

Double Pendulum

In case of the double pendulum estimations with one Taylor model of order $\rho = 6$ in time and initial state variables with QR-preconditioning were carried out. Like in Section 5.3.3 the initial conditions were assumed to be $\theta_1(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]$rad, $\theta_2(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]$rad, $\dot{\theta}_1(0) \in [-0.01; 0.01]$rad/s, $\dot{\theta}_2(0) \in [-0.01; 0.01]$rad/s. The assumed measurement errors were again $[\delta_1] = [\delta_2] = [-0.01; 0.01]$m for all t. The time between two measurements was $\Delta T_m = 0.005$s. Three results with constant step sizes $h = 0.001$s, $h = 0.0025$s, and $h = 0.005$s are depicted in Fig 5.39 and Fig. 5.40. Like for the interval observer, the results for $h = 0.001$s and $h = 0.0025$s are of similar quality. Increasing the step-size to $h = 0.005$s leads to a large widening of the bounds. The computation times are given Tab. 5.9.

Table 5.9: Double pendulum: Comparison of the computation time for different step-sizes.

Step-size h in s	Computation time in s
0.001	814
0.0025	450
0.005	321

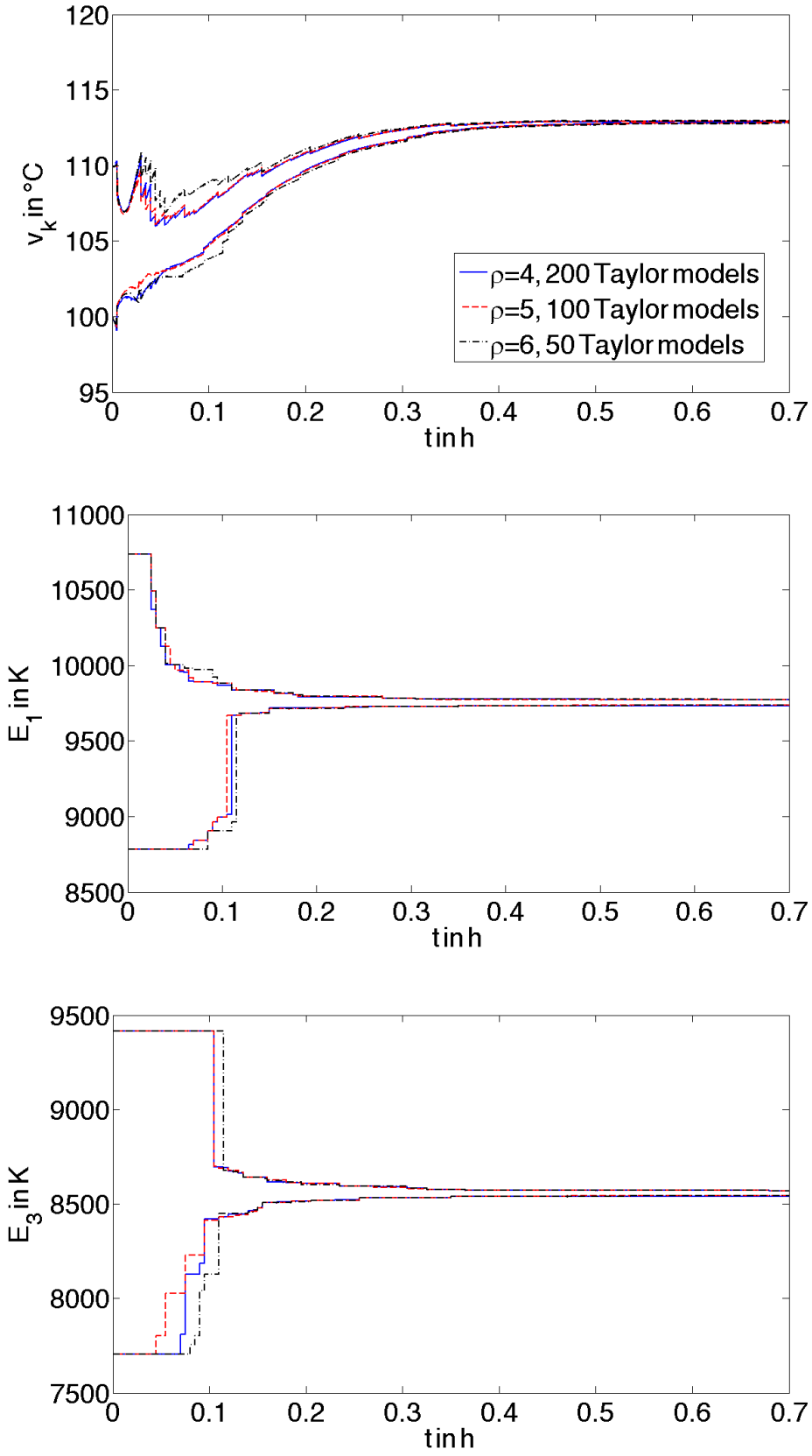


Figure 5.38: NISTR: Comparison of the interval enclosures for jacket temperature v_k, E_1 and E_3

.

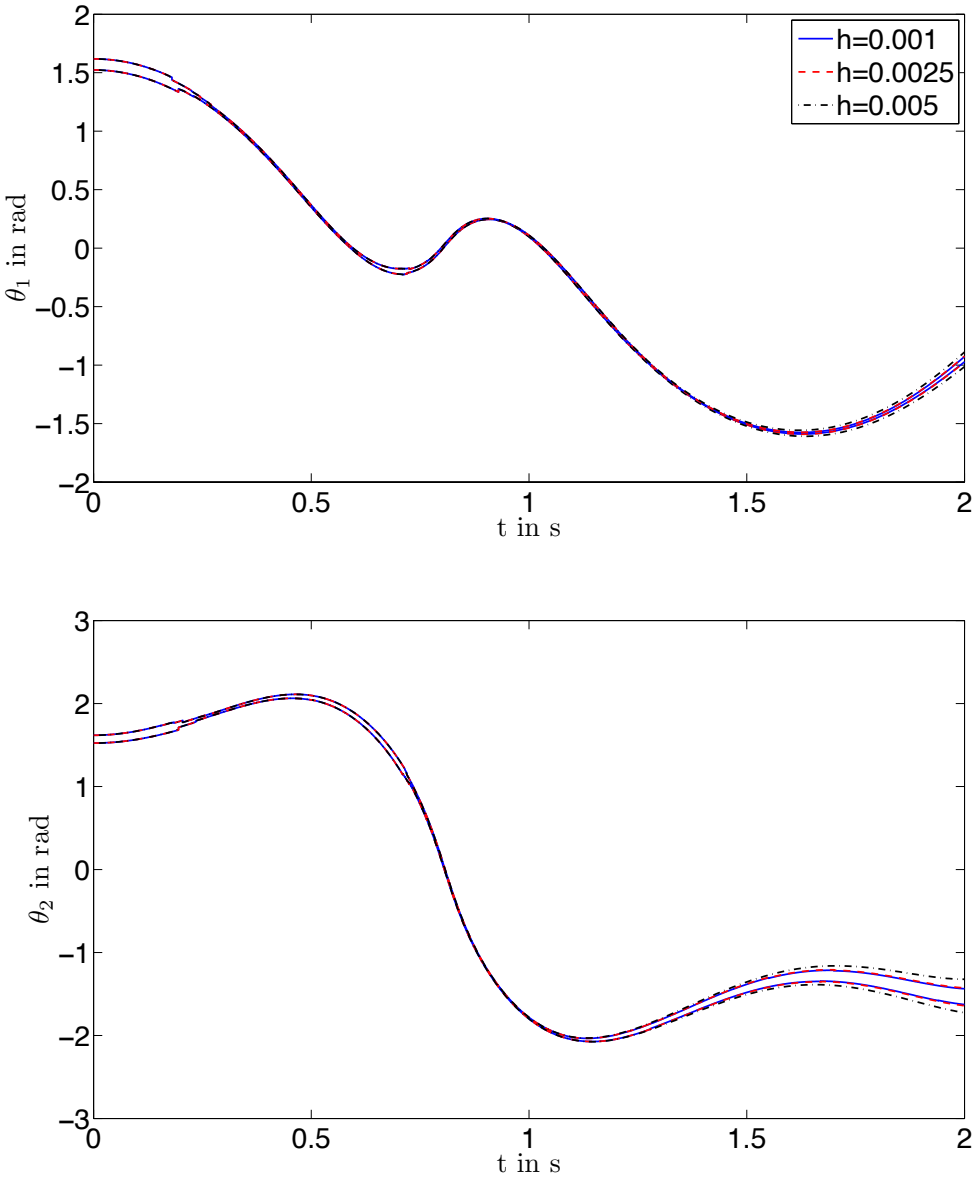


Figure 5.39: Double pendulum: Interval enclosures for θ_1 and θ_2.

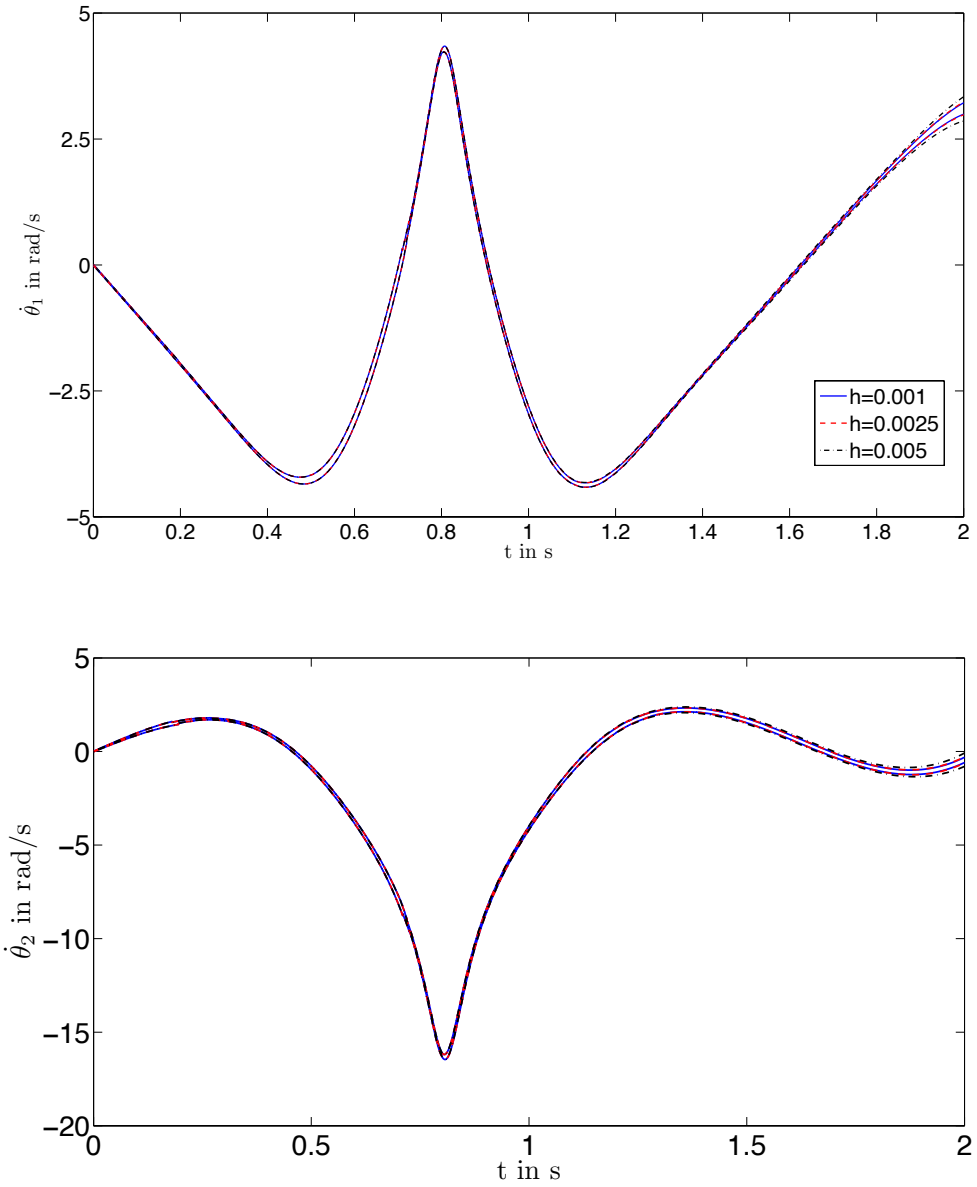


Figure 5.40: Double pendulum: Interval enclosures for $\dot{\theta}_1$ and $\dot{\theta}_2$.

Biological Waste Water Treatment Plant

For the BWTP an estimation with 40 Taylor models of $\rho = 4$ was applied. The intervals for the initial state variables and for the uncertain parameters were the same like for the interval observer in Section 5.3.3, namely $S(0) \in [0.9; 1.1] \cdot 0.616$ kg/m^3, $X(0) \in [0.97; 1.03] \cdot 0.1$ kg/m^3, $S_O(0) \in [0.99; 1.01] \cdot S_{O,sat}/2 \cdot 10^{-4}$ kg/m^3, $X_{Set}(0) \in [0.9; 1.1] \cdot 0.001$ kg/m^3, and $\hat{\mu}_H \in [0.8; 1.2] \cdot 1/14400$ 1/s. Here also QR-Preconditioning was employed additionally. However, the preconditioning with respect to the parameter had to be adapted by modifying the matrix obtained from the QR-decomposition. Otherwise the estimation failed. The reason therefore is that the preconditioning lead to a correlation of the time-invariant parameters with the system state variables leading to an increasing enclosure of the parameters, despite the fact that the parameter were assumed to be time-invariant. The uncertain parameter $\hat{\mu}_H$ is the fifth component in the extended state vector $z = [S, X, S_O, X_{Set}, \hat{\mu}_H]^T$. In the Matrix Q used in the preconditioning the fifth row was set to zero in each time-step except for the fifth component in the fifth row which was set to one. With this modification the correlation of the time-invariant parameters with the system state variables is avoided.
For the BWTP also a step-size control was employed, since at the beginning of the estimation process a very small step size was required. The maximum step-size was $h_{max} = 10$s.
In Figs. 5.41 and 5.42 results for $\delta_1 = [-0.01; 0.01]$ kg/m^3 and $\delta_2 = [-0.05; 0.05]$ kg/m^3 are depicted, with $\Delta T_m = 300$s. Figs. 5.43 and 5.44 depict results for $\Delta T_m = 300$s and $\Delta T_m = 600$s, with $\delta_1 \in [-0.01; 0.01]$ kg/m^3.
In comparison to the interval observer, the influence on the resulting enclosures of the non measured state variables X, X_{Set}, and for the estimated parameter $\hat{\mu}_H$ is not so strong, when increasing the measurement uncertainty $[\delta_1]$ or increasing the time ΔT_m between two consecutive measurements.

Table 5.10: BWTP: Comparison of the computation time for different $[\delta_1]$.

$[\delta_1]$ in kg/m^3	Computation time in s
[-0.01;0.01]	30140
[-0.05;0.05]	30636

Table 5.11: BWTP: Comparison of the computation time for different ΔT_m.

ΔT_m	Computation time in s
300 s	30140
600 s	29752

5.5 Performance Comparison

In this section both VSPE concepts are compared by means of the estimation results of the NISTR, the double pendulum, and the BWTP. In addition to the state enclosures, the time dependent diameters of the state and parameter enclosures are presented.

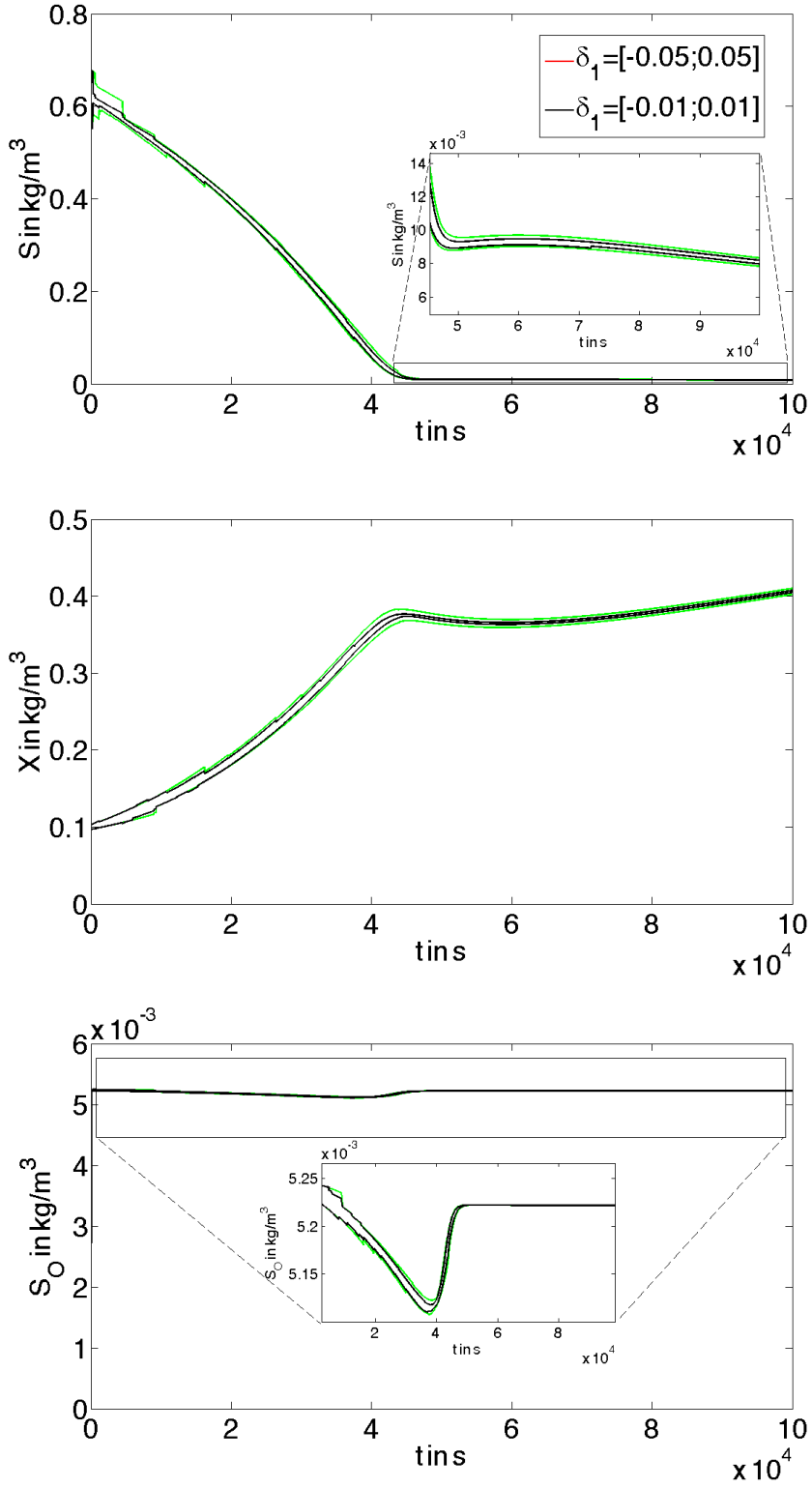


Figure 5.41: BWTP: State enclosures for $S(t)$, $X(t)$ and $S_O(t)$ for different $[\delta_1]$.

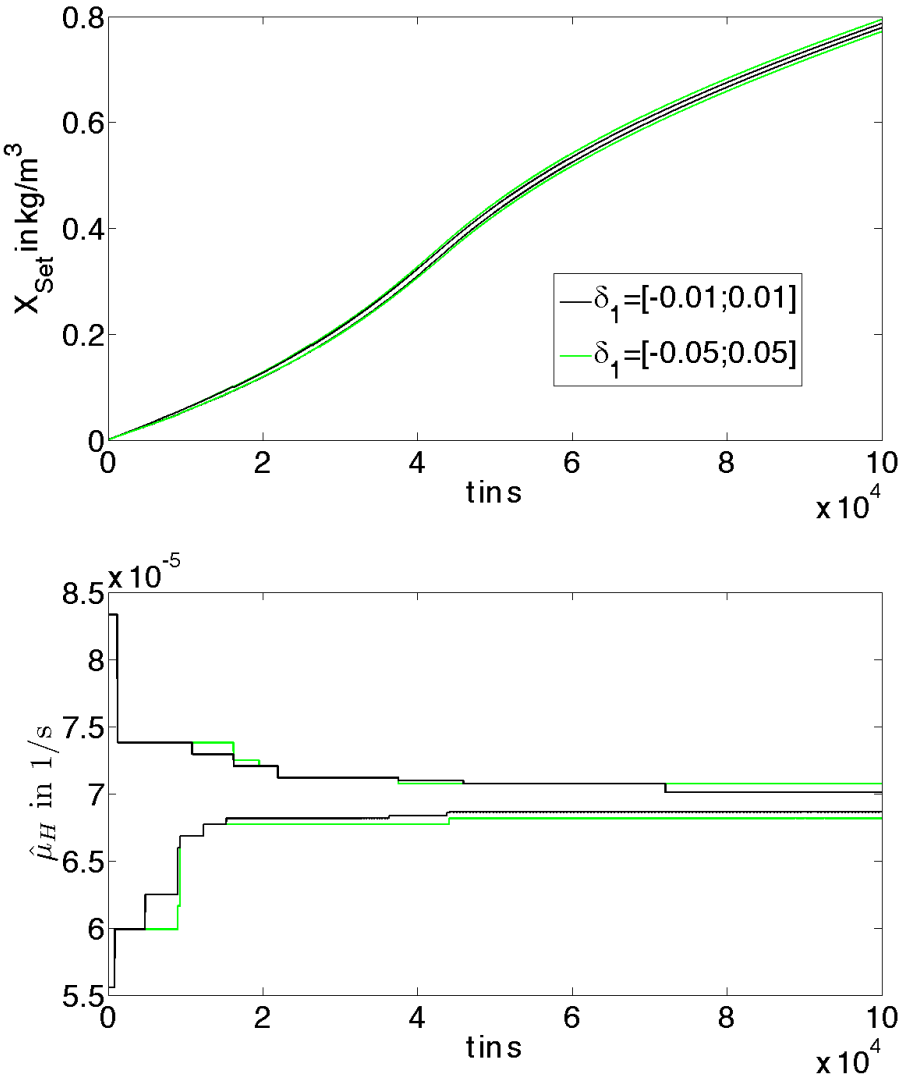


Figure 5.42: BWTP: State enclosures for $X_{Set}(t)$ and $\mu(t)$ for different $[\delta_1]$.

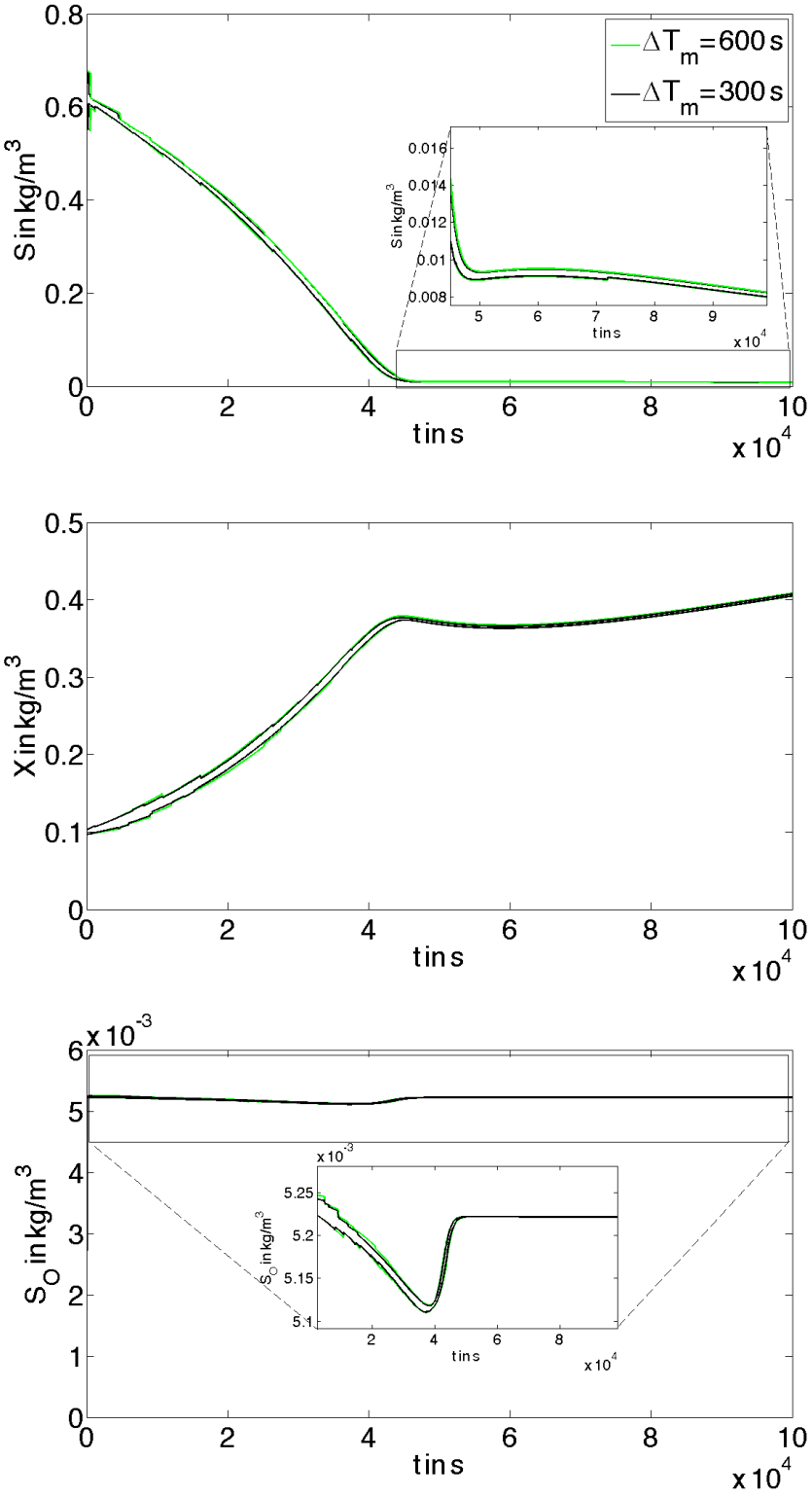


Figure 5.43: BWTP: State enclosures for $S(t)$, $X(t)$ and $S_O(t)$ for different ΔT_m.

