FACULTY OF MATHEMATICS, PHYSICS AND COMPUTER SCIENCE

INSTITUTE OF MATHEMATICS

Discrete Approximation of Feasible Sets and Direct Methods for Optimal Control Problems with State Constraints

Diploma Thesis
by
Benjamin Hell

Date: 14.4.2010

Supervisor:
Prof. Dr. Frank Lempio
Dr. Robert Baier

FACULTY OF MATHEMATICS, PHYSICS AND COMPUTER SCIENCE

INSTITUTE OF MATHEMATICS

Discrete Approximation of Feasible Sets and Direct Methods for Optimal Control Problems with State Constraints

Diploma Thesis
by
Benjamin Hell

Date: 14.4.2010

Supervisor:
Prof. Dr. Frank Lempio
Dr. Robert Baier

Acknowledgements:

At this point I would like to thank Prof. Frank Lempio for his great support for this diploma thesis and the whole time of my studies at the university of Bayreuth. I am also very greatful for the great amount of time Robert Baier spent with me on talking about the content of this article and a lot more. It has really been a pleasure for me to work with him and I hope this will continue sometime in the future. Additionally my thanks go to Jürgen Pannek, who has been a great help for the programming part of this thesis. Thank you all.

Table of Contents

Abbreviations:

ODE:	ordinary differential equation
OCP:	optimal control problem
DI:	differential inclusion
DIC:	constrained differential inclusion
DDI:	discrete differential inclusion
DDIC:	constrained discrete differential inclusion

1 Introduction

This thesis is about delivering convergence results for a pretty wide class of optimal control problems, when using direct methods for discretization. Those control problems may include pure state constraints, which is one of the major difficulties. The majority of this article deals with convergence of the discrete solution for the state. But it is the authors strong believe, that the methods presented here can be used to show convergence of the discrete solution for the control, too.

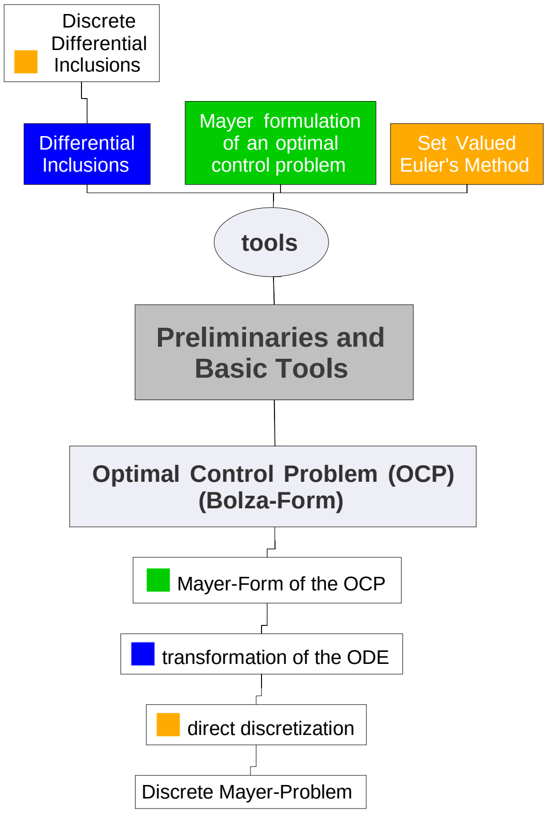
This document is made up of five core chapters:
Chapter 2 introduces the class of control problems considered in quite some variants. Each of these forms has its theoretical advantages and disadvantages, but the Mayer-Form builds the basis for most investigations. Chapter 2 also introduces the notations used throughout the paper and a lot of basic tools, that will play a great role in the following chapters but are not really part of the core concept.
Chapter 3 then derives the central convergence result of this thesis for the problem class presented in chapter 2. The strategy, used for deriving that result, might be applied to other problem classes, that are not presented in this article. Showing that this might be possible was the intention of creating chapter 4, which takes a look at the methods used in chapter 3 in a more general way. In fact showing how flexible the modularized approach is, is the main goal of that part of the article. Chapter 4 is presented after chapter 3, because it is the authors opinion, that the concept is easier to understand in a more or less concrete case. Although quite some things are repeated in chapter 3, the statements made are usually more general and do involve less assumptions.
One major role plays the so called Approximation Property, which is a statement about the approximation of feasible sets. The result delivered is already needed in chapter 3, but as the property is the core of this thesis, a whole chapter has been devoted to it. This chapter contains some of the most interesting parts of the whole article. For example, the way state constraints can be handled is included there. That also contains some yet unproven claims for multidimensional state constraints, an interesting aspect in current research.
The thesis concludes with chapter 6, which is a large chapter on examples. A majority of the work has been put into that chapter, because calculating discrete solutions for the examples shown is no easy task. The chapter itself is intended to shed some light on how the whole theory presented before may be applied to concrete problems. This includes showing how to obtain an inverse stability property, verifying central properties for the constraints to know that the Approximation Property holds and of course a lot

of numerical results, that should give more insight to what assumptions are actually needed. For some examples, properties needed to apply the theory do not hold, but good convergence results can be obtained anyway. By means of such results, ways on explaining the observed behavior are presented.

2 Preliminaries and Basic Tools

2.1 Overview

This section covers the basic tools and definitions, which the following chapters will make use of. The main part is describing the class of optimal control problems nearly the whole thesis is about. The only exception will be chapter 4, which may be applied to even more generalized problems. The first thing to do will be to present some basic notations, which includes funcitons, sets vectors and constants, that will be used throughout this article. Afterwards some variants of the continuous OCP will be introduced. Then the process of discretization will be discussed, which will conclude in a directly discretized version of the continuous OCP. The final section of this chapter will be a discussion about convergence, which of course includes discussion about appropriate distance measures and norms. Let us start with the basic notations.


Discrete Differential Inclusions
Differential Inclusions
Mayer formulation of an optimal control problem
Set Valued Euler's Method
tools
Preliminaries and Basic Tools
Optimal Control Problem (OCP) (Bolza-Form)
Mayer-Form of the OCP
transformation of the ODE
direct discretization
Discrete Mayer-Problem

outline of deriving the different forms of the OCP in this chapter

2.2 Basic Notations

The listing of all the functions below is intended to serve as a reference to come back to later on. The functions will soon make sense, when the actual OCP will be introduced in 2.3. $\tilde{\ }$ marks functions and constants that will be altered, when the form of the OCP will be changed to this articles needs. This change will take place in the section about the continuous OCP (2.3), when transforming the Bolza-Problem into a Mayer-Problem. Because there will be an Extension of the state variable involved, there might be a bit of confusion about the use of the indices $n-1$ and n, when just looking at the declarations in this section. In general I recommend not to pay too much attention to the exact indices occurring below for now. They will make sense, when this chapter is done. Let's start with the definitions and declarations.

$I = [t_0, T]$ shall be the time interval for the optimization.

The following functions represent the control function and the state function:

$$\begin{aligned}
u(.) &\in L_\infty(I)^m && \text{(m-dimensional control function)}\\
\tilde{x}(.) &\in AC(I)^{n-1} && \text{((n-1)-dimensional state function (Bolza-Formulation))}\\
z(.) &\in AC(I) && \text{(extension of the state, see 2.3.2)}\\
x(.) = \begin{pmatrix} \tilde{x}(.) \\ z(.) \end{pmatrix} &\in AC(I)^n && \text{(n-dimensional state function (Mayer-Formulation))}
\end{aligned}$$

Where AC(I) is the space of absolutely continous functions on I.

The optimal control and optimal state are denoted using $\hat{\ }$:

$$\begin{aligned}
\hat{u}(.) &\in L_\infty(I)^m && \text{(m-dimensional optimal control function)}\\
\hat{\tilde{x}}(.) &\in AC(I)^{n-1} && \text{((n-1)-dimensional optimal state function (Bolza-Formulation))}\\
\hat{z}(.) &\in AC(I) && \text{(extension of the state, see 2.3.2)}\\
\hat{x}(.) = \begin{pmatrix} \hat{\tilde{x}}(.) \\ \hat{z}(.) \end{pmatrix} &\in AC(I)^n && \text{(n-dimensional optimal state function (Mayer-Formulation))}
\end{aligned}$$

The following vectors will occur in general as a placeholder for the control u(.) and the state x(.) at a certain time:

$$\begin{aligned}
u &\in \mathbb{R}^m && \text{(placeholder for control at a certain time)}\\
\tilde{x} &\in \mathbb{R}^{n-1} && \text{(placeholder for state at a certain time)}\\
x = \begin{pmatrix} \tilde{x} \\ z \end{pmatrix} &\in \mathbb{R}^n && \text{(placeholder for state at a certain time (Mayer-Formulation), with } z \in \mathbb{R}\text{)}
\end{aligned}$$

Functions to describe a specific problem:

$\tilde{J} : AC(I)^{n-1} \times L_\infty(I)^m \to \mathbb{R}$	(objective function for the Bolza-Problem)
$J : \mathbb{R}^{n-1} \times \mathbb{R}^n \to \mathbb{R}$	(objective function for the Mayer-Problem)
$f : I \times \mathbb{R}^{n-1} \times \mathbb{R}^m \to \mathbb{R}$	(integrand in the objective function integral term)
$\varphi : \mathbb{R}^{n-1} \times \mathbb{R}^{n-1} \to \mathbb{R}$	(pointcost term in the objective function)
$\tilde{\psi} : I \times \mathbb{R}^{n-1} \times \mathbb{R}^m \to \mathbb{R}^{n-1}$	(right-hand side of the ODE for the Bolza-Problem)
$\psi : I \times \mathbb{R}^{n-1} \times \mathbb{R}^m \to \mathbb{R}^n$	(right-hand side of the ODE for the Mayer-Problem)
$g : I \times \mathbb{R}^{n-1} \times \mathbb{R}^m \to \mathbb{R}^{n_g}$	(mixed control-state constraints)
$s : I \times \mathbb{R}^{n-1} \to \mathbb{R}^{n_s}$	(pure state constraints)
$U : I \times \mathbb{R}^{n-1} \to \mathbb{R}^m$	(mixed control-state constraints, set form)

Sets used in conjunction with the ODE:

$\tilde{X}_0 \subset \mathbb{R}^{n-1}$	(set of all initial values for the state x for the Bolza-Problem)
$X_0 \subset \mathbb{R}^n$	(set of all initial values for the state x for the Mayer-Problem)
$\tilde{X}(T, t_0, X_0) \subset AC(I)^n$	(set of all feasible solutions to the unrestricted (no pure state constraints) differential inclusion on $I = [t_0, T]$ in the Bolza-Problem)
$\tilde{X}_\Theta(T, t_0, X_0) \subset AC(I)^{n-1}$	(set of all feasible solutions to the restricted differential inclusion on $I = [t_0, T]$ in the Bolza-Problem)
$X(T, t_0, X_0) \subset AC(I)^n$	(set of all feasible solutions to the unrestricted (no pure state constraints) differential inclusion on $I = [t_0, T]$ in the Mayer-Problem)
$X_\Theta(T, t_0, X_0) \subset AC(I)^n$	(set of all feasible solutions to the restricted differential inclusion on $I = [t_0, T]$ in the Mayer-Problem)
$S \subset \mathbb{R}^n$	(compact set with $x(t) \in S$ for all $x(.) \in X(T, t_0, X_0)$ and $t \in [t_0, T]$; Theorem 2.6.1 proves that such a set exists)
$U \subset \mathbb{R}^m$	(set such that $u(t) \in U$ for all admissible controls u(.) of the OCP)

Set-valued functions (see 2.3.3):

$F : I \times R^n \rightrightarrows \mathbb{R}^n$	(set-valued right-hand side of the ODE)
$\Theta : I \rightrightarrows \mathbb{R}^n$	(set-valued mapping representing pure state constraints)

Special notation for the discrete case (see 2.4):

$N \in \mathbb{N}$ (the number of steps).

$\mathbb{G}_N = (t_0, \ldots, t_N) \in \mathbb{R}^{N+1}$ (grid on the interval I, used for discretization)

$.^N$ ("discrete function" on the grid $\mathbb{G}_N$)

$\rho_N : V^k \to \mathbb{R}^{(N+1)k}$ (ordinary restriction operator to the grid, V: function space)

$\tilde{X}^N(t_j, t_0, X_0) \subset \mathbb{R}^{(j+1)(n-1)}$ (set of all feasible discrete solutions to the unrestricted (no pure state constraints) differential inclusion on $[t_0, t_j]$ in the Bolza-Problem)

$\tilde{X}^N_\Theta(t_j, t_0, X_0) \subset \mathbb{R}^{(j+1)(n-1)}$ (set of all feasible discrete solutions to the restricted differential inclusion on $[t_0, t_j]$ in the Bolza-Problem)

$X^N(T, t_0, X_0) \subset \mathbb{R}^{(N+1)n}$ (set of all feasible discrete solutions to the unrestricted (no pure state constraints) differential inclusion on $I = [t_0, T]$ in the discrete Mayer-Problem)

$X^N_\Theta(T, t_0, X_0) \subset \mathbb{R}^{(N+1)n}$ (set of all feasible discrete solutions to the restricted differential inclusion on $I = [t_0, T]$ in the discrete Mayer-Problem)

$X^N(T, t_0, X_0)(t_j) \subset \mathbb{R}^n$ (set of all reachable points of the discretized unrestricted (no pure state constraints) differential inclusion on $\{t_0, \ldots, t_j\}$ in the discrete Mayer-Problem)

$X^N_\Theta(T, t_0, X_0)(t_j) \subset \mathbb{R}^n$ (set of all feasible discrete solutions to the restricted differential inclusion on $\{t_0, \ldots, t_j\}$ in the discrete Mayer-Problem)

$\tilde{S} \subset \mathbb{R}^n$ (compact set with $x^N_j \in \tilde{S}$ for all $x^N \in X^N_\Theta(T, t_0, X_0)$, $j \in \{0, \ldots, N\}$ and $N \in \mathbb{N}$; Theorem 2.6.3 proves that such a set exists)

Discrete control and state (see 2.4.2):

$u^N = (u^N_0, \ldots, u^N_N) \in \mathbb{R}^{(N+1)m}$ (discrete control)

$\tilde{x}^N = (\tilde{x}^N_0, \ldots, \tilde{x}^N_N) \in \mathbb{R}^{(N+1)(n-1)}$ (discrete state)

$z^N = (z^N_0, \ldots, z^N_N) \in \mathbb{R}^{N+1}$ (extension of the discrete state)

$x^N = (\begin{pmatrix} \tilde{x}^N_0 \\ z^N_0 \end{pmatrix}, \ldots, \begin{pmatrix} \tilde{x}^N_N \\ z^N_N \end{pmatrix}) \in \mathbb{R}^{(N+1)n}$ (discrete state (Mayer-Formulation))

The optimal discrete control and optimal state are denoted using $\hat{\ }$:

$\hat{u}^N \in \mathbb{R}^{(N+1)m}$ (m-dimensional optimal discrete control function)

$\hat{\tilde{x}}^N \in \mathbb{R}^{(N+1)(n-1)}$ (optimal discrete state function (Bolza-Formulation))

$\hat{z}^N \in \mathbb{R}^{N+1}$ (extension of the discrete state for $\hat{\tilde{x}}^N$ and $\hat{\tilde{u}}^N$, see 2.3.2)

$\hat{x}^N = (\begin{pmatrix} \hat{\tilde{x}}_0^N \\ \hat{z}_0^N \end{pmatrix}, \dots, \begin{pmatrix} \hat{\tilde{x}}_N^N \\ \hat{z}_N^N \end{pmatrix}) \in \mathbb{R}^{(N+1)n}$ (n-dimensional optimal state function (Mayer-Formulation))

Constants:

$M :$ uniform upper bound of all feasible solutions $x(.) \in X(T, t_0, X_0)$ to the continuous problem

$\tilde{M} :$ uniform upper bound of all feasible solutions $x^N \in X^N(T, t_0, X_0)$ to the discrete problem; does not depend on N

$L :$ uniform Lipschitz constant of all feasible solutions $x(.) \in X(T, t_0, X_0)$ to the continuous problem

$\tilde{L} :$ uniform Lipschitz constant of all feasible solutions $x^N \in X(T, t_0, X_0)$ to the discrete problem; does not depend on N

$L_{\Delta x} :$ $L_{\Delta x} := 2L$

$L_J :$ Lipschitz constant of the objective function $J(.,.)$ with respect to both arguments and the supremum norm

$L_\psi :$ Lipschitz constant of the ODE function $\psi(.,.,.)$ with respect to all three arguments and the supremum norm

The following functions map solutions to a continuous problem on solutions of a discrete problem respectively the other way round. They appear in conjunction with the so called Approximation Property (see section 3.2.1)

$$\begin{aligned} \pi_N : & X_\Theta^N(T, t_0, X_0) \to X_\Theta(T, t_0, X_0) && \text{(feasible discrete solution mapping)} \\ \delta_N : & X_\Theta(T, t_0, X_0) \to X_\Theta^N(T, t_0, X_0) && \text{(feasible solution mapping)} \end{aligned}$$

2.3 Optimal Control Problem (continuous)

This subchapter starts with the basic form of an OCP, that will be considered throughout this article. Due to theoretical reasons this form will be altered afterwards to finally reach the shape, that will be used later on.
The class of optimal control problems considered in this article may be described the following way.

2.3.1 Basic Form of the Optimal Control Problem (Bolza-Problem)

Bolza-Problem

$$\text{Minimize}: \qquad \tilde{J}(\tilde{x}(.), u(.)) = \varphi(\tilde{x}(t_0), \tilde{x}(T)) + \int_{t_0}^{T} f(t, \tilde{x}(t), u(t))dt \tag{1}$$

with respect to :

$$\dot{\tilde{x}}(t) = \tilde{\psi}(t, \tilde{x}(t), u(t)) \qquad a.e. \tag{2}$$

$$\tilde{x}(t_0) \in \tilde{X}_0 \tag{3}$$

$$g_i(t, \tilde{x}(t), u(t)) \begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} \qquad a.e. \tag{4}$$

$$s_i(t, \tilde{x}(t)) \begin{cases} \leq 0 & (i = 1, \ldots, n_s') \\ = 0 & (i = n_s' + 1, \ldots, n_s) \end{cases} \qquad a.e. \tag{5}$$

$$u(t) \in U(t, \tilde{x}(t)) \qquad a.e. \tag{6}$$

with $\tilde{x}(.) \in AC(I)^{n-1}$ and $u(.) \in L_\infty(I)^m$

Remarks:

- a.e. means almost everywhere and occurs because of the specific function spaces for the state and the control
- (1) is the objective function. It consists of two terms. The first one only depends on the state on the boundary of the time interval I and is sometimes called the pointcost term. The second one is an integral. This special form is related to models from physics or engineering.
- (2) is the system equation, which is an ODE parametrized by the control $u(.)$.
- (3) are the initial value conditions, where $\tilde{X}_0$ is the set of all allowed initial values for the state.

- (4) are mixed control-state constraints.
- (5) are pure state constraints. They are treated separately from 4, to allow more complicated constraints like differential-algebraic equations. They are also treated separately for theoretical purposes that appear in conjunction with treating differential inclusions (see Definition 3).
- (6) is an alternative description for mixed control-state constraints or pure control constraints. This description uses a set constraint.

Definition 1:

*The **set of all feasible solutions** to this problem will be called* $\boldsymbol{\tilde{X}_\Theta(T, t_0, X_0)}$. *So* $\tilde{X}_\Theta(T, t_0, X_0)$ *is the set of all functions* $\tilde{x}(.) \in AC(I)^{n-1}$ *with:*

$$\begin{aligned}
\dot{\tilde{x}}(t) &= \tilde{\psi}(t, \tilde{x}(t), u(t)) && a.e.\\
\tilde{x}(t_0) &\in \tilde{X}_0 \\
g_i(t, \tilde{x}(t), u(t)) &\begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} && a.e.\\
s_i(t, \tilde{x}(t)) &\begin{cases} \leq 0 & (i = 1, \ldots, n_s') \\ = 0 & (i = n_s' + 1, \ldots, n_s) \end{cases} && a.e.\\
u(t) &\in U(t, \tilde{x}(t)) && a.e.\\
u(.) &\in L_\infty(I)^m
\end{aligned}$$

*The **set of all feasible solutions** to this problem **without pure state constraints** will be called* $\boldsymbol{\tilde{X}(T, t_0, X_0)}$. *So* $\tilde{X}(T, t_0, X_0)$ *is the set of all functions* $\tilde{x}(.) \in AC(I)^{n-1}$ *with:*

$$\begin{aligned}
\dot{\tilde{x}}(t) &= \tilde{\psi}(t, \tilde{x}(t), u(t)) && a.e.\\
\tilde{x}(t_0) &\in \tilde{X}_0 \\
g_i(t, \tilde{x}(t), u(t)) &\begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} && a.e.\\
u(t) &\in U(t, \tilde{x}(t)) && a.e.\\
u(.) &\in L_\infty(I)^m
\end{aligned}$$

This should give the reader an outline of the structure of an OCP this article deals with. As this is in no way an introduction to optimization problems, the reader is referred to a wide range of publications on that topic for further reading (for example [1], chapter 3).

2.3.2 Mayer-Form of the Optimal Control Problem (Mayer-Problem)

The main goal with this kind of formulation is to get an objective function that only depends on the state evaluated at the first and the last timepoint, i.e. t_0 and T. To achieve this, the state variable itself has to be extended by one component. The special form of the objective function will be advantageous, more precisely very important, for the theoretical investigation in chapter 3. The reason for this is that the objective funciton will be the same in the discrete case as in the continuous case.
The transformation from the Bolza-Form to the Mayer-Form is no difficult task. Nevertheless the author recommends paying attention to the exact definitions of the functions used. The basic notations can be found in 2.2. Let's start with extending the state variable, which will directly lead to the new form of the OCP afterwards.

Transformation:
The basic concept is to subordinate the integral term of the objective funciton $\tilde{J}$ to the state function by expanding the ODE. This comes down to adding one component to the state function vector. This additional state vector component shall be called $z(.)$ and is defined the following way:

$$z(t) := \int\limits_{t_0}^{t} f(\tau, \tilde{x}(\tau), u(\tau))d\tau \qquad (t \in [t_0, T] = I)$$

This is equivalent to:

$$\dot{z}(t) := f(t, \tilde{x}(t), u(t)) \qquad \text{(for almost all } t \in I) \qquad \wedge \qquad z(t_0) := 0$$

So we get one additional state vector component and two additional equations. The first equation will extend the ODE, the second one is an initial value condition. This leads to redefining the state $\tilde{x}(.)$ to $x(.)$, the ODE right-hand side $\tilde{\psi}(., \tilde{x}(.), u(.))$ to $\psi(., \tilde{x}(.), u(.))$, the initial value set $\tilde{X}_0$ to X_0 and finally the objective function $\tilde{J}(\tilde{x}(.), u(.))$ to $J(.,.)$.

$$x(t) := \begin{pmatrix} \tilde{x}(t) \\ z(t) \end{pmatrix} \in \mathbb{R}^n \tag{1}$$

$$\psi(t, \tilde{x}(t), u(t)) := \begin{pmatrix} \tilde{\psi}(t, \tilde{x}(t), u(t)) \\ f(t, \tilde{x}(t), u(t)) \end{pmatrix} \in \mathbb{R}^n \tag{2}$$

$$X_0 := \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} : v \in \tilde{X}_0 \right\} \subset \mathbb{R}^n \tag{3}$$

$$J(\tilde{x}(t_0), x(T)) := \varphi(\tilde{x}(t_0), \tilde{x}(T)) + z(T) \tag{4}$$

The other functions g, s and U stay untouched. These definitions correspond to the listing in 2.2.

Important remark for easier notations:
For the sake of simplicity $\tilde{x}(t)$ (the first $(n-1)$ components of the vector $x(t)$) will no longer be passed explicitly as an argument to one of the functions J, ψ, g, s, r and U.

Instead the whole vector $x(t)$ will be used and implicitly only the first $n-1$ components (which correspond to $\tilde{x}(t)$) will be taken into account. For example the function $s(.,.)$ takes only an $(n-1)$-dimensional vector as its second argument. But instead of passing $\tilde{x}(t)$, the whole vector $x(t)$ will be passed and $s(t, x(t))$ shall be implicitly understood as $s(t, \tilde{x}(t))$. To give the reader a bit more of a view behind the scenes, the functions ψ and J defined above only take the full vector $x(t)$ where needed, otherwise only $\tilde{x}(t)$ has been passed. For example the function $J(.,.)$ (see (4)) just needs all n components in its second argument, not in its first one. So the function J has been defined to map from $\mathbb{R}^{n-1} \times \mathbb{R}^n$ to $\mathbb{R}$. The alternative would have been to define it as a function from $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}$, but this would have suggested, that the the n-th component of the first argument would have some influence, which it does not. But nevertheless the notation will be $J(x(t_0), x(T))$ from now on, which shall be understood as $J(\tilde{x}(t_0), x(T))$.

Taking the transformation and the note about easier notations into account, we arrive at the Mayer-Formulation of the OCP:

Mayer-Problem

$$\text{Minimize}: \qquad J(x(t_0), x(T)) = \varphi(\tilde{x}(t_0), \tilde{x}(T)) + z(T)$$

with respect to :

$$\dot{x}(t) = \psi(t, x(t), u(t)) \qquad a.e.$$

$$x(t_0) \in X_0$$

$$g_i(t, x(t), u(t)) \begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} \qquad a.e.$$

$$s_i(t, x(t)) \begin{cases} \leq 0 & (i = 1, \ldots, n_s') \\ = 0 & (i = n_s' + 1, \ldots, n_s) \end{cases} \qquad a.e.$$

$$u(t) \in U(t, x(t)) \qquad a.e.$$

with $x(.) \in AC(I)^n$ and $u(.) \in L_\infty(I)^m$

Remarks:

- Take in account the remark about easier notations above.
- The objective function now just depends on the state $x(.)$ evaluated at t_0 and T. To achieve that the state vector itself had to be changed (see (1)).
- The integral term of the objective function is now represented by the n-th component of the ODE (see (2)).
- The Mayer-Problem stated above is just another formulation of the Bolza-Problem (see 2.3.1). So the above form represents the same OCP.

Definition 2:

*The **set of all feasible solutions** to this problem will be called* $\boldsymbol{X_\Theta(T,t_0,X_0)}$. *So* $X_\Theta(T,t_0,X_0)$ *is the set of all functions* $x(.) \in AC(I)^n$ *with:*

$$\begin{aligned}
\dot{x}(t) &= \psi(t,x(t),u(t)) && a.e.\\
x(t_0) &\in X_0 \\
g_i(t,x(t),u(t)) &\begin{cases} \leq 0 & (i=1,\dots,n_g') \\ = 0 & (i=n_g'+1,\dots,n_g) \end{cases} && a.e.\\
s_i(t,x(t)) &\begin{cases} \leq 0 & (i=1,\dots,n_s') \\ = 0 & (i=n_s'+1,\dots,n_s) \end{cases} && a.e.\\
u(t) &\in U(t,x(t)) && a.e.\\
u(.) &\in L_\infty(I)^m
\end{aligned}$$

*The **set of all feasible solutions** to this problem **without pure state constraints** will be called* $\boldsymbol{X(T,t_0,X_0)}$. *So* $X(T,t_0,X_0)$ *is the set of all functions* $x(.) \in AC(I)^n$ *with:*

$$\begin{aligned}
\dot{x}(t) &= \psi(t,x(t),u(t)) && a.e.\\
x(t_0) &\in X_0 \\
g_i(t,x(t),u(t)) &\begin{cases} \leq 0 & (i=1,\dots,n_g') \\ = 0 & (i=n_g'+1,\dots,n_g) \end{cases} && a.e.\\
u(t) &\in U(t,x(t)) && a.e.\\
u(.) &\in L_\infty(I)^m
\end{aligned}$$

Remarks: $X_\Theta(T,t_0,X_0)$ *is the same as* $\tilde{X}_\Theta(T,t_0,X_0)$ *with one exception. Any element of* $X_\Theta(T,t_0,X_0)$ *has the additional component* $z(.)$ *defined in the Transformation section above.*

We have now almost arrived at the final form of the OCP, that will be used later on this article. The only thing left is packing the feasible controls combined with the parametrized ODE in a set-valued formulation.

2.3.3 Set-Valued Form of the Mayer-Problem

There are only two things left to do.
The first one is to transform the parametrized ODE into a so called **Differential Inclusion**.