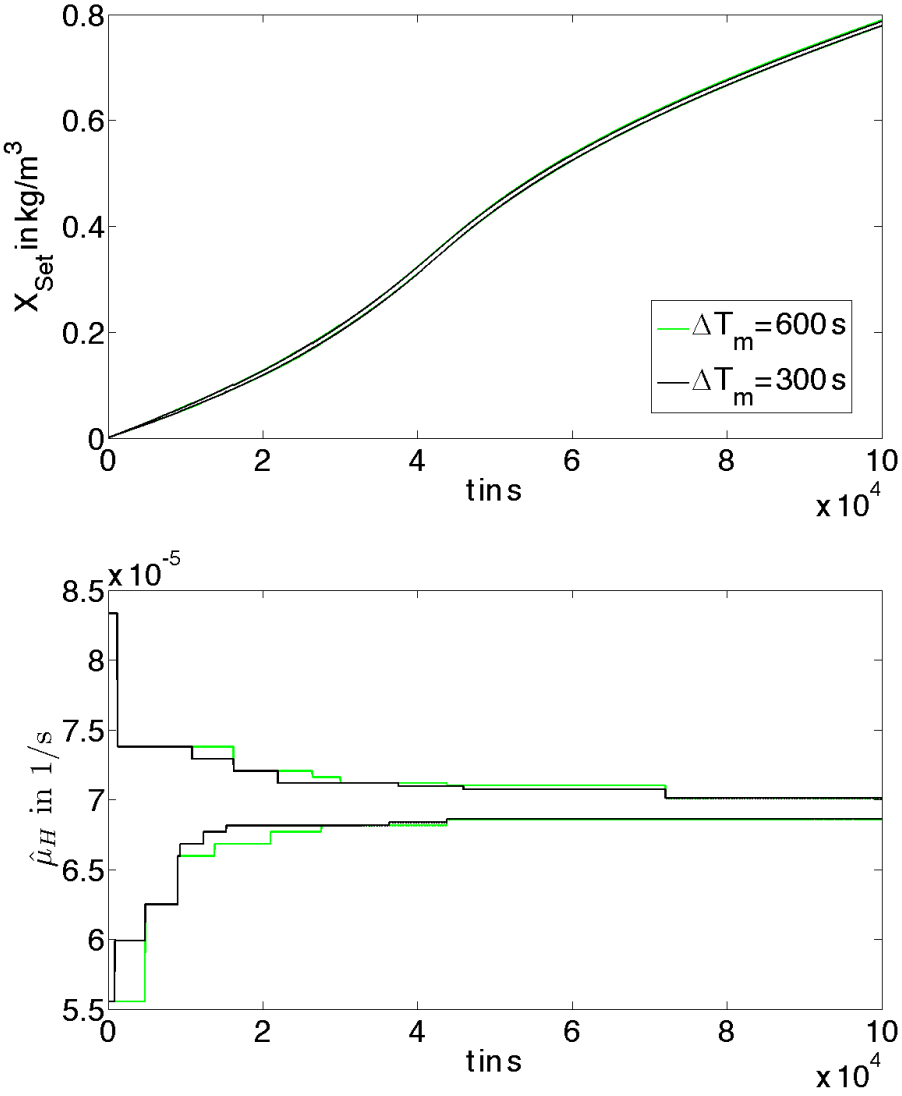


Figure 5.44: BWTP: State enclosures for $X_{Set}(t)$ and $\mu(t)$ for different ΔT_m.

5.5.1 Non-Isothermal Stirred Tank Reactor

The initial conditions were $c_a(0) \in [0;1]$mol/l, $c_b(0) \in [0;1]$mol/l, $v(0) \in [100;110]°$C, $v_K(0) \in [100;110]°$C. The parameters E_1 and E_3 were assumed to be uncertain and time-invariant, with $E_1 \in [0.9;1.1] \cdot (9758.3)$K and $E_3 \in [0.9;1.1] \cdot (8560)$K. The measurement uncertainties were assumed to be $\delta_1 \in [-0.05;0.05]$mol/l and $\delta_2 \in [-0.5;0.5]°$C for all t and it was further assumed that measurements were available every 0.005h.

The estimation parameters for the interval observer were:

- order $\nu = 2$,
- maximum number of boxes was $L_{max} = 2500$,
- maximum number of splittings was also 2500,
- hull limit $\delta_{hull,limit} = 2.5\%$,
- constant step-size $h = 0.001$,
- interval splitting and merging every 5 time-steps.

The adjustments for the Taylor model observer were:

- order $\rho = 4$ in initial state variables and time variable,
- maximum number of Taylor models was $L_{max} = 200$,
- maximum number of splittings was also 200,
- variable step-size with a maximum value $h_{max} = 0.001$ (for a successful execution of the estimation, the algorithm required a smaller step size at the beginning).

The resulting interval enclosures are depicted in Fig. 5.45 and Fig. 5.46, the corresponding diameters in Fig. 5.47 and Fig. 5.48. The interval observer leads to tighter results at the

Table 5.12: NISTR: Comparison of the computation time.

Enclosure type	Computation time in s
Taylor models	31384
Interval vectors	29908

beginning of the estimation: For c_a, E_1, and E_3 up to $t \approx 0.1$h, for c_b until $t \approx 0.15$h, for v until $t \approx 0.2$h, and for v_k up to $t \approx 0.3$. Then the results for the Taylor model observer become tighter. The state variables c_b and v are measured state variables. In contrast to the Taylor model observer, the interval observer, allows for a direct intersection of the predicted set and the measured values plus the corresponding interval of the measurement

uncertainty. That's the reason why these enclosures are much tighter at the beginning for the interval observer. Between $t = 0$ and $t \approx 0.1$ the algorithm for correction step in the Taylor model observer was only efficient in the first correction step at $t = 0.005$. Then the outer bounds of the measured state variables c_b and v in this time interval become wider than the measured value plus the measurement uncertainty. However, after $t \approx 0.1$ the width of the bounds decreases significantly. The computation times are comparable for both estimators (see Tab. 5.12).

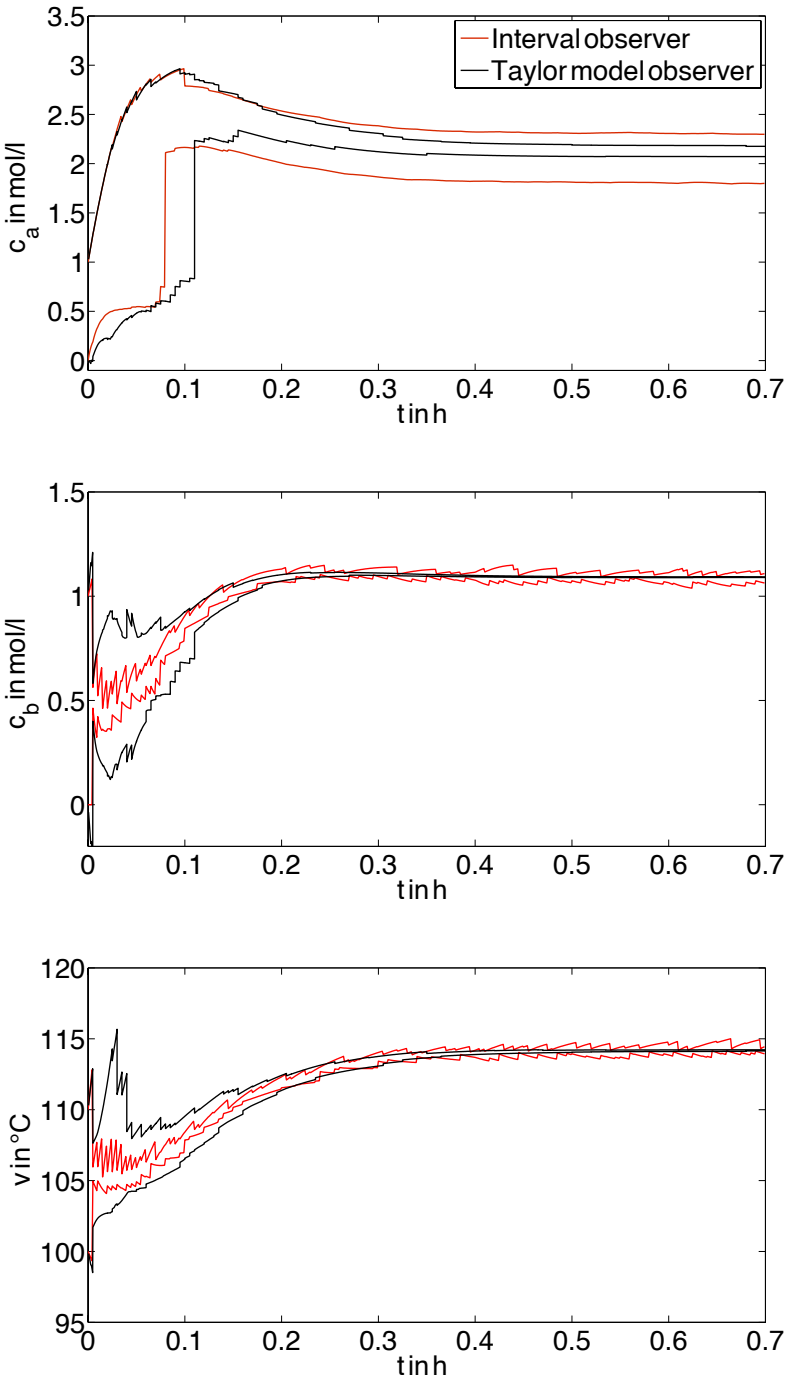


Figure 5.45: NISTR: Interval enclosures for cyclopentadien concentration c_a and cyclopentenol concentration c_b, and reactor temperature v.

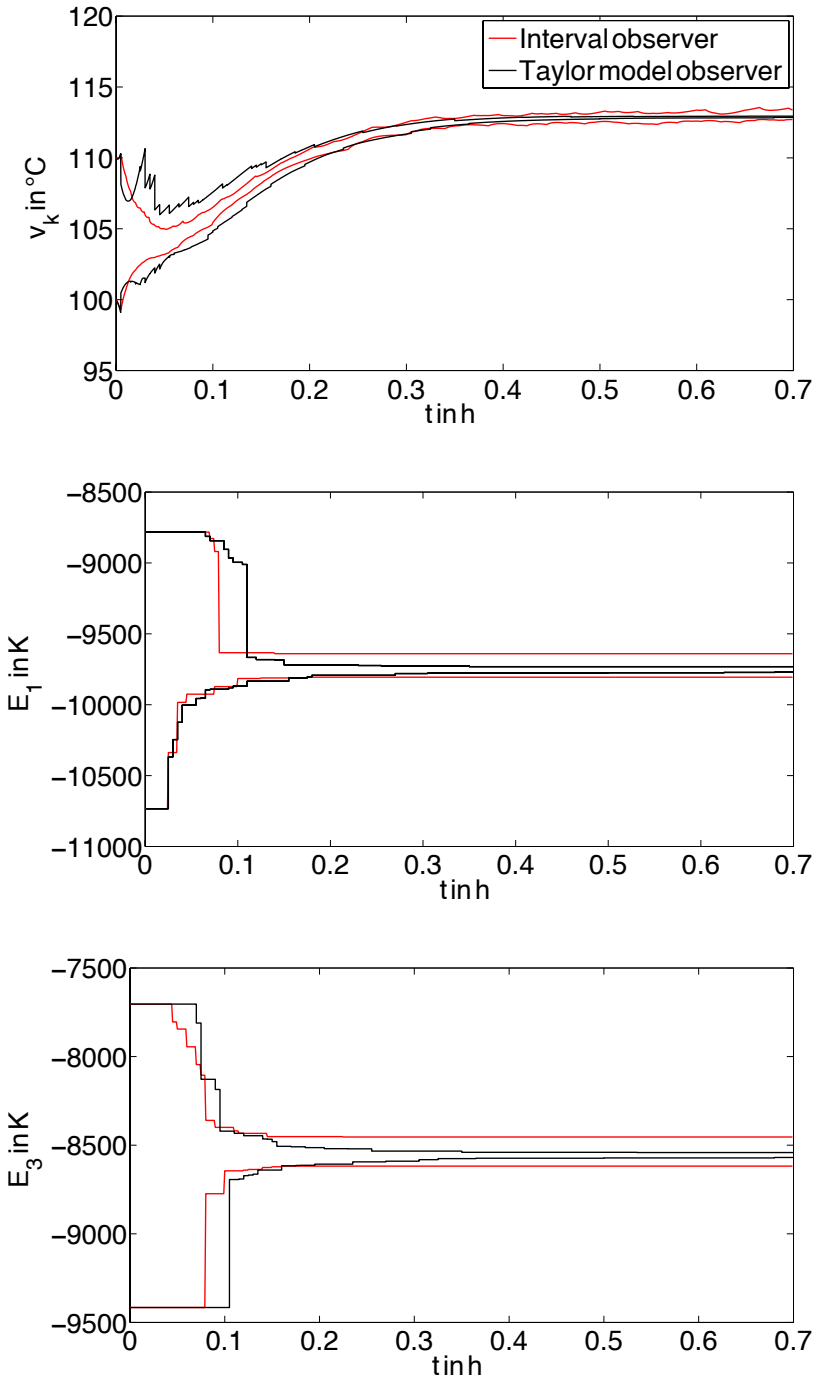


Figure 5.46: NISTR: Interval enclosures for jacket temperature v_k, for E_1, and E_3.

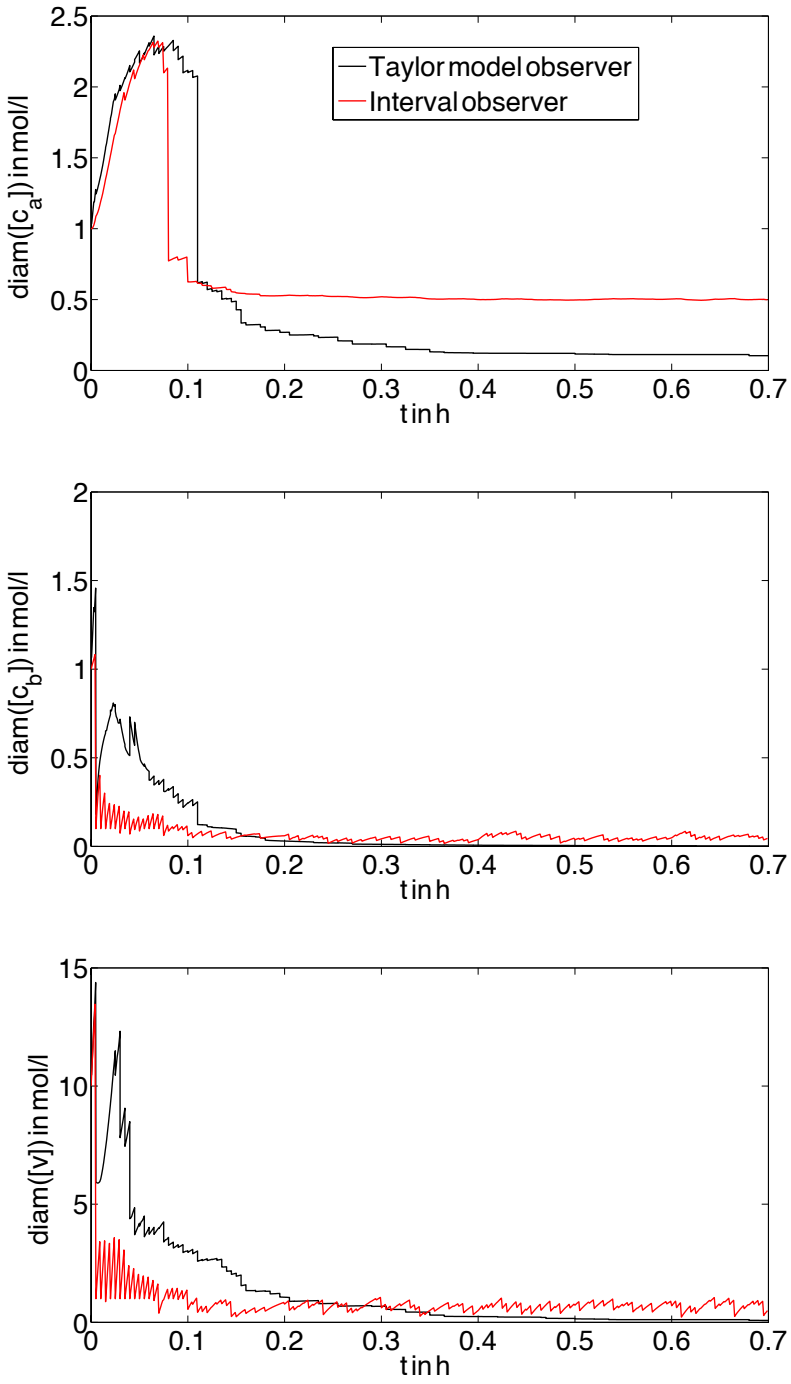


Figure 5.47: NISTR: Diameters of the outer bounds of for cyclopentadien concentration c_a and cyclopentenol concentration c_b, and reactor temperature v.

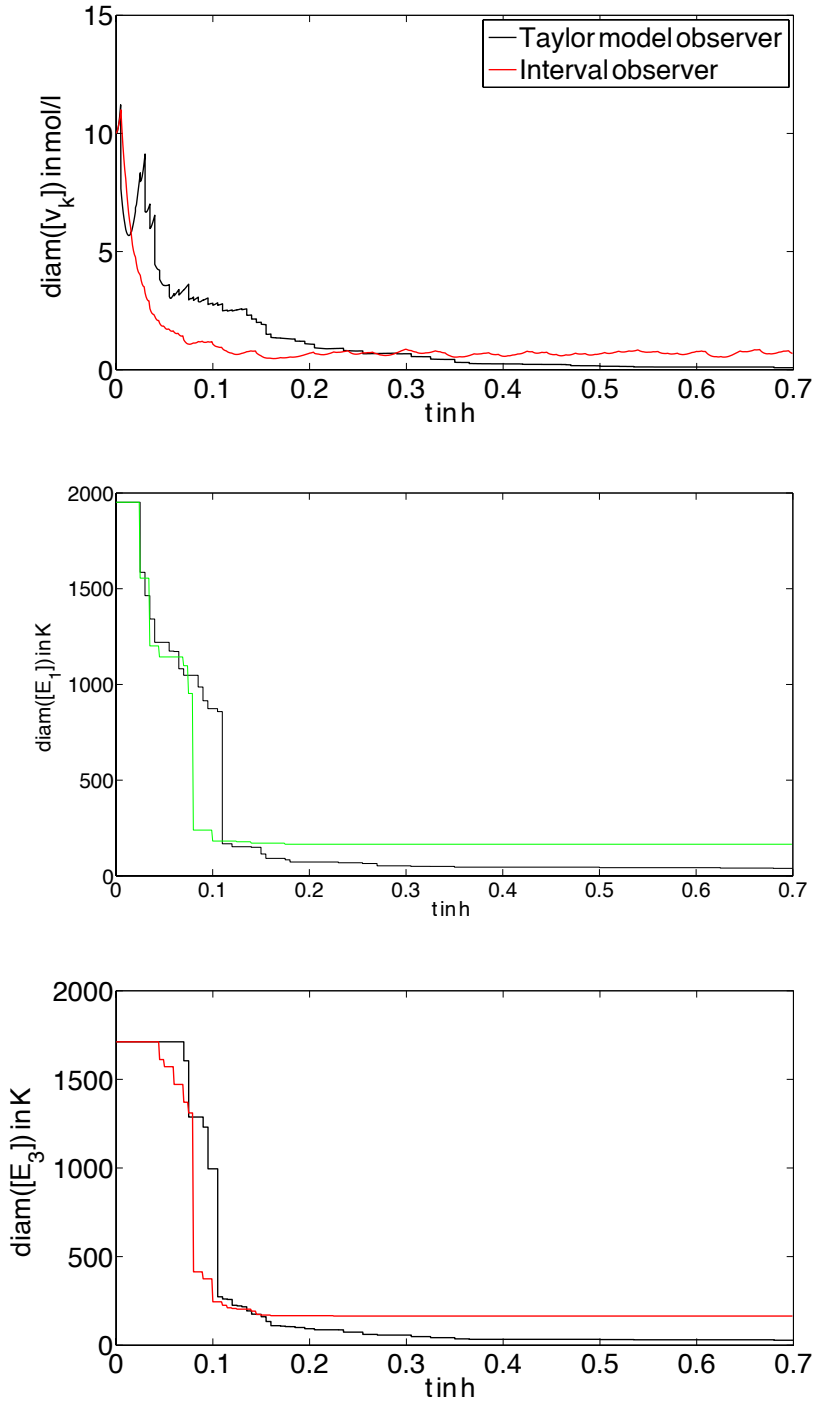


Figure 5.48: NISTR: Diameters of the outer bounds of v_k, for E_1, and E_3.

5.5.2 Double Pendulum

The initial conditions were assumed to be
$\theta_1(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]$rad, $\theta_2(0) \in [\frac{\pi}{2} - 0.03 \cdot \frac{\pi}{2}; \frac{\pi}{2} + 0.03 \cdot \frac{\pi}{2}]$rad, $\dot{\theta}_1(0) \in [-0.01; 0.01]$rad/s, and $\dot{\theta}_2(0) \in [-0.01; 0.01]$rad/s.
The assumed measurement errors were again enclosed by the intervals $[\delta_1] = [\delta_2] = [-0.01; 0.01]$m. The time between two measurements was $\Delta T_m = 0.005$s. The step-size was $h = 0.001$s. The estimation parameters for the interval observer were:

- order $\nu = 1$,
- maximum number of boxes was $L_{max} = 100$,
- maximum number of splittings was also 50,
- hull limit $\delta_{hull,limit} = 2.5\%$,
- constant step-size $h = 0.001$s,
- interval splitting and merging every 5 time-steps.

The adjustments for the Taylor model observer were:

- order $\rho = 6$ in state variables and time variable,
- maximum number of Taylor models was $L_{max} = 1$,
- no splitting of the domain interval was performed,
- constant step-size with $h = 0.001$s.

The resulting interval enclosures are depicted in Fig. 5.49 and Fig. 5.50, the corresponding diameters in Fig. 5.51 and Fig. 5.52. The quality of the enclosures of θ_1 and θ_2 are comparable for both methods, as in some time intervals the interval observer provides tighter enclosures; in some time intervals the Taylor model observer performs better. However, the bounds for $\dot{\theta}_1$ and $\dot{\theta}_2$ are tighter for the Taylor model observer for the whole considered time horizon. The computation time was also much shorter (see Tab. 5.13).

Table 5.13: Double Pendulum: Comparison of the computation time.

Enclosure type	Computation time in s
Taylor models	841
Interval vectors	2090

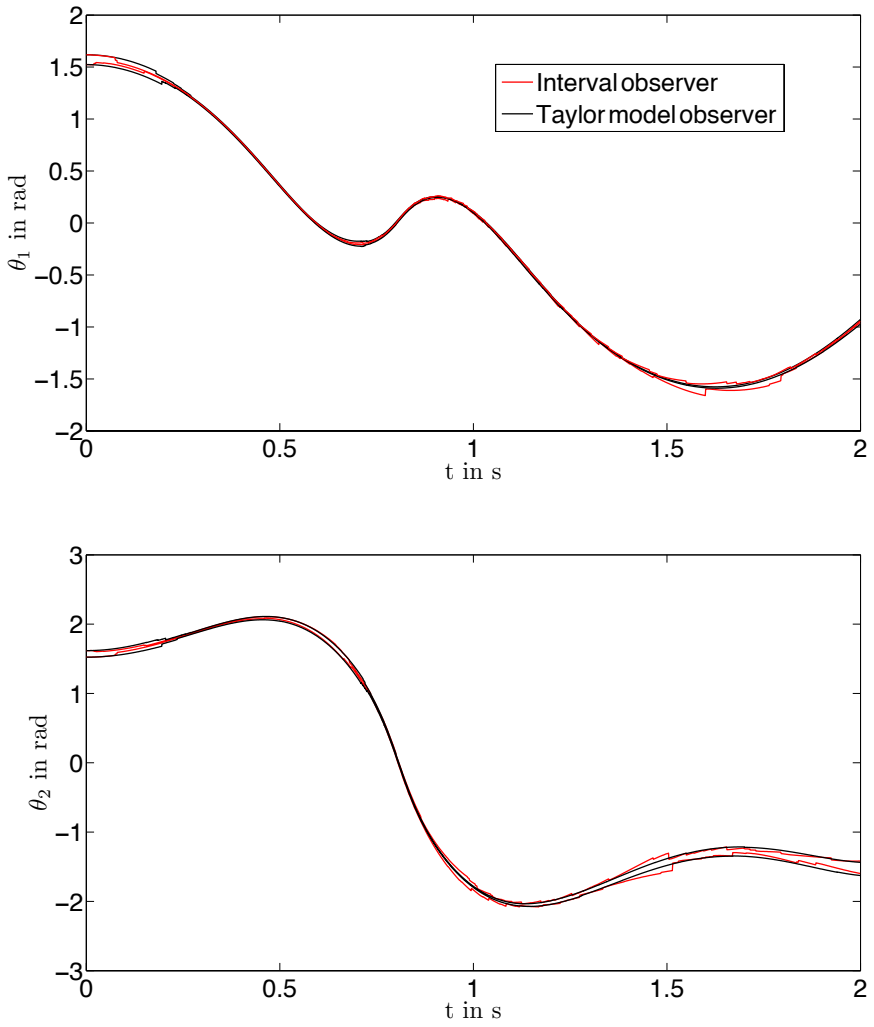


Figure 5.49: Double pendulum: Interval enclosures for θ_1 and θ_2..

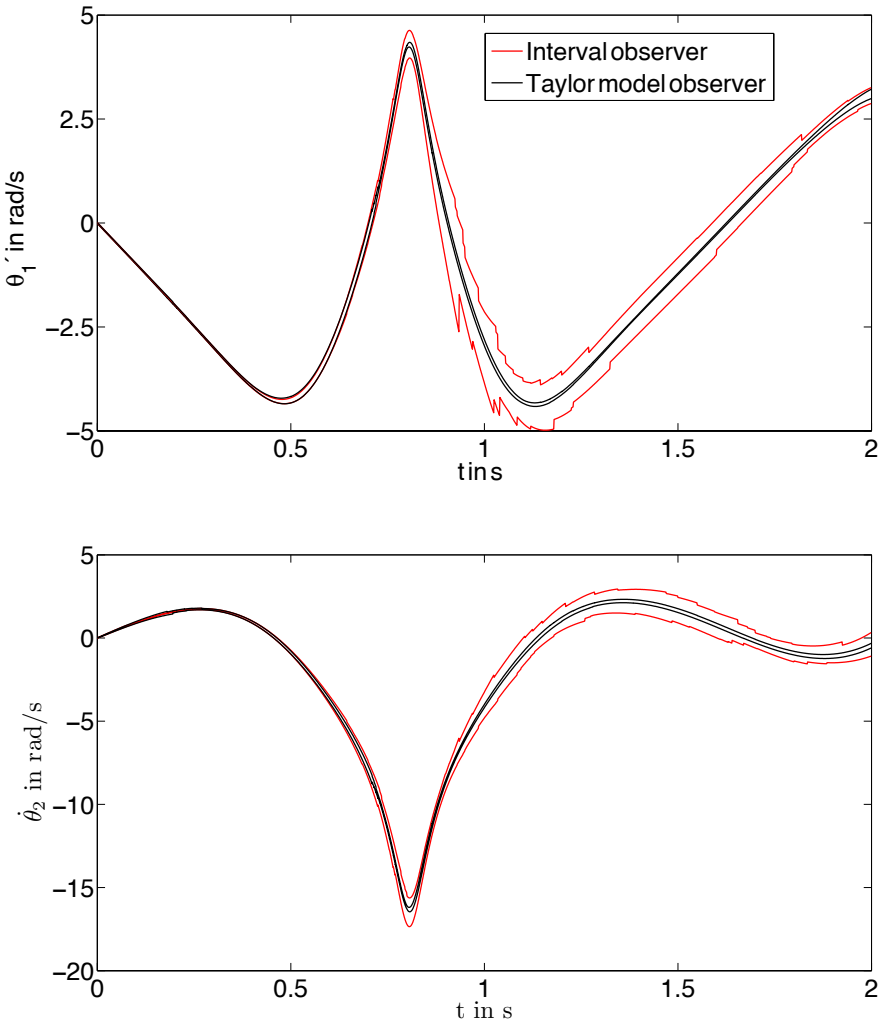


Figure 5.50: Double pendulum: Interval enclosures for $\dot{\theta}_1$ and $\dot{\theta}_2$.

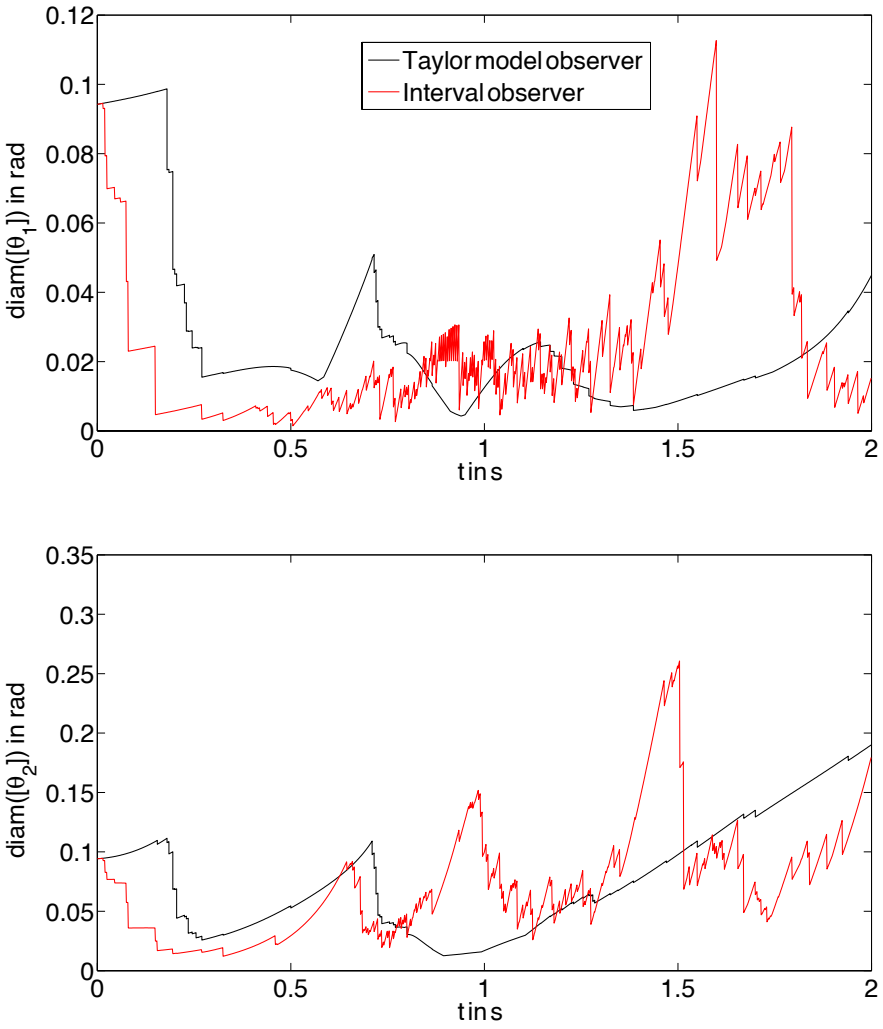


Figure 5.51: Double pendulum: Diameters of the outer bounds for θ_1 and θ_2.

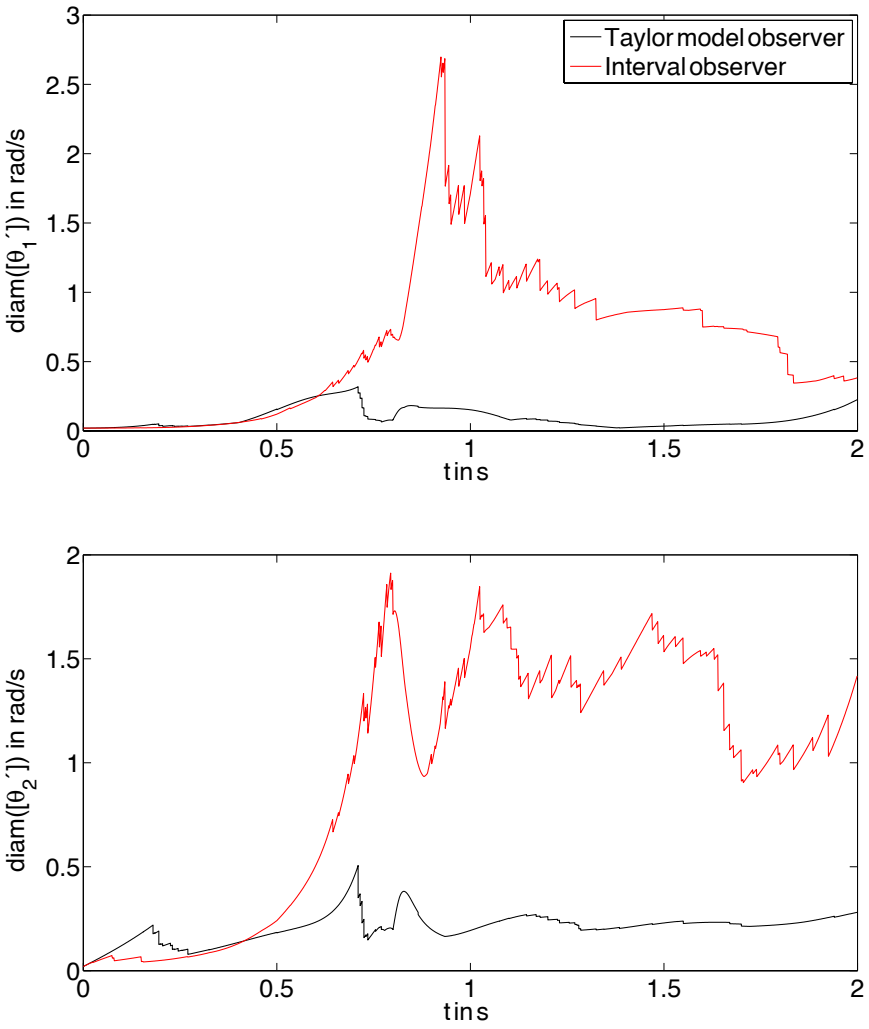


Figure 5.52: Double pendulum: Diameters of the outer bounds of $\dot{\theta}_1$ and $\dot{\theta}_2$.

5.5.3 Biological Waste Water Treatment Plant

Finally, an estimation comparison for the biological waste water treatment plant is given. The initial state variables were given by $S(0) \in [0.9; 1.1] \cdot 0.616$ kg/m^3, $X(0) \in [0.97; 1.03] \cdot 0.1$ kg/m^3, $S_O(0) \in [0.99; 1.01] \cdot S_{O,sat}/2 \cdot 10^{-4}$ kg/m^3 and $X_{Set}(0) \in [0.9; 1.1] \cdot 0.001$ kg/m^3. The maximum specific growth rate $\hat{\mu}$ was assumed to be uncertain and time-invariant with $\hat{\mu} \in [0.8; 1.2] \cdot 1/14400$ 1/s. The interval bounds for the measurement errors were given by $\delta_1 = [-0.01; 0.01]$ kg/m^3 and $\delta_2 = [-0.05; 0.05]$ kg/m^3. Measured values were available every $\Delta T_m = 300$s.

The estimation parameters for the interval observer were:

- order $\nu = 1$,
- maximum number of intervals was $L_{max} = 200$,
- maximum number of splittings was also 100,
- hull limit $\delta_{hull,limit} = 2.5\%$,
- constant step-size $h = 10$,
- interval splitting and merging every 5 time-steps.

The adjustments for the Taylor model observer were:

- order $\rho = 4$ in state variables and and time variable,
- maximum number of Taylor models was $L_{max} = 40$,
- maximum number of splittings was also 40,
- variable step-size with a maximum value $h_{max} = 10$s (for a successful execution of the estimation, the algorithm required a smaller step size at the beginning).

The resulting interval enclosures are shown in Fig. 5.55 and Fig. 5.54. The bounds of the substrate concentration S, which is measured directly, are tighter for the interval observer up to $t \approx 4.6 \cdot 10^4$s. The reason for this is that a direct intersection of the predicted set and measured values plus measurement uncertainty is only possible for the interval observer. The bounds for the bacteria concentrations X and X_{Set} are tighter up to $t \approx 3 \cdot 10^4$s. Then the Taylor model observer gives tighter bounds, especially for X. The estimation results for the oxygen concentration S_O and the maximum specific growth rate $\hat{\mu}_H$ is more precise when using the Taylor model observer. The computation time was of similar magnitude for both estimators (see Tab. 5.14).

Table 5.14: BWTP: Comparison of the computation time.

Enclosure type	Computation time in s
Taylor models	30140
Interval vectors	33456

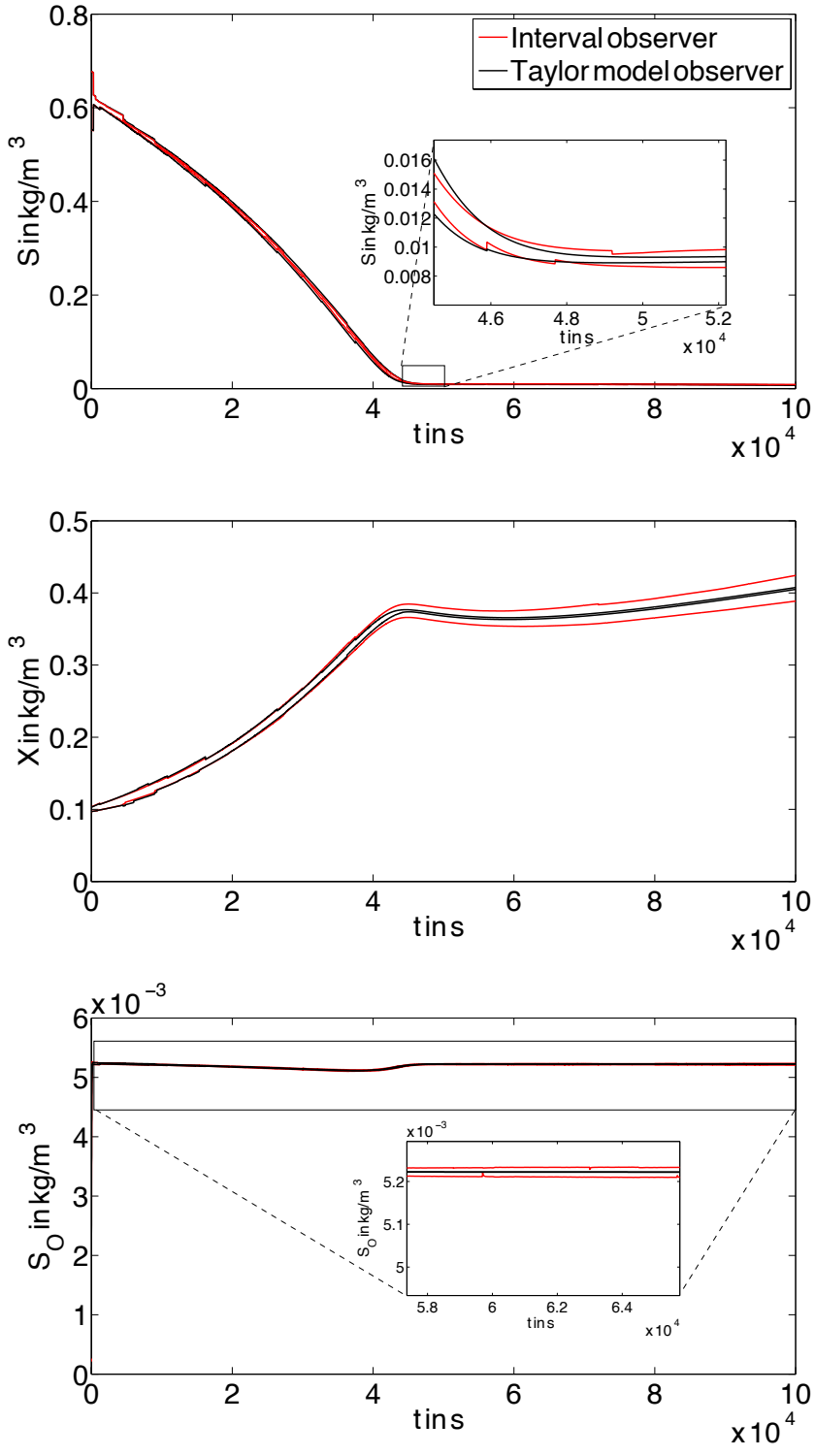


Figure 5.53: BWTP: Interval enclosures of the concentrations S, X, and S_O.

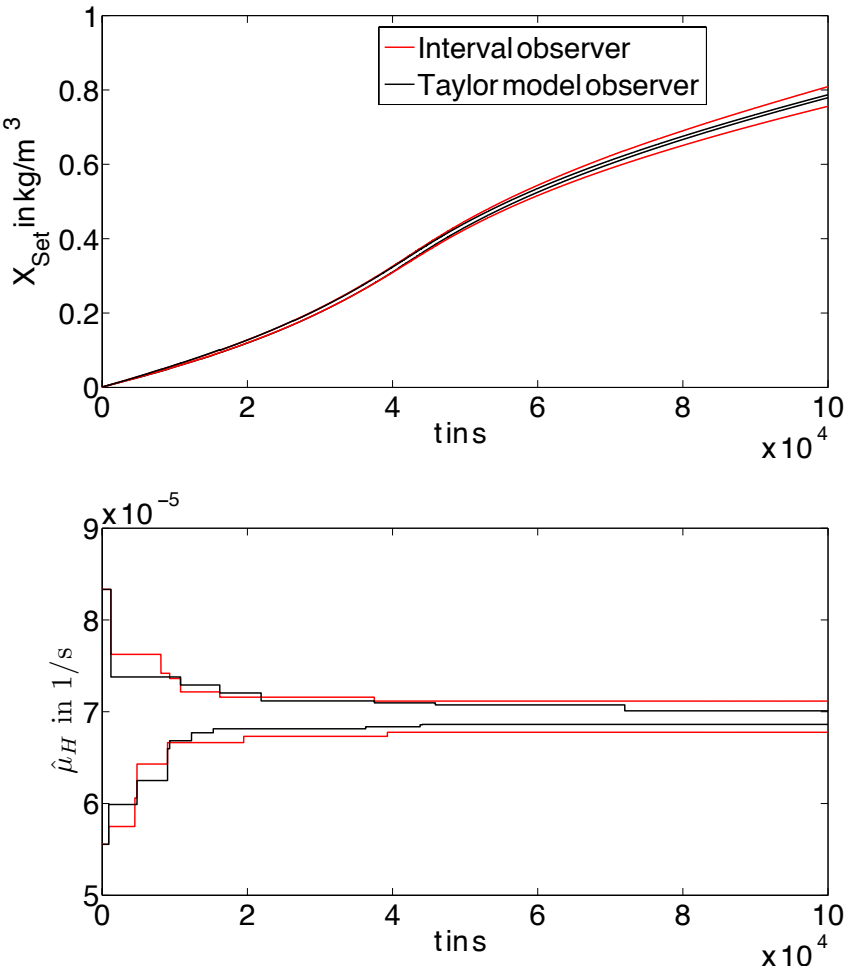


Figure 5.54: BWTP: Interval enclosures of X_{Set} and $\hat{\mu}$.

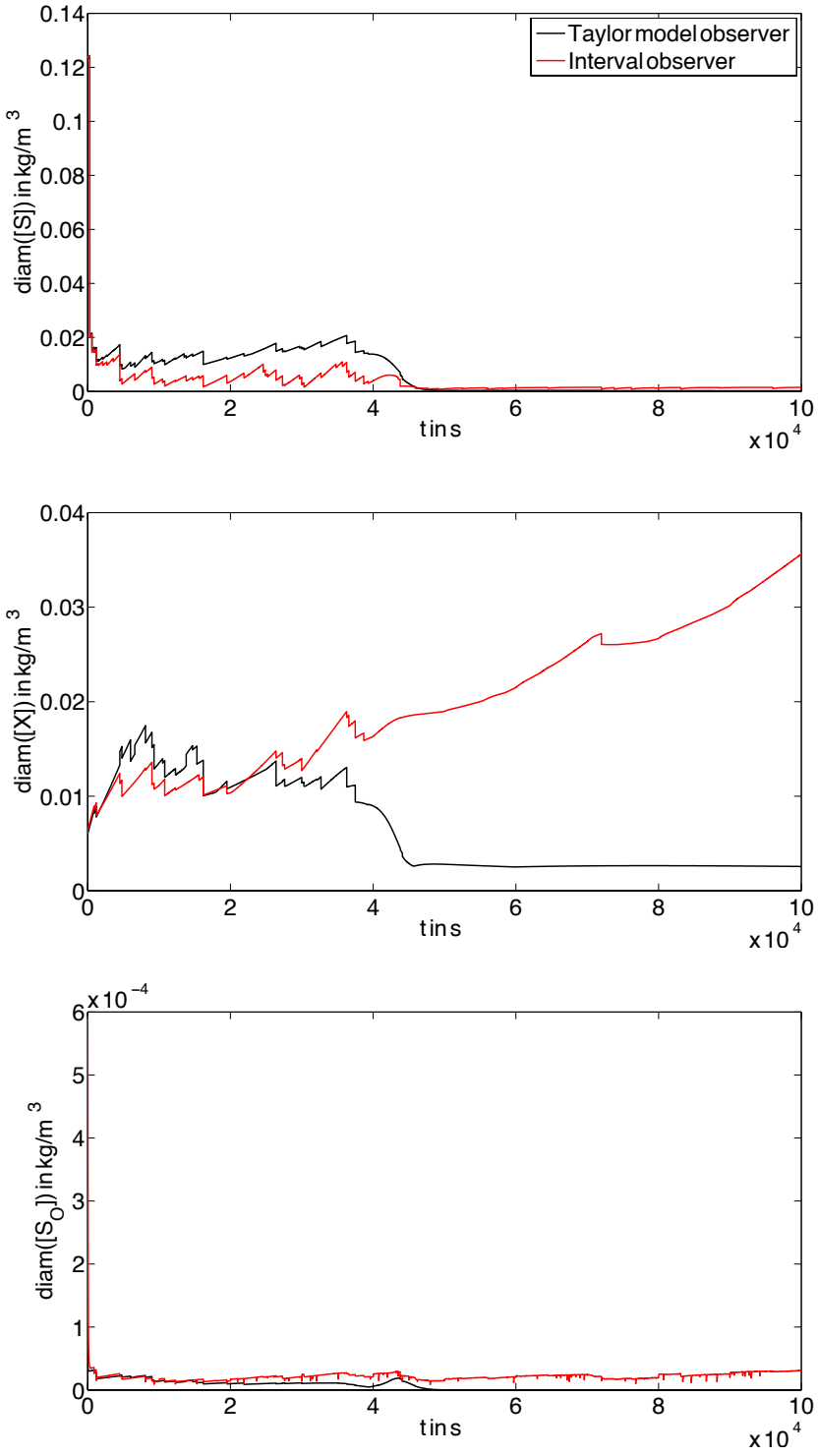


Figure 5.55: BWTP: Diameters of the outer bounds of for the concentrations S, X, and S_O.