Definition 3:

*A **Differential Inclusion (DI)** can be described as a differential equation with set-valued right-hand side and a set of feasible initial values. With the notations of this section (which will be introduced later on) the differential inclusion looks like:*

$$
\textbf{\textit{(DI)}} \quad \left| \begin{array}{rll} \dot{x}(t) \in F(t, x(t)) & & a.e. \\ x(t_0) \in X_0 & & \end{array} \right.
$$

*In our case this Differential Inclusion will include pure state constrains represented by a set-valued mapping $\Theta(.)$ defined in this section. The **Constrained Differential Inclusion (DIC)** then looks like this:*

$$
\textbf{\textit{(DIC)}} \quad \left| \begin{array}{rll} \dot{x}(t) \in F(t, x(t)) & & a.e. \\ x(t_0) \in X_0 & & \\ x(t) \in \Theta(t) & & a.e. \end{array} \right.
$$

So the first thing that needs to be done is to create a set-valued mapping, whose image consists of all feasible (without taking pure state constraints into account) right-hand sides of the ODE. This mapping will be called F.
The second one is to replace the pure state constraints $s(.,.)$ by a set-valued map Θ, which will give all feasible states at a given time t. Let's start with the right-hand side of the ODE.

Creating the set-valued right-hand side:
The set-valued right-hand side F will depend on the time t and the state at that time $x(t)$. In fact F consists of all feasible right-hand sides $\psi(t, x(t), u(t))$. This leads to F subsuming all constraints concerning the control u. In general, F is defined in the following way:

$$
F: \qquad I \times R^n \Rightarrow \mathbb{R}^n
$$

$$
\begin{aligned}
F(t, x) := \{ \ & \psi(t, x, u) : \\
& \quad g_i(t, x, u) \ \begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} \\
& \quad u \in U(t, x) \\
\} &
\end{aligned}
$$

Note: *The "$\Rightarrow$" signalises that the mapping is set-valued, i.e. the image is a set.*

Creating the set-valued pure state constraint mapping:
$\Theta(t)$ should be the set of all feasible states at the time t according to the pure state constraints $s(.,.)$. So defining the set Θ is straighforward:

$$\Theta : \qquad I \rightrightarrows \mathbb{R}^n$$

$$\Theta(t) := \{ \; x \in \mathbb{R}^n : \quad s_i(t,x) \begin{cases} \leq 0 & (i = 1, \ldots, n_s') \\ = 0 & (i = n_s' + 1, \ldots, n_s) \end{cases} \; \}$$

Note: *The "$\Rightarrow$" signalises that the mapping is set-valued, i.e. the image is a set. As one can see, the last component of the vector x, i.e. z, is not restricted in any way. This is because there are no restrictions to the integral term of the objective function from the Bolza-Problem.*

Those two modifications applied to the Mayer-Problem given in 2.3.2 delivers the final form.

Set-Valued Mayer-Problem

Minimize : $\qquad J(x(t_0), x(T))$

with respect to :

$$\dot{x}(t) \in F(t, x(t)) \qquad a.e.$$

$$x(t_0) \in X_0$$

$$x(t) \in \Theta(t) \qquad a.e.$$

with $x(.) \in AC(I)^n$ and $u(.) \in L_\infty(I)^m$

Remarks:

- The control $u(.)$ does not appear directly in the optimization problem above. Nevertheless it is an optimization variable, all the terms containing $u(.)$ are just included in the right-hand side of the ODE F. See definition of F above.
- Though this problem looks way easier than the Mayer-Problem(2.3.2) it is absolutely identical. The crucial point really is the defintion of the right-hand side F of the ODE above. So the set of all feasible solutions to the problem above is $X_\Theta(T, t_0, X_0)$, defined in Definition 2. With the formulation of the set-valued

Mayer-Problem we now get another way of defining the sets $X_\Theta(T, t_0, X_0)$ and $X(T, t_0, X_0)$:

$X_\Theta(T, t_0, X_0)$ is the set of all functions $x \in AC(I)^n$ with:

$$\begin{aligned} \dot{x}(t) &\in F(t, x(t)) && a.e. \\ x(t_0) &\in X_0 \\ x(t) &\in \Theta(t) && a.e. \end{aligned}$$

The **set of all feasible solutions** to this problem **without pure state constraints** will be called $\boldsymbol{X(T, t_0, X_0)}$. So $X(T, t_0, X_0)$ is the set of all functions $x \in AC(I)^n$ with:

$$\begin{aligned} \dot{x}(t) &\in F(t, x(t)) && a.e. \\ x(t_0) &\in X_0 \end{aligned}$$

The form of the OCP above is the final form, that will be used later on in this article. This subchapter started with introducing the Bolza-Problem. Then the Bolza-Problem has been transformed into a Mayer-Problem, which then has been slightly modified to represent the parametrized ODE as a differential inclusion. We have now reached the point where we can turn to discretizing the OCP.

2.4 Discretization

This article is about using direct discretization methods for the numerical approximation. This means that the optimal control problem itself is taken and every occurring function gets discretized without any further diversion. The same thing will be done with the ODE. The really crucial point will be the method used to do this. In this paper **Euler's Method** will be chosen for a number of reasons. One of them being that Euler's Method allows pretty weak assumptions for several results or even makes them possible in the first place. The goal of this section is to provide discrete versions of the Mayer-Problem (2.3.2) and the set-valued version of the Mayer-Problem (2.3.3). To get the latter one, the concept of the Set-Valued Euler's Method will be introduced.
The first thing to do will be to choose an appropriate grid for the discretization process. So this subchapter starts with a description of that grid and introduces some basic notations specific to the discretized problem afterwards. This is all that's needed to discretize the continuous Mayer-Problem (2.3.2). After doing so, the Set-Valued Euler's Method for solving the parametrized ODE will be introduced. Having those tools ready, the Set-Valued Mayer-Problem (2.3.3) will be discretized.

2.4.1 The Grid

Discretization takes place with respect to the time variable t. So we have to split up the interval I into appropriate fragments. This will be done by specifying an equidistant grid $\mathbb{G}_N$. As the name suggests, this grid will depend on the number of steps N chosen for the discretization, but it will remain equidistant for all the investigations done in this article later on. So the grid presented in this subsection will be the grid used throughout the whole thesis. The following definitions are all straightforward.

Definitions:
Let N be the number of steps, then:

$$\begin{aligned} h_N &:= \frac{T - t_0}{N} && \text{(steplength)} \\ t_j &:= t_0 + j \cdot h_N \quad (j = 0 \ldots, N) && \text{(discrete time points)} \\ \mathbb{G}_N &:= (t_0, t_1, \ldots, t_N) && \text{(the grid)} \end{aligned}$$

From the definition of t_j it follows directly that $t_N = T$. To stress the fact, that T is a grid point, later on t_N will be used in some places instead of T.

To restrict a function to the grid, the so called ordinary restriction operator ρ_N is introduced:

$$\rho_N : V^k \to \mathbb{R}^{(N+1)k}$$

Where V is an appropriate function space operating on I (for example $V = L_\infty(I)$ for the control $u(.)$).
Let $f(.) \in V^k$, then:

$$\rho_N(f(.)) := (f(t_0), f(t_1), \ldots, f(t_N)) \in \mathbb{R}^{(N+1)k}$$

Applying ρ_N to $\tilde{V} \subset V$ shall also be permitted:

$$\rho_N(\tilde{V}) := \left\{ \rho_N(f(.)) \;\middle|\; f(.) \in \tilde{V} \right\}$$

2.4.2 Discrete Mayer-Problem

Notations:
A superscript N $(.^N)$ denotes a “discrete function”, i.e. a vector in $\mathbb{R}^{(N+1)k}$ with k representing the space dimension. For example a discrete state would be called x^N. Written according to the grid, this would give $x^N = (x_0^N, \ldots, x_N^N)$ with $x_j \in \mathbb{R}^n$ $(j = 0, \ldots, N)$. So x^N could be seen as a function evaluated on the grid. Once again: Space dimension of the state is n due to the extension process described in the Transformation section of 2.3.2. Furthermore according to 2.3.2(2) we define $\tilde{x}^N = (\tilde{x}_0^N, \ldots, \tilde{x}_N^N)$ $(\tilde{x}_j^N \in \mathbb{R}^{n-1}$ for $j = 0, \ldots, N)$ and $z^N = (z_0^N, \ldots, z_N^N)$ $(z_j^N \in \mathbb{R}$ for $j = 0, \ldots, N)$ such that $x_j^N = \binom{\tilde{x}_j^N}{z_j^N}$ for $j = 0, \ldots, N$.

By now we have everything we need to apply the method of direct discretization to the Mayer-Problem (see 2.3.2).
The optimization variables will be represented by $x^N = (x_0^N, \ldots, x_N^N) \in \mathbb{R}^{(N+1)n}$ and $u^N = (u_0^N, \ldots, u_N^N)) \in \mathbb{R}^{(N+1)m}$. As already mentioned one major advantage of the objective function of the Mayer-Problem is the fact that it has to be evaluated just at discrete points even in the continuous case. So the objective function does not change in the process of discretization. For the discretization of the ODE **Euler’s Method** will be used. The continuous constraints will be restricted to the grid $\mathbb{G}_N$ which will result in lots of constraints depending on the number of steps N. So the directly discretized version of the Mayer-Problem looks like this:

Discrete Mayer-Problem

$$\text{Minimize}: \qquad J(x_0^N, x_N^N) = \varphi(\tilde{x}_0^N, \tilde{x}_N^N) + z_N^N$$

with respect to :

$$x_{j+1}^N = x_j^N + h_N \cdot \psi(t_j, x_j^N, u_j^N) \qquad (j = 0, \ldots, N-1)$$

$$x_0^N \in X_0$$

$$g_i(t_j, x_j^N, u_j^N) \begin{cases} \leq 0 & (i = 1, \ldots, n_g') \\ = 0 & (i = n_g' + 1, \ldots, n_g) \end{cases} \qquad (j = 0, \ldots, N)$$

$$s_i(t_j, x_j^N) \begin{cases} \leq 0 & (i = 1, \ldots, n_s') \\ = 0 & (i = n_s' + 1, \ldots, n_s) \end{cases} \qquad (j = 0, \ldots, N)$$

$$u_j^N \in U(t_j, x_j^N) \qquad (j = 0, \ldots, N)$$

with $x^N \in \mathbb{R}^{(N+1)n}$ and $u^N \in \mathbb{R}^{(N+1)m}$

Remarks:

- The objective function is the same as in the continuous case, but notation for the discrete functions changes the exact expression (see “Notations” above).

- Taking a closer look at the restrictions above yields: There are $(N+1) \cdot n_g$ mixed control-state constraints, $(N+1) \cdot n_s$ pure state constraints, $(N+1)$ set-valued mixed control-state constraints and one set-valued constraint for the initial state. This sums up to $(N+1) \cdot (n_g + n_s)$ single scalar and $(N+2)$ set-valued constraints. The latter ones have to be treated separately when counting the number of constraints, because they each may consist of a lot of constraints.

Definition 4:

The ***set of all feasible solutions*** *to this problem will be called* $\boldsymbol{X^N_\Theta(T, t_0, X_0)}$. *So* $X^N_\Theta(T, t_0, X_0)$ *is the set of all discrete functions* $x^N \in \mathbb{R}^{(N+1)n}$ *with:*

$$
\begin{aligned}
& x^N_{j+1} = x^N_j + h_N \cdot \psi(t_j, x^N_j, u^N_j) && (j = 0, \dots, N-1) \\
& x^N_0 \in X_0 && \\
& g_i(t_j, x^N_j, u^N_j) \begin{cases} \leq 0 & (i = 1, \dots, n'_g) \\ = 0 & (i = n'_g + 1, \dots, n_g) \end{cases} && (j = 0, \dots, N) \\
& s_i(t_j, x^N_j) \begin{cases} \leq 0 & (i = 1, \dots, n'_s) \\ = 0 & (i = n'_s + 1, \dots, n_s) \end{cases} && (j = 0, \dots, N) \\
& u^N_j \in U(t_j, x^N_j) && (j = 0, \dots, N) \\
& u^N \in \mathbb{R}^{(N+1)m} &&
\end{aligned}
$$

The ***set of all feasible solutions*** *to this problem* ***without pure state constraints*** *will be called* $\boldsymbol{X^N(T, t_0, X_0)}$. *So* $X^N(T, t_0, X_0)$ *is the set of all discrete functions* $x^N \in \mathbb{R}^{(N+1)n}$ *with:*

$$
\begin{aligned}
& x^N_{j+1} = x^N_j + h_N \cdot \psi(t_j, x^N_j, u^N_j) && (j = 0, \dots, N-1) \\
& x^N_0 \in X_0 && \\
& g_i(t_j, x^N_j, u^N_j) \begin{cases} \leq 0 & (i = 1, \dots, n'_g) \\ = 0 & (i = n'_g + 1, \dots, n_g) \end{cases} && (j = 0, \dots, N) \\
& u^N_j \in U(t_j, x^N_j) && (j = 0, \dots, N) \\
& u^N \in \mathbb{R}^{(N+1)m} &&
\end{aligned}
$$

The only thing that's missing to start with the discretization process of the Set-Valued Mayer-Problem is the method to discretize the set-valued ODE, in other words the differential inclusion. To solve the ODE we will use Euler's Method again, which can be easily adjusted to differential inclusions. The method is then called "Set-Valued Euler's Method".

2.4.3 Set-Valued Euler's Method

To work with sets we need to introduce addition and scalar multiplication of sets. The way this article understands addition and multiplication by a scalar is the following:

Let $\lambda \in \mathbb{R}$ and let $A, B \subset \mathbb{R}^k$ nonempty for arbitrary $k \in \mathbb{N}$.
Then addition is defined by the so called "Minkowski Sum", which is a straightforward definition:

$$A + B := \{a + b \in \mathbb{R}^k : a \in A, b \in B\} \qquad \text{(Addition)}$$

Multiplication is defined as one might guess:

$$\lambda A = \{\lambda a : a \in A\} \qquad \text{(Scalar Multiplication)}$$

With these definitions one can easily transfer the concept of Euler's method to it's set-valued form:

$$x_{j+1}^N \in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad (j = 0, \dots, N-1)$$

Definition 5:

To stress the set-valued character of the above expression we introduce $X^N(T, t_0, X_0)(t_j)$ $(j = 0, \dots, N)$ as the set of all reachable points delivered by Euler's method with starting set X_0 on $\{t_0, \dots, t_j\}$. Then $X^N(T, t_0, X_0)(t_j)$ fulfills the following recursion:

$$\begin{aligned} X^N(T, t_0, X_0)(t_0) &= X_0 \\ X^N(T, t_0, X_0)(t_{j+1}) &= \bigcup_{x \in X^N(T, t_0, X_0)(t_j)} \big(x + h_N \cdot F(t_j, x)\big) \qquad (j = 0, \dots, N-1) \end{aligned}$$

Including pure state constraints represented by the restriction set Θ in the above recursion delivers:

$$X_\Theta^N(T, t_0, X_0)(t_{j+1}) = \bigcup_{x \in X_\Theta^N(T, t_0, X_0)(t_j)} \big(x + h_N \cdot F(t_j, x)\big) \cap \Theta(t_{j+1}) \quad (j = 0, \dots, N-1)$$

Where $X_\Theta^N(T, t_0, X_0)(t_j)$ is the same as $X^N(T, t_0, X_0)(t_j)$ with the addition that every element in $X_\Theta^N(T, t_0, X_0)(t_j)$ obeys the pure state constraints represented by the set $\Theta(t_j)$, i.e. $X_\Theta^N(T, t_0, X_0)(t_j) \subset \Theta(t_j)$ (j=0,…,N)

This definition won't be used later on, but the sets mentioned above shall give the reader a better idea on how the set valued theory works.

2.4.4 Discrete Set-Valued Mayer-Problem

Obtaining the directly discretized of the Set-Valued Mayer-Problem (2.3.3) involves the same process as for the discrete Mayer-Problem 2.4.2. The only difference is the use of the set-valued Euler's method 2.4.3. So we finally obtain:

Discrete Set-Valued Mayer-Problem

Minimize : $J(x_0^N, x_N^N)$

with respect to :

$$x_{j+1}^N \in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad (j = 0, \ldots, N-1)$$

$$x_0^N \in X_0$$

$$x_j^N \in \Theta(t_j) \qquad (j = 0, \ldots, N)$$

with $x^N \in \mathbb{R}^{(N+1)n}$ and $u^N \in \mathbb{R}^{(N+1)m}$

Remarks

- The discrete control u^N does not appear directly in the optimization problem above. Nevertheless it is an optimization variable. All the terms containing parts of u^N are just included in the right-hand side of the ODE F. See definition of F in the section about the set-valued Mayer-Problem 2.3.3.

- Though this problem looks way easier than the discrete Mayer-Problem (2.4.2) it is absolutely identical. The crucial point really is the set-valued Euler's Method. So the set of all feasible solutions to the problem above is $X_\Theta(T, t_0, X_0)$, defined in Definition 2. With the formulation of the discrete set-valued Mayer-Problem we now get another way of defining the sets $X_\Theta^N(T, t_0, X_0)$ and $X^N(T, t_0, X_0)$:

  $X_\Theta^N(T, t_0, X_0)$ is the set of all discrete functions $x^N \in \mathbb{R}^{(N+1)n}$ with:

  $$\begin{aligned} x_{j+1}^N &\in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad && (j = 0, \ldots, N-1) \\ x_0^N &\in X_0 && \\ x_j^N &\in \Theta(t_j) && (j = 0, \ldots, N) \end{aligned}$$

  And $X^N(T, t_0, X_0)$ is the set of all discrete functions $x^N \in \mathbb{R}^{(N+1)n}$ with:

  $$\begin{aligned} x_{j+1}^N &\in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad && (j = 0, \ldots, N-1) \\ x_0^N &\in X_0 && \end{aligned}$$

Let's conclude with the definition of the so called Discrete Differential Inclusion, which is the discrete analog of a Differential Inclusion and appears in the set-valued Mayer-Problem.

Definition 6:

A discretized Differential Inclusion (see Definition 3) is called ***Discrete Differential Inclusion (DDI)*** *and looks like this for $x^N \in \mathbb{R}^{(N+1)n}$:*

$$\begin{aligned} x_{j+1}^N &\in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad && (j = 0, \ldots, N-1) \\ x_0^N &\in X_0 \end{aligned}$$

A ***Constrained Discrete Differential Inclusion (DDIC)*** *includes pure state constraints, in our case represented by the set-valued mapping $\Theta(.)$:*

$$\begin{aligned} x_{j+1}^N &\in x_j^N + h_N \cdot F(t_j, x_j^N) \qquad && (j = 0, \ldots, N-1) \\ x_0^N &\in X_0 \\ x_j^N &\in \Theta(t_j) && (j = 0, \ldots, N) \end{aligned}$$

2.5 Norm and Convergence Discussion

Concerning convergence analysis it is essential to consider appropriate norms. In infinite dimensional spaces norms are not equivalent. So convergence with respect to a certain norm might not lead to convergence in another one or vice versa. This will also be the case for convergence analysis concerning discrete optimal solutions and their continous counterparts. To understand that, one has to know in what sense the word convergence has to be understood. This is what this section is all about.

First off norms used in this article will be defined. Then some basic properties of these norms will be presented. Afterwards some light will be shed on the phrase "convergence of optimal solutions". The last part will be about how to measure distance between sets, which will lead to the **Hausdorff-distance**. The chapter about the Approximation Property will make heavy use of that definition, because theory about differential inclusions plays a great role there. So let's start with the definitions.

2.5.1 Norm Definitions

case $\mathbb{R}^{\mathbf{n}}$:

Let $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$, then:

$$\begin{aligned} \|v\|_\infty &= \max_{i=1,\ldots,n} |v_i| \\ \|v\|_p &= \left(\sum_{i=1}^n v_i^p \right)^{1/p} \qquad && (1 \leq p < \infty) \end{aligned}$$

continuous case:
According to the occuring function spaces we have to consider the following L_p-norms:

$$\|f(.)\|_\infty = \operatorname{ess\,sup} |f(.)| = \inf\{K \in \mathbb{R} : \|f(t)\|_\infty \le K \text{ for almost all } t \in I\} \quad (\text{for } f(.) \in L_\infty(I)^n)$$

$$\|f(.)\|_\infty = \operatorname{ess\,sup} |f(.)| = \sup_{t \in I} \|f(t)\|_\infty \qquad (\text{for } f(.) \in C(I)^n)$$

$$\|f(.)\|_p = \left(\int_{t_0}^{T} \|f(t)\|_p^p \, dt \right)^{1/p} \qquad (\text{for } f(.) \in L_p(I)^n)$$

Remarks:

- $\|.\|_\infty$ is the L_∞-norm and $\|.\|_p$ the L_p-norm.
- An alternate definition of the essential supremum ess sup is
  $\operatorname{ess\,sup} |f(.)| = \inf\limits_{N \text{ is null set}} \left\{ \sup\limits_{t \in I \setminus N} \|f(t)\|_\infty \right\}$
  For $f(.) \in C(I)^n$ the distinction between null sets falls away and one arrives directly at the expression given above for a continuous function $f(.)$.

discrete case:
For the discrete case we use the discrete L_p-norms, which are directly derived from the L_p-norms presented above. Let $f^N = (f_0, \dots, f_N) \in \mathbb{R}^{(N+1)n}$, then:

$$\|f^N\|_\infty = \sup_{j=0,\dots,N} \|f_j\|_\infty$$

$$\|f^N\|_p = \left(\sum_{j=0}^{N-1} h_N \cdot \|f_j\|_p^p \right)^{1/p} \qquad (1 \le p < \infty)$$

Note: *The discrete L_p-norm can be derived from the continuous case by taking the Riemann Sum of the Integral.*

set case:
To deal with Differential Inclusions (see Definition 3) we need to define what we understand by the norm of a set.
Let V be a Banach space with norm $\|.\|_V$ and $M \subset V$ then

$$\|M\|_V := \sup_{v \in M} \|v\|_V$$

Remarks:

- It is easy to prove that this is a norm on the vector space of subsets of V. Addition and scalar multiplication for these sets is defined as in the section about the set-valued Euler's Method (see 2.4.3).
- We will need this norm definition among others for getting an upper bound of any valid initial value. Let's say the set of initial values $X_0 \subset \mathbb{R}^n$ is bounded by a constant C, i.e. for any vector $v \in X_0$ it holds $\|v\|_\infty \leq C$. Then $\|X_0\|_\infty = \sup\limits_{v \in X_0} \|v\|_\infty \leq C$.

2.5.2 Norm Properties

This section just highlights a few properties concerning the norms defined in 2.5.1, which are useful in terms of convergence analysis.

Let $v \in \mathbb{R}^n$ and $f(.) \in L_\infty(I)^n$, then:

1. In finite dimensional spaces norms are equivalent. Only two estimations used later on will be presented here:

$$\|v\|_2 = \left(\sum_{i=1}^{n} v_i^2\right)^{1/2} \leq \left(\sum_{i=1}^{n} \|v\|_\infty^2\right)^{1/2} = \|v\|_\infty \left(\sum_{i=1}^{n} 1\right)^{1/2} = \sqrt{n}\,\|v\|_\infty$$

and (without loss of generality let $\|v\|_\infty = |v_n|$)

$$\|v\|_2 = \left(\sum_{i=1}^{n} v_i^2\right)^{1/2} = \left(\|v\|_\infty^2 + \sum_{i=1}^{n-1} v_i^2\right)^{1/2} \underset{\text{of } \sqrt{\cdot}}{\overset{\text{monotonicity}}{\geq}} \sqrt{\|v\|_\infty^2} = \|v\|_\infty$$

2. $\exists C > 0$ with $\|f(.)\|_p \leq C \cdot \|f(.)\|_\infty$

This means, that the L_∞-norm is "stronger" than the L_p-norm.
Proof :

$$\|f(.)\|_p = \left(\int_{t_0}^{T} \|f(t)\|_p^p \, dt\right)^{1/p} \leq \left(\int_{t_0}^{T} n \cdot \|f(t)\|_\infty^p \, dt\right)^{1/p} \leq \left(n \cdot \int_{t_0}^{T} \|f(.)\|_\infty^p \, dt\right)^{1/p}$$

$$= \sqrt[p]{n} \cdot \left(\|f(.)\|_\infty^p \cdot \int_{t_0}^{T} 1 \, dt\right)^{1/p} = \underbrace{\sqrt[p]{n \cdot (T - t_0)}}_{C:=} \cdot \|f(.)\|_\infty \qquad \blacksquare$$

This inequality also shows that $L_\infty(I)^n \subset L_p(I)^n$.

3. The other way round is not true, i.e. $\nexists C > 0$ with $\|f(.)\|_\infty \le C \cdot \|f(.)\|_p$ for all $f(.) \in L_\infty(I)^n$.
Proof :
This can be easily shown by giving a counterexample.
Let $f_k(t) := \max\Big((t-t_0)\cdot\Big(-\frac{k}{T-t_0}\Big)+1\ ,\ 0\Big)$. Then $\|f_k\|_\infty = 1 \quad \forall k \in \mathbb{N}$. But $\|f_k\|_p \to 0 \ (k \to \infty)$. So, for every $C > 0$ there $\exists \tilde{k} \in \mathbb{N}$ such that $\|f_{\tilde{k}}\|_\infty > C \cdot \|f_{\tilde{k}}\|_p$. ■

The discrete norms behave like their continuous counterparts. This is no surprise, because they are directly derived from them.

Let $f^N = (f_0, \ldots, f_N) \in \mathbb{R}^{(N+1)n}$, then:

4. $\exists C > 0$ **independent of N** with $\|f^N\|_p \le C \cdot \|f^N\|_\infty$

This means, that the discrete L_∞-norm is “stronger” than the discrete L_p-norm.
Proof :

$$\|f^N\|_p = \left(\sum_{j=0}^{N-1} h_N \cdot \|f_j\|_p^p\right)^{1/p} \le \left(\sum_{j=0}^{N-1} h_N \cdot n \cdot \|f_j\|_\infty^p\right)^{1/p}$$

$$\le \left(n \cdot \sum_{j=0}^{N-1} h_N \cdot \|f^N\|_\infty^p\right)^{1/p} = \sqrt[p]{n} \cdot \left(\|f^N\|_\infty^p \cdot \sum_{j=0}^{N-1} h_N\right)^{1/p}$$

$$= \underbrace{\sqrt[p]{n \cdot (T-t_0)}}_{C:=} \cdot \|f^N\|_\infty \qquad ■$$

5. The other way round is not true, i.e. $\nexists C > 0$ **independent of N** with $\|f^N\|_\infty \le C \cdot \|f^N\|_2$ for all $f(.)^N \in \mathbb{R}^{(N+1)}$ and $N \in \mathbb{N}$.
Proof : Analog to the continuous case. Just evaluate $f_k(.)$ on the grid. ■

The following calculation shall illustrate, that in any estimation of the form $\|f^N\|_\infty \le C \cdot \|f^N\|_p$ for fixed N the constant $C > 0$ depends on N. In fact, $C \to \infty$ $(N \to \infty)$. Due to the specific definition of the $\|.\|_p$-norm (which does not take f_N into account), f_N will also be cut off on the left side. (Note: One could have defined the $\|.\|_\infty$ to be more comparable to the $\|.\|_p$-norm by leaving f_N out. But as those two norms are not equivalent anyway and for some purposes later on this isn't such a good idea.

$$\|(f_0,\ldots,f_{N-1})\|_\infty = \sup_{j=0,\ldots,N-1} \|f_j\|_\infty \le \sup_{j=0,\ldots,N-1} \|f_j\|_p \le \left(\sum_{j=0}^{N-1} \|f_j\|_p^p\right)^{1/p}$$

$$= \frac{1}{\sqrt[p]{h_N}} \cdot \left(h_N \sum_{j=0}^{N-1} \|f_j\|_p^p\right)^{1/p} = \underbrace{\sqrt[p]{\frac{N}{T-t_0}}}_{C:=} \|f^N\|_p$$

2.5.3 Convergence

The phrase "the discrete solution f^N converges to the continuous one $f(.)$" shall be understood as:

$$\boxed{\lim_{N\to\infty} \|f^N - \rho_N(f(.))\| = 0 \quad \text{(discrete convergence)}}$$

Where $\|.\|$ is an appropriate discrete norm (for example the discrete $\|.\|_\infty$-norm or the discrete $\|.\|_2$-norm defined in 2.5.1). ρ_N, as defined in 2.4, is the ordinary restriction operator to the grid.
Let's consider the discrete norm properties 2.5.2.4 and 2.5.2.5. As a direct consequence of these properties the following holds:

- $\lim_{N\to\infty} \|f^N - \rho_N(f(.))\|_\infty = 0 \overset{2.5.2.4}{\Rightarrow} \lim_{N\to\infty} \|f^N - \rho_N(f(.))\|_2 = 0.$

  So convergence in the discrete L_∞-norm leads to convergence in the discrete L_2-norm.