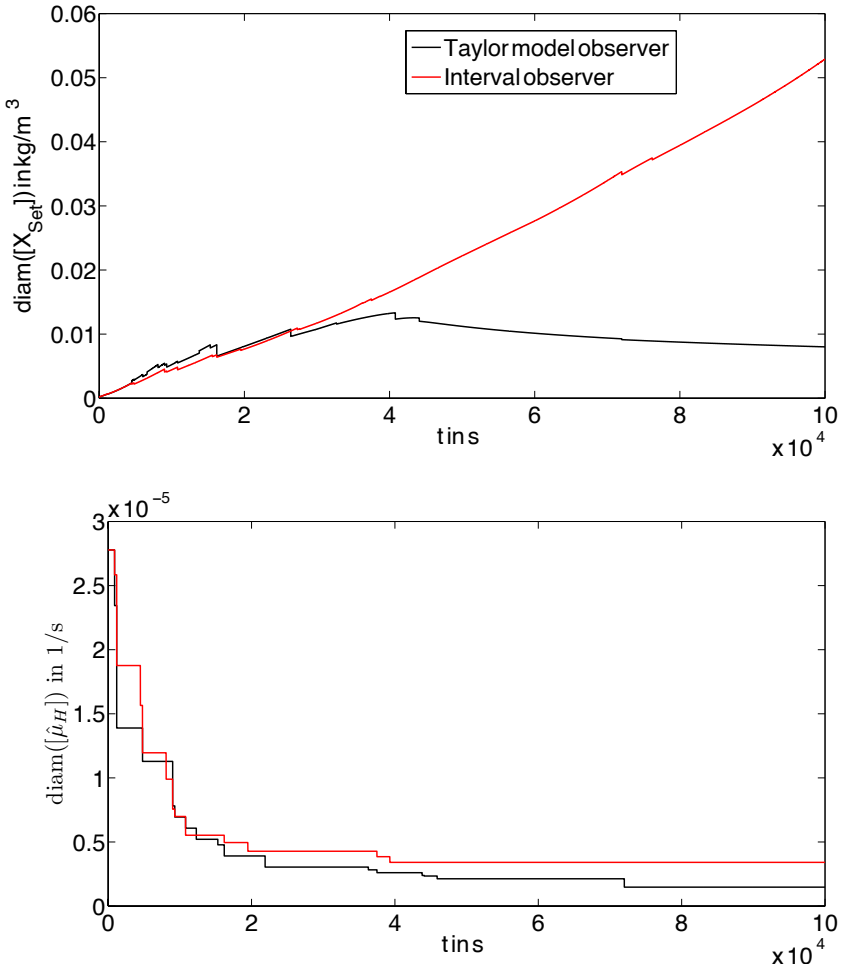


Figure 5.56: BWTP: Diameters of X_{Set} and $\hat{\mu}$.

5.6 Summary

In the following, the main results of this chapter are summarized. For the interval observer one of the main insights is that when increasing the order ν of the Taylor series expansion above order $\nu = 2$ the additional obtained improvement for each further rising of ν is decreasing very quickly. While estimation results for different orders where only presented for the NISTR, similar results are also obtained for the other applications. The main goal of increasing the order is to reduce the size of the remainder error. However, by doing so not only computation time is increased, but also more expressions have to be evaluated with interval methods. This may lead – depending on the system – to more overestimation. This is especially the case, when the diameters of the corresponding interval arguments are large. If there is no improvement by increasing the order above a given value it is more efficient to use more subboxes. This is required anyway, if like in most nonlinear systems, the solution set is complexly shaped. The results have also pointed out that an additional coordinate transformation can be useful for further reduction of overestimation.

For the Taylor model observer similar things can be observed. Increasing the order is limited when the uncertainty is large. Here splitting of the domain interval vector is much more efficient. Preconditioning is a very important tool in the verified integration based on Taylor models. The choice of the most appropriate preconditioning method is crucial. In the estimation process of the presented results QR preconditioning has been used, since it has been shown to be the most efficient for this application. However, in the example of the BWTP some modifications of the preconditioning method for a successful estimation have to be conducted.

For the NISTR the Taylor model observer required more computation effort for a successful estimation than the interval observer. However, if the number of interval boxes is increased until the computation time is approximately the same for both estimators, the Taylor model observer provided tighter bounds in average. However, for the directly measured state variables the enclosures resulting from the estimation by the interval observer are still better at the beginning of the estimation process.
For the double pendulum the Taylor model observer performed better than the interval observer.
The results in case of the BWTP are similar to the results for the NISTR. In average the bounds resulting from the Taylor model observer are better. Still, at the beginning and especially for the measured state variable S the bounds are tighter for the interval observer. The Taylor model observer required a step size control for the NISTR and the BWTP, employing smaller step-sizes at the beginning.

Thus, it can be concluded that depending on the nature of the measurement equation and on the time horizon of interest one or the other of the two estimators performs better.

6 Verified Methods for Guaranteed Robust Tracking with Flatness Based Controllers

6.1 Problem Formulation

Verified state and parameter estimation can also be understood as a verified robustness analysis of a control system, which includes a traditional observer, e.g. a tracking observer [16]. The robustness analysis then determines guaranteed enclosures of the set of the uncertain system parameters and uncertain initial state variables for which prespecified robustness requirements are fulfilled.
In this thesis a flatness based control system is considered. Flatness based controller design [14, 15, 71] is a powerful tool for motion planning and trajectory tracking for linear and nonlinear systems. Especially, for nonlinear systems there is a wide acceptance of this approach, which has been applied successfully to numerous problems of industrial relevance.
Roughly speaking, the flatness property of a nonlinear system is characterized by the existence of a — possibly fictitious — flat output that allows a differential parameterization of the state variables and inputs. Based on the differential parameterization a tracking controller for a given reference trajectory of the flat output can be designed. In general, not all system state variables which are necessary to implement the tracking controller can be measured . In this case a nonlinear tracking observer as proposed in [16] can be used.

Interval methods are used to analyze the dynamic behavior of the control system [3,33,34,60] which is described by a system of nonlinear differential equations. Applying these techniques the maximum admissible range of parameter uncertainty in the plant is determined such that the flow of the uncertain control system evolves within specified tolerances. More detailed, subboxes of the parameter uncertainty are considered for a verified integration over the desired time span. A subbox is admissible if the resulting enclosures over the complete time span lie completely inside the specified tolerances for robustness. If the enclosures are completely outside the specified tolerances for at least one point of time, the corresponding subbox is not admissible. Further splitting is required to decide about the admissibility of all remaining interval vectors. In addition to uncertainty of parameters of the plant also interval uncertainty of the initial conditions and the available measured data can be considered.

The methodology for robustness analysis is restricted here to single-input systems. However, with obvious extensions it can also be applied to multi-input systems and a wide class of controllers and dynamical systems.

6.2 Flatness Based Controller Design

6.2.1 Flatness

Flatness based controller design has been introduced e.g. in [14] (differential algebraic setting) and [15] (differential geometric setting). Various aspects of flatness are illustrated e.g. in [71]. In this contribution the following relations for nonlinear single input systems are used, where explicitely the dependence of the relations on the parameters are stated:
For a flat system

$$\dot{x} = f_x(p, x, u) \tag{6.1}$$

with $x \in \mathbb{R}^{n_x}$, the control variable $u \in \mathbb{R}$ and the parameter vector $p \in \mathbb{R}^{n_p}$ the flatness property implies the existence of a flat output $y_f \in \mathbb{R}$, such that

$$\begin{aligned} y_f &= h_f(p, x) \ , & (6.2)\\ x &= \psi_x(p, y_f, \dot{y}_f, \ldots, y_f^{(n-1)}) \ , & (6.3)\\ u &= \psi_u(p, y_f, \dot{y}_f, \ldots, y_f^{(n)}) & (6.4) \end{aligned}$$

holds, with h_f, ψ_x, ψ_u smooth at least on an open subset of $\mathbb{R}$, $\mathbb{R}^{n_x}$, and $\mathbb{R}$, respectively. Introducing the new coordinates

$$\zeta = (\zeta_1, \ldots, \zeta_n) = (y_f, \dot{y}_f, \ldots, y_f^{(n-1)}) \ , \tag{6.5}$$

the flat system (6.1) can be transformed via the well defined diffeomorphism

$$\zeta = \Phi(p, x) \tag{6.6}$$

into controller normal form

$$\begin{aligned} \dot{\zeta}_i &= \zeta_{i+1}, \qquad i = 1, 2, \ldots n-1 \ , \\ \dot{\zeta}_n &= \alpha(p, \zeta, u) \ . \end{aligned} \tag{6.7}$$

Setting $v = y_f^{(n)}$ yields

$$u = \psi_u(\zeta, v, p) \tag{6.8}$$

in view of (6.4) and (6.5). In [20] it has been shown that

$$\alpha(p, \zeta, \psi_u(p, \zeta, v)) = v \tag{6.9}$$

holds and thus by application of the feedback law (6.8), system (6.1) is diffeomorphic to the Brunovský normal form

$$\begin{aligned} \dot{\zeta}_i &= \zeta_{i+1}, \qquad i = 1, 2, \ldots n-1 \ , \\ \dot{\zeta}_n &= v \end{aligned} \tag{6.10}$$

with new input v.

6.2.2 Flatness Based Feedforward Controller

Due to the derived relations in Section 6.2.1 a (sufficiently smooth) reference trajectory $y_{f,d} : [t_0, t_0 + T] \to \mathbb{R}$ for the flat output y_f can be assigned almost arbitrarily (excluding singularities of the differential parameterization (6.3)–(6.4)). If the reference trajectory $y_{f,d}$ satisfies the boundary conditions

$$x(t_0) = \psi_x(p_{nom}, y_{f,d}(t_0), \dot{y}_{f,d}(t_0), \ldots, y_{f,d}^{(n-1)}(t_0)) \ , \tag{6.11}$$

then a corresponding feedforward controller that provides $y_f(t) = y_{f,d}(t)$ for $t \in [t_0, t_0 + T]$ is given by

$$u_d(t) = \psi_u(p_{nom}, y_{f,d}(t), \dot{y}_{f,d}(t), \ldots, y_{f,d}^{(n)}(t)) \ . \tag{6.12}$$

For (6.11) and (6.12) it has been assumed that the parameters of the plant (6.1) match a nominal parameter vector p_{nom}.

6.2.3 Flatness Based Tracking Controller design

To stabilize the tracking of a given reference trajectory $y_{f,d}$ for the flat output, the tracking error e is introduced as

$$e = y_f - y_{f,d} = \zeta_1 - \zeta_{1,d} \ . \tag{6.13}$$

In view of (6.10) it follows that

$$e^{(i)} = \zeta_{i+1} - \zeta_{i+1,d}, \quad i = 0, 1, \ldots, n-1 \ . \tag{6.14}$$

Thus, when setting the new input v in (6.10) to

$$v = \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \lambda_i (\zeta_{i+1} - \zeta_{i+1,d}) = \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \lambda_i e^{(i)} \ , \tag{6.15}$$

the tracking error obeys the differential equation

$$0 = e^{(n)} + \sum_{i=0}^{n-1} \lambda_i e^{(i)} \tag{6.16}$$

which can be achieved to be stable by suitable choice of the λ_i. Substituting (6.15) into the differential parameterization (6.4) of the input yields in view of (6.5) the feedback law

$$u = \psi_u(p, y_f, \dot{y}_f, \ldots, y_f^{(n-1)}, y_{f,d}, \dot{y}_{f,d}, \ldots, y_{f,d}^{(n)}) \ . \tag{6.17}$$

Using the diffeomorphism (6.6), the feedbacklaw (6.17) can be implemented as

$$u = \psi'_u(p_{nom}, x, y_{f,d}, \dot{y}_{f,d}, \ldots, y_{f,d}^{(n)}) = \psi''_u(p_{nom}, x, t) \ , \tag{6.18}$$

where again the plant parameters p are assumed to be equal to the nominal parameter vector p_{nom}. As a consequence, for the feedback controller (6.18), the controlled system can be summarized as

$$\dot{x} = f_x(p, x, \psi''_u(p_{nom}, x, t)) = f_{fb}(p, x, t) \ , \tag{6.19}$$

where $p \neq p_{nom}$ can occur due to not exactly known parameters. To improve the robustness of the tracking controller an integral error feedback is often introduced, i.e. the error feedback (6.15) is extended according to

$$\begin{aligned} \dot{e}_I &= \zeta_1 - \zeta_{1,d} \ , \\ v &= \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \lambda_i e^{(i)} - \lambda_{-1} e_I \ . \end{aligned} \tag{6.20}$$

This feedback can clearly be implemented as a state feedback of the kind

$$\begin{aligned} \dot{e}_I &= h_f(p,x) - y_{f,d}(t) \ , \\ u &= \psi''_{u,I}(p_{nom}, x, e_I, t) \ . \end{aligned} \tag{6.21}$$

6.2.4 Tracking using a Nonlinear Tracking Observer

For the implementation of the feedback (6.18), in general, all state variables have to be available for measurement. If only the output

$$y = h(p,x) \tag{6.22}$$

is available for measurement, a nonlinear tracking observer with time varying observer gain $G(t)$

$$\begin{aligned} \dot{\hat{x}} &= f_x(p_{nom}, \hat{x}, u) + G(t)(y - h(p_{nom}, \hat{x})) \\ &= f_{obs}(p_{nom}, x, \hat{x}, u, t) \end{aligned} \tag{6.23}$$

as proposed in [16] can be applied. The observer (6.23) basically consists of a model of the plant and a feedback of the difference of the measured output and the estimated output. For the model of the plant also the nominal parameter values p_{nom} are used. The time varying observer gain $G(t)$ is designed such that the linearization of the estimation error dynamics about the reference trajectory $y_{f,d}$ which result to

$$\Delta\dot{\hat{x}} - \Delta\dot{x} = (A(t) - G(t)C(t))(\Delta\hat{x} - \Delta x) \tag{6.24}$$

with

$$A(t) = \left.\frac{\partial f_x}{\partial x}\right|_{x_d,u_d}, \qquad C(t) = \left.\frac{\partial h}{\partial x}\right|_{x_d,u_d} \tag{6.25}$$

are stable. For the stabilization of (6.24), i.e. of the estimation error dynamics in the vicinity of the reference trajectory $y_{f,d}$, methods for linear time varying systems as proposed in [18] can be used. Using the tracking observer (6.23) the feedback (6.18) is estimated using the observer state vector $\hat{x}$

$$\hat{u} = \hat{\psi}(p_{nom}, \hat{x}, t) \ . \tag{6.26}$$

Thus, the controlled system can be summarized as

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} f_x(p, x, \hat{\psi}(p_{nom}, \hat{x}, t), t) \\ f_{obs}(p_{nom}, x, \hat{x}, \hat{\psi}(p_{nom}, \hat{x}, t), t) \end{bmatrix} = f_{fbo}(p, x, \hat{x}, t) \ . \tag{6.27}$$

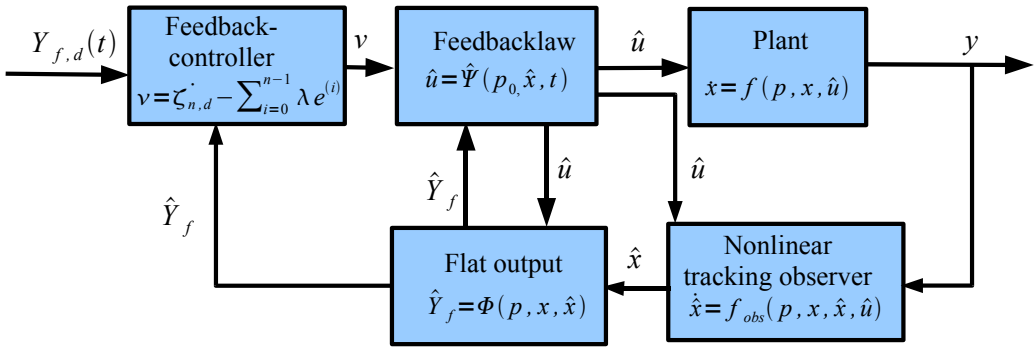


Figure 6.1: Blockdiagram of the controlled system.

A block diagram of the controlled system is shown in Fig. 6.1. Here the vector $\hat{Y}_f$ contains the flat output and its derivatives up to order $n-1$. It can easily be deduced that when using an observer together with the controller (6.21) which includes integral error feedback the following structure results

$$\begin{aligned}\begin{bmatrix}\dot{e}_I \\ \dot{x} \\ \dot{\hat{x}}\end{bmatrix} &= \begin{bmatrix} h_f(p,x,\hat{x}) - y_{f,d}(t) \\ f(p,x,\hat{\psi}(p_{nom},\hat{x},t),t) \\ f_{obs}(p_{nom},x,\hat{x},\hat{\psi}(p_{nom},\hat{x},t),t)\end{bmatrix} \\ &= f_{fbo,I}(p,x,\hat{x},e_I,t) \ . \end{aligned} \tag{6.28}$$

The controlled systems (6.27) and (6.28) have a similar structure as (6.19). This structure can be analyzed using the methods discussed in Section 6.3.

6.3 Robustness Analysis of the Tracking Controller

Assume that there are constraints for the at most tolerable deviations from the reference trajectory x_d for the controlled system which are specified in the following manner

$$|x_i(t) - x_{i,d}(t)| < d_i, \quad i = 1,2,\ldots,n_x; \ \forall\, t \in [0,T] \ . \tag{6.29}$$

In the sequel it will be shown that, using verified integration methods, it is possible to determine the admissible set Ω_{in} of parameter values p and initial conditions $x(0)$ such that the tracking controllers can meet the specification (6.29) [3, 33, 34].

The goal is to determine parameter values and values for the initial state variables

$$\Omega_{in} = \left\{ \begin{bmatrix} x(0) \\ p \end{bmatrix} \middle| \, |x_i(t) - x_{i,d}(t)| \le d_i \forall t \in [t_0, t_0+T], \ \ i = 1,2,\ldots,n_x \right\} , \tag{6.30}$$

for which it can be guaranteed that the conditions for robustness in (6.29) are fulfilled and those parameter values

$$\Omega_{out} = \left\{ \begin{bmatrix} x(0) \\ p \end{bmatrix} \middle| \, |x_i(t) - x_{i,d}(t)| > d_i \quad \text{for any} \quad t \in [t_0, t_0 + T],\ i = 1, 2, \ldots, n_x \right\}, \quad (6.31)$$

for which it can be guaranteed that these conditions are not fulfilled. This problem formulation is similar to the problem formulation for the algorithm SIVIA (set inversion via interval analysis) [27]; but it is used in a new context. Uncertain parameters are bounded by $p \in [\underline{p}, \overline{p}]$, the initial state vector is bounded by $x(0) \in [\underline{x}(0), \overline{x}(0)]$ and d_i are the allowed tolerances around the reference trajectories for the state variables x_i.

The determination of Ω_{in} and Ω_{out} can be done by splitting $[p]$ and $[x(0)]$ in subboxes. Here the state vector x, the uncertain parameters p, the estimated state vector $\hat{x}$ and if applicable the differential equation for the error e_I are combined in an extended state vector $z = [e_I, x^T, \hat{x}^T, p^T]^T$. Thus, when the $[p]$ and $[x(0)]$ are split, the interval vector $[z(0)]$ is split into subboxes

$$[\tilde{z}^{(l)}(0)], l = 1, 2, \ldots, L, \quad \bigcup_{l=1}^{L} [\tilde{z}^{(l)}(0)] = [z(0)] \ . \quad (6.32)$$

For each subbox a verified integration is performed. Here, the approach based on Taylor models as described in Section 4.4 is used. The algorithm is illustrated in Fig. 6.2. First an interval vector $[\tilde{z}^{(l)}(0)]$ is selected for the robustness analysis, then a splitting criterion is evaluated and the selected box is split. For the split subboxes a verified integration of the system model is performed. Then, three cases have to be distinguished:

1. If for some $t \in [t_0, t_0+T]$, the resulting enclosure of the trajectory is completely outside the specified tolerances the corresponding box is inconsistent and can be deleted.

2. If on the other hand the resulting enclosures of the trajectory lies completely inside the tolerance for all $t \in [t_0, t_0 + T]$, the corresponding box is admissible.

3. Subboxes which lead to enclosures of the trajectories which are partially (but not not completely) outside for some $t \in [t_0, t_0 + T]$ but also not completely inside for all $t \in [t_0, t_0 + T]$ have to be split further until a user given maximum number of splitting operations is reached.

Each subbox $[\tilde{z}^{(l)}(0)]$ can again be expressed as a Taylor model with the unit box $[-1; 1]^n$ as domain interval vector according to

$$[\tilde{z}^{(l)}(0)] = \tilde{c}_0^{(l)} + \tilde{D}^{(l)} \mathfrak{z} \quad \text{with} \quad \mathfrak{z}_i \in [-1; 1],\ i = 1, 2, \ldots, n \ , \quad l = 1, 2, \ldots, L \ , \quad (6.33)$$

where $\tilde{c}_0^{(l)}$ is the midpoint of $[\tilde{z}^{(l)}(0)]$ and $\tilde{D}^{(l)}$ is a diagonal matrix with $\tilde{D}^{(l)} = \text{rad}\left([\tilde{z}^{(l)}(0)]\right)$.
If $L > 1$, the most appropriate subbox for the splitting has to be selected at first. This could be the interval vector $[\tilde{z}^{(l)}(0)]$ with the largest pseudo-volume. Another strategy is to calculate the pseudo volume of the interval enclosure of the Taylor model $\tilde{T}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ resulting

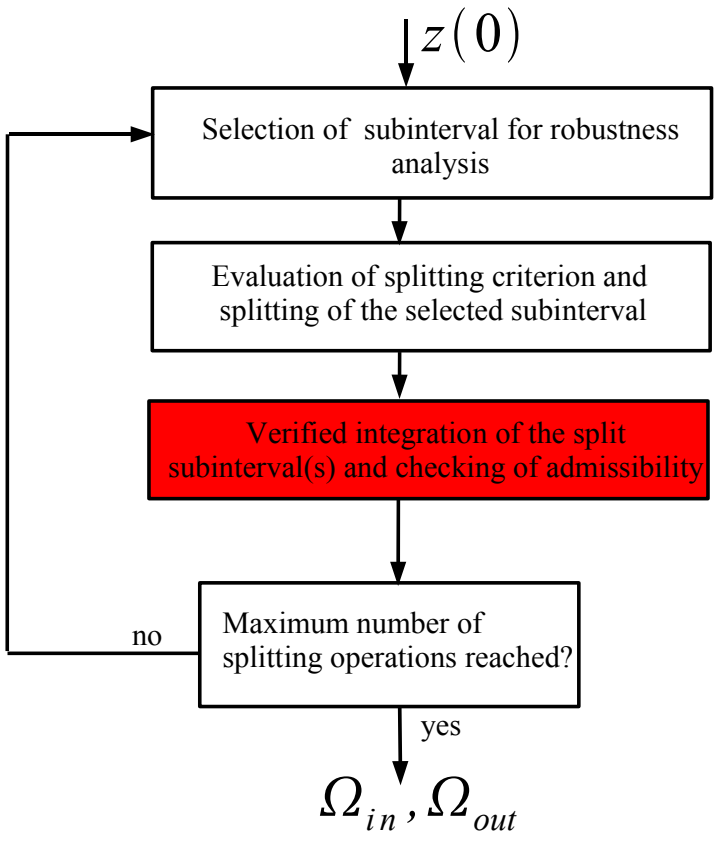


Figure 6.2: Block diagram of the algorithm for the determination of the admissible parameters.

from each subbox $[\tilde{z}^{(l)}(0)]$ in the last integration step of the preceding integration and select the subbox $[\tilde{z}^{(l)}(0)]$ which led to the largest pseudo volume. A third selection strategy is to consider the interval remainders of the Taylor models in the last integration step, and to select the subbox which led to the interval remainder with the largest pseudo volume. After the selection of an interval vector $[\tilde{z}^{(l)}(0)]$ a splitting direction has to be determined by checking the sensitivity of the Taylor model $\tilde{T}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ from the selected interval vector $[\tilde{z}^{(l)}(0)]$ at the last integration step of the previous integration with respect to each component $\mathfrak{z}_i$, $i = 1 \ldots n$ of the domain interval vector. The component μ of $\mathfrak{z}$ for which the Taylor model $\tilde{T}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ is most sensitive is determined by the heuristics described in Section 4.4.2.. As the interval vectors $[\tilde{z}^{(l)}(0)]$ and $[\mathfrak{z}]$ are related by (6.33), the interval vector $[\tilde{z}^{(l)}(0)]$ selected for splitting is also split in the component μ.

6.4 Application: Magnetic Levitation system

In this Section simulation results of a magnetic levitation system as described in Section 2.5 are shown. The system equations of the magnetic levitation system are:

$$\begin{aligned} \dot{x}_1 &= x_2 \ , \\ \dot{x}_2 &= \frac{k}{m}\frac{u^2}{(c-x_1)^2} - g \ . \end{aligned} \tag{6.34}$$

The system is already given in controller normal form and thus a flat output of (6.34) is given by

$$y_f = x_1 \ . \tag{6.35}$$

The relations (6.3)–(6.4) can directly be derived from (6.34)

$$(x_1, x_2) = (y_f, \dot{y}_f) \ , \tag{6.36}$$

$$u = (c-y_f)\sqrt{\frac{m}{k}(\ddot{y}_f + g)} \ . \tag{6.37}$$

For the load of the levitation system a set point change is considered, i.e. a tajectory has to be planned such that the following boundary conditions are satisfied

$$\begin{aligned} (x_1(0\,\mathrm{s}), x_2(0\,\mathrm{s})) &= (-0.4\,\mathrm{cm}, 0\,\frac{\mathrm{cm}}{\mathrm{s}}) \ , \\ (x_1(0.1\,\mathrm{s}), x_2(0.1\,\mathrm{s})) &= (-0.2\,\mathrm{cm}, 0\,\frac{\mathrm{cm}}{\mathrm{s}}) \ . \end{aligned} \tag{6.38}$$

In view of the differential parameterization (6.36) this yields the following boundary conditions for a corresponding trajectory $y_{f,d}$ for the flat output

$$\begin{aligned} y_{f,d}(0\,\mathrm{s}) &= -0.4\,\mathrm{cm}, & \dot{y}_{f,d}(0\,\mathrm{s}) &= 0\,\frac{\mathrm{cm}}{\mathrm{s}} \ , \\ y_{f,d}(0.1\,\mathrm{s}) &= -0.2\,\mathrm{cm}, & \dot{y}_{f,d}(0.1\,\mathrm{s}) &= 0\,\frac{\mathrm{cm}}{\mathrm{s}} \ , \end{aligned} \tag{6.39}$$

which can be satisfied by assigning for $y_{f,d}$ a third order polynomial. The resulting reference trajectory for y_f and $\dot{y}_f$ together with tolerance bands can be seen in Fig. 6.3.

Based on the results in Section 6.2.1 a tracking controller for system (6.34) is given by

$$u = (c-x_1)\cdot\sqrt{\frac{m}{k}(\ddot{y}_{f,d} - \lambda_1(x_2 - y_{f,d}) - \lambda_0(x_1 - y_{f,d}) + g)} \ . \tag{6.40}$$

It is assumed that the flat output of (6.34) is available for measurement, i.e.

$$y = h(x) = x_1 + \epsilon \ , \tag{6.41}$$

where, ϵ describes an unknown but constant sensor offset. For the given output a nonlinear tracking observer, as discussed in Section 6.2.4, can be derived. It has the form

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 - l_1(t)(y + \epsilon - \hat{x}_1) \ , \\ \dot{\hat{x}}_2 &= \frac{k}{m}\frac{u^2}{(c-\hat{x}_1)^2} - g - l_2(t)(y + \epsilon - \hat{x}_1) \ , \end{aligned} \tag{6.42}$$

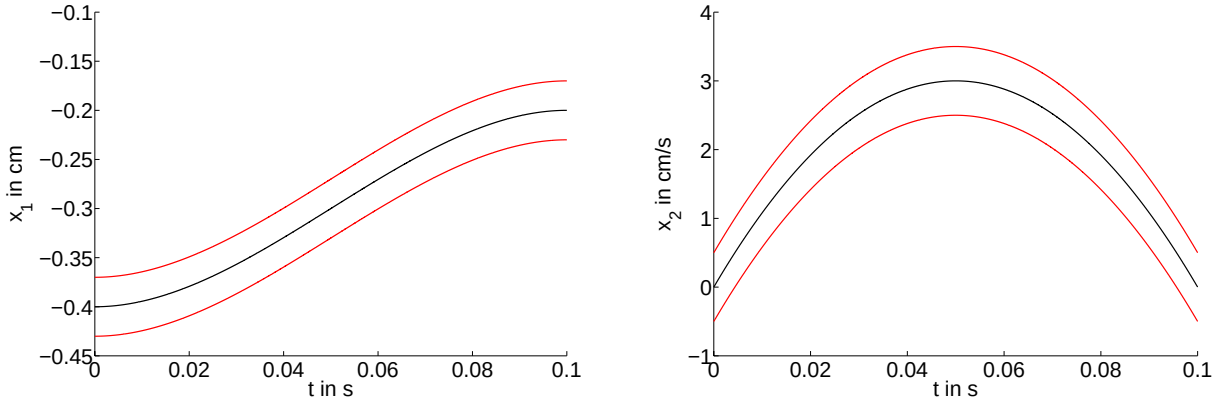


Figure 6.3: Reference trajectories for the position x_1 and the velocity x_2 and the bounds for admissible deviations (grey).

where the unknown sensor offset ϵ in (6.41) is explicitly included in the model. The feedback law (6.40) which stabilizes the tracking can then be estimated as

$$\hat{u} = (c - x_1 + \epsilon) \cdot \sqrt{\frac{m}{k}(\ddot{y}_{f,d} - \lambda_1(\hat{x}_2 - y_{f,d}) - \lambda_0(x_1 + \epsilon - y_{f,d}) + g)} \ , \tag{6.43}$$

where the measured output (6.41) was used to estimate x_1 and x_2 is estimated using the observer state $\hat{x}_2$. A tracking controller with integral error feedback as discussed in Section 6.2.1 is given by (using again the observer (6.42) and the measured output (6.41))

$$\begin{aligned} \dot{e}_I &= x_1 + \epsilon - y_{f,d} \ , \\ \hat{u} &= (c - x_1 + \epsilon) \cdot \sqrt{\Big(\frac{m}{k}(\ddot{y}_{f,d} - \lambda_1(\hat{x}_2 - y_{f,d}) - \lambda_0(x_1 + \epsilon - y_{f,d}) - \lambda_{-1} e_I) + g)\Big)} \end{aligned} \tag{6.44}$$

in view of (6.21). With the parameter vector $p = [k, m, c, g, \epsilon]^T$ the controlled systems (6.34), (6.42)–(6.43) and (6.34), (6.42), (6.44) exhibit the structure as in (6.27) and (6.28) respectively.

For the simulation the nominal parameters of system (6.34) were assumed to be (see [21])

$$\begin{aligned} k_{nom} &= 58.042 \frac{\mathrm{kg\,cm^3}}{\mathrm{s^2\,A^2}}, & g_{nom} &= 981 \, \frac{\mathrm{cm}}{\mathrm{s^2}} \ , \\ m_{nom} &= 0.0844 \, \mathrm{kg}, & c_{nom} &= 0.11 \, \mathrm{cm} \ . \end{aligned} \tag{6.45}$$

The algorithm described in the previous section was applied to the magnetic levitation system which is controlled using the tracking controllers [(6.42), (6.43)] and [(6.42), (6.44)] respectively. For simplicity the parameters of the levitation system were assumed to match then nominal ones in (6.45). Only the parameter k was assumed to be uncertain but bounded by the interval $k \in [54.042; 62.042] \, \frac{\mathrm{kg\,cm^3}}{\mathrm{s^2\,A^2}}$. The sensor was assumed to have an unknown offset ϵ which is assumed to be bounded by $\epsilon \in [-0.01; 0.01] \, \mathrm{cm}$. The order of the Taylor models was chosen to be 6 in time and initial state variables. Additionally QR-Preconditioning was applied. The tolerances for the deviations from the reference trajectory (see (6.29)) were: $d_1 \in [-0.03; 0.03]$ cm, $d_2 \in [-0.5; 0.5] \, \frac{\mathrm{cm}}{\mathrm{s}}$. The deviation of the position d_1 was chosen very small as the final equililibrium position is desired to be approached very exactly.

To speed up the convergence of the estimation error the observer (6.42) was initialized using the measured output (6.41), i.e.

$$\hat{x}_1(0) = y(0) \ . \tag{6.46}$$

As x_2 cannot be measured the observer state $\hat{x}_2$ was initialized with the reference trajectory, i.e.

$$\hat{x}_2(0) = x_{2,d}(0) = \dot{y}_{f,d}(0) \ . \tag{6.47}$$

The simulation results for the controller [(6.42), (6.43)] can be seen in Fig. 6.4. For the consistent set of parameters the robustness specifications (6.29) are met. For the inconsistent parameters the specifications are not met. Remaining undecided parameter sets could be split up further, if necessary. It can be deduced that the controller [(6.42), (6.43)] cannot meet the specification (6.29) for all values of $[k]$ and $[\epsilon]$ and thus the parameters k and ϵ would have to be determined more exactly if the specifications have to be met in any case.

In Fig. 6.5 the resulting consistent parameters are compared for the controllers [(6.42), (6.43)] and [(6.42), (6.44)]. It can be seen that for the desired specifications the two controllers yield different subsets of the intervals $[k]$ and $[\epsilon]$. It can be concluded e.g. that if there is a large uncertainty in the offset ϵ of the sensor and the parameter k can be determined very well, then the controller [(6.42), (6.43)] should be preferred (indicated by the dashed box). If on the other hand the uncertainty in the sensor offset are small but the uncertainty in the parameter k is relatively large, then the controller with integral error feedback should be preferred. Thus the proposed robustness analysis can indeed yield hints on the choice of the used controller.

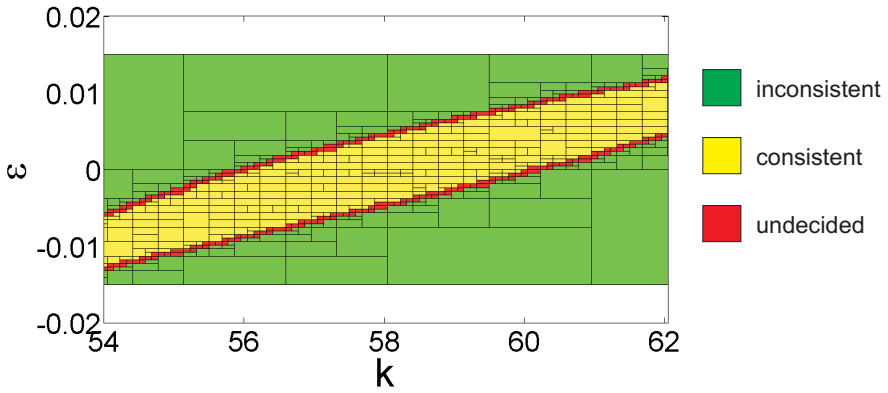


Figure 6.4: Simulation results for controller without integral error feedback.

It should be mentioned, however, that the resulting parameter sets as shown in Figures 6.4 and 6.5 strongly depend on the specifications (6.29). This can be verified when looking at Fig. 6.6. Here the resulting trajectories using the controller [(6.42), (6.44)] for $k = 56.5$ and $\epsilon = 0.004$ – a parameter combination which belongs to the set unadmissble parameter values – are depicted. It can be seen that the robustness specification is violated only by the velocity and mostly in the first part of the trajectory. As a consequence, if the tolerable deviation e.g. on the velocity in the first part of the trajectory can be relaxed or a specification which

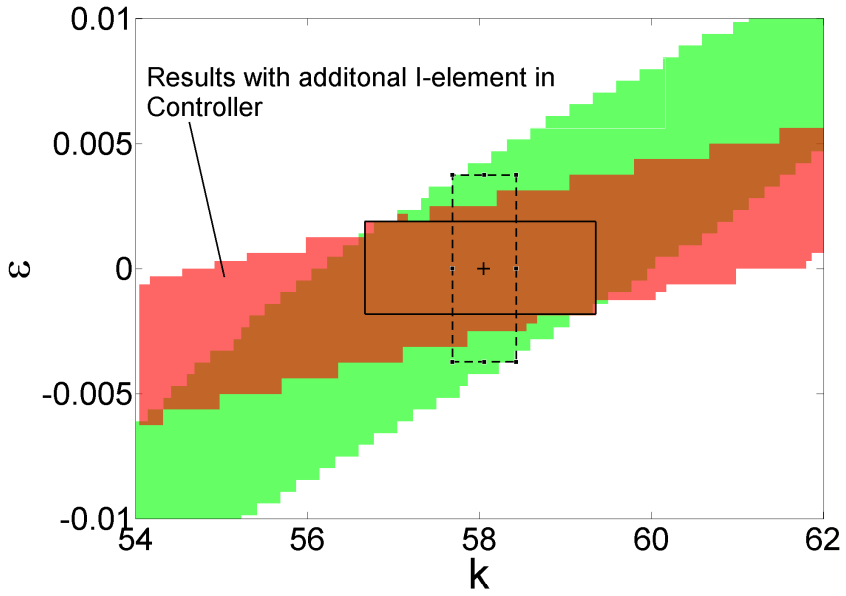


Figure 6.5: Consistent subboxes for a controller with and a controller without integral error feedback.

allows some kind of a transition time for the state variables – as also depicted in Fig. 6.6 – a much larger admissible parameter set would result for this controller. With the proposed approach also such a specification could be investigated.

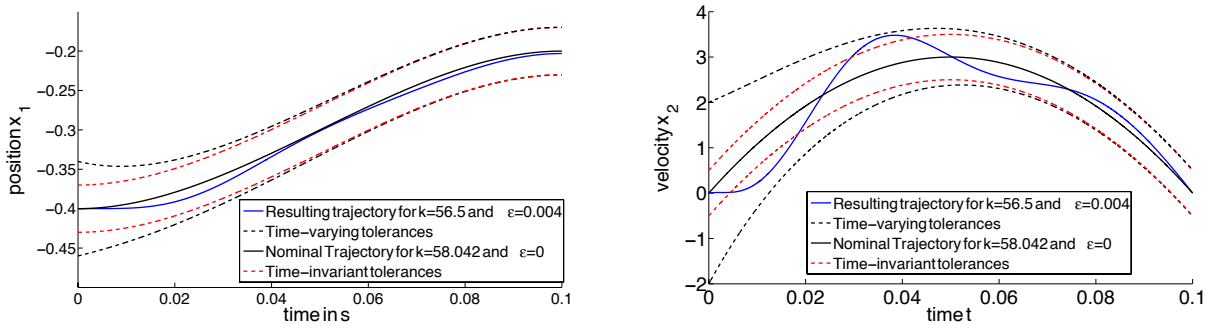


Figure 6.6: Robustness specification with transition time for the state variables.

7 Verified State and Parameter Estimators in Closed Loop Control

7.1 Problem Formulation

In Chapter 5 two different VSPE concepts are presented and compared in detail. The considered systems are open loop systems and the results of the estimates have not been fed back in sense of a feedback control. In this Chapter results of a VSPE in closed loop control are shown. The estimates required for the controller are then provided by the VSPE. Like in Chapter 6 a flatness based control system is considered. The tracking observer is replaced by a VSPE which estimates the non measured state variables required for the feedback control. In addition to the state vector also the uncertain parameters are estimated. The extended state vector of the estimated state variables and parameters is given by

$$\tilde{z}(t) = \begin{bmatrix} \tilde{x}^T(t) & \tilde{p}^T(t) \end{bmatrix}^T \tag{7.1}$$

and is enclosed either by interval vectors or Taylor models. However, in practice the required estimates for the controller have to be specific values no sets in form of intervals or Taylor models. The estimate $\hat{z}$ for the controller was chosen to be the midpoint of the estimated interval enclosure of $\tilde{z}$, hence

$$\hat{z} = \text{mid}([\tilde{z}]) \tag{7.2}$$

which results in the controller output $\hat{u}$. This selection criterion is usually only suboptimal, therefore in future work more sophisticated criteria have to be developed, for example by the formalization of an optimization problem and and calculation of a value $\hat{z} \in [\tilde{z}]$, which results in an optimum value for $\hat{u}$. The system equations and the equations for the VSPE are then given by

$$\begin{bmatrix} \dot{z} \\ \dot{\tilde{z}} \end{bmatrix} = \begin{bmatrix} f(z, \hat{\psi}_u(\hat{z}, t), t) \\ f(\tilde{z}, \hat{\psi}_u(\hat{z}, t), t) \end{bmatrix} \quad \text{with} \quad \hat{z} = \text{mid}([\tilde{z}]) \ . \tag{7.3}$$

Here, z is the state vector of the plant, $\tilde{z}$ are the estimates of z, and $\hat{z} = \text{mid}([\tilde{z}])$ is the value fed back to the controller. The block diagram of the controlled system containing a VSPE in case of a flatness based control is depicted in Fig. 7.1.

Here, the VSPE calculates guaranteed bounds for a given control variable $\hat{u}$ which is obtained by using $\hat{z} = \text{mid}([\tilde{z}])$ for the feedback control. It does not determine an outer approximation $[\hat{u}]$ with respect to the whole interval $[z]$. However, it would be easy to adapt the algorithm to this case.

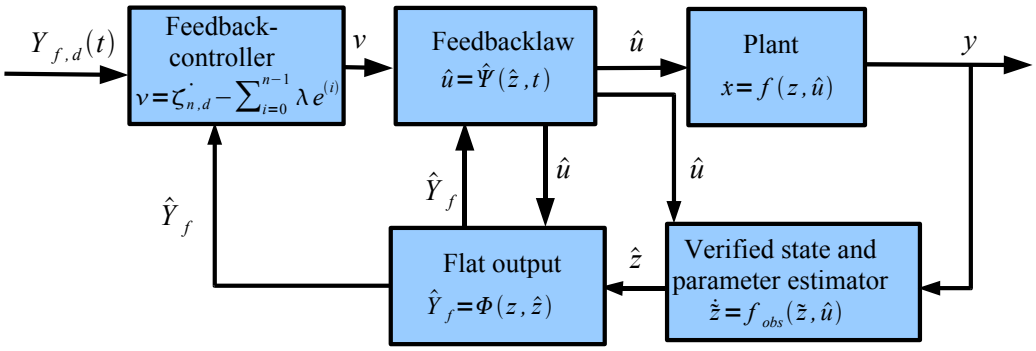


Figure 7.1: Blockdiagram of the controlled system with an verified estimation concept.

The results obtained by applying the VSPE are compared with the result, when a nonlinear tracking observer with additional estimation of the uncertain parameters is used. In view of (6.27) the equations for the controlled system with additional estimation of the parameter are

$$\begin{bmatrix} \dot{z} \\ \dot{\hat{z}} \end{bmatrix} = \begin{bmatrix} f(z, \hat{\psi}_u(\hat{z}, t), t) \\ f_{obs}(z, \hat{z}, \hat{\psi}_u(\hat{z}, t), t) \end{bmatrix} \quad . \tag{7.4}$$

7.2 Application: Magnetic Levitation System

Consider again the Amira-Pendulum from Section 2.5 and Chapter 6. It is assumed that the parameter k is uncertain with $k \in [58.042 - 0.6; 58.042 + 0.6] \frac{\text{kg cm}^3}{\text{s}^2\,\text{A}^2}$. Again x_1 is the only measured state. The sensor has no constant offset, but the measurements are affected by bounded measurement noise $\delta_1(t) \in [\delta_1]$. Two estimations were carried out, the first with $\delta_1(t) \in [\delta_1] = [-0.001; 0.001]$cm, the second with $\delta_1(t) \in [\delta_1] = [-0.01; 0.01]$cm. The observer has to estimate x_2 and the uncertain parameter k. As stated above, for comparison also an estimation with a tracking observer like in chapter 6 is presented. In contrast to chapter 6 the uncertain parameter k is estimated additionally. If the parameter k is assumed to be time-invariant, then the system equation is given by:

$$\begin{aligned} \dot{x}_1 &= x_2 \ , \\ \dot{x}_2 &= \frac{k}{m} \frac{u^2}{(c - x_1)^2} - g \ , \\ \dot{k} &= 0 \ . \end{aligned} \tag{7.5}$$

For the measurement equation

$$y = x_1 + \delta_1 \tag{7.6}$$

holds. The measured values are assumed to be available in each integration step. They are obtained by adding an uniformly distributed noise bounded by $[\delta_1]$ to the obtained value of x_1. The equations for the tracking observer are:

$$\begin{aligned}\dot{\hat{x}}_1 &= \hat{x}_2 - l_1(t)(y - \hat{x}_1) \ , \\ \dot{\hat{x}}_2 &= \frac{\hat{k}}{m}\frac{u^2}{(c - \hat{x}_1)^2} - g - l_2(t)(y - \hat{x}_1) \ , \\ \dot{\hat{k}} &= -l_3(t)(y - \hat{x}_1) \ .\end{aligned} \tag{7.7}$$

For the equations of the VSPE

$$\begin{aligned}\dot{\tilde{x}}_1 &= \tilde{x}_2 \ , \\ \dot{\tilde{x}}_2 &= \frac{\tilde{k}}{m}\frac{u^2}{(c - \tilde{x}_1)^2} - g \ , \\ \dot{\tilde{k}} &= 0\end{aligned} \tag{7.8}$$

holds.

The feedback law u (6.40) which stabilizes the tracking can then be estimated as

$$\hat{u} = (c - y) \cdot \sqrt{\frac{m}{\hat{k}}(\ddot{y}_{f,d} - \lambda_1(\hat{x}_2 - y_{f,d}) - \lambda_0(y - y_{f,d}) + g)} \tag{7.9}$$

for the tracking observer and

$$\begin{aligned}\hat{u} = &(c - \mathrm{mid}([\tilde{x}_1])) \\ &\cdot \sqrt{\frac{m}{\mathrm{mid}([\tilde{k}])}(\ddot{y}_{f,d} - \lambda_1(\mathrm{mid}([\tilde{x}_2]) - y_{f,d}) - \lambda_0(\mathrm{mid}([\tilde{x}_1]) - y_{f,d}) + g)}\end{aligned} \tag{7.10}$$

for the VSPE. The estimated feedback law $\hat{u}$ is substituted for u in (7.5), (7.7) and (7.8). The initial values of the real systems were $x_1(0) = 0.39\mathrm{cm}$, $x_2(0) = 0\mathrm{cm/s}$, $k = k_{nom} + 0.5 = 58.042 + 0.5 = 58.542 \frac{\mathrm{kg\,cm^3}}{\mathrm{s^2\,A^2}}$. The estimated position $\hat{x}_1$ for the tracking observer is initialized by the measured value $y(0) = x_1(0) + \delta(0)$, the velocity $\hat{x}_2$ with the values of the reference trajectory at $t = 0$, hence $\hat{x}_2(0) = x_2(0) = 0\,\mathrm{cm/s}$. The initial value for the parameter was the nominal parameter value $k(0) = k_{nom} = 58.042$. The initial interval enclosures for the VSPE was assumed to be $\tilde{x}_1(0) = y(0) + [\delta]$, $\tilde{x}_2(0) = 0\mathrm{cm/s}$, and $[k] = k_{nom} + [-0.6; 0.6] \frac{\mathrm{kg\,cm^3}}{\mathrm{s^2\,A^2}}$. The VSPE applied was a Taylor model observer with an expansion of order $\rho = 6$ in time and state variables. Only a single Taylor model was used. For $\delta_1(t) \in [-0.001; 0.001]\mathrm{cm}$, the results for the system state variables x_1, x_2 and the estimated parameter $\hat{k}$ are depicted in Figs. 7.2(a) – 7.2(c). The dashed lines depict the reference trajectories $x_{1,d}$ and $x_{2,d}$ and the real parameter value. The bright solid lines are the resulting trajectories with tracking observer, the black solid lines depict the results with the VSPE. Since the results for both estimators cannot clearly be distinguished in these pictures, additionally the absolute values of the deviations from the reference trajectory and the deviation from the estimated parameter to its real value

$$\begin{aligned}d_1 &= |x_1(t) - x_{1,d}(t)| \\ d_2 &= |x_2(t) - x_{2,d}(t)| \\ d_k &= |\hat{k}(t) - k|\end{aligned} \tag{7.11}$$

are depicted in Fig. 7.3. Fig. 7.3(a) shows the deviation d_1 of the position x_1. In Fig. 7.3b) the deviations of the velocity d_2 is depicted. The absolute values of the deviations of the parameter k are shown in Fig. 7.3(c). The results of the state variables are comparable, the estimation for the parameter is slightly better for the VSPE.

The results for the state variables x_1, x_2, and the estimated parameter $\tilde{k}$ for $\delta_1(t) \in [-0.01; 0.01]$ are depicted in Figs. 7.4(a) – 7.4(c). The corresponding deviations are presented in Figs. 7.5(a)–7.5(c). Here, the estimation quality in sense of the deviations from the reference trajectories and from the deviation from the estimated parameter to its corresponding real values are much better for the VSPE.

The results point out that the tracking observer is much more sensitive to the measurement noise. The VSPE is able to reduce the measurement uncertainty during the estimation process. An additional advantage of the VSPE is, that it provides guaranteed enclosures for the state variables x_1, x_2, and for the parameter k.