- From 2.5.2.5 it follows directly, that the other way round is not true. So convergence in the discrete L_2-norm does **not** lead to convergence in the discrete L_∞-norm.

2.5.4 Hausdorff-distance

As already stated the right-hand side of the ODE occurring in the considered class of optimal control problems will in general be parametrized by the control. This, as shown in 2.3.3, leads to considering set-valued right-hand sides in the ODE, which is then called a differential inclusion. Dealing with this set-valued approach will include measuring the distance between sets. This will be done using the Hausdorff-distance. In general the Hausdorff-distance is defined the following way:

general Hausdorff-distance:
Let $(M, \tilde{d})$ be a metric space with metric $\tilde{d}$. Let furthermore be $A, B \subset M$. Then the one sided distance d of the sets A and B is defined in the following way:

$$d(A,B) := \sup_{a\in A} \operatorname{dist}(a,B)$$

Where $\operatorname{dist}(x,Z) = \inf_{z\in Z} \tilde{d}(x,z)$ for $x \in M$ and $C \subset M$.

This definition is not symmetrical, i.e. in general it holds $d(A,B) \neq d(B,A)$. The Hausdorff-distance d_H of the sets A and B adds symmetry by taking the maximum:

$$d_H(A,B) := \max\left(d(A,B), d(B,A)\right)$$

Note: *$d_H(A,B) = 0$ does not mean that $A = B$, but it means, that $\bar{A} = \bar{B}$. The bar $\bar{.}$ denotes the closure of a set.*

One property of the Hausdorff-distance is the triangular inequality:

Triangular Inequality
Let $A, B, Z \subset M$. Then it holds:

$$d_H(A,B) \leq d_H(A,Z) + d_H(Z,B)$$

Proof :
The idea of the proof is to start with the triangular inequality for the dist-function and derive the triangle inequality for d. This result then leads to the triangle inequality of the Hausdorff-distance. Because of $d(A,B) = \sup\limits_{a \in A} \operatorname{dist}(a,B)$ we are taking a look at $\operatorname{dist}(a,B)$ first. Via the triangular inequality for the dist-function one gets:

① $$\operatorname{dist}(a,B) \leq \operatorname{dist}(a,\{z\}) + \operatorname{dist}(z,B) \overset{\text{Def. of } d}{\leq} \operatorname{dist}(a,\{z\}) + d(Z,B) \qquad \forall z \in Z$$

From the definition of the dist-function we get:

② $$d(a,Z) = \inf_{z \in Z} \operatorname{dist}(a,\{z\}) \Rightarrow \exists\, (z_k)_{k \in \mathbb{N}} \subset Z \text{ such that } \operatorname{dist}(a,z_k) \xrightarrow{k \to \infty} \operatorname{dist}(a,Z)$$

As ① holds for all z in Z this leads to:

① $$\operatorname{dist}(a,B) \overset{\textcircled{1}}{\leq} \operatorname{dist}(a,\{z_k\}) + d(Z,B) \xrightarrow{k \to \infty} \operatorname{dist}(a,Z) + d(Z,B)$$

So we have:

① $$\operatorname{dist}(a,B) \leq \operatorname{dist}(a,Z) + d(Z,B)$$

Applying the definition of d leads to:

③ $$d(A,B) = \sup_{a \in A} \operatorname{dist}(a,B) \overset{\textcircled{1}}{\leq} \sup_{a \in A} \operatorname{dist}(a,Z) + d(Z,B) = d(A,Z) + d(Z,B) \overset{\text{Def. } d_H}{\leq} d_H(A,Z) + d_H(Z,B)$$

Of course A and B can be interchanged, which gives:

④ $$d(B,A) \overset{\textcircled{3}}{\leq} d(B,Z) + d(Z,A) \overset{\text{Def. } d_H}{\leq} d_H(B,Z) + d_H(Z,A) \overset{d_H \text{ symmetric}}{=} d_H(A,Z) + d_H(Z,B)$$

Alltogether we have:

$$d_H(A,B) \overset{\text{Def. } d_H}{=} \max\left(d(A,B), d(B,A)\right) \overset{\textcircled{3},\textcircled{4}}{\leq} d_H(A,Z) + d_H(Z,B)$$

■

One goal of this section is to use this concept to measure the distance between the sets $X^N(T,t_0,X_0)$ and $X(T,t_0,X_0)$ respectively the sets $X^N_\Theta(T,t_0,X_0)$ and $X_\Theta(T,t_0,X_0)$. Recall, that $X^N(T,t_0,X_0)$ is the set of all feasible solutions to the discrete Mayer-Problem respectively $X(T,t_0,X_0)$ the set of all feasible solutions to the continuous

Mayer-Problem, both without pure state constraints Θ. $X_\Theta^N(T, t_0, X_0)$ is the set of all feasible solutions to the discrete Mayer-Problem respectively $X_\Theta(T, t_0, X_0)$ the set of all feasible solutions to the continuous Mayer-Problem with pure state constraints. $X^N(T, t_0, X_0)$ and $X_\Theta^N(T, t_0, X_0)$ are treated in 2.4.3.
Measuring the distance between a set $X \subset AC(I)^n$ and $X^N \in \mathbb{R}^{(N+1)n}$ will always be done on the grid. This means that the set X will be restricted to the grid by using the ordinary restriction operator ρ_N (see 2.4.1) and afterwards the Hausdorff-distance will be applied. So the Hausdorff-distance of X and X^N has to be understood as $d_H(\rho_N(X), X^N)$.

Of course the above definition of the Hausdorff-distance depends on the metric $\tilde{d}$, which will always be a norm in this article. Three special cases, that will be considered throughout this article are:

$$\begin{aligned}
\operatorname{dist}_\infty^N(b, A) &:= \inf \left\{ \sup_{j=0,\dots,N} \|b - a\|_2 : a \in A \right\} && (A \subset \mathbb{R}^{(N+1)n}, b \in \mathbb{R}^{(N+1)n}) \\
\operatorname{dist}_\infty(b, A) &:= \inf \{ \|b - a\|_\infty : a \in A \} && (A \subset (M, \|.\|_\infty), b \in (M, \|.\|_\infty)) \\
\operatorname{dist}_2(b, A) &:= \inf \{ \|b - a\|_2 : a \in A \} && (A \subset (M, \|.\|_2), b \in (M, \|.\|_2))
\end{aligned}$$

This dist-functions lead directly to the definition of the corresponding Hausdorff-distances.

$$\begin{aligned}
d_{H,\infty}^N(A, B) &:= \max \left(\sup_{a \in A} \operatorname{dist}_\infty^N(a, B), \sup_{b \in B} \operatorname{dist}_\infty^N(b, A) \right) && (A, B \subset \mathbb{R}^{(N+1)n}) \\
d_{H,\infty}(A, B) &:= \max \left(\sup_{a \in A} \operatorname{dist}_\infty(a, B), \sup_{b \in B} \operatorname{dist}_\infty(b, A) \right) && (A, B \subset (M, \|.\|_\infty)) \\
d_{H,2}(A, B) &:= \max \left(\sup_{a \in A} \operatorname{dist}_2(a, B), \sup_{b \in B} \operatorname{dist}_2(b, A) \right) && (A, B \subset (M, \|.\|_2))
\end{aligned}$$

So for $X \subset AC(I)^n$ and $X^N \in \mathbb{R}^{(N+1)n}$ we have:

$$d_{H,\infty}^N(\rho_N(X), X^N) = \max \left(\sup_{x(.) \in X} \operatorname{dist}_\infty^N(\rho_N(x), X^N), \sup_{x^N \in X^N} \operatorname{dist}_\infty^N(x^N, \rho_N(X)) \right)$$

For $X, Y \subset AC(I)^n$ we get:

$$d_{H,\infty}(X, Y) = \max \left(\sup_{x(.) \in X} \operatorname{dist}_\infty(x(.), Y), \sup_{y(.) \in Y} \operatorname{dist}_\infty(y(.), X) \right)$$

And as a last example for $X, Y \subset \mathbb{R}^n$ the above definitions deliver:

$$d_{H,2}(X, Y) = \max \left(\sup_{x \in X} \operatorname{dist}_2(x, Y), \sup_{y \in Y} \operatorname{dist}_2(y, X) \right)$$

Note: *The examples above have been chosen to represent usage of the Hausdorff-distance later on. The Hausdorff-distances occur especially in chapter 5. That chapter is based on [2]. Note that $d_{H,\infty}$ used in [2] corresponds to $d_{H,\infty}^N$ in this thesis.*

2.6 Differential Inclusion Theory

This section presents some basic results concerning differential inclusions. These results will play a great role in chapter 5. That chapter is about the so called Approximation Property and delivers one of the major results in this thesis, i.e. a relation of the solution set of the Constrained Differential Inclusion and the solution set of the Constrained Discrete Differential Inclusion. In chapter 3 the results about Lipschitz-continuity of solutions of Differential Inclusions with global Lipschitz constant are needed to get the so called Compatibility Property (3.2.6). The uniform boundedness theorems are essential for restricting Lipschitz-continuity premises on compact sets (see Theorem 3.2.2 and Theorem 3.2.4). Let's begin with a few assumptions. Recall the definition of the norm of a set (for example $\|F(t,x)\|_2 = \sup\limits_{y \in F(t,x)} \|y\|_2$, see 2.5.1) and the definition of the Hausdorff-distance (see 2.5.4).

Assumptions:

(A1') F satisfies a linear growth condition in integrable form, i.e.:
There exists a nonnegative function $C_F(.) \in L_1(I)$ such that with $t \in I$ and $x \in \mathbb{R}^n$ it holds

$$\|F(t,x)\|_2 \leq C_F(t)\ (\|x\|_2 + 1)$$

(A1) F satisfies a linear growth condition, i.e.:
There exists a constant $C_F \geq 0$ such that with $t \in I$ and $x \in \mathbb{R}^n$ it holds

$$\|F(t,x)\|_2 \leq C_F\ (\|x\|_2 + 1)$$

Note: *It should be easy to see that the assumption (A1') is a weaker formulation of (A1), i.e. (A1) implies (A1').*

These assumptions are enough to obtain the following results concerning the boundedness and Lipschitz-continuity of feasible solutions to the Differential Inclusions.

2.6.1 Theorem (Uniform Boundedness)

Let the set of all valid initial values $\boldsymbol{X_0}$ **be bounded** and let **(A1')** be fulfilled. Then all solutions $x(.)$ of the Differential Inclusion presented in Definition 3, i.e. $x(.) \in X(T, t_0, X_0)$, are **uniformly bounded** by the constant $M := (\|X_0\|_2 + C_L)e^{C_L}$ with $C_L := \|C_F(.)\|_1$.

Note: *This shows that there exists a compact set $S \in \mathbb{R}^n$ such that $x(t) \in S$ for all $x(.) \in X_\Theta(T, t_0, X_0)$ and $t \in [t_0, T]$.*

Proof :
The strategy for this proof is exactly the same as for the well known Gronwall Lemma. The only difference is the occurrence of the time dependent constant $C_F(.)$.

Let $x(.) \in X(T, t_0, X_0)$. This means that $\dot{x}(t) \in F(t, x(t))$ *a.e.* and $x(t_0) \in X_0$. So we get:

①
$$\begin{aligned}
&\|x(t)\|_2 = \|x(t_0) + \int_{t_0}^{t} \dot{x}(\tau)\; d\tau\|_2 \le \|x(t_0)\|_2 + \int_{t_0}^{t} \|\dot{x}(\tau)\|_2\; d\tau \overset{\text{Def. } \|F(t,x(t))\|_2}{\le} \\
&\|x(t_0)\|_2 + \int_{t_0}^{t} \|F(\tau, x(\tau))\|_2\; d\tau \overset{\text{(A1')}}{\le} \|x(t_0)\|_2 + \int_{t_0}^{t} C_F(\tau)\left(\|x(\tau)\|_2 + 1\right) d\tau \le \\
&\qquad \|X_0\|_2 + \int_{t_0}^{t} C_F(\tau)\; d\tau + \int_{t_0}^{t} C_F(\tau)\, \|x(\tau)\|_2\; d\tau \overset{C_F(.) \text{ nonnegative}}{\le} \\
&\qquad\qquad \underbrace{\|X_0\|_2 + C_L}_{\tilde{C}:=} + \int_{t_0}^{t} C_F(\tau)\, \|x(\tau)\|_2\; d\tau \;(t \in I)
\end{aligned}$$

Due to $x(.)$ being continuous on I there exists a constant $C_x \ge 0$ with $\|x(t)\|_2 \le C_x$ $(t \in I)$ (which means $x(.)$ is bounded on I). Note that the constant C_x depends on the specific function $x(.)$. This proof is about showing that there exists a constant M, which is an upper bound to $\|x(.)\|_\infty$ for all $x(.) \in X(T, t_0, X_0)$, so we have to be careful not to mix that up. From combining $\|x(t)\|_2 \le C_x$ $(t \in I)$ with ① we get:

②
$$\|x(t)\|_2 \overset{①}{\le} \tilde{C} + \int_{t_0}^{t} C_F(\tau)\, \|x(\tau)\|_2\; d\tau \overset{\|x(\tau)\|_2 \le C_x}{\le} \tilde{C} + C_x \int_{t_0}^{t} C_F(\tau)\; d\tau$$

Placing ② in ① leads to:

③
$$\|x(t)\|_2 \overset{①}{\le} \tilde{C} + \int_{t_0}^{t} C_F(\tau)\, \|x(\tau)\|_2\; d\tau \overset{②}{\le} \tilde{C} + \tilde{C} \int_{t_0}^{t} C_F(\tau)\; d\tau + C_x \int_{t_0}^{t} C_F(\tau_1) \int_{t_0}^{\tau_1} C_F(\tau_2)\; d\tau_2 d\tau_1$$

Using the above inequality and placing it into ① and repeating that process k-times gives us the following estimation:

③
$$\begin{aligned}
\|x(t)\|_2 \le \tilde{C} + \tilde{C} \int_{t_0}^{t} C_F(\tau)\; d\tau + \cdots + \tilde{C} \underbrace{\int_{t_0}^{t} C_F(\tau_1) \ldots \int_{t_0}^{\tau_{k-1}} C_F(\tau_k)\; d\tau_k \ldots d\tau_1}_{k \text{ nested integral terms}} \\
+ C_x \underbrace{\int_{t_0}^{t} C_F(\tau_1) \ldots \int_{t_0}^{\tau_k} C_F(\tau_{k+1})\; d\tau_{k+1} \ldots d\tau_1}_{k+1 \text{ nested integral terms}} \qquad (t \in I,\, k \in \mathbb{N})
\end{aligned}$$

The right-hand side of the estimation depends on C_x. To reach our goal we have to get rid of that dependency. It should be clear, that we have to deal with the nested integral terms of the form $\int_{t_0}^{t} C_F(\tau_1) \ldots \int_{t_0}^{\tau_{k-1}} C_F(\tau_k)\; d\tau_k \ldots d\tau_1$. Via simple integration by parts one gets:

$$\begin{aligned}
&\int_{t_0}^{t} C_F(\tau_1) \int_{t_0}^{\tau_1} C_F(\tau_2)\; d\tau_2 d\tau_1 = \left(\int_{t_0}^{t} C_F(\tau) d\tau\right)^2 - \int_{t_0}^{t} C_F(\tau_1) \int_{t_0}^{\tau_1} C_F(\tau_2)\; d\tau_2 d\tau_1 \\
\Leftrightarrow &\int_{t_0}^{t} C_F(\tau_1) \int_{t_0}^{\tau_1} C_F(\tau_2)\; d\tau_2 d\tau_1 = \frac{1}{2}\left(\int_{t_0}^{t} C_F(\tau) d\tau\right)^2
\end{aligned}$$

This gives the idea for completing the proof. Indeed it holds:

④

$$\int_{t_0}^{t} C_F(\tau_1) \dots \int_{t_0}^{\tau_{k-1}} C_F(\tau_k)\, d\tau_k \dots d\tau_1 = \frac{1}{k!} \left(\int_{t_0}^{t} C_F(\tau) d\tau \right)^k$$

It is easy to show that result via induction. The initial step has already been done. So we consider ④ to be true for $\tilde{k} \in \mathbb{N}$, then it follows:

$$\int_{t_0}^{t} C_F(\tau_1) \dots \int_{t_0}^{\tau_{\tilde{k}}} C_F(\tau_{\tilde{k}+1})\, d\tau_{\tilde{k}+1} \dots d\tau_1 \underset{\text{assumption}}{\overset{\text{inductive}}{=}} \frac{1}{\tilde{k}!} \int_{t_0}^{t} C_F(\tau_1) \left(\int_{t_0}^{\tau_1} C_F(\tau_2) d\tau_2 \right)^{\tilde{k}} \, d\tau_1$$

With integration by parts we get:

$$\int_{t_0}^{t} C_F(\tau_1) \left(\int_{t_0}^{\tau_1} C_F(\tau_2) d\tau_2 \right)^{\tilde{k}} \, d\tau_1 = \left(\int_{t_0}^{t} C_F(\tau) d\tau \right)^{\tilde{k}+1} - \tilde{k} \int_{t_0}^{t} C_F(\tau_1) \left(\int_{t_0}^{\tau_1} C_F(\tau_2) d\tau_2 \right)^{\tilde{k}} \, d\tau_1$$

$$\Leftrightarrow \int_{t_0}^{t} C_F(\tau_1) \left(\int_{t_0}^{\tau_1} C_F(\tau_2) d\tau_2 \right)^{\tilde{k}} \, d\tau_1 = \frac{1}{\tilde{k}+1} \left(\int_{t_0}^{t} C_F(\tau) d\tau \right)^{\tilde{k}+1}$$

Combining both results we complete the induction step:

$$\int_{t_0}^{t} C_F(\tau_1) \dots \int_{t_0}^{\tau_{\tilde{k}}} C_F(\tau_{\tilde{k}+1})\, d\tau_{\tilde{k}+1} \dots d\tau_1 = \frac{1}{(\tilde{k}+1)!} \left(\int_{t_0}^{t} C_F(\tau) d\tau \right)^{\tilde{k}+1}$$

Together, the initial step and the induction step prove, that ④ ist true.

Combining ③ and ④ yields:

③

$$\begin{aligned} \|x(t)\|_2 &\leq \tilde{C} + \tilde{C} \int_{t_0}^{t} C_F(\tau)\, d\tau + \dots + \tilde{C} \frac{1}{k!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^k + C_x \frac{1}{(k+1)!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^{k+1} \\ &= \tilde{C} \sum_{l=0}^{k} \frac{1}{l!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^l + C_x \frac{1}{(k+1)!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^{k+1} \qquad (t \in I,\ k \in \mathbb{N}) \end{aligned}$$

To obtain a uniform upper bound for all $x(.) \in X(T, t_0, X_0)$ we still need to get rid of C_x. But as the above inequality holds for all $k \in \mathbb{N}$ we can take the limit of $k \to \infty$ on the right-hand side. From investigation of the exponential series it then follows:

$$\sum_{l=0}^{k} \frac{1}{l!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^l \to e^{\left(\int_{t_0}^{t} C_F(\tau)\, d\tau\right)} \text{ and } C_x \frac{1}{(k+1)!} \left(\int_{t_0}^{t} C_F(\tau)\, d\tau \right)^{k+1} \to 0 \qquad (k \to \infty)$$

So we finally got rid of C_x and obtain the final result:

③

$$\|x(t)\|_2 \leq \tilde{C} e^{\left(\int_{t_0}^{t} C_F(\tau)\, d\tau\right)} \leq \tilde{C} e^{C_L} = \underbrace{\left(\|X_0\|_2 + C_L\right) e^{C_L}}_{M:=} \qquad (t \in I)$$

■

As a simple result of the Uniform Boundedness Theorem 2.6.1 we get that all solutions $x(.) \in X(T, t_0, X_0)$ are uniformly Lipschitz-continuous, i.e. Lipschitz-continuous with uniform Lipschitz constant. We will call that Lipschitz constant L.

2.6.2 Theorem (Uniform Lipschitz Continuity)

Let the set of all feasible initial values $\boldsymbol{X_0}$ **be bounded** and let **(A1)** be fulfilled. Then all feasible solutions of the Differential Inclusion presented in Definition 3, i.e. $x(.) \in X(T, t_0, X_0)$, are **uniformly Lipschitz-continuous** with Lipschitz constant L (with respect to the $\|.\|_2$-norm).

Proof : Let $x(.) \in X(T, t_0, X_0)$ and $\tilde{t}, t \in I$ with $t \geq \tilde{t}$ then:

$$\begin{aligned} \|x(t) - x(\tilde{t})\|_2 &= \| \int_{\tilde{t}}^{t} \dot{x}(\tau)\ d\tau \|_2 \leq \int_{\tilde{t}}^{t} \|\dot{x}(\tau)\|_2\ d\tau \overset{\text{Def. } \|F(t,x(t))\|_2}{\leq} \int_{\tilde{t}}^{t} \|F(\tau, x(\tau))\|_2\ d\tau \\ &\overset{\text{(A1)}}{\leq} C_F \int_{\tilde{t}}^{t} \|x(\tau)\|_2 + 1\ d\tau \overset{\text{Theorem 2.6.1}}{\leq} C_F\ (M+1)\ (t - \tilde{t}) \overset{t \geq \tilde{t}}{=} C_F\ (M+1)\ |t - \tilde{t}| \end{aligned}$$

For $t < \tilde{t}$ we get with the above inequality:

$$\|x(t) - x(\tilde{t})\|_2 = \|x(\tilde{t}) - x(t)\|_2 \leq C_F\ (M+1)\ (\tilde{t} - t) \overset{t \leq \tilde{t}}{=} C_F\ (M+1)\ |t - \tilde{t}|$$

So overall we have for $x(.) \in X(T, t_0, X_0)$ and $\tilde{t}, t \in I$:

$$\|x(t) - x(\tilde{t})\|_2 \leq \underbrace{C_F\ (M+1)}_{L:=}\ |t - \tilde{t}|$$

■

For the discrete case we get similar results with the assumptions (A1) and (A1'). The only thing that needs to be strengthened is that we need $C_F(.)$ to be Riemann integrable instead of "just" Lebesgue integrable. For our future investigations in the following chapters, it will be of extreme importance that these results deliver a uniform upper bound and an uniform Lipschitz constant independent of the grid, i.e. independent of N. Let's take a look at the discrete analog of the Uniform Boundedness Theorem 2.6.1 on the next page.

2.6.3 Theorem (Discrete Uniform Boundedness)

Let the set of all valid initial values $\boldsymbol{X_0}$ **be bounded** and let **(A1')** be fulfilled. In addition let $C_F(.)$ be Riemann integrable. Then all solutions x^N of the Discrete Differential Inclusion presented in Definition 6, i.e. $x^N \in X^N(T, t_0, X_0)$, are **uniformly bounded** by the constant $\tilde{M} := (\|X_0\|_2 + C_R)e^{C_R}$ **independent of N** with C_R being the upper bound of all Riemann sums $h_N \sum_{k=0}^{N-1} C_F(t_k)$ (this means $h_N \sum_{k=0}^{N-1} C_F(t_k) \leq C_R \;\; \forall N \in \mathbb{N}$). This means that $\|x_j^N\|_2 \leq \tilde{M}$ $(j = 0, \ldots, N)$ for all $x^N \in X^N(T, t_0, X_0)$ and $N \in \mathbb{N}$.

Remarks:

- *Be aware, that in general $\tilde{M} \neq M$. This is because C_R is an upper bound to all Riemann sums $h_N \sum_{k=0}^{N-1} C_F(t_k)$ $(N \in \mathbb{N})$, which in general does not coincide with $\|C_F(.)\|_1 = C_L$.*
- *This theorem shows that there exists a compact set $\tilde{S} \in \mathbb{R}^n$ such that $x_j^N \in \tilde{S}$ for all $x^N \in X_\Theta^N(T, t_0, X_0)$, $j \in \{0, \ldots, N\}$ and $N \in \mathbb{N}$.*

Proof :
The ideas of this proof are exactly the same as for the proof of Theorem 2.6.1. The major difference is that we have to deal with Riemann sums instead of integrals.

Let $x^N \in X^N(T, t_0, X_0)$. This means that there exists $\xi_l^N \in F(t_l, x_l^N)$ such that $x_{l+1}^N = x_l^N + h_N \xi_l^N$. This leads to:

①
$$\begin{aligned}
&\|x_j^N\|_2 = \|x_0^N + h_N \sum_{l=0}^{j-1} \xi_l^N\|_2 \leq \|x_0^N\|_2 + h_N \sum_{l=0}^{j-1} \|\xi_l^N\|_2 \, d\tau \overset{\text{Def. } \|F(t_l,\xi_l^N)\|_2}{\leq} \\
&\|x_0^N\|_2 + h_N \sum_{l=0}^{j-1} \|F(t_l, \xi_l^N)\|_2 \overset{\text{(A1')}}{\leq} \|x_0^N\|_2 + h_N \sum_{l=0}^{j-1} C_F(t_l) \left(\|x_l^N\|_2 + 1 \right) \leq \\
&\|X_0\|_2 + h_N \sum_{l=0}^{j-1} C_F(t_l) + h_N \sum_{l=0}^{j-1} C_F(t_l) \, \|x_l^N\|_2 \underset{\text{integrable}}{\overset{C_F(.) \text{ Riemann}}{\leq}} \\
&\underbrace{\|X_0\|_2 + C_R}_{\tilde{C} :=} + h_N \sum_{l=0}^{j-1} C_F(t_l) \, \|x_l^N\|_2
\end{aligned}$$

Note: *There exists C_R with $h_N \sum_{j=0}^{N-1} C_F(t_j) \leq C_R \;\; \forall N \in \mathbb{N}$ because the Riemann sums $h_N \sum_{j=0}^{N-1} C_F(t_j)$ converge for $(N \to \infty)$. That's why we needed $C_F(.)$ to be Riemann integrable for this proof.*

Let $C_{x^N} := \|x^N\|_2 \geq \|x_l^N\|_2$ $(l \in \{0, \ldots, N\})$. We then get:

②
$$\|x_j^N\|_2 \leq \tilde{C} + h_N \sum_{l=0}^{j-1} C_F(t_l) \, \|x_l^N\|_2 \leq \tilde{C} + C_{x^N} \; h_N \sum_{l=0}^{j-1} C_F(t_l)$$

Placing ② in ① leads to:

③
$$\|x_j^N\|_2 \overset{\text{①,②}}{\leq} \tilde{C} + \tilde{C}\ h_N \sum_{l_1=0}^{j-1} C_F(t_{l_1}) + C_{x^N}\ h_N^2 \sum_{l_1=0}^{j-1} \left(C_F(t_{l_1}) \sum_{l_2=0}^{l_1-1} C_F(t_{l_2}) \right)$$

Using the above inequality and placing it into ① and repeating that process k-times gives us the following estimation:

③
$$\begin{aligned} \|x_j^N\|_2 \leq \tilde{C} + \tilde{C}\ h_N \sum_{l_1=0}^{j-1} C_F(t_{l_1}) + \cdots + \tilde{C}\ h_N^k \underbrace{\sum_{l_1=0}^{j-1} \left(C_F(t_{l_1}) \cdots \sum_{l_k=0}^{l_{k-1}-1} C_F(t_{l_k}) \right)}_{k \text{ nested sums}} \\ + C_{x^N}\ h_N^{k+1} \underbrace{\sum_{l_1=0}^{j-1} \left(C_F(t_{l_1}) \cdots \sum_{l_k=0}^{l_{k-1}-1} C_F(t_{l_k}) \right)}_{k \text{ nested sums}} \qquad (j \in \{0, \ldots, N\},\, k \in \mathbb{N}) \end{aligned}$$

Note: $\sum_l^k a_l = 0$ *if* $k < l$*. This occurs in the inequality above, especially if the nesting depth is greater than* $j + 1$*.*

Again, our goal is to get rid of C_{x^N}. Unlike the proof in the continuous case, we have to deal with nested sums instead of nested integral terms here. The idea now is to work with the formula for integrals we have already from ④ in the proof of Theorem 2.6.1, i.e.:

④
$$\int_{t_0}^{t} f(\tau_1) \ldots \int_{t_0}^{\tau_{k-1}} f(\tau_k)\ d\tau_k \ldots d\tau_1 = \frac{1}{k!} \left(\int_{t_0}^{t} f(\tau) d\tau \right)^k \qquad (f \in L_1(I))$$

To use it the idea is to rewrite the sums occurring in ③ as integrals. To do so we introduce:

$$C_F^N(t) := \sum_{l=0}^{N-1} C_F(t_l)\, \chi_{[t_l, t_{l+1}[}(t)$$

with χ being the characteristic function, i.e.:

$$\chi_{[t_l, t_{l+1}[}(t) := \begin{cases} 1 & t \in [t_l, t_{l+1}[\\ 0 & \text{else} \end{cases}$$

So it holds:

⑤
$$\int_{t_0}^{t_j} C_F^N(\tau)\ d\tau = h_N \sum_{l=0}^{j-1} C_F(t_l)$$

To get the connection to ④ just replacing the sums in ③ via the formula above is not enough.

We need an additional estimation to work with ④:
Let $\tilde{k} \in \{0, \dots, N\}$ and let C^+ be Lebesgue integrable with $C^+(t) \geq 0$ for all $t \in I$, then:

$$h_N \sum_{l=0}^{\tilde{k}-1} C_F(t_l) \int_{t_0}^{t_l} C^+(\tau)\, d\tau \leq \int_{t_0}^{t_{\tilde{k}}} C_F^N(t) \int_{t_0}^{t} C^+(\tau)\, d\tau dt \tag{\textcircled{6}}$$

This follows directly from

$$h_N \sum_{l=0}^{\tilde{k}-1} C_F(t_l) \int_{t_0}^{t_l} C^+(\tau)\, d\tau = \int_{t_0}^{t_{\tilde{k}}} \left(\sum_{l=0}^{\tilde{k}-1} C_F(t_l) \left(\int_{t_0}^{t_l} C^+(\tau)\, d\tau \right) \chi_{[t_l, t_{l+1}[}(t) \right) dt$$

and inspection of the integrand

$$\begin{aligned}\sum_{l=0}^{\tilde{k}-1} C_F(t_l) \left(\int_{t_0}^{t_l} C^+(\tau)\, d\tau \right) \chi_{[t_l, t_{l+1}[}(t) &\overset{C^+ \text{nonnegative}}{\leq} \sum_{l=0}^{\tilde{k}-1} C_F(t_l) \left(\int_{t_0}^{t} C^+(\tau)\, d\tau \right) \chi_{[t_l, t_{l+1}[}(t) \\ &\overset{\text{Def. } C_F^N(.)}{=} C_F^N(t) \left(\int_{t_0}^{t} C^+(\tau)\, d\tau \right) \qquad (t \in I)\end{aligned}$$

With

$$C_F^N(t) \underbrace{\int_{t_0}^{t} C_F^N(\tau_1) \dots \int_{t_0}^{\tau_{k-1}} C_F^N(\tau_k)\, d\tau_k \dots d\tau_1}_{k \text{ nested integrals}} \geq 0 \qquad (t \in I,\ k \in \mathbb{N} \cup \{0\})$$

By successively applying ⑥ we get the result we were looking for to be able to use ④:

$$\begin{aligned}&\underbrace{h_N^k \sum_{l_1=0}^{j-1} \left(C_F(t_{l_1}) \cdots \sum_{l_k=0}^{l_{k-1}-1} C_F(t_{l_k}) \right)}_{k \text{ nested sums}} \overset{\textcircled{6}}{\leq} \int_{t_0}^{t_j} C_F^N(\tau_1) \dots \int_{t_0}^{\tau_{k-1}} C_F^N(\tau_k)\, d\tau_k \dots d\tau_1 \overset{\textcircled{4}}{=} \\ &\qquad \frac{1}{k!} \left(\int_{t_0}^{t_j} C_F^N(\tau) d\tau \right)^k \overset{\textcircled{5}}{=} \frac{1}{k!} \left(h_N \sum_{l=0}^{j-1} C_F(t_l) \right)^k \ (j \in \{0, \dots, N\},\, k \in \mathbb{N})\end{aligned}$$

In the final step we applied ④ which transforms the integral representation back into the sums representation. This result can now be substituted in ③ to get an exponential series like in Theorem 2.6.1:

$$\begin{aligned}\|x_j^N\|_2 &\leq \tilde{C} + \tilde{C}\, h_N \sum_{l_1=0}^{j-1} C_F(t_{l_1}) + \dots + \tilde{C}\, \frac{1}{k!} \left(h_N \sum_{l=0}^{j-1} C_F(t_l) \right)^k + C_{x^N}\, \frac{1}{(k+1)!} \left(h_N \sum_{l=0}^{j-1} C_F(t_l) \right)^{k+1} \\ &= \tilde{C} \sum_{i=0}^{k} \frac{1}{i!} \left(h_N \sum_{l=0}^{j-1} C_F(t_l) \right)^i + C_{x^N}\, \frac{1}{(k+1)!} \left(h_N \sum_{l=0}^{j-1} C_F(t_l) \right)^{k+1} \qquad (j \in \{0, \dots, N\},\, k \in \mathbb{N})\end{aligned} \tag{\textcircled{3}}$$

To obtain a uniform upper bound for all $x^N \in X^N(T, t_0, X_0)$ we still need to get rid of C_{x^N}. But as the above inequality holds for all $k \in \mathbb{N}$ we can take the limit of $k \to \infty$ on the right-hand side. From investigation of the exponential series it then follows:

$$\sum_{i=0}^{k}\frac{1}{i!}\left(h_N\sum_{l=0}^{j-1}C_F(t_l)\right)^{i} \to e^{\left(h_N\sum_{l=0}^{j-1}C_F(t_l)\right)} \text{ and } C_{x^N}\,\frac{1}{(k+1)!}\left(h_N\sum_{l=0}^{j-1}C_F(t_l)\right)^{k+1} \to 0 \qquad (k\to\infty)$$

So we finally got rid of C_{x^N} and obtain the final result:

③
$$\|x_j^N\|_2 \le \tilde{C}e^{\left(h_N\sum_{l=0}^{j-1}C_F(t_l)\right)} \le \tilde{C}e^{C_R} = \underbrace{(\|X_0\|_2 + C_R)\,e^{C_R}}_{\tilde{M}:=} \qquad (j\in\{0,\dots,N\})$$

■

Like in the continuous case we get that all solutions $x^N \in X^N(T,t_0,X_0)$ $(N\in\mathbb{N})$ are uniformly Lipschitz-continuous, i.e. Lipschitz-continuous with uniform Lipschitz constant independent of N. We will call that Lipschitz constant $\tilde{L}$.

2.6.4 Theorem (Discrete Uniform Lipschitz Continuity)

Let the set of all feasible initial values $\boldsymbol{X_0}$ **be bounded** and let **(A1)** be fulfilled. Then all feasible solutions of the Discrete Differential Inclusion presented in Definition 6, i.e. $x^N \in X^N(T,t_0,X_0)$, are **uniformly Lipschitz-continuous** with Lipschitz constant $\tilde{L}$ (with respect to the $\|.\|_2$-norm).

Proof :

Note: *(A1) implies that C(.) is Riemann integrable, because it is assumed to be constant. So Theorem 2.6.3 can be applied in the proof.*

Let $x^N \in X^N(T,t_0,X_0)$ and let $l>k$:

$$\begin{aligned}
\|x_l^N - x_k^N\|_2 \quad &= \quad h_N\,\Big\|\sum_{j=k}^{l-1}\frac{1}{h_N}(x_{j+1}^N - x_j^N)\Big\|_2 \le h_N\sum_{j=k}^{l-1}\Big\|\overbrace{\frac{1}{h_N}(x_{j+1}^N - x_j^N)}^{\in F(t_j,x_j^N)}\Big\|_2 \\
&\overset{\text{Def. }\|F(t_j,x_j^N)\|_2}{\le} \quad h_N\sum_{j=k}^{l-1}\|F(t_j,x_j^N)\|_2 \overset{\text{(A1)}}{\le} C_F\,h_N\sum_{j=k}^{l-1}(\|x_j^N\|_2+1) \\
&\overset{\text{Theorem 2.6.3}}{\le} \quad C_F\,(\tilde{M}+1)\,(l-k)h_N \overset{l\geq k}{=} C_F\,(\tilde{M}+1)\,|t_l-\tilde{t_k}|
\end{aligned}$$

For $l<k$ we get with the above inequality:

$$\|x_l^N - x_k^N\|_2 = \|x_k^N - x_l^N\|_2 \le C_F\,(\tilde{M}+1)\,(k-l)h_N \overset{l\leq k}{=} C_F\,(\tilde{M}+1)\,|t_l-t_k|$$

So overall we have for $x^N\in X^N(T,t_0,X_0)$ and $k,l\in\{0,\dots,N\}$:

$$\|x_l^N - x_k^N\|_2 \le \underbrace{C_F\,(\tilde{M}+1)}_{\tilde{L}:=}\,|t_l-t_k|$$

■

3 Convergence Theorem (Problem Specific Approach)

3.1 Overview

In this chapter the main result $\boldsymbol{\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq Ch_N^{1/2}}$ is shown, where $\hat{x}^N$ is the optimal solution to the discrete Mayer-Problem 2.4.2 respectively 2.4.4, $\hat{x}(.)$ the optimal solution to the continuous Mayer-Problem 2.3.2 respectively 2.3.3 and C some constant independent of N. Recall that ρ_N is the ordinary restriction operator to the grid (see 2.4.1). The approach presented here makes use of results specific to the problem class itself and the chosen numerical method. A more general view on the approach used in this chapter is presented in chapter 4. That chapter is included to give the reader a better sense on how modular and flexible the method used here really is. In fact there is the core concept of using **Value Convergence** in conjunction with a so called **Approximation Property**, a **Compatibility Property** and an **Inverse Stability Property**. All these properties are modules, which need to deliver certain results, but may be interchanged to fit to specific problems and discretization methods.
We will start off directly with examining $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$. The key to success is to bear in mind that we ultimately want to make use of Value Convergence (convergence of the objective function).
To give the reader an idea of the whole concept, the whole estimation will directly be presented, which involves the 4 major steps mentioned above. These steps will be explained later on in detail, but for now a short overview of these steps shall be given:

1. The **Approximation Property** ensures, that for any valid discrete solution x^N for the state of problem 2.4.2 there exists a solution $\pi_N(x^N)(.)$ to the continous problem 2.3.2 close enough to x^N (on the grid) and vice versa. Although it might seem that only the first mentioned direction is needed, we will see (when considering Value Convergence) that indeed both directions are the key to success. The result we exploit here is $\boldsymbol{\|x^N - \rho_N(\pi_N(x^N)(.))\|_\infty \leq ch_N}$. As the reader might see, this is the major component for connecting the discrete and the continuous case.

2. The **Compatibility Property** ensures, that the value of the discrete L_∞-norm of a L_∞-function $f(.) \in L_\infty(I)^k$ restricted to the grid is close enough to the value of the L_∞-norm of $f(.)$. With the additional property of Lipschitz-continuity for the function $f(.)$ (with Lipschitz-constant L_f) one gets $\boldsymbol{|\|\rho_N(f(.))\|_\infty - \|f(.)\|_\infty| \leq L_f h_N}$. Together with the Approximation Property this result will serve as a bridge between the discrete and the continuous case. In our case the function $f(.)$ will be $\pi_N(\hat{x}^N)(.) - \hat{x}(.)$ and the corresponding Lipschitz-constant will be denoted by $L_{\Delta x}$.

3. The **Inverse Stability Property** serves as a connection between the distance of a feasible state to the optimal state together with the distance of a feasible control to the optimal control and the difference of the corresponding values of the objective function. That's the point, where second order sufficient optimality conditions come into play. The final result we will exploit here is $\boldsymbol{\alpha \left(\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty + \|\tilde{\pi}_N(\hat{u}^N) - \hat{x}(.)\|_2 \right)^2 \leq J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))}$, where $\pi_N(\hat{x}^N)(.)$ is an appropriate solution for the state to the continuous problem according to the Approximation Property. $\tilde{\pi}_N(\hat{u}^N)$ is a corresponding feasible control to the state $\pi_N(\hat{x}^N)(.)$ and $\alpha > 0$. Note the use of the different norms, which is crucial here.

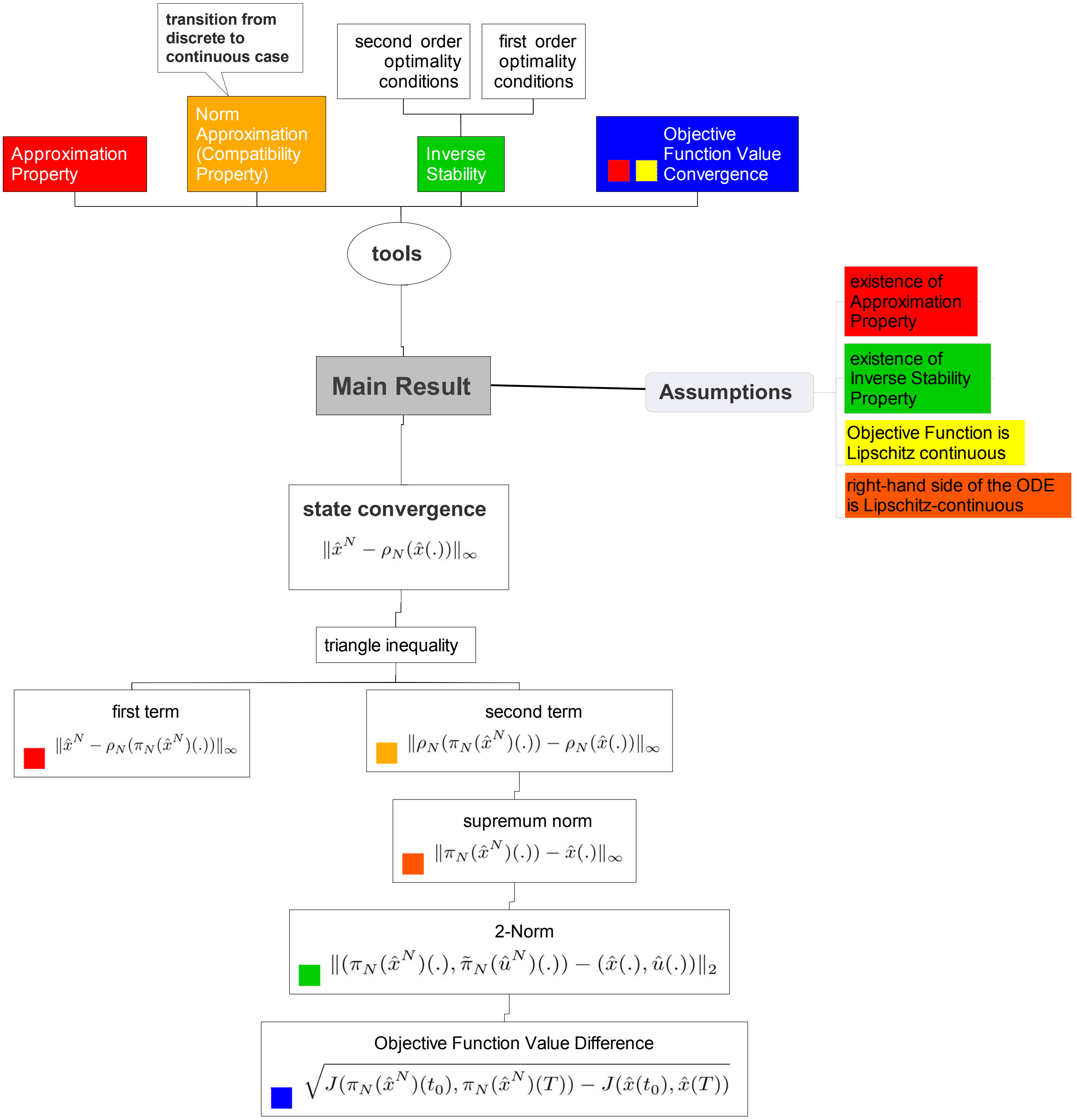
4. **Value Convergence** is convergence of the value of the objective function $J(\hat{x}_0^N, \hat{x}_N^N)$ to $J(\hat{x}(t_0), \hat{x}(T))$ for $N \to \infty$. Asuming Lipschiz-continuity of J (with corresponding Lipschitz-constant L_J) leads to $|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))| \leq L_J\, ch_N$. Based on that Value Convergence and the Approximation Property we will later on obtain the result $\boldsymbol{J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) \leq 2L_J\, ch_N}$. Together with the Inverse Stability Property this yields the important result $\boldsymbol{\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \sqrt{\frac{2L_J\, c}{\alpha}}\sqrt{h_N}}$.

Remember, the following estimation shall just serve as a starting point to see where this chapter is going. The details crucial for understanding the whole picture will be presented later on. Nevertheless the whole estimation process will be shown. To estimate $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$ we start off by using the Approximation Property, then the Compatibility Property and finally the Inverse Stability Property in conjunction with Value Convergence, i.e. the result in 4. So the whole process looks the following way:

$$\begin{aligned}
\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty &\underset{\text{according to 1.}}{\overset{\text{choose } \pi_N(\hat{x}^N)(.)}{\leq}} \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty \\
&\overset{1.}{\leq} ch_N + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty \overset{2.}{\leq} ch_N + L_{\Delta x} h_N + \|\pi_N(\hat{x}^N)(.)) - \hat{x}(.)\|_\infty \\
&\overset{3.,4.}{\leq} (c + L_{\Delta x}) h_N + \sqrt{\frac{2L_J\, c}{\alpha}}\sqrt{h_N} \overset{h_N \leq 1}{\leq} \underbrace{\left(c + L_{\Delta x} + \sqrt{\frac{2L_J\, c}{\alpha}} \right)}_{C:=} \sqrt{h_N}
\end{aligned}$$

The only reason for presenting the estimation process this way is that it is much more readable. The way following the ideas, which lead to the final result, would be the other way round. If we would go that way, we'd have to start with the inverse stability combined with Value Convergence, because Value Convergence is the result we want to make use of. And that is exactly what will be done in the next section. To understand the ideas behind the approach above the estimation process will be presented in detail there.

The following figure shows the estimation process of the following section in detail.



transition from discrete to continuous case
second order optimality conditions
first order optimality conditions
Approximation Property
Norm Approximation (Compatibility Property)
Inverse Stability
🟥 🟨 Objective Function Value Convergence
tools
Main Result
Assumptions
existence of Approximation Property
existence of Inverse Stability Property
Objective Function is Lipschitz continuous
right-hand side of the ODE is Lipschitz-continuous
state convergence
$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$
triangle inequality
first term
🟥 $\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty$
second term
🟧 $\|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty$
supremum norm
🟧 $\|\pi_N(\hat{x}^N)(.)) - \hat{x}(.)\|_\infty$
2-Norm
🟩 $\|(\pi_N(\hat{x}^N)(.), \tilde{\pi}_N(\hat{u}^N)(.)) - (\hat{x}(.), \hat{u}(.))\|_2$
Objective Function Value Difference
🟦 $\sqrt{J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))}$

3.2 Detailed Estimation Process

In the previous section, in order to give a quick overview, only the surface of the concept has been touched. This section deals with the details.

The central idea is based on the assumption of Value Convergence, i.e. convergence of the value of the objective function $J(\hat{x}_0^N, \hat{x}_N^N)$ to $J(\hat{x}(t_0), \hat{x}(T))$ for $N \to \infty$. If this is the case, there is hope that the corresponding states and maybe even controls will converge with respect to a certain discrete norm, too. For this to happen there is the need for some sort of Inverse Stability Property, but more on that topic later.
Indeed under certain circumstances Value Convergence can be proven, but to do so we need one major result, the so called Approximation Property. As we will see later on in the prove of Value Convergence (see 3.2.2) there will be a need for a feasible solution to the continuous Mayer-Problem (see 2.3.2 respectively 2.3.3) close enough to the optimal discrete solution of the discrete Mayer-Problem (see 2.4.2 respectively 2.4.4) and vice versa. So before taking a look at Value Convergence we shall consider the Approximation Property. This property builds the core of the whole estimation process and is in no way a trivial result. That's why the whole chapter 5 of this thesis has been devoted to it. For now only the result shown in chapter 5 will be presented.

3.2.1 Theorem (Approximation Property)

Recall Definition 2 (definition of feasible solution sets to the Mayer-Problem). Let $x(.) \in X_\Theta(T, t_0, X_0)$, $x^N \in X_\Theta^N(T, t_0, X_0)$ and let all the assumptions from chapter 5 ((A1), (A2), (A3), (C1) and (C2)) be satisfied. Then $\exists \tilde{N} \in \mathbb{N}$, such that for any $N \in \mathbb{N}$ with $N \geq \tilde{N}$ there exist functions $\pi_N : X_\Theta^N(T, t_0, X_0) \to X_\Theta(T, t_0, X_0)$ and $\delta_N : X_\Theta(T, t_0, X_0) \to X_\Theta^N(T, t_0, X_0)$, such that:

$$\|\rho_N(\pi_N(x^N)(.)) - x^N\|_\infty \leq ch_N$$
$$\|\delta_N(x(.)) - \rho_N(x(.))\|_\infty \leq ch_N$$

Remarks:

- Note the image sets of the functions. They indicate that $\pi_N(x^N)(.)$ is a feasible solution to the continuous Mayer-Problem and $\delta_N(x(.))$ a feasible solution to the discrete Mayer-Problem.
- An alternative description of the Approximation Property without introducing the functions π_N and δ_N would be by using the Hausdorff-distance $d_{H,\infty}^N$: defined in 2.5.4
$$d_{H,\infty}^N \left(\rho_N(X_\Theta(T, t_0, X_0)), X_\Theta^N(T, t_0, X_0)\right) \leq ch_N$$
- As this is a general result for differential inclusions with certain properties it also applies to the solution sets $\tilde{X}_\Theta(T, t_0, X_0)$ and $\tilde{X}_\Theta^N(T, t_0, X_0)$ defined in Definition 1. But we just need it for the solution sets to the Mayer-Problem.

Now we are ready to take a closer look at Value Convergence of the Mayer-Problem objective function. Indeed the only reason for using the Mayer-Problem formulation in this article is that with only one additional assumption we get Value Convergence for this special kind of objective function. The key components will be the Approximation Property described above, the **optimality** of the discrete solution $\hat{x}^N$ in the discrete case and the optimality of $\hat{x}(.)$ in the continuous case.

3.2.2 Theorem (Value Convergence)

Consider the objective function J of the Mayer-Problem 2.3.2 respectively 2.3.3. Let $J(.,.)$ be **Lipschitz-continous** in both arguments on $X_0 \times (S \cup \tilde{S})$ (with Lipschitz-constant L_J with respect to the supremum norm) and let all the assumptions of Theorem 3.2.1 (Approximation Property) be fulfilled. Then it holds:

$$\left| J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \right| \leq L_J \, ch_N$$

Note: *Lipschitz-continuity of $J(.,.)$ is only needed on $X_0 \times (S \cup \tilde{S})$. $S \subset \mathbb{R}^n$ is the compact set that contains all vectors $x(t)$ for $t \in [t_0, T]$ and $x(.) \in X_\Theta(T, t_0, X_0)$ (see Theorem 2.6.1). $\tilde{S}$ is the discrete counterpart to S, so it contains x_j^N for all $j \in \{0, \ldots, N\}$ and $x^N \in X_\Theta^N(T, t_0, X_0)$ for all $N \in \mathbb{N}$ (see Theorem 2.6.3).*

Proof :
We will estimate $J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))$ in both directions. To get the desired result we make use of the optimality of $\hat{x}^N$ respectively $\hat{x}(.)$, i.e.:

$$J(\hat{x}_0^N, \hat{x}_N^N) \leq J(x_0^N, x_N^N) \qquad \forall x^N \in X_\Theta^N(T, t_0, X_0)$$

respectively

$$J(\hat{x}(t_0), \hat{x}(T)) \leq J(x(t_0), x(T)) \qquad \forall x(.) \in X_\Theta(T, t_0, X_0)$$

From the Approximation Property (see 3.2.1) we get the existence of

$$\delta_N(\hat{x}(.)) \in X_\Theta^N(T, t_0, X_0) \text{ and } \pi_N(\hat{x}^N)(.) \in X_\Theta(T, t_0, X_0)$$

with

$$\|\delta_N(\hat{x}(.)) - \rho_N(\hat{x}(.))\|_\infty \leq ch_N \text{ and } \|\rho_N(\pi_N(\hat{x}^N)(.)) - \hat{x}^N\|_\infty \leq ch_N$$

Combining both properties we get:

$$\begin{aligned}
J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) &\leq J(\delta_N(\hat{x}(.))_0, \delta_N(\hat{x}(.))_N) - J(\hat{x}(t_0), \hat{x}(T)) \\
&\overset{J \text{ Lipschitz}}{\leq} L_J \left\| (\delta_N(\hat{x}(.))_0, \delta_N(\hat{x}(.))_N) - (\hat{x}(t_0), \hat{x}(T)) \right\|_\infty \\
&\leq L_J \left\| \delta_N(\hat{x}(.)) - \rho_N(\hat{x}(.)) \right\|_\infty \leq L_J \, ch_N
\end{aligned}$$

and

$$\begin{aligned}
J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) &\geq J(\hat{x}_0^N, \hat{x}_N^N) - J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) \\
&\overset{J \text{ Lipschitz}}{\geq} - L_J \left\| (\hat{x}_0^N, \hat{x}_N^N) - (\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) \right\|_\infty \\
&\geq - L_J \left\| \hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.)) \right\|_\infty \geq -L_J \, ch_N
\end{aligned}$$

which leads to

$$\left| J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \right| \leq L_J \, ch_N$$

Note: *The estimation above is only possible, because the objective function in the Mayer-Problem and the objective function in the discrete Mayer-Problem are the same. That way we could make use of the Lipschitz-continuity of J here.*

■

Now that we can make use of Value Convergence, the question of how to deduce convergence of the corresponding states $\hat{x}^N$ and $\hat{x}(.)$ arises. The answer is to use some sort of Inverse Stability Property, hence the name of the next property being introduced. Specifically we will make use of **second order sufficient optimality conditions** in this chapter. This is not the only way one could go for an inverse stability result, but maybe the best in most cases. First order sufficient optimality conditions would deliver an even better result, but unfortunately they do not apply in most cases. Again, a more general way will be explained in chapter 4. The following is a well known result from optimization theory in function spaces. We will first take a look at this form of the second order sufficient optimality conditions and then adjust it slightly according to our needs.

3.2.3 Theorem (Inverse Stability Property)

Let second order sufficient optimality conditions be fulfilled for the continuous problem. Then $\exists \tilde{\alpha} > 0$ and $\epsilon > 0$ such that

$$\tilde{\alpha} \left(\|x(.) - \hat{x}(.)\|_2^2 + \|u(.) - \hat{u}(.)\|_2^2 \right) = \tilde{\alpha} \, \|(x(.), u(.)) - (\hat{x}(.), \hat{u}(.))\|_2^2 \leq J(x(t_0), x(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

for all **admissible** pairs $(x(.), u(.))$ with $\|u(.) - \hat{u}(.)\|_\infty \leq \epsilon$ and $\|x(.) - \hat{x}(.)\|_\infty \leq \epsilon$

Remarks

- *The main part in second order sufficient optimality conditions is that $L''(\hat{x}, \hat{u})((v, w), (v, w)) \geq \beta\|(v, w)\|_2^2$ has to be fulfilled for all $(v, w) \in L(\Sigma, (\hat{x}, \hat{u}))$. Notations: L is the Lagragian, L'' its second Fréchet and $L(\Sigma, (\hat{x}, \hat{u}))$ the so called linearizing cone of the feasible set Σ. For further information see [1].*

- *The whole chapter deals only with Mayer-Problems, and indeed the above inequality is needed for the extended state $x(.)$. For verifying sufficient optimality conditions it might be advantageous to stick with the Bolza-Problem and show that the sufficient conditions hold. If the integrand of the objective function $f(.,.,.)$ is Lipschitz-continuous in all of its arguments on the feasible set, the desired inverse stability property for the extended state can be obtained from the inverse stability property delivered by analyzing the Bolza-Problem. Note that the Lipschitz-continuity of $f(.,.,.)$ follows directly from the Lipschitz-continuity of $\psi(.,.,.)$, which will be postulated in Theorem 3.2.4 anyway. So needing the Lipschitz-continuity of $f(.,.,.)$ is no additional restriction. For details see 6.1.1 in the examples chapter. For an example of how to verify second order optimality conditions see section 6.2.2 about applying the Convergence Theorem to Example 6.2.*

- *This result might be obtained for another norm than the L_2-norm, which would be fine. But the L_2-norm is the natural norm for proving that second order sufficient optimality conditions are fulfilled. This has to do with the fact, that the second Fréchet derivative of the Lagrangian is a bilinear form.*

- *Consider the different kind of norms used here. The phenomenon appearing here is the so called **Two-Norm-Discrepancy** (in our case L_2- and L_∞-norm). Also norms might be altered to some extent in the above statement, they may not be easily exchanged by each other. The reason for that has to do with existence of Fréchet derivatives with respect to certain norms. For example, this result is only valid on an ϵ-ball with respect to the L_∞-norm with center $(\hat{x}(.), \hat{u}(.))$. It would be much more preferable to have that result with respect to an L_p-norm $(1 \leq p < \infty)$ than to the L_∞-norm. This would widen the range of pairs $(x(.), u(.))$ for which the above inequality holds. But to the author's knowledge such a result has not been proven yet. For further details on second order optimality conditions in function spaces and the Two-Norm-Discrepancy see [1].*

Remember the final result $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq Ch_N^{1/2}$ we want to prove in this chapter. This result states convergence in the discrete L_∞-norm. When considering the above second order optimality condition 3.2.3, at first sight one might think that the result is too weak to use it to prove convergence of the state in the discrete L_∞-norm. And indeed, just looking at the inequality given in 3.2.3 supposes only convergence in the discrete L_2-norm. This is because the left-hand side only delivers convergence in the

L_2-norm, which is in general weaker than the L_∞-norm, and the L_2-norm is related to the discrete L_2-norm, not the discrete L_∞-norm (see 2.5.2). Also we haven't yet made any connection between the optimal discrete solution $\hat{x}^N$ and the optimal solution for the continuous case $\hat{x}(.)$, a gut feeling or maybe a look in the overview section of this chapter should tell us that we can't proceed directly with the result given above if we actually want to obtain the convergence result with respect to the L_∞-norm. But fortunately the fact that $\dot{x}(t) = \psi(t, x(t), u(t))$ *a.e.* for feasible states of our problem 2.3.2, i.e. $x(.) \in X_\Theta(T, t_0, X_0)$, will help us out. With the additional assumption of Lipschitz-continuity of ψ in all of its arguments, we will be able to estimate $\|x(.) - \hat{x}(.)\|_\infty$ by $\|(x(.), u(.)) - (\hat{x}(.), \hat{u}(.))\|_2$ (see the proof of the following Theorem), which leads us to the stronger result we need.