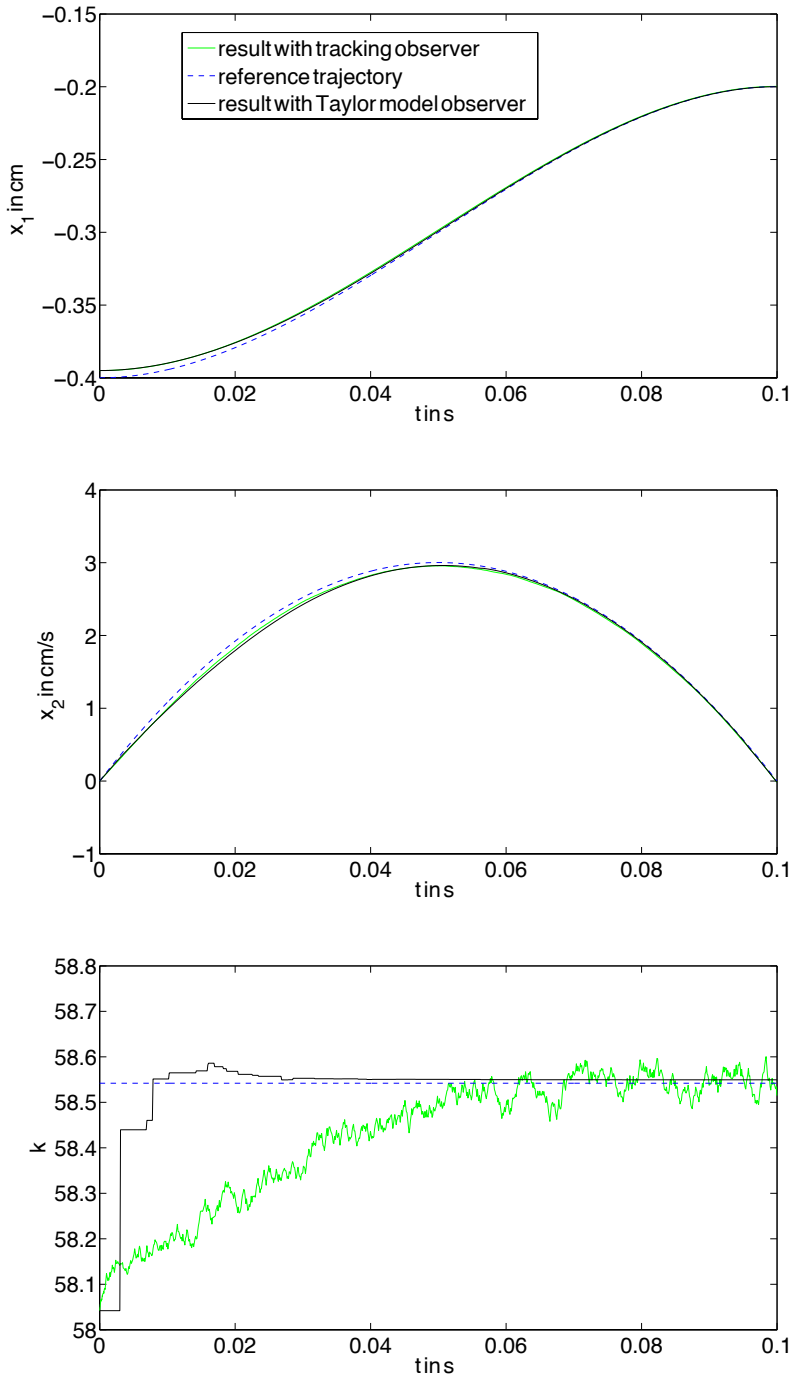
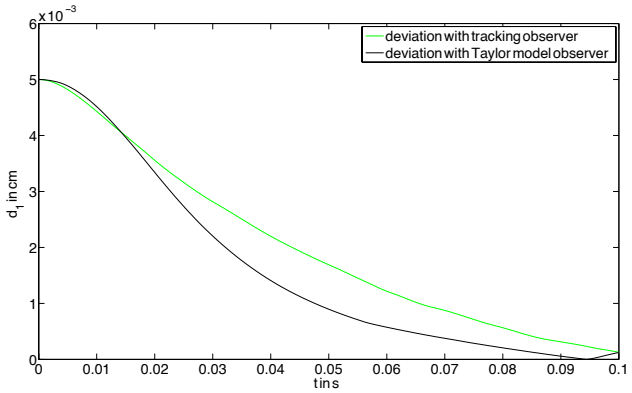
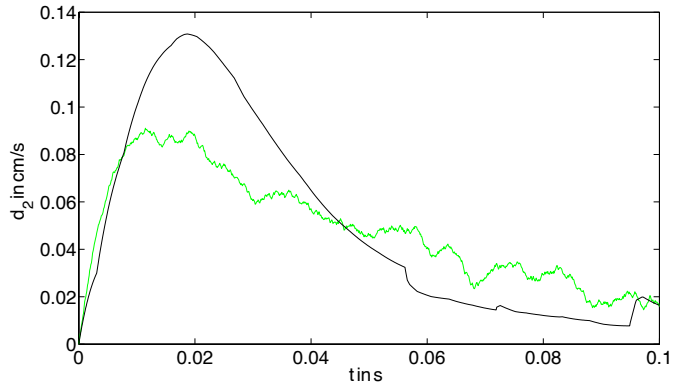


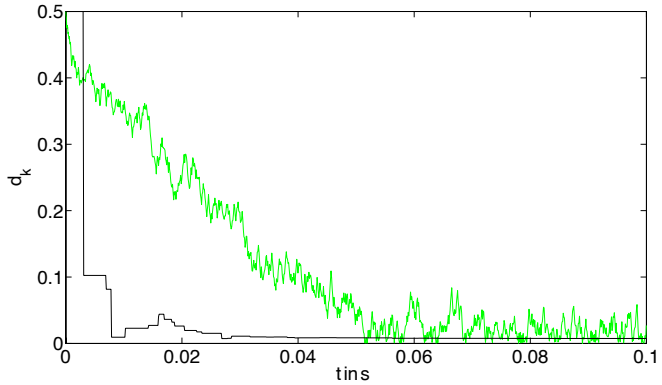
Figure 7.2: Comparison of the Results for the tracking observer and the verified state and parameter estimator for $[\delta_1] = [-0.001; 0.001]$.



(a) Deviations from the reference trajectory for the position.



(b) Deviations from the reference trajectory for the velocity.



(c) Deviations from the estimated parameter to its actual value.

Figure 7.3: Comparison of the deviations from the reference trajectories for $[\delta_1] = [-0.001; 0.001]$.

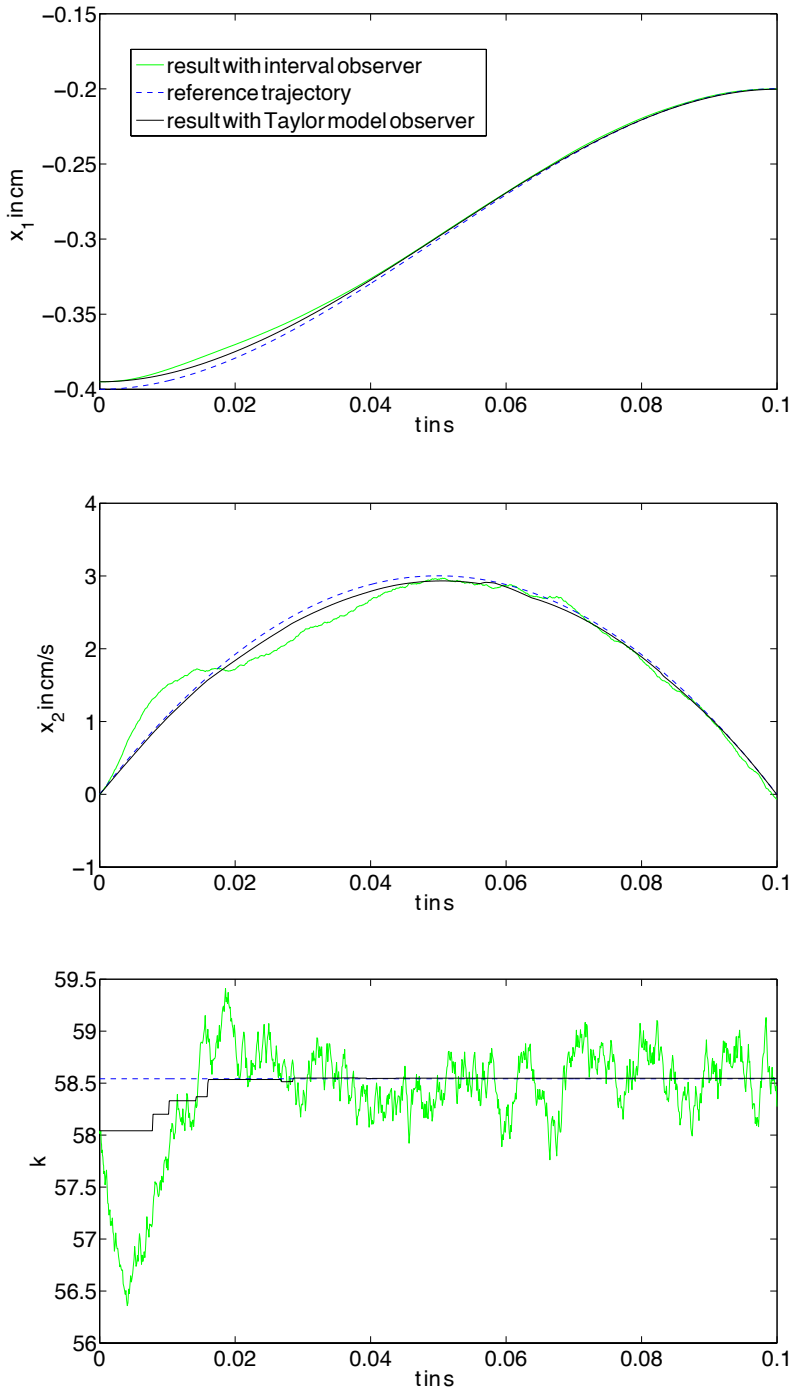
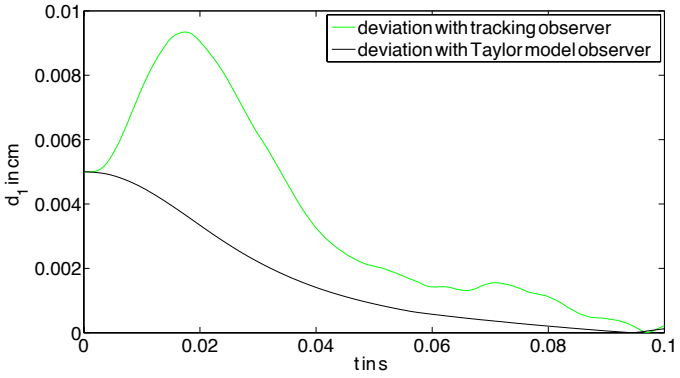
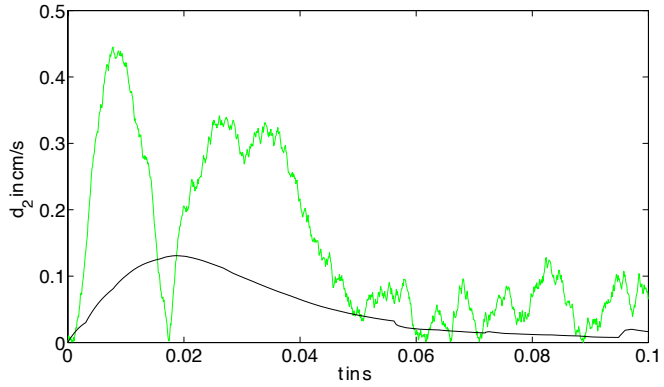


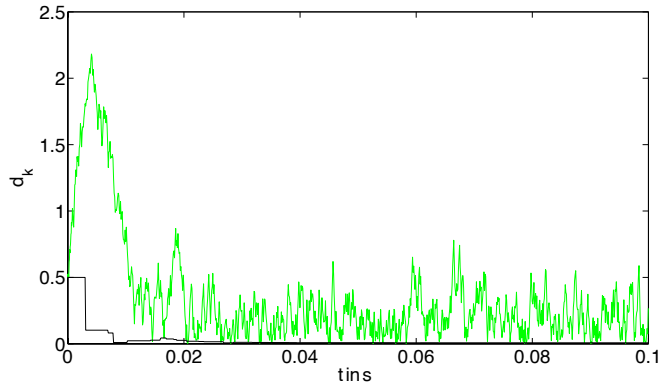
Figure 7.4: Comparison of the Results for the tracking observer and the verified state and parameter estimator for $[\delta_1] = [-0.01; 0.01]$.



(a) Deviations from the reference trajectory for the position.



(b) Deviations from the reference trajectory for the velocity.



(c) Deviations from the estimated parameter to its actual value.

Figure 7.5: Comparison of the deviations from the reference trajectories for $[\delta_1] = [-0.01; 0.01]$.

8 Conclusions and Outlook on Future Research

In this thesis two different approaches for verified state and parameter estimation have been compared. In the first approach the extended state vector consisting of system state variables and uncertain system parameters is enclosed by multiple interval vectors. The second approach employs Taylor models. Taylor models have the advantage of providing tight enclosures of non convex regions often with a single Taylor model. Interval vectors are easy to manipulated e.g., when the intersection of two interval vectors has to be determined. The manipulation of Taylor model is, however, rather difficult if the set described by a Taylor models should be intersected with an interval vector. This leads to the fact that the correction step is more difficult to implement in this approach. Another drawback is the lack of merging routines which enables efficient merging of two similar Taylor models.
Depending on the nature of the measurement equation and on the time horizon of interest one or the other of the two estimators is doing better. The comparison of the results for three applications point out that in average the Taylor model provided tighter enclosures of the estimated states and parameters. Given these results, a combination of both approaches is strongly recommended.
Reapproximation during the estimation by an interval observer was carried out at prespecified points of time – but not adaptively. In further research heuristics for determining points of time when the reapproximation is carried out have to be developed. The consistency tests presented in Section 4.2 have not been applied when performing the presented estimations with the interval observer in Chapter 5. Future research should include these techniques in the estimation process. Similar to the application of the reapproximation, criteria have to be developed for determining points of time when the consistency tests should be applied.
Further work relating the Taylor model observer should include a comparison of the different preconditioning techniques in the prediction step and an optimization of the correction step. The consistency tests in the correction step have only be applied in the illustrative example. The method has to be extended to more complex applications. For that purpose criteria have to be developed when to apply the consistency test.
The development of algorithms which allow not only for tightening of the domain interval vector in the correction step, but also for a reduction of the width of the interval remainder would be an important contribution for improving the estimation quality.
In future research other integration methods in the prediction step have to be employed and compared with the existing methods. For example the Taylor series expansion in time for the interval observer can be replaced by ValEncIA-IVP [4] and instead of COSY-VI, VSPODE [42] can be used, which is also based on Taylor models. The enclosures in ValEncIA-IVP are determined by an approximate solution and a time-depending interval enclosure of the deviation from the exact solution. It belongs in a wider sense to the so called *defect based*

methods. In VSPODE the expansion in time is done as described in Section 4.1. The dependency on the initial states and parameters are represented by Taylor models. It is therefore a combination of the approaches from Section 4.1 and 4.4.

The verified observability analysis in Section 5.1 determines a guaranteed enclosure of those state and parameter values for which the system is observable. To use this method for systems with higher dimensionality, automatic differentiation methods to calculate the higher Lie-derivatives have to be applied, since the analytical expressions grow very quickly with each further differentiation.

In Chapter 6 it has been shown how interval methods can be used successfully to verify for which parameter values a controlled system with a traditional observer concept leads to trajectories, which are within specified bounds around the reference trajectory. The methods can help to chose between different controllers depending on the situation. The algorithm for the robustness analysis described in Chapter 6 is based on Taylor models. Since Taylor models perform an expansion in the initial state vector, it is possible to split the domain interval vector of the initial state vector also for Taylor models at $t_k > 0$ and continue the integration for the new Taylor models at t_k. This will lead to a drastic reduction of computation time.
The robustness analysis is of course also feasible with verified integration methods based on interval vectors.

Chapter 7 contains results for a VSPE in closed loop control, where the estimated results are directly used in the control law. The mid-points of the interval enclosures of the estimated state variables and parameters are used in the controller. In future work more efficient criteria for choosing the values for the controller have to be developed.

Also, a comparison with stochastic estimation approaches has to be considered in future research. A comparison of these two completely different estimation approaches is inherently difficult. It can be expected that depending on how the given uncertainty can be better modeled the corresponding estimation approach will perform better. However, an investigation in this direction will give a confirmation of this assumption.
Also a combination of set-based and stochastic estimation approaches should be considered, since many systems are affected by both set valued and stochastic uncertainty.

Many real world systems are described by *differential algebraic equations* (DAEs) or *partial differential equations*(PDEs). To extent the presented estimation techniques to these kind of systems a lot of work has to be done, since interval methods for DAEs and PDEs are not as developed as the algorithms for the treatment of ODEs.
Also more complicated friction models including hysteresis effects have to be taken into account.

Interval methods are still far from being accepted as standard tools in engineering. The work presented in this thesis demonstrates that they can be successfully applied to real world applications. The presented estimation methods have been run in *cygwin* under Windows. By the time this work was completed some of the presented algorithms have been run on both cygwin and Linux. Under Linux also a newer compiler could be used. The computation was

5 to 10 times faster when using Linux. And for the main part of the algorithms developed during this work a parallelization can easily be done, enabling real time applications for a larger class of systems.
This will lead to an increasing acceptance of interval methods in the field of engineering.

A Examples for Taylor Model Based Verified Integration of ODEs

A.1 Nonlinear Example

The Lotka-Volterra equations are given by:

$$\begin{aligned} \dot{z}_1 &= z_1(a - bz_2) \ , \\ \dot{z}_2 &= z_2(cz_1 - d) \ . \end{aligned} \tag{A.1}$$

For $a = b = c = d = 1$ the equations become

$$\begin{aligned} \dot{z}_1 &= z_1(1 - z_2) \ , \\ \dot{z}_2 &= z_2(z_1 - 1) \ . \end{aligned} \tag{A.2}$$

The initial state variables are $z_1(0), z_2(0) \in [45; 55]$. The order of the Taylor model is set to $\rho = 3$. A constant step-size $h_k = h = 0.001$ is used. The initial state variables are rewritten according to

$$\begin{aligned} z_1(0) &\in 50 + 5\mathfrak{z}_1 \ , \\ z_2(0) &\in 50 + 5\mathfrak{z}_2 \ , \end{aligned} \tag{A.3}$$

with $\mathfrak{z}_1, \mathfrak{z}_2 \in [-1; 1]$ and $\hat{\mathfrak{z}}_1 = 0$ and $\hat{\mathfrak{z}}_2 = 0$. The initial initial Taylor model $T_{\rho,0}$ is then defined by

$$\begin{aligned} T_{\rho,1,0}(\mathfrak{z}) &= 50 + 5\mathfrak{z}_1 \ , \\ T_{\rho,2,0}(\mathfrak{z}) &= 50 + 5\mathfrak{z}_2 \ . \end{aligned} \tag{A.4}$$

The First Time Step

In the first integration step the ODE is integrated for $t \in [t_0; t_1] = [0; h]$. The results are displayed up to the 6-th decimal digit.

Calculation of the Polynomial Part.

First the polynomial part is determined. The initial polynomial part $P^{(0)}$ with its components $P_1^{(0)}$ and $P_2^{(0)}$ in the first integration step is given by

$$\begin{aligned} P_1^{(0)}(\mathfrak{z}, t - t_0) &= 50 + 5\mathfrak{z}_1 \ , \\ P_2^{(0)}(\mathfrak{z}, t - t_0) &= 50 + 5\mathfrak{z}_2 \ . \end{aligned} \tag{A.5}$$

With $t_0 = 0$ the first Picard iteration yields

$$\begin{aligned} P_1^{(1)}(\mathfrak{z},t) &= P_1^{(0)}(\mathfrak{z},t) + \int_0^t P_1^{(0)}(\mathfrak{z},t')(1 - P_2^{(0)}(\mathfrak{z}-\hat{\mathfrak{z}},t'))dt' \\ &= 50 - 2450t + 5\mathfrak{z}_1 - 245t\mathfrak{z}_1 - 250t\mathfrak{z}_2 - 25t\mathfrak{z}_1\mathfrak{z}_2 \ , \\ P_2^{(1)}(\mathfrak{z},t) &= P_2^{(0)}(\mathfrak{z},t) + \int_0^t P_2^{(0)}(\mathfrak{z},t')(P_1^{(0)}(\mathfrak{z},t') - 1)dt' \\ &= 50 + 2450t + 5\mathfrak{z}_2 + 250t\mathfrak{z}_1 + 25t\mathfrak{z}_1\mathfrak{z}_2 + 245t\mathfrak{z}_2 \ . \end{aligned} \tag{A.6}$$

Omitting terms higher than $\rho = 1$ results in

$$\begin{aligned} P_1^{(1)}(\mathfrak{z},t) &= 50 + 5\mathfrak{z}_1 - 2450t \ , \\ P_2^{(1)}(\mathfrak{z},t) &= 50 + 5\mathfrak{z}_2 + 2450t \ . \end{aligned} \tag{A.7}$$

The second Picard iteration is then given by

$$\begin{aligned} P_1^{(2)}(\mathfrak{z},t) &= P_1^{(0)} + \int_0^t P_1^{(1)}(\mathfrak{z},t')(1 - P_2^{(1)}(\mathfrak{z},t'))dt' \ , \\ P_2^{(2)}(\mathfrak{z},t) &= P_2^{(0)} + \int_0^t P_2^{(1)}(\mathfrak{z},t')(P_1^{(1)}(\mathfrak{z},t') - 1)dt' \ . \end{aligned} \tag{A.8}$$

Omitting terms higher than $\rho = 2$ leads to

$$\begin{aligned} P_1^{(2)}(\mathfrak{z},t) &= 50 - 2450t + 5\mathfrak{z}_1 - 245t\mathfrak{z}_1 - 250t\mathfrak{z}_2 - 1225t^2 \ , \\ P_2^{(2)}(\mathfrak{z},t) &= 50 + 2450t + 5\mathfrak{z}_2 + 250t\mathfrak{z}_1 + 245t\mathfrak{z}_2 - 1225t^2 \ . \end{aligned} \tag{A.9}$$

After the third Picard iteration (omitting terms higher than order $\rho = 3$) the polynomial part becomes

$$\begin{aligned} P_1^{(3)}(\mathfrak{z},t) &= 50 - 2450t + 5\mathfrak{z}_1 - 1225t^2 - 245t\mathfrak{z}_1 - 250t\mathfrak{z}_2 \\ &\quad -25t\mathfrak{z}_1\mathfrak{z}_2 - 12745/2t^2\mathfrak{z}_1 + 6125t^2\mathfrak{z}_2 + 6123775/3t^3 \ , \\ P_2^{(3)}(\mathfrak{z},t) &= 50 + 2450t + 5\mathfrak{z}_2 - 1225t^2 + 250t\mathfrak{z}_1 + 245t\mathfrak{z}_2 + 25t\mathfrak{z}_1\mathfrak{z}_2 \\ &\quad +6125t^2\mathfrak{z}_1 - 12745/2t^2\mathfrak{z}_2 - 6123775/3t^3 \ , \end{aligned} \tag{A.10}$$

and the desired polynomial $P_\rho(\mathfrak{z},t) = P_1^{(3)}(\mathfrak{z},t)$ has been determined. Note, that for $\hat{\mathfrak{z}} \neq 0$ powers of $(\mathfrak{z}_i - \hat{\mathfrak{z}}_i)$ would occur, e.g. $(\mathfrak{z}_1 - \hat{\mathfrak{z}}_1)(\mathfrak{z}_2 - \hat{\mathfrak{z}}_2)^2$.
Now an appropriate interval remainder bound $I_{\rho,1}$ has to be found such that

$$\begin{aligned} &\mathcal{O}(P_\rho(\mathfrak{z},t) + I_{\rho,1}) \subset P_\rho(\mathfrak{z},t) + I_{\rho,1} \ , \\ &\forall \mathfrak{z} \in [\mathfrak{z}] \quad \text{and} \quad \forall t \in [t_0;t_1] = [0;h] \ . \end{aligned} \tag{A.11}$$