3.2.4 Theorem (Adjusted Inverse Stability Property)

Let all the assumptions of 3.2.3 be fulfilled, let $\dot{x}(t) = \psi(t, x(t), u(t))$ *a.e.* and $\dot{\hat{x}}(t) = \psi(t, \hat{x}(t), \hat{u}(t))$ *a.e.* and let $\psi(.,.,.)$ be **Lipschitz-continuous** in all of its arguments on $[t_0, T] \times S \times U$ (with Lipschitz-constant L_ψ with respect to the supremum norm).

Note: *Lipschitz-continuity of $\psi(.,.,.)$ is only needed on $[t_0, T] \times S \times U$. $U \subset \mathbb{R}^m$ has to be chosen in such a way that $u(t) \in U$ for all admissible controls $u(.)$. $S \subset \mathbb{R}^n$ is the compact set that contains all vectors $x(t)$ for $t \in [t_0, T]$ and $x(.) \in X_\Theta(T, t_0, X_0)$ (see Theorem 2.6.1).*

Then with $\tilde{\alpha}$ and ϵ from 3.2.3 it holds:

$$\alpha \left(\|x(.) - \hat{x}(.)\|_\infty + \|u(.) - \hat{u}(.)\|_2\right)^2 \le J(x(t_0), x(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

with

$$\alpha = \frac{\tilde{\alpha}}{\left(\sqrt{\frac{1}{T - t_0}} + 2L_\psi\sqrt{T - t_0} + 1\right)^2}$$

for all **admissible** pairs $(x(.), u(.))$ with $\|u(.) - \hat{u}(.)\|_\infty \le \epsilon$ and $\|x(.) - \hat{x}(.)\|_\infty \le \epsilon$

Note: *The Lipschitz-continuity of ψ stands in close relation to the assumption (A3) for the Approximation Property (see chapter 5). So this is not much of an additional requirement.*

Proof :
Our main goal is to estimate $\|x(.) - \hat{x}(.)\|_\infty$ by $\|x(.) - \hat{x}(.)\|_2$ and $\|u(.) - \hat{u}(.)\|_2$. The result we want looks like $\|x(.) - \hat{x}(.)\|_\infty \le C\|(x(.), u(.)) - (\hat{x}(.), \hat{u}(.))\|_2$ with some constant C.
Let $t \in I$, then:

①

$$\begin{aligned} \|x(t) - \hat{x}(t)\|_\infty &= \|x(t_0) + \int_{t_0}^{t} \psi(\tau, x(\tau), u(\tau))\, d\tau - \hat{x}(t_0) - \int_{t_0}^{t} \psi(\tau, \hat{x}(\tau), \hat{u}(\tau))\, d\tau\|_\infty \\ &\le \underbrace{\|x(t_0) - \hat{x}(t_0)\|_\infty}_{②} + \underbrace{\|\int_{t_0}^{t} \psi(\tau, x(\tau), u(\tau)) - \psi(\tau, \hat{x}(\tau), \hat{u}(\tau))\, d\tau\|_\infty}_{③} \end{aligned}$$

We start off with estimating the second term (③). The result we will get will also prove useful to estimate the first term (②).

③

$$\begin{aligned}
&\|\int_{t_0}^{t} \psi(\tau,x(\tau),u(\tau)) - \psi(\tau,\hat{x}(\tau),\hat{u}(\tau))\ d\tau\|_\infty \leq \int_{t_0}^{t} \|\psi(\tau,x(\tau),u(\tau)) - \psi(\tau,\hat{x}(\tau),\hat{u}(\tau))\|_\infty\ d\tau \\
&\overset{t\in I}{\leq} L_\psi \int_{t_0}^{T} \|(x(\tau),u(\tau)) - (\hat{x}(\tau),\hat{u}(\tau))\|_\infty \overset{\psi\ \text{Lipschitz}}{\leq} L_\psi \int_{t_0}^{T} \|(x(\tau),u(\tau)) - (\hat{x}(\tau),\hat{u}(\tau))\|_\infty\ d\tau \\
&\overset{2.5.2.1.}{\leq} L_\psi \int_{t_0}^{T} \|(x(\tau),u(\tau)) - (\hat{x}(\tau),\hat{u}(\tau))\|_2\ d\tau \\
&\overset{\text{Hölder}}{\leq} L_\psi \left(\int_{t_0}^{T} \|(x(\tau),u(\tau)) - (\hat{x}(\tau),\hat{u}(\tau))\|_2^2\ d\tau\right)^{1/2} \left(\int_{t_0}^{T} 1\ d\tau\right)^{1/2} \\
&= L_\psi \sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2
\end{aligned}$$

Summary:

$$\|\int_{t_0}^{t} \psi(\tau,x(\tau),u(\tau)) - \psi(\tau,\hat{x}(\tau),\hat{u}(\tau))\ d\tau\|_\infty \leq L_\psi\sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2$$

We are now equipped to estimate ②. We start with the same equality used at the beginning of the prove, but as we want to estimate ② we use the triangle inequality of the norm in the other direction. So we get for all $t \in I$:

$$\begin{aligned}
\|x(t)-\hat{x}(t)\|_\infty &= \|x(t_0) + \int_{t_0}^{t} \psi(\tau,x(\tau),u(\tau))\ d\tau - \hat{x}(t_0) - \int_{t_0}^{t} \psi(\tau,\hat{x}(\tau),\hat{u}(\tau))\ d\tau\|_\infty \\
&\geq \|x(t_0)-\hat{x}(t_0)\|_\infty - \|\int_{t_0}^{t} \psi(\tau,x(\tau),u(\tau)) - \psi(\tau,\hat{x}(\tau),\hat{u}(\tau))\ d\tau\|_\infty \\
&\overset{③}{\geq} \|x(t_0)-\hat{x}(t_0)\|_\infty - L_\psi\sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2
\end{aligned}$$

Using this result and taking a closer look at $\|x(.)-\hat{x}(.)\|_2$ yields:

②

$$\begin{aligned}
\|x(.)-\hat{x}(.)\|_2 &= \left(\int_{t_0}^{T} \|x(\tau)-\hat{x}(\tau)\|_2^2\ d\tau\right)^{1/2} \geq \left(\int_{t_0}^{T} \|x(\tau)-\hat{x}(\tau)\|_\infty^2\ d\tau\right)^{1/2} \\
&\geq \Big(\|x(t_0)-\hat{x}(t_0)\|_\infty - L_\psi\sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2\Big) \left(\int_{t_0}^{T} 1\ d\tau\right)^{1/2} \\
&= \sqrt{T-t_0}\ \Big(\|x(t_0)-\hat{x}(t_0)\|_\infty - L_\psi\sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2\Big)
\end{aligned}$$

So we get the following estimation for $\|x(t_0)-\hat{x}(t_0)\|_\infty$:

$$\|x(t_0)-\hat{x}(t_0)\|_\infty \leq \sqrt{\frac{1}{T-t_0}}\ \|x(.)-\hat{x}(.)\|_2 + L_\psi\sqrt{T-t_0}\ \|(x(.),u(.)) - (\hat{x}(.),\hat{u}(.))\|_2$$

Turning back to ① and estimating further with the results of ② and ③ yields for all $t \in I$:

$$\textcircled{1}\quad \begin{aligned} \|x(t)-\hat{x}(t)\|_\infty &\leq \|x(t_0)-\hat{x}(t_0)\|_\infty + \|\int_{t_0}^{t} \psi(\tau, x(\tau), u(\tau)) - \psi(\tau, \hat{x}(\tau), \hat{u}(\tau))\, d\tau\|_\infty \\ &\overset{\textcircled{2},\textcircled{3}}{\leq} \sqrt{\frac{1}{T-t_0}}\, \|x(.)-\hat{x}(.)\|_2 + 2L_\psi\sqrt{T-t_0}\, \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2 \end{aligned}$$

Once again, this result is true for all $t \in I$, which gives us:

$$\textcircled{4}\quad \|x(.)-\hat{x}(.)\|_\infty \leq \sqrt{\frac{1}{T-t_0}}\, \|x(.)-\hat{x}(.)\|_2 + 2L_\psi\sqrt{T-t_0}\, \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2$$

We now have the estimation we wanted for $\|x(.)-\hat{x}(.)\|_\infty$. To obtain the result stated in this theorem we just need to take a look at $(\|x(.)-\hat{x}(.)\|_\infty + \|u(.)-\hat{u}(.)\|_2)^2$, apply the above result, estimate a little further and involve $\tilde{\alpha}$ to use Theorem 3.2.3.

Note: *We will make use of the trivial estimates* $\|u(.)-\hat{u}(.)\|_2 \leq \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2$ *and* $\|x(.)-\hat{x}(.)\|_2 \leq \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2$.

$$\begin{aligned} &(\|x(.)-\hat{x}(.)\|_\infty + \|u(.)-\hat{u}(.)\|_2)^2 \overset{\textcircled{4}}{\leq} \\ &\left(\sqrt{\frac{1}{T-t_0}}\, \|x(.)-\hat{x}(.)\|_2 + 2L_\psi\sqrt{T-t_0}\, \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2 + \|u(.)-\hat{u}(.)\|_2\right)^2 \leq \\ &\left(\sqrt{\frac{1}{T-t_0}} + 2L_\psi\sqrt{T-t_0} + 1\right)^2 \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2^2 \end{aligned}$$

By involving $\tilde{\alpha}$ from Theorem 3.2.3 this leads to:

$$\begin{aligned} &\underbrace{\frac{\tilde{\alpha}}{\left(\sqrt{\frac{1}{T-t_0}} + 2L_\psi\sqrt{T-t_0} + 1\right)^2}}_{\alpha :=} (\|x(.)-\hat{x}(.)\|_\infty + \|u(.)-\hat{u}(.)\|_2)^2 \\ &\leq \tilde{\alpha}\, \|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2^2 \overset{\text{Theorem 3.2.3}}{\leq} J(x(t_0), x(T)) - J(\hat{x}(t_0), \hat{x}(T)) \end{aligned}$$

■

The result from Theorem 3.2.4 definitely brings us one step closer to the final result of this chapter. Now we would like to combine the Adjusted Inverse Stability Property with Value Convergence, which we have already proven. Again, taking a closer look yields that this is not directly possible. This is because on the left hand side of the inequality in Theorem 3.2.4 we have to deal with functions from the continuous problem. There is no way to replace $x(.)$ by $\hat{x}^N$ in that equality, but we can substitute $x(.)$ with a function that is close to x^N on the grid $\mathbb{G}_N$. And this is the point, where the

Approximation Property appears for the second time (the first time was in the proof of Value Convergence). Recall the name of the function we are looking for is $\pi_N(\hat{x}^N)(.)$ and it holds $\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty \leq ch_N$. With that estimation and the Adjusted Inverse Stability Property (that we can make use of with $\pi_N(\hat{x}^N)(.)$), there is hope that we can combine both estimations to gain the final result. Indeed this will be possible, but to do so we have to overcome the obstacle that one estimate involves the discrete L_∞-norm and the other one the L_∞-norm. But more on that later. Let's deal with fitting $\pi_N(\hat{x}^N)(.)$ into Theorem 3.2.4 first. The only problem with this is that we need linear convergence for the difference of the objective functions on the right-hand side of the inequality. But this shouldn't be hard to prove, because we can make use of Value Convergence, the Approximation Property and Lipschitz-continuity of the objective function J (see prove of the following Corollary). So all this considerations lead us to the important result below.

3.2.5 Corollary to Theorem 3.2.4 (Applied Inverse Stability Property)

Let's consider the Mayer-Problem (2.3.2) and the discrete Mayer-Problem(2.4.2) again. Let all the assumptions of Theorem 3.2.1 (Approximation Property), Theorem 3.2.2 (Value Convergence) and Theorem 3.2.4 (Adjusted Inverse Stability Property) be fulfilled. Then with α and ϵ from 3.2.4 it holds:

$$\begin{aligned}\alpha\left(\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty + \|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)\|_2\right)^2 &\leq J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))\\ &\leq 2L_J\, ch_N\end{aligned}$$

as soon as $\|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)\|_\infty \leq \epsilon$ and $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \epsilon$.

This implies for $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$:

$$\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty + \|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)\|_2 \leq \sqrt{\frac{2L_J\, ch_N}{\alpha}}$$

as soon as $\|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)\|_\infty \leq \epsilon$ and $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \epsilon$

Remarks:

- $\pi_N(\hat{x}^N)(.)$ is a feasible state for the continuous problem according to the Approximation Property and $\tilde{\pi}_N(\hat{u}^N)(.)$ denotes a corresponding feasible control.

- The additional condition inherited from Theorem 3.2.4 $\|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{x}(.)\|_\infty \leq \epsilon$ and $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \epsilon$ is unfortunately present. ϵ depends on the specific problem, so this leads to an additional assumption one has to make. As already mentioned in a note for Theorem 3.2.3 it would be desirable to only have to make that assumption with respect to an L_p-norm ($1 \leq p < \infty$). There are investigations using an $L_{2+\beta}$-norm (with $\beta > 0$) going on, but these are not part of this thesis.

Proof :
The only thing that needs to be proven is that

$$J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) \leq 2L_J\, ch_N$$

This is a direct consequence of the Approximation Property (Theorem 3.2.1), Value Convergence (Theorem 3.2.2) and the Lipschitz-continuity of J postulated in Theorem 3.2.2:

$$\begin{aligned} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) = & \underbrace{J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}_0^N, \hat{x}_N^N)}_{\text{①}} \\ & + \underbrace{J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))}_{\text{②}} \end{aligned}$$

The first term (①) can be estimated using the Lipschitz-continuity and the Approximation Property:

①

$$\begin{aligned} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}_0^N, \hat{x}_N^N) & \leq L_J\, \|(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - (\hat{x}_0^N, \hat{x}_N^N)\|_\infty \\ & \leq L_J\, \|\rho_N(\pi_N(\hat{x}^N)(.)) - \hat{x}^N\|_\infty \leq L_J\, ch_N \end{aligned}$$

The second term (②) can be estimated directly using the Value Convergence Theorem:

②

$$|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))| \leq L_J\, ch_N \Rightarrow J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \leq L_J\, ch_N$$

Due to the fact that $\pi_N(\hat{x}^N)(.) \in X_\Theta(T, t_0, X_0)$ is a feasible solution of the Mayer-Problem 2.3.2 and $\hat{x}(.) \in X_\Theta(T, t_0, X_0)$, it holds:

$$J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) \geq J(\hat{x}(t_0), \hat{x}(T)) \Leftrightarrow 0 \leq J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

This, of course, follows directly from the second order sufficient optimality conditions, too.

So overall we get:

$$\begin{aligned} 0 \leq J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) & = J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}_0^N, \hat{x}_N^N) \\ & + J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \overset{\text{①,②}}{\leq} L_J\, ch_N + L_J\, ch_N = 2L_J\, ch_N \end{aligned}$$

$\blacksquare$

So let's sum things up. We are now left with two major results. The first one is the Approximation Property result $\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty \leq ch_N$. The second one is the result from the corollary above, i.e. $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \sqrt{\frac{2L_J\, ch_N}{\alpha}}$. The goal now is to combine those results to gain $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq Ch_N$ with some constant C independent of N. The only problem doing so is that we have to deal with the discrete L_∞-norm in $\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty$ and the L_∞-norm in $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$. To be able to combine them, the idea is to prove that $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$ is close enough to $\|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty$ to be able to work with $\|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty$

instead of $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$. We are now going to introduce that result, which will be called Compatibility Property, in its general form and show afterwards, that it applies to the function $(\pi_N(\hat{x}^N)(.) - \hat{x}(.))$. It's then trivial to apply the triangle inequality to $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$ and then use the two major results wrapped up before.

3.2.6 Theorem (Compatibility Property)

Let $f(.) : I \to \mathbb{R}^k$ $(k \in \mathbb{N})$ be **Lipschitz-continuous** (with Lipschitz-constant L_f with respect to the supremum norm). Then it holds:

$$\Big| \|\rho_N(f(.))\|_\infty - \|f(.)\|_\infty \Big| \leq L_f h_N$$

Proof :
As $f(.)$ is continuous on I there exits $\tilde{t} \in I$ such that $\|f(.)\|_\infty = |f_i(\tilde{t})|$ for some $i \in \{1, \ldots, k\}$ (see definition of the $\|.\|_\infty$-norm for $f(.) \in C(I)^k$). With $\tilde{t} \in I$ there exists $j \in \{0, \ldots, N-1\}$ such that $\tilde{t} \in [t_j, t_{j+1}]$. The Lipschitz-continuity of $f(.)$ then delivers:

①

$$|f_i(\tilde{t})| - |f_i(t_j)| \leq |f_i(\tilde{t}) - f_i(t_j)| \leq \|f(\tilde{t}) - f(t_j)\|_\infty \leq L_f |\tilde{t} - t_j| \leq L_f h_N$$

From the definition of the discrete L_∞-norm and the ordinary restriction operator to the grid ρ_N, it directly follows that:

②

$$\|\rho_N(f(.))\|_\infty = \sup_{j=0,\ldots,N} \|f(t_j)\|_\infty = \sup_{j=0,\ldots,N} \left(\sup_{l=1,\ldots,k} |f_l(t_j)| \right) \geq |f_i(t_j)|$$

Combining ① and ② leads to:

$$\|f(.)\|_\infty - \|\rho_N(f(.))\|_\infty = |f_i(\tilde{t})| - \|\rho_N(f(.))\|_\infty \overset{\text{②}}{\leq} |f_i(\tilde{t})| - |f_i(t_j)| \overset{\text{①}}{\leq} L_f h_N$$

Due to $\{t_0, \ldots, t_N\} \subset I$ we get by considering the norm definitions that

$$\|f(.)\|_\infty - \|\rho_N(f(.))\|_\infty \geq 0$$

So overall we have:

$$0 \leq \|f(.)\|_\infty - \|\rho_N(f(.))\|_\infty \leq L_f h_N$$

■

Let's apply the result of the compatibility property to the function $(\pi_N(\hat{x}^N)(.) - \hat{x}(.))$.

3.2.7 Corollary to Theorem 3.2.6 (Applied Compatibility Property)

Let assumption (A1) from section 2.6 be fulfilled, which leads to the fact that Theorem 2.6.2 (Uniform Lipschitz Continuity) and Theorem 2.6.4 (Discrete Uniform Lipschitz Continuity) hold.

Then

$$\Big| \|\rho_N \pi_N(\hat{x}^N)(.) - \rho_N \hat{x}(.)\|_\infty - \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \Big| \leq L_{\Delta x} h_N$$

with

$$L_{\Delta x} = 2L \qquad\qquad \text{(with } L \text{ from section 2.6)}$$

Proof :
This is actually pretty simple, because the hard work already has been done in Theorem 2.6.2 (Uniform Lipschitz Continuity) and Theorem 3.2.6 (Compatibility Property). So we just need to show that $(\pi_N(\hat{x}^N)(.) - \hat{x}(.))$ is Lipschitz continuous with Lipschitz constant $2L$.
Let $t, \tilde{t} \in I$, then:

$$\| \Big(\pi_N(\hat{x}^N)(t) - \hat{x}(t)\Big) - \Big(\pi_N(\hat{x}^N)(\tilde{t}) - \hat{x}(\tilde{t})\Big) \|_\infty \leq \|\pi_N(\hat{x}^N)(t) - \pi_N(\hat{x}^N)(\tilde{t})\|_\infty + \|\hat{x}(t) - \hat{x}(\tilde{t})\|_\infty$$
$$\overset{2.5.2.1}{\leq} \|\pi_N(\hat{x}^N)(t) - \pi_N(\hat{x}^N)(\tilde{t})\|_2 + \|\hat{x}(t) - \hat{x}(\tilde{t})\|_2 \overset{\text{Theorem 2.6.2}}{\leq} L|t - \tilde{t}| + L|t - \tilde{t}| = 2L|t - \tilde{t}|$$

We can now replace $f(.)$ in Theorem 3.2.6 with $(\pi_N(\hat{x}^N)(.) - \hat{x}(.))$ and are done. ■

Finally we have reached the point, where we have all major results ready to prove the main result of this chapter, i.e. $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq C h_N^{1/2}$. So let's wrap things up:
The initial idea was to use Value Convergence (Theorem 3.2.2) in conjunction with Inverse Stability (Theorem 3.2.3) to deduce convergence of the corresponding states from convergence of the objective function values. This did not lead to the final result, but brought us to Corollary 3.2.5, which is pretty close. To get a connection to the discrete norm, Corollary 3.2.7 came in handy. And finally one can make use of the Approximation Property (Theorem 3.2.1) to fill in the gap between $\hat{x}^N$ and $\rho_N(\pi_N(\hat{x}^N)(.))$. This describes the whole estimation process from right to left, but it is not convenient to write things down that way. It is way better to go from left to right. So we start directly with estimating $\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$ while bearing in mind, that we ultimately want to apply the result of Corollary 3.2.5, i.e. $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \sqrt{\frac{2L_J\, ch_N}{\alpha}}$.

The first step is getting $\pi_N(\hat{x}^N)(.)$ into play:

$$① \quad \|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty$$

The first term can be estimated by using the Approximation Property (Theorem 3.2.1):

②
$$\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty \le ch_N$$

The second term $\|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty$ is close to $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$, which is guaranteed by Corollary 3.2.7 (Applied Compatibility Property):

③
$$\begin{aligned} &\Big| \|\rho_N \pi_N(\hat{x}^N)(.) - \rho_N \hat{x}(.)\|_\infty - \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \Big| \le 2L\, h_N \\ \Rightarrow\ &\|\rho_N \pi_N(\hat{x}^N)(.) - \rho_N \hat{x}(.)\|_\infty \le 2L\, h_N + \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \end{aligned}$$

So finally the term $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty$ occurs, for which we have from Corollary 3.2.5 (Applied Inverse Stability):

④
$$\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \le \sqrt{\frac{2L_J\, ch_N}{\alpha}}$$

Combining ② and ③ with ① and finally using ④ delivers:

$$\begin{aligned} \|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty &\overset{\text{①}}{\le} \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty \\ &\overset{\text{②}}{\le} ch_N + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|_\infty \overset{\text{③}}{\le} ch_N + 2L\, h_N + \|\pi_N(\hat{x}^N)(.)) - \hat{x}(.)\|_\infty \\ &\overset{\text{④}}{\le} (c + 2L)h_N + \sqrt{\frac{2L_J\, c}{\alpha}}\sqrt{h_N} \overset{h_N \le 1}{\le} \underbrace{\left(c + 2L + \sqrt{\frac{2L_J\, c}{\alpha}}\right)}_{C:=} \sqrt{h_N} \end{aligned}$$

Note: *In the last step we assumed $h_N \le 1$ so that $h_N \le \sqrt{h_N}$. So the estimation above holds for $N \ge T - t_0$.*

So overall we have proven the following major result.

3.2.8 Convergence Theorem

Let all the assumptions from chapter 5 (X_0 bounded, (A1), (A2), (A3), (C1) and (C2)) be satisfied. This leads to the fact, that the Approximation Property 3.2.1 holds with constant c. From (A1) we also get that the Applied Compatibility Property (Corollary 3.2.7) can be used with constant $L_{\Delta x} = 2L$. Also let the objective function $J(.,.)$ be Lipschitz-continuous on $X_0 \times (S \cup \tilde{S})$ in both arguments with Lipschitz constant L_J. This leads to Value Convergence (Theorem 3.2.2). Furthermore, let second order sufficient optimality conditions be fulfilled (see Theorem 3.2.3) and let the right-hand side of the ODE $\psi(.,.,.)$ be Lipschitz-continuous in all of its arguments on the set $[t_0, T] \times S \times U$ with Lipschitz constant L_ψ. This implies that the Adjusted Inverse Stability Property (Theorem 3.2.4) applies with constant α.

With all these assumptions it then holds:

$$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty \leq \underbrace{\left(c + 2L + \sqrt{\frac{2L_J\,c}{\alpha}}\right)}_{C:=} \sqrt{h_N}$$

Remarks:

- *This result only holds for sure as soon as* $\|\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)\|_\infty \leq \epsilon$ *and* $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|_\infty \leq \epsilon$*. For details on that matter and the constant* ϵ *see Theorem 3.2.3 (Inverse Stability Property) and Corollary 3.2.5 (Applied Inverse Stability Property).*
- *Lipschitz-continuity of* $J(.,.)$ *is only needed on* $X_0 \times (S \cup \tilde{S})$*.* $S \subset \mathbb{R}^n$ *is the compact set that contains all vectors* $x(t)$ *for* $t \in [t_0, T]$ *and* $x(.) \in X_\Theta(T, t_0, X_0)$ *(see Theorem 2.6.1).* $\tilde{S}$ *is the discrete counterpart to S, so it contains* x_j^N *for all* $j \in \{0, \dots, N\}$ *and* $x^N \in X_\Theta^N(T, t_0, X_0)$ *(see Theorem 2.6.3).*
- *Lipschitz-continuity of* $\psi(.,.,.)$ *is only needed on* $[t_0, T] \times S \times U$*.* $U \subset \mathbb{R}^m$ *has to be chosen in such a way that* $u(t) \in U$ *for all admissible controls* $u(.)$*.*

As this chapter presented the whole concept for a specific class of optimal control problems, the reader should be somehow familiar with the techniques used. This should make following along the more abstract view presented in the next chapter easier.

4 Convergence Theorem (General Approach)

4.1 Overview

This chapter presents the estimation process of chapter 3 in a more general, hence problem independent, way. We are no longer talking directly about the Mayer-Problem presented in 2.3.2 here. Chapter 3 has been presented before this one, because in the author's opinion the core concepts are easier to understand when working with a specific problem class, which makes it possible to obtain distinct results. In the more general approach we will just be able to make the assumptions for achieving certain goals, but we won't be able to prove that they are fulfilled without considering a specific problem class. This chapter is only about presenting certain modules, usually in a weaker form than in chapter 3, that will lead to a convergence result. Again, whether these modules can be applied or not depends on the specific problem considered. The essential modules are the Approximation Property, the Inverse Stability Property and the Compatibility Property. In the previous chapter these were theorems we were able to prove under certain conditions. Now these are modules making weaker statements, which should make it possible to apply them to a wider range of problem classes. Of course these weaker statements will lead to weaker results. Especially assumptions on convergence will be weaker than the results obtained in chapter 3. In general we will just presume convergence with respect to a certain norm, whereas we were able to obtain linear convergence for most modules when considering the specific problem 2.3.2.
The problems we are talking about in this chapter will just be called **continuous problem** and **discrete problem**. To get an idea of what those problems may look like, the reader is strongly advised to take a look at the continuous Mayer-Problem 2.3.2 and its directly discretized counterpart, the discrete Mayer-Problem 2.4.2.
This time we will gain a weaker result, which is

$$\boldsymbol{\lim_{N\to\infty} \|\hat{x}^N - \rho_N(\hat{x}(.))\| = 0}$$

where $\hat{x}(.)$ is the optimal solution to the continuous problem, $\hat{x}^N$ the optimal solution to the discrete problem and $\|.\|$ an appropriate discrete norm. Choice of this norm heavily depends on the Inverse Stability Property and the Compatibility Property available.

In general all notations will stay the same, except the following ones.

Notations:

As we are talking about general optimal control problems in this chapter it makes sense not to use the same notation for solution sets, although notation will be pretty similar:

X: Set of all feasible solutions to the continuous problem.
X^N: Set of all feasible solutions to the discrete problem.

Note that we are not distinguishing between pure state constraints included or not (which was denoted by a subscript Θ) in this chapter. This makes sense, because there is no specific problem considered here.

$\|.\|$: Appropriate norm. Depends on the Inverse Stability Property and the Compatibility Property available. The corresponding (relating to the Compatibility Property) discrete norm will be denoted the same way. The arguments make clear which norm is meant.

$|||.|||$: Another appropriate norm. Usually stronger than $\|.\|$.
For example $\|.\|$ might be the L_2-norm and $|||.|||$ the L_∞-norm.

Once again, the following section essentially presents a similar estimation process as section 3.2 in the previous chapter. All the ideas are exactly the same. This chapter builds upon the previous one so the basic ideas will be explained way shorter. The reader is strongly advised to read Chapter 3 first.

4.2 Estimation Process (General Form)

The main idea of the whole concept is making use of Value Convergence of the objective function $J(.,.)$, which shall look like the objective function of the Mayer-Problem presented in the last chapter. It is essential that the objective function does not change when switching from the continuous to the discrete case. This is because both functions need to be comparable. In fact, it would be possible to consider an objective function that depends on the state at an arbitrary number of time points, but it is convenient to let J just depend on the state at t_0 and T respectively t_0 and t_N (the first and the last time point). As shown in the previous chapter this can be easily achieved for objective functions with integral term by using the Mayer formulation.
To obtain Value Convergence we need some sort of Approximation Property, which we have to premise here. It will be postulated in a weaker form than in 3.2.1 so that it is more likely to be fulfilled. For obtaining the result in 3.2.1 the optimal control problem had to fit quite a lot of needs (see chapter 5).

4.2.1 Approximation Property (General Form)

Recall the Notation about solution sets in this chapter. Let $x(.) \in X$, $x^N \in X^N$. Then $\exists \tilde{N} \in \mathbb{N}$, such that for any $N \in \mathbb{N}$ with $N \geq \tilde{N}$ there exist functions $\pi_N : X^N \to X$ and $\delta_N : X \to X^N$, such that:

$$\begin{aligned} \lim_{N\to\infty} \|\rho_N(\pi_N(x^N)(.)) - x^N\|_\infty &= 0 \\ \lim_{N\to\infty} \|\delta_N(x(.)) - \rho_N(x(.))\|_\infty &= 0 \end{aligned}$$

Remarks:

- Note the image sets of the functions. This means, that $\pi_N(x^N)(.)$ is a feasible solution to the continuous problem and $\delta_N(x(.))$ a feasible solution to the discrete problem.
- Unfortunately this proposition has to be made with respect to the $\|.\|_\infty$-norm to obtain Value Convergence of the objective function.
- An alternative description of the Approximation Property without introducing the functions π_N and δ_N would be

  $$\lim_{N\to\infty} d_{H,\infty}\Big(\rho_N(X), X^N\Big) = 0$$

  where $d_{H,\infty}$ is the applied Hausdorff-distance defined in 2.5.4.

The Approximation Property helps us gain Value Convergence.

4.2.2 Theorem (Value Convergence, General Form)

Consider the objective function $J(.,.)$: as described before. Let $\boldsymbol{J(.,.)}$ **be continuous** and let the Approximation Property 4.2.1 be fulfilled. Then it holds:

$$\lim_{N\to\infty} J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) = 0$$

Proof :
We will estimate $J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))$ in both directions. To get the desired result we make use of the optimality of $\hat{x}^N$ respectively $\hat{x}(.)$, i.e.:

$$J(\hat{x}_0^N, \hat{x}_N^N) \leq J(x_0^N, x_N^N) \qquad \forall x^N \in X^N$$

respectively

$$J(\hat{x}(t_0), \hat{x}(T)) \leq J(x(t_0), x(T)) \qquad \forall x(.) \in X$$

From the Approximation Property 4.2.1 we get the existence of $\delta_N(\hat{x}(.)) \in X^N$ and $\pi_N(\hat{x}^N)(.) \in X$ with $\lim\limits_{N\to\infty} \|\rho_N(\pi_N(x^N)(.)) - x^N\|_\infty = 0$ and $\lim\limits_{N\to\infty} \|\delta_N(x(.)) - \rho_N(x(.))\|_\infty = 0$. Together with the optimality, this will deliver the statement of this theorem.

For the first direction we get:

$$J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \leq J(\delta_N(\hat{x}(.))_0, \delta_N(\hat{x}(.))_N) - J(\hat{x}(t_0), \hat{x}(T))$$

From the Approximation Property 4.2.1 we get:

$$\|(\delta_N(\hat{x}(.))_0, \delta_N(\hat{x}(.))_N) - (\hat{x}(t_0), \hat{x}(T))\|_\infty \leq \lim_{N\to\infty} \|\delta_N(\hat{x}(.)) - \rho_N(\hat{x}(.))\|_\infty \xrightarrow[4.2.1]{N\to\infty} 0$$

① With $J(.,.)$ being continuous this leads to

$$J(\delta_N(\hat{x}(.))_0, \delta_N(\hat{x}(.))_N) - J(\hat{x}(t_0), \hat{x}(T)) \to 0 \quad (N \to \infty)$$

So taking the limit in the inequality for the first direction delivers:

$$\limsup_{N\to\infty} J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \leq 0$$

And for the second direction:

$$J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \geq J(\hat{x}_0^N, \hat{x}_N^N) - J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T))$$

From the Approximation Property 4.2.1 we get:

$$\|(\hat{x}_0^N, \hat{x}_N^N) - (\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T))\|_\infty \leq \lim_{N\to\infty} \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty \xrightarrow[4.2.1]{N\to\infty} 0$$

② With $J(.,.)$ being continuous this leads to

$$J(\hat{x}_0^N, \hat{x}_N^N) - J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) \to 0 \quad (N \to \infty)$$

So taking the limit in the inequality for the second direction delivers:

$$\liminf_{N\to\infty} J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \geq 0$$

Combining both directions ① and ② leads to:

$$\lim_{N\to\infty} J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) = 0$$

Note: *The estimation above is only possible, because the objective function in the continuous problem and the objective function in the discrete problem are the same. That way we could make use of the continuity of J here.*

■

Again we want to deduce convergence of the corresponding states from Value Convergence. To do so we need some sort of Inverse Stability Property. With results from optimization theory in mind, like second order sufficient optimality conditions presented in 3.2.3, it makes sense to take a look at $J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))$ first. With Theorem 4.2.2 we get the following corollary.

4.2.3 Corollary to Theorem 4.2.2 (Extended Value Convergence)

Let $\boldsymbol{J(.,.)}$ **be continuous** and let the Approximation Property 4.2.1 be fulfilled. Then it holds:

$$\lim_{N\to\infty} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) = 0$$

Proof :
This is a direct consequence of the Approximation Property (Theorem 4.2.1), Value Convergence (Theorem 4.2.2) and the continuity of J postulated in Theorem 4.2.2:

$$\begin{aligned} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) = & \underbrace{J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}_0^N, \hat{x}_N^N)}_{\textcircled{1}} \\ & + \underbrace{J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))}_{\textcircled{2}} \end{aligned}$$

For the first term (①) we make use of the Approximaiton Property and the continuity of J.

① From the Approximation Property 4.2.1 we get:

$$\|(\hat{x}_0^N, \hat{x}_N^N) - (\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T))\|_\infty \leq \lim_{N\to\infty} \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|_\infty \xrightarrow[4.2.1]{N\to\infty} 0$$

With $J(.,.)$ being continuous this leads to

$$J(\hat{x}_0^N, \hat{x}_N^N) - J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) \to 0 \qquad (N \to \infty)$$

The second term (②) is directly covered by the Value Convergence Theorem 4.2.2:

②

$$J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T)) \xrightarrow[4.2.2]{N\to\infty} 0$$

Combining ① and ② delivers:

$$\lim_{N\to\infty} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) = 0$$

$\blacksquare$

We are now able to formulate an abstract Inverse Stability Property, which should guarantee $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \to 0$ $(N \to \infty)$ at least locally. The reader is strongly advised to take a look at the Inverse Stability Property shown in the last chapter, which makes use of second order sufficient optimality conditions.

4.2.4 Inverse Stability Property (General Form)

It exists a function $\sigma : I \to \mathbb{R}$ such that:

- For $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \le \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \le \epsilon$ with $\epsilon > 0$ chosen appropriately it holds:

  $$\sigma\left(\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|\right) \le J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

  where $|||.|||$ is a usually stronger norm than $\|.\|$.

- σ is **strictly monotonously increasing**, hence invertible. In addition its **inverse is continuous**. This property may be restricted to the interval $[0, a]$ with $a := \sup\limits_{N\in\mathbb{N}}\left\{ \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \;\middle|\; |||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \le \epsilon\right\}$

Remarks:

- $\pi_N(\hat{x}^N)(.)$ is a feasible state for the continuous problem according to the Approximation Property and $\tilde{\pi}_N(\hat{u}^N)(.)$ denotes a corresponding feasible control.

- $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \le \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \le \epsilon$ represent the local restriction of the Inverse Stability Property. Unfortunately results from optimization theory usually include such constraints. Second order sufficient optimality conditions presented in 3.2.3 shall serve as an example.

- The reason why σ needs to be continuously invertible can be seen in the next theorem.

Combining Extended Value Convergence (4.2.3) with the general Inverse Stability Property (4.2.4) assures $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \to 0$ $(N \to \infty)$ locally:

4.2.5 Theorem (Local Convergence)

Let all assumptions of the Extended Value Convergence Theorem (4.2.3) and let the general Inverse Stability Property be fulfilled. Then it holds:

$$\lim_{N\to\infty} \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| = 0$$

As long as there exists $\tilde{N} \in \mathbb{N}$ such that $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \le \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \le \epsilon$ for all $N \ge \tilde{N}$, with ϵ from the Inverse Stability Property 4.2.4.

Proof :
Let $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \le \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \le \epsilon$.
From the Inverse Stability Property 4.2.4 we get

$$\sigma\left(\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|\right) \le J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

With σ being strictly monotonously increasing it follows that

① $$\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \leq \sigma^{-1}\left(J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))\right)$$

From Extended Value Convergence (4.2.3) we have

② $$\lim_{N\to\infty} J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T)) = 0$$

As σ^{-1} was premised to be continuous in the Inverse Stability Property 4.2.4 we get from combining ① and ② that

$$\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \overset{①}{\leq} \sigma^{-1}\left(J(\pi_N(\hat{x}^N)(t_0), \pi_N(\hat{x}^N)(T)) - J(\hat{x}(t_0), \hat{x}(T))\right) \xrightarrow[4.2.4,\ ②]{N\to\infty} 0$$

So we have

$$\lim_{N\to\infty} \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| = 0$$

■

The only thing left now is to connect $\|\hat{x}^N - \rho_N(\hat{x}(.))\|$ with $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|$ via the Approximation Property 4.2.1. To do so we need a connection between the norm in the continuous case $\|.\|$ and its "corresponding" discrete norm, which we called $\|.\|$, too. The discrete norm appears in $\|\hat{x}^N - \rho_N(\hat{x}(.))\|$ and the continuous one in $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|$ which can be clearly distinguished by the arguments being functions respectively vectors with a finite number of components. What connects the discrete norm and the norm in the continuous case is the so called Compatibility Property. With the Local Convergence Theorem 4.2.5 we have $\lim_{N\to\infty} \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| = 0$ with respect to a certain norm. Depending on the kind of Compatibility Property available we have to choose the corresponding discrete norm, which then leads to a convergence result for $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|$ in that specific discrete norm. So let's take a look at the last module we need.

4.2.6 Compatibility Property (General Form)

For $f(.) := \pi_N(\hat{x}^N)(.) - \hat{x}(.)$ it holds:

$$\lim_{N\to\infty} \Big| \|\rho_N(f(.))\| - \|f(.)\| \Big| = 0$$

Note: *If we know, that f is Lipschitz-continuous then we get from Theorem 3.2.6 that*

$$\Big| \|\rho_N(f(.))\|_\infty - \|f(.)\|_\infty \Big| \leq L_f h_N$$

This should lead to a similar result for the $\|.\|$-norm, which we are considering here. Once again, the specific norm that is represented by $\|.\|$ in this chapter, depends on the Inverse Stability Property available.

Now we have all the modules ready for use. The whole process then looks like:

The first step is getting $\pi_N(\hat{x}^N)(.)$ into play:

①
$$\|\hat{x}^N - \rho_N(\hat{x}(.))\| \leq \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\| + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|$$

For the first term we get by directly using the Approximation Property 4.2.1:

②
$$\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\| \to 0 \quad (N \to \infty)$$

The second term $\|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\|$ is close to $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|$, which is guaranteed by the general Compatibility Property 4.2.6:

③
$$\begin{aligned} & \Big| \|\rho_N \pi_N(\hat{x}^N)(.) - \rho_N \hat{x}(.)\| - \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \Big| \leq \varphi(N) \\ \Rightarrow \ & \|\rho_N \pi_N(\hat{x}^N)(.) - \rho_N \hat{x}(.)\| \leq \varphi(N) + \|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \\ & \text{with } \varphi(N) \to 0 \ \ (N \to \infty) \end{aligned}$$

So finally the term $\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\|$ occurs, for which we have from the Local Convergence Theorem 4.2.5:

④
$$\|\pi_N(\hat{x}^N)(.) - \hat{x}(.)\| \to 0 \quad (N \to \infty)$$

As long as there exists $\tilde{N} \in \mathbb{N}$ such that $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \leq \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \leq \epsilon$ for all $N \geq \tilde{N}$, with ϵ from the Inverse Stability Property 4.2.4.

Combining ② and ③ with ① and finally using ④ delivers:

$$\begin{aligned} \|\hat{x}^N - \rho_N(\hat{x}(.))\| & \overset{①}{\leq} \|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\| + \|\rho_N(\pi_N(\hat{x}^N)(.)) - \rho_N(\hat{x}(.))\| \\ & \overset{③}{\leq} \underbrace{\|\hat{x}^N - \rho_N(\pi_N(\hat{x}^N)(.))\|}_{\underset{②}{\xrightarrow{N\to\infty}} 0} + \underbrace{\varphi(N)}_{\underset{③}{\xrightarrow{N\to\infty}} 0} + \underbrace{\|\pi_N(\hat{x}^N)(.)) - \hat{x}(.)\|}_{\underset{④}{\xrightarrow{N\to\infty}} 0} \\ & \xrightarrow{N\to\infty} 0 \end{aligned}$$

This means it holds:

$$\lim_{N\to\infty} \|\hat{x}^N - \rho_N(\hat{x}(.))\| = 0$$

■

So overall we have shown that the following theorem holds.

4.2.7 Convergence Theorem (General Form)

Assume that the Approximation Property 4.2.1 holds and let $J(.,.)$ be continuous in both arguments. This leads to Value Convergence in the form of Theorem 4.2.2. Furthermore let the Inverse Stability Property 4.2.4 and the compatibility property 4.2.6 be fulfilled.

With all these assumptions it then holds:

$$\lim_{N\to\infty} \|\hat{x}^N - \rho_N(\hat{x}(.))\| = 0$$

Note: *This result only holds if there exists $\tilde{N} \in \mathbb{N}$ such that $|||\pi_N(\hat{x}^N)(.) - \hat{x}(.)||| \leq \epsilon$ and $|||\tilde{\pi}_N(\hat{u}^N)(.) - \hat{u}(.)||| \leq \epsilon$ for all $N \geq \tilde{N}$. For details on that matter and the constant ϵ see Theorem 4.2.4 (general Inverse Stability Property).*

Chapter 3 and 4 do rely heavily on some sort of Approximation Property for the feasible solution sets. The next chapter shows that under certain circumstances such a property really does exist.

5 Approximation Property

5.1 Overview

This chapter is intended to prove the Approximation Property presented in 3.2.1. It is based on article [2]. However we won't need any convexity assumptions, which is quite an improvement. The idea for proving results without the need of convexity is based on [4] and [5].
We will consider the solution sets to the Constrained Differential Inclusion introduced in Definition 3 and the Constrained Discrete Differential Inclusion introduced in Definition 6 with bounded starting set X_0. Recall that they are denoted by $X_\Theta(T, t_0, X_0)$ respectively $X_\Theta^N(T, t_0, X_0)$. X_0 being bounded is needed to apply the theorems of section 2.6. Furthermore the reader should be aware of the fact, that the Constrained Differential Inclusions involve pure state constraints, which is a major difficulty. This makes obtaining the desired result, already presented in Theorem 3.2.1, way more complex.

Note: *The notations in this chapter are the same as in the set-valued Mayer-Problem sections of chapter 2.*

The proof consists of two major parts. The first one is about proving the result without pure state constraints. The second one is the part based on [2], which uses the result without state constraints and combines it with a result obtained for the state constrained case. Both parts are in no way trivial and this thesis cannot cover any detail. This chapter is intended to wrap up all the results that lead to 3.2.1. One thing, that will be explored in detail, is the extension of the result from [3] to get rid of the convexity assumption made in [3]. For the approach to obtain the Approximation Property in this chapter the set-valued right-hand side F of the Differential Inclusion needs to fulfill the following assumptions:

Assumptions:

(A1) F satisfies a linear growth condition, i.e.:
There exists a constant $C_F \geq 0$ such that with $t \in I$ and $x \in \mathbb{R}^n$ it holds

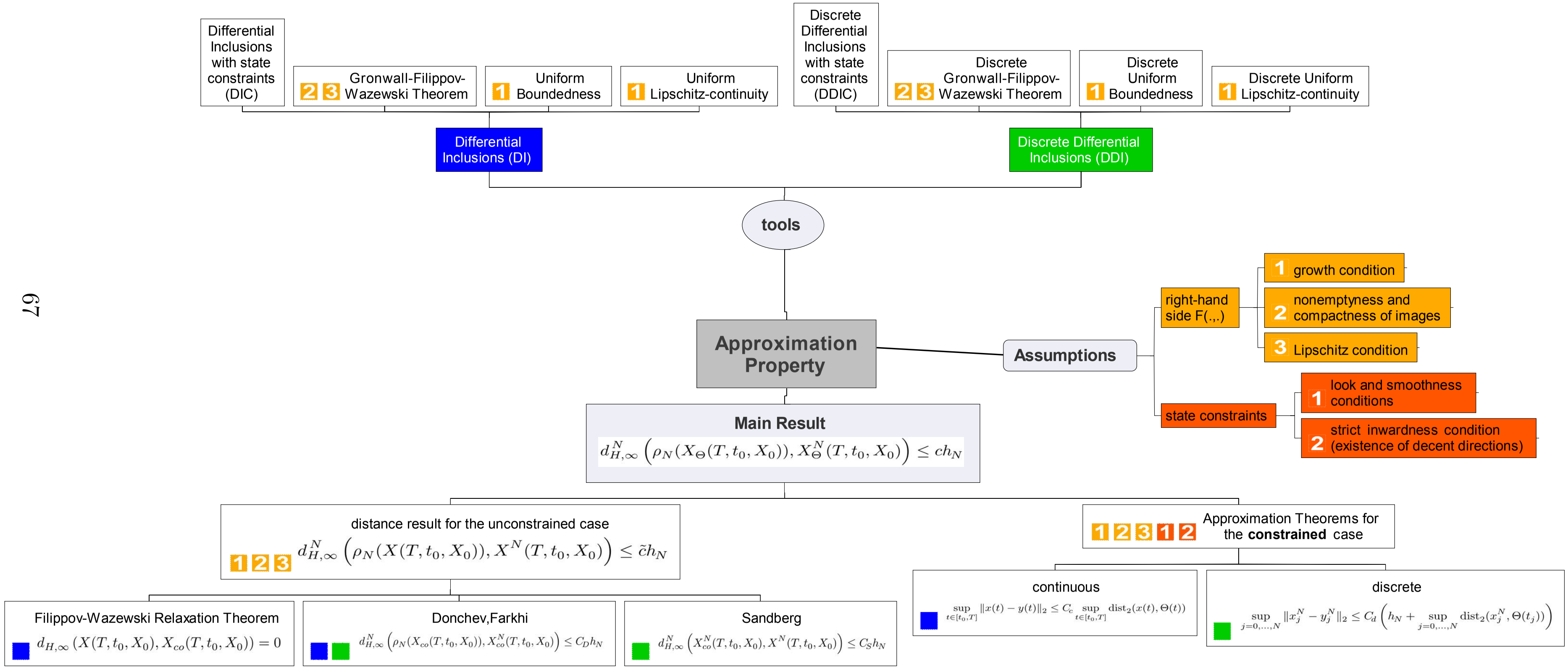
$$\|F(t,x)\|_2 \leq C_F \left(\|x\|_2 + 1\right)$$

(A2) F has nonempty compact images.

(A3) F is Lipschitz-continuous in (t,x) for all $t \in I$ and $x \in \mathbb{R}^n$ with Lipschitz constant L_F with respect to the Hausdorff-distance, i.e.

$$d_{H,2}\left(F(t,x), F(\tilde{t},y)\right) \leq L_F(|t-\tilde{t}| + \|x-y\|_2) \quad (t,\tilde{t} \in I,\ x,y \in \mathbb{R}^n)$$

The figure on the next side shows the whole concept and even some results only presented in [2].


Differential Inclusions with state constraints (DIC)
2 3 Gronwall-Filippov-Wazewski Theorem
1 Uniform Boundedness
1 Uniform Lipschitz-continuity
Discrete Differential Inclusions with state constraints (DDIC)
2 3 Discrete Gronwall-Filippov-Wazewski Theorem
1 Discrete Uniform Boundedness
1 Discrete Uniform Lipschitz-continuity
Differential Inclusions (DI)
Discrete Differential Inclusions (DDI)
tools
Approximation Property
Assumptions
right-hand side F(.,.)
1 growth condition
2 nonemptyness and compactness of images
3 Lipschitz condition
state constraints
1 look and smoothness conditions
2 strict inwardness condition (existence of decent directions)
Main Result
$d_{H,\infty}^N\left(\rho_N(X_\Theta(T,t_0,X_0)), X_\Theta^N(T,t_0,X_0)\right) \leq ch_N$
distance result for the unconstrained case
1 2 3 $d_{H,\infty}^N\left(\rho_N(X(T,t_0,X_0)), X^N(T,t_0,X_0)\right) \leq \tilde{c}h_N$
Filippov-Wazewski Relaxation Theorem
🟦 $d_{H,\infty}(X(T,t_0,X_0), X_{co}(T,t_0,X_0)) = 0$
Donchev,Farkhi
🟦 🟩 $d_{H,\infty}^N\left(\rho_N(X_{co}(T,t_0,X_0)), X_{co}^N(T,t_0,X_0)\right) \leq C_D h_N$
Sandberg
🟩 $d_{H,\infty}^N\left(X_{co}^N(T,t_0,X_0), X^N(T,t_0,X_0)\right) \leq C_S h_N$
1 2 3 1 2 Approximation Theorems for the **constrained** case
continuous
🟦 $\sup_{t\in[t_0,T]} \|x(t)-y(t)\|_2 \leq C_c \sup_{t\in[t_0,T]} \mathrm{dist}_2(x(t),\Theta(t))$
discrete
🟩 $\sup_{j=0,\dots,N} \|x_j^N - y_j^N\|_2 \leq C_d\left(h_N + \sup_{j=0,\dots,N} \mathrm{dist}_2(x_j^N,\Theta(t_j))\right)$

5.2 Approximation Property for the unconstrained case

This section proves the Approximaiton Property for the case without pure state constraints. So the solution sets one has to consider here are $X(T, t_0, X_0)$ (see Definition 2) for the Differential Inclusion introduced in Definition 3 and $X^N(T, t_0, X_0)$ (see Definition 4) for the Discrete Differential Inclusion introduced in Defintion 6. The final result of this section will be:

$$\boxed{d_{H,\infty}^N \left(\rho_N(X(T, t_0, X_0)), X^N(T, t_0, X_0)\right) \leq \tilde{c}h_N}$$

where $d_{H,\infty}^N$ is the Hausdorff-distance introduced in 2.5.4.

Note: *As mentioned in a remark to the Theorem 3.2.1 this way of representing the Approximation Property is completely identical to the way used in 3.2.1, which involved the introduction of the functions $\pi_N(.)$ and $\delta_N(.)$.*

This section on its own involves three major results. The first and the second one (both presented in [3]) are well known in literature and will be presented without proof. The third one was published by Mattias Sandberg with proof in paper [4] and [5]. This thesis presents a slightly alternated proof. This proof won't go for convergence of reachable sets, which makes it a lot more readable.

The core of this concept will be the convergence result from [3] which needs the images of F to be convex, which this chapter tries to avoid. So we introduce the following Differential Inclusions and corresponding solution sets, which are the same as in Definition 3 and Definition 6 apart from the fact that the right-hand side has been convexified.

Definition 8:

$\boldsymbol{X_{co}(T, t_0, X_0)}$ *shall be the solution set to the following Differential Inclusion with* $x \in AC(I)^n$*:*

$$\begin{aligned} \dot{x}(t) &\in co\, F(t, x(t)) && a.e. \\ x(t_0) &\in X_0 \end{aligned}$$

$\boldsymbol{X^N_{co}(T, t_0, X_0)}$ *shall be the solution set to the following Discrete Differential Inclusion with* $x^N \in \mathbb{R}^{(N+1)n}$*:*

$$\begin{aligned} x^N_{j+1} &\in x^N_j + h_N \cdot co\, F(t_j, x^N_j) && (j = 0, \ldots, N-1) \\ x^N_0 &\in X_0 \end{aligned}$$

Remarks:

- *co delivers the convex hull of a set.*
- *If the images of* F *are already convex, it of course holds* $co\, F(t, x) = F(t, x)$*, which leads to* $X_{co}(T, t_0, X_0) = X(T, t_0, X_0)$ *and* $X^N_{co}(T, t_0, X_0) = X^N(T, t_0, X_0)$

With this definition, the convergence result with convex right-hand sides looks like this:

5.2.1 Convergence Theorem for convex Differential Inclusions

Let (A1), (A2) and (A3) be fulfilled. Then it holds:

$$d^N_{H,\infty}\left(\rho_N(X_{co}(T,t_0,X_0)), X^N_{co}(T,t_0,X_0)\right) \leq C_D h_N$$

Proof : For a proof see [3].

The goal now is to obtain a similar estimation for $d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), \rho_N(X_{co}(T,t_0,X_0))\right)$ and $d^N_{H,\infty}\left(X^N_{co}(T,t_0,X_0), X^N(T,t_0,X_0)\right)$ and then use the triangular inequality of the Hausdorff distance (see 2.5.4), i.e. in this case:

$$d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), X^N(T,t_0,X_0)\right) \leq \underbrace{d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), \rho_N(X_{co}(T,t_0,X_0))\right)}_{\textcircled{1}} +$$
$$\underbrace{d^N_{H,\infty}\left(\rho_N(X_{co}(T,t_0,X_0)), X^N_{co}(T,t_0,X_0)\right)}_{\textcircled{2}} + \underbrace{d^N_{H,\infty}\left(X^N_{co}(T,t_0,X_0), X^N(T,t_0,X_0)\right)}_{\textcircled{3}}$$

For ①, i.e. $d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), \rho_N(X_{co}(T,t_0,X_0))\right)$, one can directly use Filippovs relaxation theorem, a result well known in literature (see [6]).

5.2.2 Filippovs Relaxation Theorem

Let (A2) and (A3) be fulfilled. Then for every $x(.) \in X_{co}(T,t_0,X_0)$ and $\epsilon > 0$ there exists $y(.) \in X(T,t_0,X_0)$ such that

$$\|x(.) - y(.)\|_\infty \leq \epsilon$$

which is equivalent, to

$$d_{H,\infty}\left(X(T,t_0,X_0), X_{co}(T,t_0,X_0)\right) = 0$$

Note: *The equivalence follows from:*

For every $x(.) \in X(T,t_0,X_0)$ and $\epsilon > 0$ there exists $y(.) \in X_{co}(T,t_0,X_0)$ such that

$$\|x(.) - y(.)\|_\infty = 0$$

which is a trivial result, due to $F(t,x) \subset co\,F(t,x)$ $(t \in I,\ x \in \mathbb{R}^n)$ leading to $X(T,t_0,X_0) \subset X_{co}(T,t_0,X_0)$. Together with the relaxation theorem this leads to $\bar{X}(T,t_0,X_0) = \bar{X}_{co}(T,t_0,X_0)$, where the bar $\bar{.}$ denotes the closure of a set (in this case the closure with respect to the supremum norm). When the closure of two sets is equal, the Hausdorff-distance is 0.