Calculation of the Interval Remainder

First an initial error estimate $I_{\rho,1}^{(0)}$ with its components $I_{\rho,1,1}^{(0)}$ and $I_{\rho,2,1}^{(0)}$ has to be determined according to

$$\begin{aligned}
P_{\rho,1}(\mathfrak{z},t)+I_{\rho,1,1}^{(0)} &= z_1(0)+\int_0^t \underbrace{(P_{\rho,1}(\mathfrak{z},t')+[0;0])(1-(P_{\rho,2}(\mathfrak{z},t')+[0;0]))}_{\hat{P}_{\rho,1}+\hat{I}_{\rho,1,1}^{(0)}}\,dt' \\
&= T_{\rho,1,0}(\mathfrak{z})+\int_0^t \hat{P}_{\rho-1,1}(\mathfrak{z},t')dt' \\
&\quad +\left\{B(\hat{P}_{\rho,1}(\mathfrak{z},t')-\hat{P}_{\rho-1,1}(\mathfrak{z},t'))+\hat{I}_{\rho,1,1}^{(0)}\right\}\cdot([t_0;t_1]-t_0) \\
&= P_{\rho,1}(\mathfrak{z},t)+\left\{B(\hat{P}_{\rho,1}(\mathfrak{z},t)-\hat{P}_{\rho-1,1}(\mathfrak{z},t))+\hat{I}_{\rho,1,1}^{(0)}\right\}\cdot\underbrace{([t_0;t_1]-t_0)}_{=[0;h]} \\
&= 50-2450t+5\mathfrak{z}_1-1225t^2-245t\mathfrak{z}_1-250t\mathfrak{z}_2 \\
&\quad -25t\mathfrak{z}_1\mathfrak{z}_2-12745/2t^2\mathfrak{z}_1+6125t^2\mathfrak{z}_2+6123775/3t^3+[-0.002627;0.002597]\ , \\
P_{\rho,2}(\mathfrak{z},t)+I_{\rho,2,1}^{(1)} &= z_2(0)+\int_0^t \underbrace{(P_{\rho,2}(\mathfrak{z},t')+[0;0])(P_{\rho,1}(\mathfrak{z},t')+[0;0]-1)}_{\hat{P}_{\rho,2}+\hat{I}_{\rho,2,1}^{(0)}}\,dt' \\
&= T_{\rho,2,0}(\mathfrak{z})+\int_0^t \hat{P}_{\rho-1,2}(\mathfrak{z},t')dt' \\
&\quad +\left\{B(\hat{P}_{\rho,2}(\mathfrak{z},t')-\hat{P}_{\rho-1,2}(\mathfrak{z},t'))+\hat{I}_{\rho,2,1}^{(0)}\right\}\cdot(t_1-t_0) \\
&= P_{\rho,2}(\mathfrak{z},t)+\left\{B(\hat{P}_{\rho,2}(\mathfrak{z},t')-\hat{P}_{\rho-1,2}(\mathfrak{z},t'))+\hat{I}_{\rho,2,1}^{(0)}\right\}\cdot\underbrace{(t_1-t_0)}_{=h} \\
&= 50+2450t+5\mathfrak{z}_2-1225t^2+250t\mathfrak{z}_1+245t\mathfrak{z}_2+25t\mathfrak{z}_1\mathfrak{z}_2 \\
&\quad +6125t^2\mathfrak{z}_1-12745/2t^2\mathfrak{z}_2-6123775/3t^3+[-0.002596;0.002628]\ . \qquad \text{(A.12)}
\end{aligned}$$

Here only one iteration is performed for the calculation of the initial interval remainder. For $h=0.001$ the following error bounds are obtained:

$$I_{\rho,1}^{(0)}=\begin{pmatrix} I_{\rho,1,1}^{(0)} \\ I_{\rho,2,1}^{(0)}\end{pmatrix}=\begin{pmatrix} [-0.002627;0.002597] \\ [-0.002596;0.002628]\end{pmatrix}\ . \qquad \text{(A.13)}$$

This initial estimation is inflated by the factor 1.1

$$I_{\rho,1}^{(1)}=\begin{pmatrix} I_{\rho,1,1}^{(1)} \\ I_{\rho,2,1}^{(1)}\end{pmatrix}=\begin{pmatrix} [-0.002889;0.002857] \\ [-0.002855;0.002891]\end{pmatrix}\ . \qquad \text{(A.14)}$$

Now the Picard iteration is repeated for this remainder:

$$\begin{aligned}
P_{\rho,1}(\mathfrak{z},t)+I_{\rho,1,1}^{*(1)} &= \mathcal{O}_1(P_\rho(\mathfrak{z},t)+I_{\rho,1}^{(1)})=P_{\rho,1}(\mathfrak{z},t)+[-0.002656;0.002626] \\
&\subseteq P_{\rho,1}(\mathfrak{z},t)+I_{1,1}^{(1)}=P_{\rho,1}(\mathfrak{z},t)+[-0.002889;0.002857]\ , \\
P_{\rho,2}(\mathfrak{z},t)+I_{\rho,2,1}^{*(1)} &= \mathcal{O}_2(P_\rho(\mathfrak{z},t)+I_{\rho,1}^{(1)})=P_{\rho,2}(\mathfrak{z},t)+[-0.002625;0.002658] \\
&\subseteq P_{\rho,2}(\mathfrak{z},t)+I_{2,1}^{(1)}=P_{\rho,2}(\mathfrak{z},t)+[-0.002855;0.002891]\ .
\end{aligned} \qquad \text{(A.15)}$$

Hence, the inclusion requirement is already fulfilled after the first inflation and the corresponding Taylor model provides a enclosure of the flow for $0 \leq t \leq 0.001$.

Now the refinement procedure is applied, which implies the recursive application of the Picard iteration. Here, two further iterations are applied. Denoting $\tilde{I}^{(0)}_{\rho,1} = I^{(1)}_{\rho,1}$ the first iteration gives

$$\begin{aligned} P_{\rho,1}(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,1,1} &= \mathcal{O}_1(P_\rho(\mathfrak{z},t) + \tilde{I}^{(0)}_{\rho,1}) = P_{\rho,1}(\mathfrak{z},t) + [-0.00263; 0.0026] \\ &\subseteq P_{\rho,1}(\mathfrak{z},t) + \tilde{I}^{(0)}_{\rho,1,1} = P_{\rho,1}(\mathfrak{z},t) + [-0.002656; 0.002626] \ , \\ P_{\rho,2}(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,2,1} &= \mathcal{O}_2(P_\rho(\mathfrak{z},t) + \tilde{I}^{(0)}_{\rho,1}) = P_{\rho,2}(\mathfrak{z},t) + [-0.0026; 0.002632] \\ &\subseteq P_{\rho,2}(\mathfrak{z},t) + \tilde{I}^{(0)}_{\rho,2,1} = P_{\rho,2}(\mathfrak{z},t) + [-0.002625; 0.002658] \ . \end{aligned} \tag{A.16}$$

After one more iteration the refined Taylor model

$$\begin{aligned} P_{\rho,1}(\mathfrak{z},t) + \tilde{I}^{(2)}_{\rho,1,1} &= \mathcal{O}_1(P_\rho(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,1}) = P_{\rho,1}(\mathfrak{z},t) + [-0.002627; 0.002597] \\ &\subseteq P_{\rho,1}(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,1,1} = P_{\rho,1}(\mathfrak{z},t) + [-0.00263; 0.0026] \ , \\ P_{\rho,2}(\mathfrak{z},t) + \tilde{I}^{(2)}_{\rho,2,1} &= \mathcal{O}_2(P_\rho(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,1}) = P_{\rho,2}(\mathfrak{z},t) + [-0.002597; 0.002628] \\ &\subseteq P_{\rho,2}(\mathfrak{z},t) + \tilde{I}^{(1)}_{\rho,2,1} = P_{\rho,2}(\mathfrak{z},t) + [-0.002656; 0.002632] \end{aligned} \tag{A.17}$$

is obtained. With $I_{\rho,1} = \tilde{I}^{(2)}_{\rho,1}$ The final Taylor model enclosing the flow for $0 \leq t \leq 0.001$ is

$$T_{\rho,1}(\mathfrak{z},t) = P_\rho(\mathfrak{z},t) + I_{\rho,1} \quad \text{with} \quad \mathfrak{z} \in [-1;1]^2 \quad \text{and} \quad t \in [0; 0.001] \ . \tag{A.18}$$

This Taylor model is now evaluated for $t = 0.001$ and the enclosure of the flow of the ODE after the first time-step is then represent by the Taylor model

$$T_{\rho,1}(\mathfrak{z}) = P_{\rho,1}(\mathfrak{z}) + I_{\rho,1} \tag{A.19}$$

with

$$\begin{aligned} T_{\rho,1,1}(\mathfrak{z}) &= P_{\rho,1,1}(\mathfrak{z}) + I_{\rho,1,1} \\ &= 47.550816 + 4.748627\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 - 0.025\mathfrak{z}_1\mathfrak{z}_2 + [-0.002627; 0.002597] \ , \\ T_{\rho,2,1}(\mathfrak{z}) &= P_{\rho,2,1}(\mathfrak{z}) + I_{\rho,2,1} \\ &= 52.446734 + 0.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 + 0.025\mathfrak{z}_1\mathfrak{z}_2 + [-0.002597; 0.002628] \ . \end{aligned} \tag{A.20}$$

The Second Time Step

In the second integration step the ODE is integrated for $[t_1; t_2] = [t_1; t_1 + h] = [0.001; 0.002]$. The initial polynomial $P^{(0)}$ for the determination of the polynomial part $P_\rho(\mathfrak{z}, t - t_1)$ is given by the polynomial part of the Taylor model $T_{\rho,1}$ at $t = 0.001$, hence

$$\begin{aligned} P^{(0)}_1(\mathfrak{z}, t - t_1) &= 47.550816 + 4.748627\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 - 0.025\mathfrak{z}_1\mathfrak{z}_2 \ , \\ P^{(0)}_2(\mathfrak{z}, t - t_1) &= 52.446734 + 0.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 + 0.025\mathfrak{z}_1\mathfrak{z}_2 \ . \end{aligned} \tag{A.21}$$

Then, the first Picard iteration

$$\begin{aligned} P_1^{(1)}(\mathfrak{z}, t-t_1) &= P_1^{(0)}(\mathfrak{z}) + \int_{t_1}^{t} P_1^{(0)}(\mathfrak{z}, t'-t_1)(1 - P_2^{(0)}(\mathfrak{z}, t'-t_1))dt' \ , \\ P_2^{(1)}(\mathfrak{z}, t-t_1) &= P_2^{(0)}(\mathfrak{z}) + \int_{t_1}^{t} P_2^{(0)}(\mathfrak{z}, t'-t_1)(P_1^{(0)}(\mathfrak{z}, t'-t_1) - 1)dt' \end{aligned} \tag{A.22}$$

is applied. Omitting terms higher than $\rho = 1$ after the first iteration results in

$$\begin{aligned} P_1^{(1)}(\mathfrak{z}, t) &= 47.550816 - 2446.334183(t-t_1) + 4.748627\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 \ , \\ P_2^{(1)}(\mathfrak{z}, t) &= 52.446734 + 2441.433826(t-t_1) + 0.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 \ . \end{aligned} \tag{A.23}$$

The second Picard iteration

$$\begin{aligned} P_1^{(2)}(\mathfrak{z}, t-t_1) &= P_1^{(0)} + \int_{t_1}^{t} P_1^{(1)}(\mathfrak{z}, t'-t_1)(1 - P_2^{(1)}(\mathfrak{z}, t'-t_1))dt' \ , \\ P_2^{(2)}(\mathfrak{z}, t-t_1) &= P_2^{(0)} + \int_{t_1}^{t} P_2^{(1)}(\mathfrak{z}, t'-t_1)(P_1^{(1)}(\mathfrak{z}, t'-t_1) - 1)dt \end{aligned} \tag{A.24}$$

yields (after truncation of terms higher than $\rho = 2$)

$$\begin{aligned} P_1^{(2)}(\mathfrak{z}, t-t_1) &= 47.550816 + 4.748627\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 - 2446.333418(t-t_1) \\ &\quad -0.025\mathfrak{z}_1\mathfrak{z}_2 - 256.480327(t-t_1)\mathfrak{z}_1 \\ &\quad -236.553444(t-t_1)\mathfrak{z}_2 + 4881.760493(t-t_1)^2 \ , \\ P_2^{(2)}(\mathfrak{z}, t-t_1) &= 52.446734 + 340.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 + 2441.438265(t-t_1) \\ &\quad +0.025\mathfrak{z}_1\mathfrak{z}_2 + 260.972829(t-t_1)\mathfrak{z}_1 \\ &\quad +231.071939(t-t_1)\mathfrak{z}_2 - 7325.646718(t-t_1)^2 \ . \end{aligned} \tag{A.25}$$

After the third Picard iteration the desired polynomial $P^{(3)}$ is obtained with

$$\begin{aligned} P_1^{(3)}(\mathfrak{z}, t-t_1) &= 47.550816 + 4.748627\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 - 2446.333418(t-t_1) - 0.025\mathfrak{z}_1\mathfrak{z}_2 \\ &\quad -256.480327(t-t_1)\mathfrak{z}_1 - 236.554441(t-t_1)\mathfrak{z}_2 + 4881.760493(t-t_1)^2 \\ &\quad -1.216242\mathfrak{z}_1^2(t-t_1) - 24.71643\mathfrak{z}_1\mathfrak{z}_2(t-t_1) \\ &\quad +1.2775\mathfrak{z}_2^2(t-t_1) - 5090.6547\mathfrak{z}_1(t-t_1)^2 \\ &\quad +7296.566668\mathfrak{z}_2(t-t_1)^2 + 0.202325 \cdot 10^7(t-t_1)^3 \ , \\ P_2^{(3)}(\mathfrak{z}, t-t_1) &= 52.446734 + 0.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 + 2441.438265(t-t_1) + 0.025\mathfrak{z}_1\mathfrak{z}_2 \\ &\quad +260.972829(t-t_1)\mathfrak{z}_1 \\ &\quad +231.071939(t-t_1)\mathfrak{z}_2 - 7325.646718(t-t_1)^2 + \\ &\quad +1.216242\mathfrak{z}_1^2(t-t_1) - 24.664301\mathfrak{z}_1\mathfrak{z}_2(t-t_1) - 1.27757\mathfrak{z}_2^2 t \\ &\quad +4831.928178\mathfrak{z}_1(t-t_1)^2 - 7530.379858\mathfrak{z}_2(t-t_1)^2 \\ &\quad -0.201918 \cdot 10^7(t-t_1)^3 \ . \end{aligned} \tag{A.26}$$

Now again an appropriate interval remainder bound $I_{\rho,2}$ has to be found such that

$$\begin{aligned} &\mathcal{O}(P_\rho(\mathfrak{z}, t-t_1) + I_{\rho,2}) \subseteq P_\rho(\mathfrak{z}, t-t_1) + I_{\rho,2} \\ &\quad \text{with } P_\rho(\mathfrak{z}, t-t_1) = P^{(3)}(\mathfrak{z}, t-t_1) \\ &\forall \mathfrak{z} \in [\mathfrak{z}] \quad \text{and} \quad \forall t \in [t_1; t_2] = [h; 2h] = [0.001; 0.002] \ . \end{aligned} \tag{A.27}$$

The application of

$$\begin{aligned}
&P_{\rho,1}(\mathfrak{z},t-t_1)+I^{(0)}_{\rho,1,2}=\mathcal{O}_1(P_\rho(\mathfrak{z},t-t_1)+[0;0])\\
&=T_{\rho,1,1}(\mathfrak{z})+\int_{t_1}^{t}(P_{\rho,1}(\mathfrak{z},t'-t_1)+[0;0])(1-(P_{\rho,2}(\mathfrak{z},t'-t_1)+[0;0]))dt'\\
&=P_{\rho,1,1}(\mathfrak{z})+I_{\rho,1,1}+\int_{t_1}^{t}(P_{\rho,1}(\mathfrak{z},t'-t_1)+[0;0])(1-(P_{\rho,2}(\mathfrak{z},t'-t_1)+[0;0]))dt'\ ,\\
&P_{\rho,2}(\mathfrak{z},t)+I^{(0)}_{\rho,2,2}=\mathcal{O}_2(P_\rho(\mathfrak{z},t-t_1)+[0;0])\\
&=T_{\rho,2,1}(\mathfrak{z})+\int_{t_1}^{t}(P_{\rho,2}(\mathfrak{z},t'-t_1)+[0;0])((P_{\rho,1}(\mathfrak{z},t'-t_1)+[0;0])-1)dt'\\
&=P_{\rho,2,1}(\mathfrak{z})+I_{\rho,2,1}+\int_{t_1}^{t}(P_{\rho,2}(\mathfrak{z},t'-t_1)+[0;0])((P_{\rho,1}(\mathfrak{z},t'-t_1)+[0;0])-1)dt'
\end{aligned}\tag{A.28}$$

results in the initial estimate

$$I^{(0)}_{\rho,2}=\begin{pmatrix} I^{(0)}_{\rho,1,2}\\ I^{(0)}_{\rho,2,2}\end{pmatrix}=\begin{pmatrix}[-0.006213;0.006205]\\ [-0.006226;0.006237]\end{pmatrix}\ .\tag{A.29}$$

This initial estimation is inflated by the factor 1.1:

$$I^{(1)}_{\rho,2}=\begin{pmatrix} I^{(1)}_{\rho,1,2}\\ I^{(1)}_{\rho,2,2}\end{pmatrix}=\begin{pmatrix}[-0.006834;0.006825]\\ [-0.006881;0.006861]\end{pmatrix}\ .\tag{A.30}$$

Another Picard iteration yields in

$$\begin{aligned}
&P_{\rho,1}(\mathfrak{z},t-t_1)+I^{*(1)}_{\rho,1,2}=\mathcal{O}_1(P_\rho(\mathfrak{z},t-t_1)+I^{(1)}_{\rho,2})\\
&=T_{\rho,1,1}+\int_{t_1}^{t}(P_{\rho,1}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,1,2})(1-(P_{\rho,2}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,2,2}))dt'\\
&=P_{\rho,1,1}(\mathfrak{z},t-t_1)+I_{\rho,1,1}+\int_{t_1}^{t}(P_{\rho,1}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,1,1})(1-(P_{\rho,2}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,2,2}))dt'\\
&=P_{\rho,1,1}(\mathfrak{z},t-t_1)+[-0.006284;0.006275]\subset P_{\rho,1,1}(\mathfrak{z},t-t_1)+I_{\rho,1,1}\\
&=P_{\rho,1,1}(\mathfrak{z},t-t_1)+[-0.006834;0.006825]\ ,\\
&P_{\rho,2}(\mathfrak{z},t)+I^{*(1)}_{\rho,2,2}=\mathcal{O}_2(P_\rho(\mathfrak{z},t-t_1)+I^{(1)}_{\rho,2})\\
&=T_{\rho,2,1}(\mathfrak{z})+\int_{t_1}^{t}(P_{\rho,2}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,2,2})((P_{\rho,1}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,1,2})-1)dt'\\
&=P_{\rho,2,1}(\mathfrak{z},t-t_1)+I_{\rho,2,1}+\int_{t_1}^{t}(P_{\rho,2}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,2,2})((P_{\rho,1}(\mathfrak{z},t'-t_1)+I^{(1)}_{\rho,1,2})-1)dt'\\
&=P_{\rho,1,1}(\mathfrak{z},t-t_1)+[-0.006296;0.006308]\subset P_{\rho,1,1}(\mathfrak{z},t-t_1)+I_{\rho,1,1}\\
&=P_{\rho,1,1}(\mathfrak{z},t-t_1)+[-0.006848;0.006861]\ .
\end{aligned}\tag{A.31}$$

Again the inclusion requirement is fulfilled after the first inflation. After two additional applications of the Picard iteration for the refinement of the remainder bounds, the final remainder error is given by

$$I_{\rho,2}=\begin{pmatrix} I_{\rho,1,2}\\ I_{\rho,2,2}\end{pmatrix}=\begin{pmatrix}[-0.006215;0.006207]\\ [-0.006227;0.006239]\end{pmatrix}\ .\tag{A.32}$$

The final Taylor model enclosing the flow for $t \in [t_1; t_2] = [t_1; t_1 + h] = [h; 2h]$ is

$$T_{\rho,2}(\mathfrak{z}, t-t_1) = P_{\rho,2}(\mathfrak{z}, t-t_1)+I_{\rho,2} \quad \text{with} \quad \mathfrak{z} \in [-1;1]^2 \quad \text{and} \quad t \in [t_1; t_2] = [t_1; t_1+h] = [h; 2h] \ . \tag{A.33}$$

To obtain the Taylor model enclosing the flow at $t = t_2 = t_1 + h = 0.001 + 0.001 = 0.002$ this Taylor model has to be evaluated for $t = t_2 = 0.002$ or alternatively the expressions $t - t_1$ are replaced by h. This yields the Taylor model after the second integration step:

$$T_{\rho,2}(\mathfrak{z}) = P_{\rho,2}(\mathfrak{z}) + I_{\rho,2} \tag{A.34}$$

with

$$\begin{aligned} T_{\rho,1,2}(\mathfrak{z}) &= P_{\rho,1,2}(\mathfrak{z}) + I_{\rho,1,2} \\ &= 45.111387 + 4.487056\mathfrak{z}_1 - 0.473132\mathfrak{z}_2 - 0.001216\mathfrak{z}_1^2 - 0.049716\mathfrak{z}_1\mathfrak{z}_2 + 0.0012775\mathfrak{z}_2^2 \\ &\quad +[-0.006215; 0.006207] \ , \\ T_{\rho,2,2}(\mathfrak{z}) &= P_{\rho,2,2}(\mathfrak{z}) + I_{\rho,2,2} \\ &= 54.878827 + 0.519297\mathfrak{z}_1 + 5.462169\mathfrak{z}_2 + 0.001216\mathfrak{z}_1^2 + 0.049666\mathfrak{z}_1\mathfrak{z}_2 - 0.0012775\mathfrak{z}_2^2 \\ &\quad +[-0.006227; 0.006239] \ . \end{aligned} \tag{A.35}$$

A.2 Nonlinear Example with Preconditioning

Consider again the Lotka-Volterra equations (A.2). In each integration step, the left Taylor models are constructed via QR factorization of the linear parts of the integrated Taylor models of the previous integration step. The initial state variables are given by $z_1(0), z_2(0) \in [45; 55]$. The order is again set to $\rho = 3$ in time and initial conditions and the step-size to $h = 0.001$. All numbers are again displayed rounded to six decimal digits.