From $d_{H,\infty}\left(X(T,t_0,X_0), X_{co}(T,t_0,X_0)\right) = 0$ it follows directly by the definition of $d_{H,\infty}$ and $d^N_{H,\infty}$ that $d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), \rho_N(X_{co}(T,t_0,X_0))\right) = 0$.
Proof : See [6].

We now have sufficient estimations for the Hausdorff-distances ① and ②. The only term left now is ③, which will be explored in detail. The result we need is the same as Filippovs Relaxation Theorem, but this time for discrete convex Differential Inclusions. The proof will be quite lengthy and at the beginning pretty technical. But the core concept is nevertheless quite interesting.

5.2.3 Convergence Theorem for convex discrete Differential Inclusions

Let the set of initial values X_0 be bounded and let (A1), (A2) and (A3) be fulfilled. Then for every $x^N \in X^N_{co}(T, t_0, X_0)$ there exists $y^N \in X^N(T, t_0, X_0)$ such that

$$\sup_{j=0,\dots,N} \|x^N_j - y^N_j\|_2 \leq C_S h_N$$

which is equivalent to

$$d^N_{H,\infty}\left(X^N_{co}(T, t_0, X_0), X^N(T, t_0, X_0)\right) \leq C_S h_N$$

Note: *The equivalence follows like in Theorem 5.2.2:*
Due to

$$F(t, x) \subset co\, F(t, x) \ (t \in \ I, \ x \in \mathbb{R}^n)$$

one gets $X^N(T, t_0, X_0) \subset X^N_{co}(T, t_0, X_0)$. So $d(X^N(T, t_0, X_0), X^N_{co}(T, t_0, X_0)) = 0$, with d defined in 2.5.4.

Proof :
The proof of this theorem is based on [4] and [5]. It will be presented in detail in section 5.4.

Jumping back to the triangular inequality presented at the beginning of the section and using Theorem 5.2.2 to estimate ①, Theorem 5.2.1 to estimate ② and Theorem 5.2.3 to estimate ③ we get the final result for this section.

5.2.4 Theorem (Convergence Result for the unconstrained case)

Let the set of initial values X_0 be bounded and let (A1), (A2) and (A3) be fulfilled. Then it holds:

$$\boxed{d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), X^N(T,t_0,X_0)\right) \le \tilde{c}h_N}$$

Proof :
The proof is a simple consequence of Theorem 5.2.2, Theorem 5.2.1 and Theorem 5.2.3:

$$\begin{aligned} d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), X^N(T,t_0,X_0)\right) &\le d^N_{H,\infty}\left(\rho_N(X(T,t_0,X_0)), \rho_N(X_{co}(T,t_0,X_0))\right) + \\ &\quad d^N_{H,\infty}\left(\rho_N(X_{co}(T,t_0,X_0)), X^N_{co}(T,t_0,X_0)\right) + d^N_{H,\infty}\left(X^N_{co}(T,t_0,X_0), X^N(T,t_0,X_0)\right) \\ &\overset{5.2.2,\ 5.2.1,\ 5.2.3}{\le} 0 + C_D h_N + C_S h_N = \underbrace{(C_D + C_S)}_{\tilde{c}:=} h_N \end{aligned}$$

■

5.3 Approximation Property for the constrained case

This section extends the result of section 5.2 for the constrained case, which means that pure state constraints are involved. So the sets considered in this section are $X_\Theta(T,t_0,X_0)$ and $X^N_\Theta(T,t_0,X_0)$. The main result of this chapter will be

$$\boxed{d^N_{H,\infty}\left(\rho_N(X_\Theta(T,t_0,X_0)), X^N_\Theta(T,t_0,X_0)\right) \le ch_N}$$

To deal with the constrained case in addition to (A1), (A2) and (A3), some quite restrictive assumptions are needed. They might be weakened, but for the proofs in [2] they are indespensible. These assumptions will make sense when looking at the proofs of Theorem 3.1 and Theorem 3.2 in [2].

Assumptions:

(C1) $\Theta : I \rightrightarrows \mathbb{R}^n$ *has nonempty images explicitly given by*

$$\Theta(t) := \left\{x \in \mathbb{R}^N \mid s(t,x) \le 0\right\}$$

with $s(.,.) \in C^{1,L}(I \times R^n)$ *being a single scalar function.*
Furthermore $x \in \partial\Theta(t) \Leftrightarrow s(t,x) = 0$ *shall be fulfilled.*

(C2) The boundary of $\Theta(.)$ *fulfills the "strict inwardness condition". This means that there exist* $\alpha, \mu > 0$ *such that for all* $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta(.)) \cap (I \times \mathbb{R}^n)$ *it holds*

$$\min_{v \in F(t,x)} \langle \nabla s(t,x), \left(\begin{smallmatrix}1\\v\end{smallmatrix}\right)\rangle \le -\alpha$$

Remarks:

- $\partial\Theta(t)$ is the boundary of $\Theta(t)$.
- $C^{1,L}(I \times \mathbb{R}^n)$ is the space of all differentiable functions, whose partial derivatives are Lipschitz-continuous. So with $s(.,.) \in C^{1,L}(I \times R^n)$, $\nabla s(.,.)$ is Lipschitz-continuous on $I \times \mathbb{R}^n$. Let the corresponding Lipschitz-constant be $L_{\nabla s}$.
- The postulation that $s(.,.)$ has to be a single scalar function is very restrictive. After analyzing the proof of Theorem 3.2 in [2] it is the authors strong believe that this restriction is not needed. The proof of that theorem might be extended to fit the multidimensional case, but it would be even more lengthy and technical than the proof for the single scalar case already is. So it won't be presented in this thesis. Nevertheless there will be a short section on ideas on how to treat the multidimensional case, which involves extending (C1) and (C2) (see the extended variants (C1E) and (C2E) below or 5.3.4 for further details on the multidimensional case).
- $x \in \partial\Theta(t) \Leftrightarrow s(t,x) = 0$ has to be postulated, because this is in general not the case. With $s(.,.)$ being continuos it holds that $x \in \partial\Theta(t) \Rightarrow s(t,x) = 0$. But the other way round does not have to be true even if $s(.,.) \in C^{1,L}(I \times R^n)$.
  For example with
  $$s(t,x) := \begin{cases} -(x+1)^2 & x \leq 1 \\ 0 & x \in (-1,1] \\ (x-1)^2 & x > 1 \end{cases}$$
  we have $\partial\Theta(t) = \{1\}$ $(t \in I)$, but $s(t,x) = 0$ $(x \in [-1,1], t \in I)$.
- B_μ is a ball with radius μ with respect to the $\|.\|_2$-norm, i.e.
  $$B_\mu := \{(t,x) \in I \times \mathbb{R}^n \bigm| \|(t,x)\|_2 \leq \mu\}$$
- $\operatorname{graph} \partial\Theta(.) = \{(t,x) \in I \times \mathbb{R}^n \bigm| x \in \partial\Theta(t)\}$
- The "strict inwardness condition" makes it possible to "redirect" any feasible solution that comes close to the boundary of $\Theta(.)$ inwards. One might also say that the condition delivers a valid direction of descent at "critical" points (those close to the boundary of $\Theta(.)$). The idea of the proof of Theorem 3.2 in [2] is to redirect a solution of the unconstrained Differential Inclusion (DI) to get a solution of the constrained Differential Inclusion (DIC).

For multidimensional state constraints $s(.,.)$ the following assumptions should be enough to deliver the desired results from the following theorems. It is the authors strong believe, that replacing (C1) with (C1E) and (C2) with (C2E) will make the statements of Theorem 5.3.1, Theorem 5.3.2 and Theorem 5.3.3 available for the case involving more than one single scalar function for the state constraints. But as there are no formulated proofs available yet, the following extended assumptions should be considered "experimental". For details see the following section 5.3.4.

Assumptions:

(C1E) $\Theta : I \Rightarrow \mathbb{R}^n$ *has nonempty images explicitly given by*

$$\Theta(t) = \bigcap_{i=1}^{n_s} \Theta_i(t)$$

with $n_s \in \mathbb{N}$ *and*

$$\Theta_i(t) := \Big\{ x \in \mathbb{R}^N \mid s_i(t,x) \leq 0 \Big\} \qquad\qquad (i = 1, \dots, n_s)$$

where

$$s_i(.,.) \in C^{1,L}(I \times R^n) \qquad\qquad (i = 1, \dots, n_s)$$

Furthermore $x \in \partial\Theta_i(t) \Leftrightarrow s_i(t,x) = 0$ $(i = 1, \dots, n_s)$ *shall be fulfilled.*

(C2E) The boundary of $\Theta(.)$ *fulfills the "strict inwardness condition". This means that there exist* $\alpha, \mu > 0$ *such that for each* $i \in \{1, \dots, n_s\}$ *it holds:*
For all $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_i(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap (I \times \mathbb{R}^n)$ *the following inequality applies:*

$$\min_{v \in F(t,x)} \langle \nabla s_i(t,x), \left(\begin{smallmatrix} 1 \\ v \end{smallmatrix}\right) \rangle \leq -\alpha$$

The central idea of this section is to make use of Theorem 5.2.4, which is the main result from section 5.2.
This result will be applied to a solution $x(.) \in X_\Theta(T, t_0, X_0))$ to obtain $\tilde{x}^N \in X^N(T, t_0, X_0)$, which is close to $x(.)$ on the grid. Though being close to $x(.)$ the solution $\tilde{x}^N$ may not obey the pure state constraints, but it definitely does not harm them too much. Assuming that (C1) and (C2) hold, we shall see in this section that for such a solution $\tilde{x}^N$ there exists a solution $x^N \in X_\Theta^N(T, t_0, X_0)$ (so x^N obeys the pure state constraints represented by Θ), which is close enough to $\tilde{x}^N$ on the grid to give the desired estimation. The other way round, i.e. starting with a solution $x^N \in X_\Theta^N(T, t_0, X_0)$ and finding a solution $x(.) \in X_\Theta(T, t_0, X_0))$ uses the same method as described above, but a different result for finding a feasible solution that obeys the pure state constraints $\Theta(.)$ will be required. This is because after applying Theorem 5.2.4 one has to deal with the continuous case.
The two following theorems will provide these results. As already mentioned the proofs are rather lengthy and technical, so they won't be provided here. For further details see [2].

The first one will help us gain a solution $x(.) \in X_\Theta(T, t_0, X_0)$ when we already have a solution $\tilde{x}(.) \in X(T, t_0, X_0)$ that is close to fulfilling the pure state constraints represented by $\Theta(.)$. The dist_2-function is the distance function based on the $\|.\|_2$-norm (see 2.5.4).

5.3.1 Theorem

Consider the constrained differential inclusion (DIC) (see Definition 3). Assume that (A1), (A2), (A3) and in addition (C1) and (C2) hold.
Then there exists $C_c > 0$ such that for every $x_0 \in X_0 \cap \Theta(t_0)$ and $x(.) \in X(T, t_0, \{x_0\})$ there exists $y(.) \in X_\Theta(T, t_0, \{x_0\})$ with

$$\sup_{t \in [t_0, T]} \|x(t) - y(t)\|_2 \leq C_c \sup_{t \in [t_0, T]} \operatorname{dist}_2(x(t), \Theta(t))$$

Note: $x_0 \in X_0 \cap \Theta(t_0)$ *just expresses the fact, that the initial value has to be feasible. This is automatically fulfilled for any solution* $y(.) \in X_\Theta(T, t_0, X_0)$ *but not for any solution* $x(.) \in X(T, t_0, X_0)$. *So this has to be postulated seperately.*

Proof :
For further details see Theorem 3.1 in [2].

The second one will help us gain a solution $x^N \in X^N_\Theta(T, t_0, X_0)$ when we already have a solution $\tilde{x}^N \in X^N(T, t_0, X_0)$ that is close to fulfilling the pure state constraints represented by $\Theta(.)$.

5.3.2 Theorem

Consider the constrained discrete differential inclusion (DIC) (see Definition 6). Assume that (A1), (A2), (A3) and in addition (C1) and (C2) hold.
Then there exists $N_0 \in \mathbb{N}$ and $C_d > 0$ such that for every $x_0 \in X_0 \cap \Theta(t_0)$, $N \geq N_0$ and $x^N \in X^N(T, t_0, \{x_0\})$ there exists $y^N \in X^N_\Theta(T, t_0, \{x_0\})$ with

$$\sup_{j=0,\ldots,N} \|x^N_j - y^N_j\|_2 \leq C_d \left(h_N + \sup_{j=0,\ldots,N} \operatorname{dist}_2(x^N_j, \Theta(t_j)) \right)$$

Note: $x_0 \in X_0 \cap \Theta(t_0)$ *just expresses the fact, that the initial value has to be feasible. This is automatically fulfilled for any solution* $y^N \in X^N_\Theta(T, t_0, X_0)$ *but not for any solution* $x^N \in X^N(T, t_0, X_0)$. *So this has to be postulated seperately.*

Proof :
For further details see Theorem 3.2 in [2].

Combining Theorem 5.3.1, Theorem 5.3.2 and Theorem 5.2.4 then delivers the final result. The only major step to proof that will be to estimate $\sup\big\{\mathrm{dist}_2(x(t), \Theta(t)) \;\big|\; t \in [t_0, T]\big\}$ and $\sup\big\{\mathrm{dist}_2(x_j^N, \Theta(t_j)) \;\big|\; j \in \{0, \dots, N\}\big\}$.

5.3.3 Theorem (Convergence Result for the constrained case)

Let the set of initial values X_0 be bounded. Furthermore let (A1), (A2), (A3) and in addition (C1) and (C2) be fulfilled. Then there exists $N_0 \in \mathbb{N}$ such that for all $N \geq N_0$ it holds:

$$\boxed{d_{H,\infty}^N \Big(\rho_N(X_\Theta(T, t_0, X_0)), X_\Theta^N(T, t_0, X_0)\Big) \leq ch_N}$$

Note: *Once again: The major difference between Theorem 4.2 in [2] and this Theorem is that for the latter one the images of F must not be convex. This can be seen by comparing assumption (A2) in this thesis to assumption (H2) in [2]. This is due to the fact that [2] uses the result of Dontchev Farkhi (this is Theorem 5.2.1), which needs convexity. In section 5.2 this result has been extended for the nonconvex case without the need for additional assumptions, which lead to Theorem 5.2.4.*

Proof :

First direction:

Let us first consider a solution $y(.) \in X_\Theta(T, t_0, \{x_0\})$ with $x_0 \in X_0$. The goal now is to obtain a solution $y^N \in X_\Theta(T, t_0, \{x_0\})$ which is close enough to $y(.)$ on the grid. The key to success is the use of Theorem 5.2.4 in conjunction with Theorem 5.3.2.

For $y(.)$ Theorem 5.2.4 delivers $\eta^N \in X^N(T, t_0, \{x_0\})$ (for $N \geq N_0$) with

$$\text{①}\quad \sup_{j=0,\dots,N} \|y(t_j) - \eta_j^N\|_2 \leq \tilde{c}h_N$$

Theorem 5.3.2 guarantees the existence of $y^N \in X_\Theta^N(T, t_0, \{x_0\})$ with

$$\text{②}\quad \sup_{j=0,\dots,N} \|\eta_j^N - y_j^N\|_2 \leq C_d \left(h_N + \sup_{j=0,\dots,N} \mathrm{dist}_2(\eta_j^N, \Theta(t_j))\right)$$

Analyzing $\mathrm{dist}_2(\eta_j^N, \Theta(t_j))$ $(j = 0, \dots, N)$ by using the fact that $y(t_j) \in \Theta(t_j)$ $(j = 0, \dots, N)$ delivers

$$\text{③}\quad \mathrm{dist}_2(\eta_j^N, \Theta(t_j)) \leq \|\eta_j^N - y(t_j)\|_2 + \mathrm{dist}_2(y(t_j), \Theta(t_j)) = \|\eta_j^N - y(t_j)\|_2 \quad (j = 0, \dots, N)$$

Combining ①, ② and ③ yields for $j \in \{0, \dots, N\}$ that

$$\begin{aligned}\|y(t_j) - y_j^N\|_2 &\leq \|y(t_j) - \eta_j^N\|_2 + \|\eta_j^N - y_j^N\|_2 \overset{\text{①,②}}{\leq} \tilde{c}h_N + C_d \left(h_N + \sup_{j=0,\dots,N} \mathrm{dist}_2(\eta_j^N, \Theta(t_j))\right) \\ &\overset{\text{③}}{\leq} \tilde{c}h_N + C_d \left(h_N + \sup_{j=0,\dots,N} \|\eta_j^N - y(t_j)\|_2\right) \overset{\text{①}}{\leq} (\tilde{c} + C_d(1 + \tilde{c}))\, h_N\end{aligned}$$

So we have the desired result for the first direction, i.e.:
For any $y(.) \in X_\Theta(T, t_0, \{x_0\})$ there exists $y^N \in X_\Theta(T, t_0, \{x_0\})$ with $N \geq N_0$ such that

$$\sup_{j=0,\ldots,N} \|y(t_j) - y_j^N\|_2 \leq (\tilde{c} + C_d(1 + \tilde{c}))\, h_N$$

Second direction:
This time we start off with $y^N \in X_\Theta(T, t_0, \{x_0\})$ and try to get $y(.) \in X_\Theta(T, t_0, \{x_0\})$, where $x_0 \in X_0 \cap \Theta(t_0)$. For this direction the key to success is the use of Theorem 5.2.4 in conjunction with Theorem 5.3.1.
For y^N Theorem 5.2.4 delivers $\eta(.) \in X(T, t_0, \{x_0\})$ with

④
$$\sup_{j=0,\ldots,N} \|y_j^N - \eta(t_j)\|_2 \leq \tilde{c} h_N$$

Theorem 5.3.1 then guarantees the existence of $y(.) \in X_\Theta(T, t_0, \{x_0\})$ with

⑤
$$\sup_{t \in [t_0,T]} \|\eta(t) - y(t)\|_2 \leq C_c \sup_{t \in [t_0,T]} \mathrm{dist}_2(\eta(t), \Theta(t))$$

So this time we have to estimate $\mathrm{dist}_2(\eta(t), \Theta(t))$ for all $t \in [t_0, T]$ instead of $\mathrm{dist}_2(\eta_j^N, \Theta(t_j))$ $(j = 0, \ldots, N)$. This is more complicated, because we only have $y_j^N \in \Theta(t_j)$ $(j = 0, \ldots, N)$ as a reference. To bring $y_j^N \in \Theta(t_j)$ $(j = 0, \ldots, N)$ into play we need to use reference points on the grid. This gives us the following estimation for $t \in [t_j, t_{j+1})$ and all $j \in \{0, \ldots, N-1\}$

⑥
$$\begin{aligned} \mathrm{dist}_2(\eta(t), \Theta(t)) &\leq \|\eta(t) - \eta(t_j)\|_2 + \|\eta(t_j) - y_j^N\|_2 + \mathrm{dist}_2(y_j^N, \Theta(t)) \\ &\overset{2.6.2,\textcircled{4}}{\leq} L|t - t_j| + \tilde{c} h_N + \mathrm{dist}_2(y_j^N, \Theta(t)) \leq (L + \tilde{c}) h_N + \mathrm{dist}_2(y_j^N, \Theta(t)) \end{aligned}$$

So all that's left to do is to estimate $\mathrm{dist}_2(y_j^N, \Theta(t))$ for $t \in [t_j, t_{j+1})$. Although $y_j^N \in \Theta(t_j)$ $(j = 0, \ldots, N)$ this turns out to by quite tricky.
The central idea is the following:
If $y^N \in B_\mu(\mathrm{graph}\, \partial\Theta(.))$ use (C2) to create a function $y^N(.)$ on $[t_j, t_{j+1})$ with $y^N(t) \in \Theta(t)$ and $\|y^N(t) - y_j^N\|_2 \leq C h_N$ for $t \in [t_j, t_{j+1})$ and $C > 0$. Then we can make use of $\mathrm{dist}_2(y_j^N, \Theta(t)) \leq \|y_j^N - y^N(t)\|_2 \leq C h_N$ for $t \in [t_j, t_{j+1})$.
If $y_j^N \notin B_\mu(\mathrm{graph}\, \partial\Theta(.))$ then y_j^N is far enough in the interior of $\Theta(t_j)$ such that for $N \geq N_0$ and $t \in [t_j, t_{j+1})$ it still holds $y_j^N \in \Theta(t)$. Hence $\mathrm{dist}_2(y_j^N, \Theta(t)) = 0$ for $t \in [t_j, t_{j+1})$.
So we have to consider the following two cases:

- Let's begin with the case that $y_j^N \notin B_\mu(\mathrm{graph}\, \partial\Theta(.))$. The goal is to show that $s(t, y_j^N) \leq 0$ for $t \in [t_j, t_{j+1})$. Let's assume the opposite, i.e. it exists $\tilde{t} \in [t_j, t_{j+1})$ such that $s(\tilde{t}, y_j^N) > 0$. Because of $s(t_j, y_j^N) \leq 0$ it exists $\hat{t} \in [t_j, \tilde{t}]$ such that $\gamma(\hat{t}) = 0$, where $\gamma(t) := s(t, y_j^N)$. This follows directly from the intermediate value theorem. So $y_j^N \in \Theta(\hat{t})$, which means $(\hat{t}, y_j^N) \in \mathrm{graph}\, \partial\Theta(.)$. With N_0 chosen high enough this leads to

$$\mathrm{dist}_2((t_j, y_j^N), \mathrm{graph}\, \partial\Theta(.)) \leq \|(t_j, y_j^N) - (\hat{t}, y_j^N)\|_2 = |\hat{t} - t_j| \leq h_N \overset{N \geq N_0}{\leq} \mu$$

Note: *If we have already set N_0 high enough such that Theorem 5.3.2 holds, $h_N \leq \mu$ is already fulfilled. This can be seen by taking a look at (3.5) in the proof of Theorem 3.2 in [2], which is a postulation. Theorem 3.2 in [2] corresponds to Theorem 5.3.2 in this thesis.*

So the above inequality leads to a contradiction to $y_j^N \notin B_\mu(\operatorname{graph}\partial\Theta(.))$. This means that indeed $s(t, y_j^N) \leq 0$ for $t \in [t_j, t_{j+1})$. So for $y_j^N \notin B_\mu(\operatorname{graph}\partial\Theta(.))$ we have $y_j^N \in \Theta(t)$, hence $\operatorname{dist}_2(y_j^N, \Theta(t)) = 0$ (for $t \in [t_j, t_{j+1})$).

- If $y_j^N \in B_\mu(\operatorname{graph}\partial\Theta(.))$ it follows from (C2) that there exists $v_j \in F(t_j, y_j^N)$ such that

$$\langle \nabla s(t_j, y_j^N), \left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\rangle \leq -\alpha$$

With $y^N(t) := y_j^N + (t - t_j)v_j$ for $t \in [t_j, t_{j+1})$ we get for $t \in [t_j, t_{j+1})$ that

$$\begin{aligned}
s(t, y^N(t)) &= s(t_j, y_j^N) + \int\limits_{t_j}^{t} \frac{d}{d\tau} s(\tau, y^N(\tau))\, d\tau \\
&\leq \int\limits_{t_j}^{t} \frac{d}{d\tau} s(\tau, y^N(\tau))\, d\tau = \int\limits_{t_j}^{t} \langle \nabla s(\tau, y^N(\tau)), \left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\rangle\, d\tau \\
&= \int\limits_{t_j}^{t} \langle \nabla s(t_j, y_j^N), \left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\rangle\, d\tau + \int\limits_{t_j}^{t} \langle \nabla s(\tau, y^N(\tau)) - \nabla s(t_j, y_j^N), \left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\rangle\, d\tau \\
&\underset{\text{Schwarz}}{\overset{\text{(C2)}}{\leq}} -\alpha(t - t_j) + \int\limits_{t_j}^{t} \|\nabla s(\tau, y^N(\tau)) - \nabla s(t_j, y_j^N)\|_2 \, \|\left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\|_2\, d\tau
\end{aligned}$$

With

$$\begin{aligned}
&\|\nabla s(\tau, y^N(\tau)) - \nabla s(t_j, y_j^N)\|_2 \, \|\left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\|_2 \overset{\text{Lipschitz}}{\leq} L_{\nabla s}\left(|\tau - t_j| + \|y^N(\tau) - y_j^N\|_2\right)\left(1 + \|v_j\|_2\right) \\
&= \quad L_{\nabla s}\left(|\tau - t_j| + |\tau - y_j|\, \|v_j\|_2\right)\left(1 + \|v_j\|_2\right) \\
&\overset{|\tau - t_j| \leq h_N}{\leq} \quad L_{\nabla s} h_N \left(1 + \|v_j\|_2\right)^2 \overset{v_j \in F(t_j, y_j^N)}{\leq} L_{\nabla s} h_N \left(1 + \|F(t_j, y_j^N)\|_2\right)^2 \\
&\overset{\text{(A1)}}{\leq} \quad L_{\nabla s} h_N \left(1 + C_F(1 + \|y_j^N\|_2)\right)^2 \overset{2.6.3}{\leq} L_{\nabla s}\left(1 + C_F(1 + \tilde{M})\right)^2 h_N
\end{aligned}$$

Combining those two inequalities we get

$$\begin{aligned}
s(t, y^N(t)) &\leq -\alpha(t - t_j) + \int\limits_{t_j}^{t} \|\nabla s(\tau, y^N(\tau)) - \nabla s(t_j, y_j^N)\|_2 \, \|\left(\begin{smallmatrix}1\\ v_j\end{smallmatrix}\right)\|_2\, ds \\
&\leq -\alpha(t - t_j) + L_{\nabla s}\left(1 + C_F(1 + \tilde{M})\right)^2 h_N (t - t_j) \overset{N \geq N_0}{\leq} -\frac{\alpha}{2}(t - t_j) \leq 0
\end{aligned}$$

So we have $y^N(t) \in \Theta(t)$ for $t \in [t_j, t_{j+1})$ which leads to

$$\operatorname{dist}_2(y_j^N, \Theta(t)) \leq \|y_j^N - y^N(t)\|_2 = (t - t_j)\|v_j\|_2 \leq C_F(1 + \tilde{M})h_N \qquad (t \in [t_j, t_{j+1}))$$

Combining both cases delivers

$$\operatorname{dist}_2(y_j^N, \Theta(t)) \leq C_F(1+\tilde{M})h_N \qquad (t \in [t_j, t_{j+1}),\ j = 0, \dots, N-1)$$

So with ⑥ we get

⑦
$$\operatorname{dist}_2(\eta(t), \Theta(t)) \leq \big(L + \tilde{c} + C_F(1+\tilde{M})\big)h_N \qquad (t \in [t_0, T])$$

Using the triangular inequality and applying ④, ⑤ and ⑦ leads to

$$\begin{aligned} \|y_j^N - y(t_j)\|_2 \leq \|y_j^N - \eta(t_j)\|_2 + \|\eta(t_j) - y(t_j)\|_2 &\overset{\text{④,⑤}}{\leq} \tilde{c}h_N + C_c \operatorname{dist}_2(\eta(t), \Theta(t)) \\ &\overset{\text{⑦}}{\leq} \Big(\tilde{c} + C_c\big(L + \tilde{c} + C_F(1+\tilde{M})\big)\Big)\, h_N \qquad (j = 0, \dots, N) \end{aligned}$$

As the above inequality holds for all $j \in \{0, \dots, N\}$ we get for the second direction

$$\sup_{j=0,\dots,N} \|y_j^N - y(t_j)\|_2 \leq \Big(\tilde{c} + C_c\big(L + \tilde{c} + C_F(1+\tilde{M})\big)\Big)\, h_N$$

Combining the first and the second direction
Combining the results from the first and the second direction delivers

$$d_{H,\infty}^N \Big(\rho_N(X_\Theta(T, t_0, X_0)), X_\Theta^N(T, t_0, X_0)\Big) \leq \underbrace{\max\Big(\tilde{c} + C_d(1+\tilde{c}), \tilde{c} + C_c\big(L + \tilde{c} + C_F(1+\tilde{M})\big)\Big)}_{c:=} h_N$$

■

Let's conclude this section with ideas on how to handle multidimensional state constraints.