In the first integration step the initial set at $t = 0$ is described by the left Taylor model $T_{l,0}$ with

$$\begin{aligned} T_{l,1,0} &= 50 + 5\bar{\mathfrak{z}}_1 \ , \\ T_{l,2,0} &= 50 + 5\bar{\mathfrak{z}}_2 \quad \text{with} \quad \bar{\mathfrak{z}}_1, \bar{\mathfrak{z}}_2 \in [-1; 1] \ . \end{aligned} \tag{A.36}$$

The right Taylor model $T_{r,0}$ a $t = 0$ is the identity Taylor model

$$\begin{aligned} T_{r,1,0} &= \mathfrak{z}_1 \ , \\ T_{r,2,0} &= \mathfrak{z}_2 \quad \text{with} \quad \mathfrak{z}_1, \mathfrak{z}_2 \in [-1; 1] \ . \end{aligned} \tag{A.37}$$

The first integration step is performed as in Section A.1 , but as described in Section 4.4.2 the left Taylor model is affected, hence the left Taylor model at $t = 0.001$ is given by

$$\begin{aligned} \check{T}_{l,1,1}(\bar{\mathfrak{z}}) &= \check{P}_{l,1,1}(\bar{\mathfrak{z}}) + \check{I}_{l,1,1} \\ &= 47.550816 + 4.748627\bar{\mathfrak{z}}_1 - 0.243875\bar{\mathfrak{z}}_2 - 0.025\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2 + [-0.002627; 0.002597] \ , \\ \check{T}_{l,2,1}(\bar{\mathfrak{z}}) &= \check{P}_{l,2,1}(\bar{\mathfrak{z}}) + \check{I}_{l,2,1} \\ &= 52.446734 + 0.256125\bar{\mathfrak{z}}_1 + 5.238627\bar{\mathfrak{z}}_2 + 0.025\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2 + [-0.002597; 0.002628] \ . \end{aligned} \tag{A.38}$$

The right Taylor model has been unaffected by the integration so far. Now the left Taylor model is preconditioned before the integration is continued. The linear parts of $\check{T}_{l,2,1}$ and $\check{T}_{l,1,1}$ are extracted and the Matrix $\check{C}_1$ is obtained, with

$$\check{C}_1 = \begin{pmatrix} 4.748627 & -0.243875 \\ 0.256125 & 5.238627 \end{pmatrix} \ . \tag{A.39}$$

Sorting of $\check{C}_1$ yields

$$\tilde{\check{C}}_1 = \begin{pmatrix} -0.243875 & 4.748627 \\ 5.238627 & 0.256125 \end{pmatrix} \ . \tag{A.40}$$

The QR-decomposition leads to

$$\begin{aligned} \tilde{\check{C}} &:= \begin{pmatrix} -0.243875 & 4.748627 \\ 5.238627 & 0.256125 \end{pmatrix} \\ &= \begin{pmatrix} -0.046502 & 0.998918 \\ 0.998918 & 0.046502 \end{pmatrix} \cdot \begin{pmatrix} 5.244300 & 0.035023 \\ 0 & 4.7553400 \end{pmatrix} =: QR \ . \end{aligned} \tag{A.41}$$

The left Taylor model in the second integration step is constructed from the constant parts of $\check{T}_{l,1,1}$ and $\check{T}_{l,2,1}$ and from Q. This leads to

$$\begin{aligned} \bar{T}_{l,1,1} &= 47.550816 - 0.046502\bar{\mathfrak{z}}_1 + 0.998918\bar{\mathfrak{z}}_2 \ , \\ \bar{T}_{l,2,1} &= 52.446734 + 0.998918\bar{\mathfrak{z}}_1 + 0.046502\bar{\mathfrak{z}}_2 \ . \end{aligned} \tag{A.42}$$

The nonlinear parts and the interval remainders of $\check{T}_{l,1,1}$ and $\check{T}_{l,2,1}$ are collected in the right Taylor models. According to (4.174) the right Taylor models become

$$\begin{aligned} \bar{T}_{r,1,1} &= 0.03502\mathfrak{z}_1 + 5.244273\mathfrak{z}_2 + 0.00261\mathfrak{z}_1\mathfrak{z}_2 + \bar{I}_{r,1,1} \ , \\ \bar{T}_{r,2,1} &= 4.75537\mathfrak{z}_1 - 0.00238\mathfrak{z}_1\mathfrak{z}_2 + \bar{I}_{r,2,1} \end{aligned} \tag{A.43}$$

with $\bar{I}_{r,1,1} = [-0.002714; 0.002748]$ and $\bar{I}_{r,2,1} = [-0.002745; 0.002717]$. The composition of the left Taylor model $\bar{T}_l$ and the right Taylor model $\bar{T}_r$

$$\begin{aligned} \bar{T}_1 &= \bar{T}_{l,1,1}(\bar{T}_{r,1,1}, \bar{T}_{r,2,1}) = \bar{P}_{l,1,1}(\bar{T}_{r,1,1}, \bar{T}_{r,2,1}) + \bar{I}_{l,1,1} \\ &= 47.550816 - 0.046502 \cdot \bar{T}_{r,1,1} + 0.998918 \cdot \bar{T}_{r,2,1} \\ &= 47.550816 + 4.7486\mathfrak{z}_1 - 0.243875\mathfrak{z}_2 - 0.025\mathfrak{z}_1\mathfrak{z}_2 + \bar{I}_{l,1,1} \ , \\ \bar{T}_2 &= \bar{T}_{l,2,1}(\bar{T}_{r,1,1}, \bar{T}_{r,2,1}) = \bar{P}_{l,2,1}(\bar{T}_{r,1,1}, \bar{T}_{r,2,1}) + \bar{I}_{l,2,1} \\ &= 52.446734 + 0.998918 \cdot \bar{T}_{r,1,1} + 0.046502 \cdot \bar{T}_{r,1,2} \\ &= 52.446734 + 0.256125\mathfrak{z}_1 + 5.238627\mathfrak{z}_2 + 0.025\mathfrak{z}_1\mathfrak{z}_2 + \bar{I}_{l,2,1} \ . \end{aligned} \tag{A.44}$$

gives indeed the same polynomial part like $\check{T}_l$ however the remainder bounds

$$\begin{pmatrix} \bar{I}_{l,1,1} \\ \bar{I}_{l,2,1} \end{pmatrix} = Q \cdot \left[Q^T \cdot \begin{pmatrix} \check{I}_{l,1,0} \\ \check{I}_{l,2,0} \end{pmatrix} \right] = \begin{pmatrix} [-0.00286; 0.00284] \\ [-0.00283; 0.002871] \end{pmatrix} \tag{A.45}$$

are slightly wider:

$$\begin{aligned} \bar{I}_{1,1} &= [-0.00286; 0.00284] \supset \check{I}_{l,1,1} = [-0.002627; 0.002597] \ , \\ \bar{I}_{2,1} &= [-0.00283; 0.002871] \supset \check{I}_{l,2,1} = [-0.002597; 0.002628] \ . \end{aligned} \tag{A.46}$$

Still, in most nonlinear systems preconditioned integration is the superior method with respect to accuracy in the long run. The advantage of preconditioning becomes only apparent

after several integration steps. Note, that for the integration algorithm the computation of the composition is not required.
For the right Taylor models in (A.43)

$$\begin{aligned}\mathrm{Rg}(\bar{T}_{r,1,1}) &\subseteq 0.03502\cdot[-1;1]+5.244273\cdot[-1;1]\\ &\quad+0.00261\cdot[-1;1]\cdot[-1;1]+[-0.002714;0.002748]=[-5.30543;5.30544]\ ,\\ \mathrm{Rg}(\bar{T}_{r,2,1}) &\subseteq 4.75537\cdot[-1;1]-0.00238\cdot[-1;1]\cdot[-1;1]+[-0.00274;0.00272]\\ &\quad=[-4.77918;4.77919]\end{aligned} \tag{A.47}$$

holds. To continue the integration of the flow, with only the left hand Taylor model, the flow of the ODE has to be enclosed by left Taylor model. Therefore either the domain of the independent variables $\bar{\mathfrak{z}}$ of the left Taylor model are modified and the integration be continued with

$$\begin{aligned}&\bar{T}_{l,1,1}=47.5508-0.04650\bar{\mathfrak{z}}_1+0.998918\bar{\mathfrak{z}}_2\ ,\\ &\bar{T}_{l,2,1}=52.4467+0.99892\bar{\mathfrak{z}}_1+0.046502\bar{\mathfrak{z}}_2\\ &\text{with}\quad \bar{\mathfrak{z}}_1\in[-5.30543;5.30544],\quad\text{and}\quad \bar{\mathfrak{z}}_2\in[-4.77918;4.77918]\end{aligned} \tag{A.48}$$

or the left and right Taylor model can be modified by an additional Transformation using a Scaling matrix S as described in Section 4.4.2 such that the domain interval vector for $\bar{\mathfrak{z}}$ remains $[-1;1]^n$. To ensure this a linear transformation on the left and right Taylor model according to

$$\bar{T}_{l,1}\circ\bar{T}_{r,1}=\bar{T}_{l,1}\circ(S\circ S^{-1})\circ\bar{T}_{r,1}=(\bar{T}_{l,1}\circ S)\circ(S^{-1}\circ\bar{T}_{r,1}) \tag{A.49}$$

has to be applied such that

$$\mathrm{Rg}((S^{-1}\circ\bar{T}_{r,1}))\subseteq[-1;1]^n \tag{A.50}$$

holds. In this example

$$S^{-1}=\begin{pmatrix}0.1885 & 0\\ 0 & -0.2092\end{pmatrix} \tag{A.51}$$

is obtained. Thus for S,

$$S=\begin{pmatrix}5.30544 & 0\\ 0 & -4.77918\end{pmatrix} \tag{A.52}$$

holds. The new left Taylor model $T_{l,1}$ is then given by

$$\begin{aligned}&T_{l,1,1}=47.5508-0.24685\bar{\mathfrak{z}}_1-4.77678\bar{\mathfrak{z}}_2\ ,\\ &T_{l,2,1}=52.4467+5.30246\bar{\mathfrak{z}}_1-0.22237\bar{\mathfrak{z}}_2\\ &\text{with}\quad \bar{\mathfrak{z}}_1,\bar{\mathfrak{z}}_2\in[-1;1]\ .\end{aligned} \tag{A.53}$$

The corresponding right Taylor models are

$$T_{r,1}=S^{-1}\circ\bar{T}_{r,1}=\begin{pmatrix}0.1885 & 0\\ 0 & -0.2092\end{pmatrix}\cdot\begin{pmatrix}0.03502\bar{\mathfrak{z}}_1+5.244273\bar{\mathfrak{z}}_2+0.00261\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2+\bar{I}_{r,1,1}\\ 4.75537\bar{\mathfrak{z}}_1-0.00238\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2+\bar{I}_{r,2,1}\end{pmatrix}\ . \tag{A.54}$$

For the second integration step the initial set defined by $T_{l,1}$. The integrated left Taylor model $\check{T}_{l,2}$ becomes

$$\begin{aligned}\check{T}_{l,1,2} &= 45.111387-0.478998\bar{\mathfrak{z}}_1-4.515291\bar{\mathfrak{z}}_2+0.001308\bar{\mathfrak{z}}_1^2-0.025273\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2\\ &\quad-0.001062\bar{\mathfrak{z}}_2^2+\check{I}_{l,1,1}\ ,\\ \check{T}_{l,2,2} &= 54.878827+5.528730\bar{\mathfrak{z}}_1-0.488162\bar{\mathfrak{z}}_2-0.001388\bar{\mathfrak{z}}_1^2\\ &\quad-0.025273\bar{\mathfrak{z}}_1\bar{\mathfrak{z}}_2+0.001062\bar{\mathfrak{z}}_2^2+\check{I}_{l,2,1}\ .\end{aligned} \tag{A.55}$$

with $\check{I}_{l,1,1} = [-0.002901; 0.002929]$ and $\check{I}_{l,2,1} = [-0.002950; 0.002926]$.

The flow at t_2 is then given by the composition of the integrated Taylor model $\check{T}_{l,2}$ and the previous Taylor model $T_{r,1}$, which results in

$$T_{\rho,2} = \check{T}_{l,2} \circ T_{r,1} \tag{A.56}$$

with

$$\begin{aligned} T_{\rho,1,2} = \check{T}_{l,1,2}(T_{r,1,1}, T_{r,2,1}) &= 45.111387 + 4.487056\mathfrak{z}_1 - 0.4731328\mathfrak{z}_2 - 0.001216\mathfrak{z}_1^2 \\ &\quad -0.049654\mathfrak{z}_1\mathfrak{z}_2 + 0.001275\mathfrak{z}_2^2 - 0.000112\mathfrak{z}_1^2\mathfrak{z}_2 \\ &\quad +0.000137\mathfrak{z}_1\mathfrak{z}_2^2 + [-0.0057772; 0.005769] \ , \\ T_{\rho,2,2} = \check{T}_{l,2,2}(T_{r,1,1}, T_{r,2,1}) &= 54.878827 + 0.521929\mathfrak{z}_1 + 5.462169\mathfrak{z}_2 + 0.001216\mathfrak{z}_1^2 \\ &\quad +0.049604\mathfrak{z}_1\mathfrak{z}_2 - 0.001275\mathfrak{z}_2^2 + 0.000112\mathfrak{z}_1^2\mathfrak{z}_2 \\ &\quad -0.000137\mathfrak{z}_1\mathfrak{z}_2^2 + [-0.005777; 0.005769] \ . \end{aligned} \tag{A.57}$$

As mentioned above, for continuing the integration the calculation of the composition of the left and right Taylor model is not required.

Bibliography

[1] T. Alamo, J.M. Bravo, and E. F. Camacho. Guaranteed State Estimation by Zonotopes. Proc. of the 42nd IEEE Conference on Decision and Control, pages 5831–5836, Maui,Hawai, USA, 2003.

[2] G. Alefeld. Inclusion Methods for Systems of Nonlinear Equations - The Interval Newton Method and Modifications. *Topics in Validated Computation, Elsevier Science*, pages 7–26, 1994.

[3] F. Antritter, M. Kletting, and E. P. Hofer. Robustness Analysis of Flatness Based Control Using Interval Methods. *Intl. J. of Control*, 80:816–823, 2007.

[4] E. Auer, A. Rauh, E. P. Hofer, and W. Luther. Validated Modeling of Mechanical Systems with SmartMOBILE: Improvement of Performance by ValEncIA-IVP. Proc. Dagstuhl-Seminar 06021: Reliable Implementation of Real Number Algorithms: Theory and Practice, volume 5045 of *Lecture Notes on Computer Science*, pages 1–28, Springer Verlag, Dagstuhl, Germany, 2008.

[5] M. Berz and K. Makino. Verified Integration of ODEs and Flows Using Differential Algebraic Methods on High-Order Taylor Models. *Reliable Computing*, 4:361–369, 1998.

[6] J. C. Burkill. Functions of Intervals. pages 375–446 , Proc. of the London Mathematical Society, 1924.

[7] C. Canudas de Wit, H. Olsson, K. J. Åström, and P. Lischinsky. A New Model for Control of Systems with Friction. *IEEE Transactions on Automatic Control*, 40(3):419–425, 1995.

[8] H. Cornelius and R. Lohner. Computing the Range of Values of Real Functions with Accuracy Higher than Second Order. *Computing*, 33:331–347, 1984.

[9] T. Csendes and D. Ratz. Subdivision Direction Selection in Interval Methods for Global Optimization. *SIAM Journal on Numerical Analysis*, 34(3):922–938, 1997.

[10] F. Daum. Nonlinear Filters: Beyond the Kalman Filter . *IEEE Aerospace and Electronic Systems Magazine*, 20(8):57–69, 2005.

[11] Y. Deville, M. Janssen, and P. van Hentenryck. Consistency Techniques for Ordinary Differential Equations. *Constraint*, 7(3–4):289–315, 2002.

[12] I. Eble. *Über Taylormodelle*. PhD thesis, Universität Karlsruhe, 2006.

[13] S. Engell. *Entwurf Nichtlinearer Regelungen*. Oldenbourg Verlag, Munich, Germany, 2nd edition, 1995.

[14] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples. *Int. J. of Control*, 61:1327–1361, 1995.

[15] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. A Lie-Bäcklund Approach to Equivalence and Flatness of Nonlinear Systems. *IEEE Trans. Aut. Control*, 44:922–937, 1999.

[16] M. Fliess and J. Rudolph. Local Tracking Observers for Flat Systems. Proc. of the Symposium on Control, Optimization and Supervision, CESA '96 IMACS Multiconference, pages 213–217, Lille, France, 1996.

[17] O. Föllinger. *Regelungstechnik*. Hüthig Buch Verlag Heidelberg, Heidleberg, 7th edition, 1992.

[18] E. Freund. *Zeitvariable Mehrgrößensysteme*. Lecture notes in operations and mathematical science 57, Springer-Verlag, New York, 1971.

[19] J. L. Gouze, A. Rapaport, and Z .M. Hadj-Sadok. Interval observers for uncertain biological systems. *J. of Ecological Modelling, 133:45-56, 2000*.

[20] V. Hagenmeyer and E. Delaleau. Exact Feedforward Linearisation Based on Differential Flatness: The SISO Case. In *Nonlinear and Adaptive Control (NCN4 2001), Lecture notes in Control and Information Sciences*, volume 281, pages 161–170. Springer, Berlin, Heidelberg, 2001.

[21] V. Hagenmeyer and E. Delaleau. Exact Feedforward Linearization Based on Differential Flatness. *Int. J. of Control*, 76:537–556, 2003.

[22] E. Hansen and S. Sengupta. Bounding Solutions of Systems of Equations Using Interval Analysis. *BIT*, pages 203–211, 1981.

[23] E. R. Hansen. *Global Optimization Using Interval Analysis*. Marcel Dekker, New York, 1992.

[24] J. Heeks. *Charakterisierung Unsicherer Systeme mit Intervallarithmetischen Methoden*. PhD thesis, Universität Ulm, Abteilung Mess-, Regel- und Mikrotechnik, 2002.

[25] M. Henze, P. Harremoës, E. Arvin, and J. la Cour Jansen. *Wastewater Treatment*. Springer–Verlag, (Berlin, 2002).

[26] L. Jaulin. Nonlinear Bounded-Error State Estimation of Continuous-Time Systems. *Automatica*, 38:1079–1082, 2002.

[27] L. Jaulin, M. Kieffer, O. Didrit, and É. Walter. *Applied Interval Analysis*. Springer–Verlag, London, U.K., 2001.

[28] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002.

[29] M. Köhne. Analyse und Regelung Biologischer Abwasserreinigungsprozesse in Kläranlagen. *at-Automatisierungstechnik*, 46(5):215–234, 1998. R. Oldenbourg Verlag, Munich.

[30] M. Kieffer, L. Jaulin, and E. Walter. Guaranteed Recursive Nonlinear State Estimation Using Interval Analysis Proc. 37th IEEE Conference on Decision and Control. pages 3966–3971, Tampa, USA, 1998.

[31] M. Kieffer, L. Jaulin, and E. Walter. Guaranteed Recursive Nonlinear State Bounding Using Interval Analysis. *International Journal of Adaptative Control and Signal Processing*, 6(3):193–218, 2002.

[32] M. Kieffer and E. Walter. Nonlinear Parameter and State Estimation for Cooperative Systems in a Bounded-Error Context. Proc. International Dagstuhl Seminar: Software with Result Verification. Revised Papers edited by R. Alt, A. Frommer, R.B. Kearfott, W. Luther, Springer, Lecture Notes in Computer Science 2991,pages 107-123,2004.

[33] M. Kletting, F. Antritter, and E. P. Hofer. Guaranteed Robust Tracking with Flatness Based Controllers Applying Interval Methods. Proc. of 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany, 2006.

[34] M. Kletting, F. Antritter, and E. P. Hofer. Robust Flatness Based Controller Design Using Interval Methods,Proc. of 7th IFAC Symposium on Nonlinear Control Systems (NOLCOS),Pretoria, South Africa. 2007.

[35] M. Kletting, A. Rauh, H. Aschemann, and E. P. Hofer. Consistency Techniques for Simulation of Wastewater Treatment Processes with Uncertainties, DVD Proc. 16th IFAC World Congress, Prague, Czech Republic, No. M19-TO/3, 2005.

[36] M. Kletting, A. Rauh, H. Aschemann, and E. P. Hofer. Interval Observer Design for Nonlinear Systems with Uncertain Time-Varying Parameters, Proc. 12th IEEE Intl. Conference on Methods and Models in Automation and Robotics MMAR, Miedzyzdroje, Poland, IEEE 0-7803-9514-X/06, pages 361–366, 2006.

[37] M. Kletting, A. Rauh, H. Aschemann, and E. P. Hofer. Interval Observer Design Based on Taylor Models for Nonlinear Uncertain Continuous-Time Systems. Proc. of 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany, 2006.

[38] M. Kletting, A. Rauh, H. Aschemann, and E. P. Hofer. Consistency Tests in Guaranteed Simulation of Nonlinear Uncertain Systems with Application to an Activated Sludge Process. *Journal of Computational and Applied Mathematics*, 199(2):213–219, 2007.

[39] O. Knüppel. PROFIL/BIAS, A Fast Interval Library. *Computing*, 53:277–287, 1994.

[40] I. Krasnochtanova. Optimized Interval Algorithms for Simulation and Controller Design for Nonlinear Uncertain Systems Applied to Processes in Biological Wastewater Treatment, Master Thesis, Institute of Measurement Control and Microthechnology,University of Ulm, Germany, 2007.

[41] J. Levine, J. Lottin, and J. C. Ponsart. A Nonlinear Approach to the Control of Magnetic Bearings. *IEEE Trans. on Control Systems Technology*, pages 545 – 552, 1996.

[42] Y. Lin and M. A. Stadtherr. Deterministic Global Optimization for Dynamic Systems Using Interval Analysis. Proc. of 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany, 2006.

[43] R. Lohner. *Einschließung der Lösung Gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen*. PhD thesis, Universität Karlsruhe (TH), Fakultät für Mathematik, 1988.

[44] K. Makino. Linear ODE example. bt.pa.msu.edu/TM/Miami/L-ODEexample.pdf.

[45] K. Makino. *Rigorous analysis of nonlinear motion in particle accelerators*. PhD thesis, Michigan State University, 1998.

[46] K. Makino and M. Berz. Suppression of the Wrapping Effect by Taylor Model-Based Verified Integrators: Long-Term Stabilization by Preconditioning. *International Journal of Differential Equations and Applications*, 2005.

[47] K. Makino and M. Berz. Suppression of the Wrapping Effect by Taylor Model-Based Verified Integrators: Long-Term Stabilization by Shrink Wrapping. *International Journal of Differential Equations and Applications*, 2005.

[48] K. Makino and M. Berz. Suppression of the Wrapping Effect by Taylor Model-Based Verified Integrators: The Single Step. *International Journal of Pure and Applied Mathematics*, 2006.

[49] I. Mayergoyz. *Mathematical Models of Hysteresis and Their Applications*. Elsevier, Amsterdam, The Netherlands, 2003.

[50] R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs N. J.,USA, 1966.

[51] R. E. Moore. *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, 1979.

[52] N. Nedialkov, Kenneth. R. Jackson, and John. D. Pryce. An Effective High-Order Interval Method for Validating Existence and Uniqueness of the Solution of an IVP for an ODE. *Reliable Computing*, 7:449–465, 2001.

[53] N. S. Nedialkov. *Computing Rigorous Bounds on the Solution of an Initial Value Problem for an Ordinary Differential Equation*. PhD thesis, Graduate Department of Computer Science, University of Toronto, 1999.

[54] M. Neher, K. R. Jackson, and N.S. Nedialkov. On Taylor Model Based Integration of ODEs. *Technical Report, Department of Computer Science, University of Toronto*, 2005.

[55] A. Neumaier. *Interval Methods for Systems of Equations*. Cambridge University Press, Encyclopedia of Mathematics, (Cambridge, 1990).

[56] A. Neumaier. Taylor forms - use and limits. *Technical Report, Universität Wien*, 2002.

[57] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, USA, 1991.

[58] T. Raissi, N. Ramdani, and Y. Candau. Set Membership State and Parameter Estimation for Systems Described by Nonlinear Differential Equations. *Automatica*, 40:1771–1777, 2004.

[59] A. Rauh, M. Kletting, H. Aschemann, and E. P. Hofer. Application of Interval Arithmetic Simulation Techniques to Wastewater Treatment Processes. Proc. MIC 2004, pages 287–293, Grindelwald, Switzerland, 2004.

[60] A. Rauh, M. Kletting, H. Aschemann, and E. P. Hofer. Robust Controller Design for Bounded State and Control Variables and Uncertain Parameters Using Interval Methods. Proc. 5th Intl. Conference on Control and Automation ICCA, pages 777–782, Budapest, Hungary. 2005.

[61] A. Rauh, M. Kletting, H. Aschemann, and E. P. Hofer. Interval Methods for Simulation of Dynamical Systems with State-Dependent Switching Characteristics. Proc. IEEE International Conference on Control Applications CCA, pages 355–360, Munich, Germany. 2006.

[62] A. Rauh, M. Kletting, H. Aschemann, and E. P. Hofer. Reduction of Overestimation in Interval Arithmetic Simulation of Biological Wastewater Treatment Processes. *Journal of Computational and Applied Mathematics*, 199(2):207–212, 2007.

[63] R. Rihm. *Über Einschließungsverfahren für gewöhnliche Anfangswertprobleme und ihre Anwendung auf Differentialgleichungen mit unstetiger rechter Seite.* PhD thesis, Universität Karlsruhe, 1993.

[64] R. Rihm. Interval Methods for Initial Value Problems. *Topics in Validated Computing*, pages 173–207, 1994.

[65] R. Rihm. Implicit Methods for Enclosing Solutions of ODEs. *Journal of Universal Computer Science*, 4(2):202–209, 1998.

[66] J. Rohn. A Handbook of Results Interval Linear Problems. Technical report.

[67] S. M. Rump. IntLab. Version 4.1.1 (22.01.2003): `http://www.ti3.tu-harburg.de/~rump/intlab/`.

[68] S. M. Rump. INTLAB — INTerval LABoratory. In Tibor Csendes, editor, *Developments in Reliable Computing*, pages 77–104. Kluver Academic Publishers, 1999.

[69] G. Schrabereger, H. Olsson, K. J. Åström, and P. Lischinsky. A New Model for Control of Systems with Friction. *IEEE Transactions on Automatic Control*, 40(3):419–425, 1995.

[70] H. Schwarz. *Nichtlineare Regelungssysteme.* R. Oldenbourg Verlag, Munich, Germany, 1991.

[71] H. Sira-Ramirez and S. K. Agrawal. *Differentially Flat Systems.* Marcel Dekker, New York, 2004.

[72] O. Stauning. *Automatic Validation of Numerical Solutions.* PhD thesis, Technical University of Denmark,DK-2800, Lyngby, Denmark, 1997.

[73] E. W. Weisstein. Double Pendulum.
`http://scienceworld.wolfram.com/physics/DoublePendulum.html`.

[74] R. C. Young. The Algebra of Many-Valued Quantities. *Math. Ann.*, 104:260–290, 1931.