5.3.4 Multidimensional State Constraints

As already mentioned in the remarks to the assumptions (C1) and (C2), the fact that they only allow a single scalar state constraint is very restrictive. Even very simple examples involve more than one scalar state constraint (see chapter 6). It is more than desirable to get rid of that restriction. The core concept is to adjust (C1) and (C2) in such a way, that the proofs of Theorem 5.3.1 and Theorem 5.3.2 presented in [2] can be slightly altered to cover the multidimensional case. As this thesis does not cover this proofs, this section will focus on modifying (C1) and (C2) instead of explaining the modifications, that need to be done in the proofs, in detail.

modifying (C1)
The first thing to do should be obvious: The definition of $\Theta(.)$ will stay the same, but we will allow $s(.,.) \in \big(C^{1,L}(I \times R^n)\big)^{n_s}$ with $n_s \in \mathbb{N}$. So n_s functions in $C^{1,L}(I \times R^n)$ are allowed to describe the pure state constraints $\Theta(.)$. So we have:

$$\Theta(t) := \Big\{x \in \mathbb{R}^N \mid s_i(t,x) \leq 0 \ (i = 1, \dots, n_s)\Big\} \text{ with } s(.,.) \in \big(C^{1,L}(I \times R^n)\big)^{n_s}$$

Defining

$$\Theta_i(t) := \Big\{ x \in \mathbb{R}^N \mid s_i(t,x) \leq 0 \Big\} \text{ with } s_i(.,.) \in C^{1,L}(I \times R^n) \qquad (i = 1, \ldots, n_s)$$

leads to

$$\Theta(t) = \bigcap_{i=1}^{n_s} \Theta_i(t)$$

This definition comes in handy when replacing the postulation of when $x \in \partial\Theta(t)$. We now postulate that

$$x \in \partial\Theta_i(t) \Leftrightarrow s_i(t,x) = 0 \qquad (i = 1, \ldots, n_s)$$

This means for $\partial\Theta(t)$:

① $$x \in \partial\Theta(t) \Leftrightarrow \Big(\exists i \in \{1, \ldots, n_s\} \text{ with } s_i(t,x) = 0 \text{ and } x \in \Theta(t) \Big)$$

This property is important for the second direction in the proof of Theorem 5.3.3: Let's say we have $y_j^N \in \Theta(t)$ with $y_j^N \notin B_\mu(\operatorname{graph} \partial\Theta(.))$. Furthermore there exists $\tilde{t} \in [t_j, t_{j+1})$ such that $s_i(\tilde{t}, y_j^N) > 0$ $(i \in I)$ with $\emptyset \neq I \subset \{1, \ldots, n_s\}$. Then there exist $\tilde{t}_i$ $(i \in I)$ with $s_i(\tilde{t}_i, y_j^N) = 0$ and at least one index $\tilde{i} \in I$ with $y_j^N \in \Theta(\tilde{t}_{\tilde{i}})$. So from ① we get that $y_j^N \in \partial\Theta(\tilde{t}_{\tilde{i}})$.

So we have arrived at an extended form of (C1), which should be suitable for the multidimensional case. It shall be called (C1E), where E means extended.

(C1E) $\Theta : I \rightrightarrows \mathbb{R}^n$ *has nonempty images explicitly given by*

$$\Theta(t) = \bigcap_{i=1}^{n_s} \Theta_i(t)$$

with $n_s \in \mathbb{N}$ *and*

$$\Theta_i(t) := \Big\{ x \in \mathbb{R}^N \mid s_i(t,x) \leq 0 \Big\} \qquad (i = 1, \ldots, n_s)$$

where

$$s_i(.,.) \in C^{1,L}(I \times R^n) \qquad (i = 1, \ldots, n_s)$$

Furthermore $x \in \partial\Theta_i(t) \Leftrightarrow s_i(t,x) = 0$ $(i = 1, \ldots, n_s)$ *shall be fulfilled.*

An alternate description for (C1E) would be:

(C1E) $\Theta : I \Rightarrow \mathbb{R}^n$ *has nonempty images explicitly given by*

$$\Theta(t) = \Big\{x \in \mathbb{R}^N \;\Big|\; s_i(t,x) \leq 0 \; (i = 1, \ldots, n_s)\Big\}$$

with $n_s \in \mathbb{N}$ *and* $s(.,.) \in \big(C^{1,L}(I \times R^n)\big)^{n_s}$.

Furthermore

$$x \in \partial\Theta(t) \Leftrightarrow \Big(\exists i \in \{1, \ldots, n_s\} \text{ with } s_i(t,x) = 0 \text{ and } x \in \Theta(t)\Big)$$

shall be fulfilled.

modifying (C2)

When extending (C2) for the multidimensional case, the concept of active constraints comes into play. A constraint i, represented by $\Theta_i(.) = \{x \in \mathbb{R}^N \mid s_i(t,x) \leq 0\}$, is called active when $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_i(.)) \cap (I \times \mathbb{R}^n)$. The strict inwardness condition, stated in (C2), shall serve as an opportunity to redirect a solution that is close to the boundary of $\Theta(.)$ inwards. So for $(t,x) \in I \times \mathbb{R}^n$ this condition is only needed for constraints being active at that point. In addition the inward steering condition just needs to hold on $B_\mu(\operatorname{graph} \partial\Theta(.)) \cap (I \times \mathbb{R}^n)$. This is because points, which do not obey the state constraints, i.e. points (t,x) with $x \notin \Theta(t)$, do not have to be considered for redirecting solutions of the Differential Inclusion.
So for each $i \in \{1, \ldots, n_s\}$ it must hold: For all $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_i(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap (I \times \mathbb{R}^n)$ the following condition is fulfilled:

$$\min_{v \in F(t,x)} \langle \nabla s_i(t,x), \left(\begin{smallmatrix}1\\v\end{smallmatrix}\right) \rangle \leq -\alpha$$

So we have arrived at an extended form of (C2), which should be suitable for the multidimensional case. It shall be called (C2E), where E means extended.

(C2E) *The boundary of* $\Theta(.)$ *fulfills the "strict inwardness condition". This means that there exist* $\alpha, \mu > 0$ *such that for each* $i \in \{1, \ldots, n_s\}$ *it holds:*
For all $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_i(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap (I \times \mathbb{R}^n)$ *the following inequality applies:*

$$\min_{v \in F(t,x)} \langle \nabla s_i(t,x), \left(\begin{smallmatrix}1\\v\end{smallmatrix}\right) \rangle \leq -\alpha$$

extending the proofs

The proofs that need to be adjusted are the ones that belong to the Theorems involving pure state constraints. These are Theorem 5.3.1, Theorem 5.3.2 and Theorem 5.3.3. In modifying (C1) it has already been explained how to apply (C1E) to the proof of the Convergence Result given in 5.3.3. In general it is crucial for the proofs of the Theorems involving the discrete case, that the maximum step length is chosen small enough. The way to choose this maximum steplength usually involves the constants $\max\limits_{i=1\ldots,n_s}\{L_{\nabla s_i}\}$ and $\max\limits_{i=1\ldots,n_s}\{\max\limits_{(t,x)\in I\times S}\|\nabla s_i(t,x)\|\}$. For a single scalar state constraint there was no need to use the maximum. Those constants also play a great role in many estimations appearing in the proofs the mentioned theorems. Those estimations have to be done for every scalar state constraint involved and constants have to be chosen to fit all those cases. But the general way to proof things does not have to be altered.

5.4 Proof of the Convergence Theorem for Convex Discrete Differential Inclusions

This section contains the proof of Theorem 5.2.3.

5.4.1 Notations for the proof

To proof of the result involves quite a few steps. For the sake of readability, the following notations are introduced:
For easier function definitions it makes sense to take a look at a modified version of the Differential Inclusion (DI) introduced in Definition 3, which shall be named **(DIE)** (extended Differential Inclusion):

(DIE)

$$\begin{aligned} &z(t) := (t, x(t)) \in \mathbb{R}\times\mathbb{R}^n \text{ and fulfills}\\ &\dot{z}(t) \in \tilde{\boldsymbol{F}}(\boldsymbol{z(t)}) \boldsymbol{:= \{1\}} \times \boldsymbol{F(t, x(t))}\\ &z(t_0) \in \boldsymbol{\{t_0\}} \times \boldsymbol{X_0 := Z_0} \end{aligned}$$

with corresponding solution set $\boldsymbol{Z(T, t_0, Z_0)}$

This is essentially the same Differential Inclusion as (DI), except for the extension by the time variable. The corresponding discrete Differential Inclusion will be named **(DDIE)** and looks like:

(DDIE)

$$\begin{aligned} &z_j^N = (t_j, x_j^N) \in \mathbb{R}\times\mathbb{R}^n \ (j = 0, \ldots, N) \text{ and fulfills}\\ &z_{j+1}^N \in z_j^N + h_N\tilde{F}(z_j)\ (\tilde{F}(z_j) = \{1\}\times F(t_j, x_j^N))\\ &z_0^N = z_0 \in \{t_0\}\times X_0 := Z_0 \end{aligned}$$

with corresponding solution set $\boldsymbol{Z^N(T, t_0, Z_0)}$

The corresponding convexified discrete Differential Inclusion named **(CDDIE)** then looks like:

(CDDIE)

$$\begin{aligned} &z_j^N = (t_j, x_j^N) \in \mathbb{R}\times\mathbb{R}^n \ (j = 0, \ldots, N) \text{ and fulfills}\\ &z_{j+1}^N \in z_j^N + h_N\ co\tilde{F}(z_j)\ (\tilde{F}(z_j) = \{1\}\times F(t_j, x_j^N))\\ &z_0^N = z_0 \in \{t_0\}\times X_0 := Z_0 \end{aligned}$$

with corresponding solution set $\boldsymbol{Z_{co}^N(T, t_0, Z_0)}$ and *co* delivering the convex hull

The advantage of packing the time variable in z lies in the fact that the timepoint does not have to be passed explicitly to the following functions, which represent one step in Euler's Method in (DDIE) and the corresponding convexified discrete Differential Inclusion (CDDIE).

A lot of the following results depend on the space dimension of the space in which $\tilde{F}(z)$ is included. In this article the space dimension is $n+1$, but as the results do hold in general we introduce

$$\tilde{n} := n+1$$

and work with $\tilde{n}$ from here on.

Let $z \in R^{\tilde{n}}$, $A \subset R^{\tilde{n}}$ and $i \in \mathbb{N}$ then:

$$\Phi(z) := z + h_N \ \tilde{F}(z)$$
$$\Psi(z) := z + h_N \ co\tilde{F}(z)$$

$$\Phi(A) := \bigcup_{z \in A} \Phi(z) \qquad \Psi(A) := \bigcup_{z \in A} \Psi(z)$$
$$\Phi^i(z) := \underbrace{\Phi \circ \cdots \circ \Phi \circ \Phi}_{i\text{-times}}(z) \qquad \Psi^i(z) := \underbrace{\Psi \circ \cdots \circ \Psi \circ \Psi}_{i\text{-times}}(z)$$

The unit ball in $\mathbb{R}^{\tilde{n}}$ with respect to the $\|.\|_2$-norm will be called B:

$$B := \{z \in \mathbb{R}^{\tilde{n}} \mid \|z\|_2 \leq 1\}$$

Amongst others we will have to analyze (CDDIE), which involves convexified right-hand sides $co\tilde{F}$. To deal with $co\tilde{F}$ some results from convex analysis are needed.

5.4.2 Some results from convex analysis

Theorem
With the Minkowski Sum and scalar multiplication for sets defined in 2.4.3 it holds for a convex set A and $\alpha, \beta \geq 0$:

$$(\alpha+\beta)A = \alpha A + \beta A$$

Proof :
$(\alpha+\beta)A \subset \alpha A + \beta A$ should be clear.
The other way round follows from:

$$\alpha A + \beta A = (\alpha+\beta)\left(\frac{\alpha}{\alpha+\beta}A + \frac{\beta}{\alpha+\beta}A\right) \overset{\frac{\alpha}{\alpha+\beta}+\frac{\beta}{\alpha+\beta}=1}{\subset} (\alpha+\beta)A$$

Theorem (Carathéodory)
For $A \subset \mathbb{R}^d$ it holds

$$coA = \left\{ \sum_{i=1}^{d+1} \lambda_i a_i \;\middle|\; a_i \subset A, \ \sum_{i=1}^{d+1} \lambda_i = 1 \right\}$$

A proof can be found in [7].

Theorem
For any nonempty compact subsets $A, B \subset \mathbb{R}^d$ it holds:

$$d_{H,2}(coA, coB) \le d_{H,2}(A, B)$$

For a proof see [8].

For this proof we introduced $\tilde{F}$, which replaces F in this proof. Introducing $\tilde{F}$ would be worthless, if we could not make use of the properties of F. As the next section shows, the properties of F can be transferred one to one even to $co\tilde{F}$. The only thing that needs to be adjusted are the constants.

5.4.3 Passing properties of F on to $\tilde{F}$ and $co\tilde{F}$

If (A1) and (A2) hold for F, then it should be clear, that they hold for $\tilde{F}$, too. In this section some boundedness and Lipschitz-continuity results will be derived.

Boundedness
From the uniform boundedness of $y^N \in X^N(T, t_0, X_0)$ (see 2.6.3) and (A1) it follows that for $j \in \{0, \dots, N\}$

$$\|\tilde{F}(z_j^N)\|_2 = \|(1, F(t_j, y_j^N))\|_2 \le 1 + \|F(t_j, y_j^N)\|_2 \overset{\text{(A1)}}{\le} 1 + C_F(\|y_j^N\|_2 + 1) \overset{2.6.3}{\le} \underbrace{1 + C_F(\tilde{M} + 1)}_{K:=}$$

This means that ***$\tilde{F}$* is bounded for all valid solutions of (DDIE) by a constant *K***, which is independent of N.

To gain the same boundedness for $co\tilde{F}$ we first have to consider coF. Any $\xi \in coF(t, x)$ can be represented by

$$\xi = \sum_{i=1}^{k} \lambda_i \eta_i \qquad \left(\eta_i \in F(t,x)\ (i = 1, \dots, k),\ \sum_{i=1}^{k} \lambda_i = 1,\ \lambda_i \ge 0\ (i = 1, \dots, k),\ k \in \mathbb{N} \right)$$

So with (A1') we get

$$\|\xi\|_2 \le \sum_{i=1}^{k} \lambda_i \|\eta_i\|_2 \overset{(A1')}{\le} \sum_{i=1}^{k} (\lambda_i C_F(t)(\|x\|_2 + 1)) = C_F(t)(\|x\|_2 + 1) \sum_{i=1}^{k} \lambda_i = C_F(t)(\|x\|_2 + 1)$$

Hence

$$\|coF(t,x)\|_2 = \sup_{\xi \in coF(t,x)} \|\xi\|_2 \le C_F(t)(\|x\|_2 + 1)$$

So with the proof of Theorem 2.6.3 we get that for $x^N \in X_{co}^N(T, t_0, X_0)$ it holds $\|x_j^N\|_2 \le \tilde{M}$. This means that $X_{co}^N(T, t_0, X_0)$ is bounded by the same constant as $X^N(T, t_0, X_0)$.
Let $x^N \in X_{co}^N(T, t_0, X_0)$ and $z_j^N = (t_j, y_j^N)$ $(j = 0, \dots, N)$. Considering the fact that $co\tilde{F}(z_j^N) = (1, coF(t_j, x_j^N))$ we then get the same estimation for $\|co\tilde{F}(z_j^N)\|_2$ as we already have for $\|\tilde{F}(z_j^N)\|_2$:
Let $j \in \{0, \dots, N\}$ then

$$\|co\tilde{F}(z_j^N)\|_2 = \|(1, coF(t_j, x_j^N))\|_2 \le 1 + \|coF(t_j, x_j^N)\|_2 \le 1 + C_F(\|x_j^N\|_2 + 1) \overset{2.6.3}{\le} 1 + C_F(\tilde{M} + 1) = K$$

This means that $\tilde{F}$ **is bounded for all valid solutions of (CDDIE) by the constant** $\boldsymbol{K}$, which is independent of N.

Lipschitz-continuity
The Lipschitz-continuity property (A3) of F leads to Lipschitz-continuity of $\tilde F$, but with Lipschitz constant $\boldsymbol{L_{\tilde F} = \sqrt{2}\, L_F}$. This follows from:

Let $\tilde z = (\tilde t, y)$, $z = (t, x)$, $t, \tilde t \in I$ and $x, y \in \mathbb{R}^n$, then:

$$d_{H,2}\left(\tilde F(z), \tilde F(\tilde z)\right) \overset{\text{Def.}\tilde F}{=} d_{H,2}\left((1, F(t,x)), (1, F(\tilde t, y))\right) = d_{H,2}\left(F(t,x), F(\tilde t, y)\right)$$

$$\le L_F(|t - \tilde t| + \|x - y\|_2) = 2\, L_F(\frac{1}{2}\sqrt{|t-\tilde t|^2} + \frac{1}{2}\sqrt{\|x-y\|_2^2}) \overset{\text{concavity of } \sqrt{\cdot}}{\le}$$

$$2\, L_F \left(\frac{1}{2}|t - \tilde t|^2 + \frac{1}{2}\|x - y\|_2^2\right)^{1/2} = \sqrt{2}\, L_F \left(|t - \tilde t|^2 + \|x - y\|_2^2\right)^{1/2} = \underbrace{\sqrt{2}\, L_F}_{L_{\tilde F} :=} \|z - \tilde z\|_2$$

One additional property that is needed for the proof, concerns the Lipschitz-continuity of $co\tilde F$. Indeed, it follows directly from the Hausdorff-distance property in 5.4.2 that

$$\mathbf{d_{H,2}\left(co\tilde{F}(z), co\tilde{F}(\tilde{z})\right) \overset{5.4.2}{\le} L_{\tilde{F}} \|z - \tilde{z}\|_2}$$

We are now ready to start with the core part of the proof.

5.4.4 Overview of the proof

The basic idea is to use the general result: $(d+1)\, coA = d\, coA + A$ (for any set $A \subset \mathbb{R}^d$). In this case A will be $\tilde F(z_0)$ and the result from above will be applied repeatedly. It won't be possible, of course, to do this directly, because one step of the Euler scheme in (CDDIE) is represented by $z_{j+1}^N \in z_j^N + h_N\, co\,\tilde F(z_j)$ ($\tilde F$ has different arguments, only z_0 in the first step). But with the properties (A1) to (A3) for $\tilde F$ one can derive the deviation result $co\tilde F(\Psi^i(z_0)) \subset co\tilde F(z_0) + iKL_{\tilde F}\, h_N B$ with B being the unit ball with respect to the $\|.\|_2$-norm.

The base of the whole proof will be the following result.

5.4.5 Theorem (reduction of convexification for constant sets)

Let $A \subset \mathbb{R}^d$, then:

$$(d+1)\, coA = d\, coA + A$$

Proof :
The proof is quite lengthy and technical. In the author's view it would disturb the flow of reading to present it at this juncture. That is why the proof has been postponed to the end of this section (see Proof of 5.4.5).

This result applied to $A = \tilde{F}(z_0) \subset \mathbb{R}^{\tilde{n}}$ delivers

5.4.6 Lemma (convex deviation)

$$\Psi^{\tilde{n}+1}(z) \subset \Psi^{\tilde{n}}(\Phi(z)) + KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 B \qquad \text{for any } z \in \mathbb{R}^{\tilde{n}}$$

Note: *The structure of the above expression reflects Theorem 5.4.5: On the left hand side we have the convexified Euler step applied $\tilde{n}+1$ times whereas on the right-hand side it is only applied $\tilde{n}$ times and in the first step the not convexified Euler step is used. The term $KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 B$ is the deviation term that appears because $\tilde{F}$ is not constant. In fact this term represents the deviation from Euler's Method with the constant set $\tilde{F}(z)$.*

Proof :
From the fact that $\tilde{F}(\Psi^i(z)) \subset \bigcup_{z \in \Psi^i(z)} \tilde{F}(z)$ for $i \in \{1, \dots, N\}$ it follows from the definition of Ψ that:

① $$\Psi^{\tilde{n}+1}(z) = z + h_N\, co\tilde{F}(z) + h_N\, co\tilde{F}(\Psi(z)) + \cdots + h_N\, co\tilde{F}(\Psi^{\tilde{n}}(z))$$

From the boundedness of $\tilde{F}$ we get

$$\Psi^i(z) \subset \Psi^{i-1}(z) + Kh_N \Rightarrow d_{H,2}(\Psi^i(z), \Psi^{i-1}(z)) \leq Kh_N$$

Combining this result with the triangular inequality of the Hausdorff-distance (see 2.5.4) we get

$$d_{H,2}(\Psi^k(z), z) \leq \sum_{i=1}^{k} d_{H,2}(\Psi^i(z), \Psi^{i-1}(z)) \leq kKh_N$$

To use this result for estimating ① we make use of the Lipschitz-continuity of $co\tilde{F}$ (see 5.4.3), which yields

② $$\begin{aligned} &d_{H,2}\left(co\tilde{F}(\Psi^k(z)), co\tilde{F}(z)\right) \leq L_{\tilde{F}} d_{H,2}(\Psi^k(z), z) \leq L_{\tilde{F}} kKh_N \\ &\Rightarrow co\tilde{F}(\Psi^k(z)) \subset co\tilde{F}(z) + kL_{\tilde{F}}Kh_N B \end{aligned}$$

Applying this result to ① delivers

$$\begin{aligned} \Psi^{\tilde{n}+1}(z) \overset{\textcircled{2}}{\subset} z + (\tilde{n}+1)h_N\, co\tilde{F}(z) + \sum_{k=1}^{\tilde{n}} kL_{\tilde{F}}Kh_N^2 B = \\ z + (\tilde{n}+1)h_N\, co\tilde{F}(z) + \frac{\tilde{n}(\tilde{n}+1)}{2} L_{\tilde{F}}Kh_N^2 B \end{aligned}$$

This can now be further modified by applying **Theorem 5.4.5**

③
$$\Psi^{\tilde{n}+1}(z) \subset z + (\tilde{n}+1)h_N\, co\tilde{F}(z) + \frac{\tilde{n}(\tilde{n}+1)}{2} L_{\tilde{F}} K h_N^2 B \overset{5.4.5}{=}$$
$$z + h_N\, \tilde{F}(z) + \tilde{n}\, h_N\, co\tilde{F}(z) + \frac{\tilde{n}(\tilde{n}+1)}{2} L_{\tilde{F}} K h_N^2 B$$

Like in the derivation of ② we obtain

④
$$co\tilde{F}(z) \subset co\tilde{F}\big(\Psi^k(\Phi(z))\big) + (k+1)L_{\tilde{F}} K h_N B$$

Note: *$k+1$ appears due to the fact that $\Psi^k(\Phi(z))$ represents $k+1$ Euler steps. Also recall that the Lipschitz constants of $co\tilde{F}$ and $\tilde{F}$ are the same.*

Applying ④ to ③ delivers

$$\begin{aligned}
\Psi^{\tilde{n}+1}(z) &\overset{③}{\subset} z + h_N\, \tilde{F}(z) + \tilde{n}\, h_N\, co\tilde{F}(z) + \frac{\tilde{n}(\tilde{n}+1)}{2} L_{\tilde{F}} K h_N^2 B \\
&\overset{④}{\subset} z + h_N\, \tilde{F}(z) + h_N \sum_{k=0}^{\tilde{n}-1} \Big(co\tilde{F}\big(\Psi^k(\Phi(z))\big) + (k+1)L_{\tilde{F}} K h_N B \Big) + \frac{\tilde{n}(\tilde{n}+1)}{2} L_{\tilde{F}} K h_N^2 B \\
&= \left(z + h_N\, \tilde{F}(z) + \sum_{k=0}^{\tilde{n}-1} h_N\, co\tilde{F}\big(\Psi^k(\Phi(z))\big) \right) + \tilde{n}(\tilde{n}+1) L_{\tilde{F}} K h_N^2 B \\
&= \Psi^{\tilde{n}}(\Phi(z)) + \tilde{n}(\tilde{n}+1) L_{\tilde{F}} K h_N^2 B
\end{aligned}$$

■(Proof of Lemma 5.4.6)

Extending this result a little further gives us an essential result for this proof.

5.4.7 Theorem (consecutive convex deviation)

For $z \in \mathbb{R}^{\tilde{n}}$ and $\epsilon > 0$ it holds

$$\Psi\big(\Psi^{\tilde{n}}(z) + \epsilon B\big) \subset \Psi^{\tilde{n}}(\Phi(z)) + \Big(K L_{\tilde{F}} \tilde{n}(\tilde{n}+1) h_N^2 + \epsilon(1 + L_{\tilde{F}} h_N) \Big) B$$

Proof :
Like in the proof of Lemma 5.4.6 we obtain for $A \subset \mathbb{R}^{\tilde{n}}$

$$d_{H,2}\left(co\tilde{F}\big(A + \epsilon B\big),\, co\tilde{F}\big(A\big) \right) \leq L_{\tilde{F}}\ d_{H,2}\left(A + \epsilon B,\, A\right) = L_{\tilde{F}}\epsilon$$

So we have for $A = \Psi^{\tilde{n}}(z)$

①
$$d_{H,2}\left(co\tilde{F}\big(\Psi^{\tilde{n}}(z) + \epsilon B\big),\, co\tilde{F}\big(\Psi^{\tilde{n}}(z)\big) \right) \leq L_{\tilde{F}}\epsilon$$

Using this result after applying Ψ to $\Psi^{\tilde{n}}(z) + \epsilon B$ delivers

$$\begin{aligned}
\Psi\big(\Psi^{\tilde{n}}(z) + \epsilon B\big) &= \Psi^{\tilde{n}}(z) + \epsilon B + h_N co\tilde{F}\big(\Psi^{\tilde{n}}(z) + \epsilon B\big) \\
&\overset{\text{①}}{\subset} \Psi^{\tilde{n}}(z) + \epsilon B + h_N \Big(co\tilde{F}\big(\Psi^{\tilde{n}}(z)\big) + L_{\tilde{F}}\epsilon B\Big) \\
&= \Big(\Psi^{\tilde{n}}(z) + h_N co\tilde{F}\big(\Psi^{\tilde{n}}(z)\big)\Big) + \epsilon(1 + L_{\tilde{F}}h_N)B \\
&= \Psi^{\tilde{n}+1}(z) + \epsilon(1 + L_{\tilde{F}}h_N)B
\end{aligned}$$

Applying Lemma 5.4.6 to the result above delivers

$$\begin{aligned}
\Psi\big(\Psi^{\tilde{n}}(z) + \epsilon B\big) \;&\subset\; \Psi^{\tilde{n}+1}(z) + \epsilon(1 + L_{\tilde{F}}h_N)B \\
&\overset{5.4.6}{\subset}\; \Psi^{\tilde{n}}(\Phi(z)) + \tilde{n}(\tilde{n}+1)L_{\tilde{F}}Kh_N^2B + \epsilon(1 + L_{\tilde{F}}h_N)B \\
&=\; \Psi^{\tilde{n}}(\Phi(z)) + \Big(KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 + \epsilon(1 + L_{\tilde{F}}h_N)\Big)\, B
\end{aligned}$$

■(Proof of Theorem 5.4.7)

By considering solution paths and using Theorem 5.4.7 for going one Euler step further, it is now possible to conclude the proof.

5.4.8 Concluding the proof of the Convergence Theorem for convex discrete Differential Inclusions

Let $(\eta_k^N)_{k\in\{1,\dots,N\}}$ be a solution to (CDDIE) with starting point $\eta_0^N = z_0 \in Z_0$. In other words $(\eta^N \in Z_{co}^N(T, t_0, \{z_0\})$. What needs to be proven is that there exists a solution $(\xi_k^N)_{k\in\{1,\dots,N\}}$ to (DDIE) such that $\|\eta_k^N - \xi_k^N\|_2 \le C_S h_N$ $(k = 1, \dots, N)$. This will be done using induction. It makes sense to choose the same starting point for ξ^N as for η^N, so we have $\xi^N \in Z^N(T, t_0, \{z_0\})$.

Due to the fact that $(\eta_k^N)_{k\in\{1,\dots,N\}}$ is a solution of (CDDIE) we have

$$\eta_j^N \in \Psi^j(z_0) \qquad\qquad (j \in \{0, \dots, N\})$$

Initial Step
So we have for $j = \tilde{n}$

$$\eta_{\tilde{n}}^N \in \Psi^{\tilde{n}}(z_0) + \epsilon_0 B \qquad\qquad \text{with } \epsilon_0 := 0$$

From Theorem 5.4.7 it then follows that

$$\begin{aligned}
\eta_{\tilde{n}+1}^N \in \Psi(\eta_{\tilde{n}}^N) \;&\subset\; \Psi\big(\Psi^{\tilde{n}}(z_0) + \epsilon_0 B\big) \overset{\xi_0^N = \xi_0^N}{=} \Psi\big(\Psi^{\tilde{n}}(\xi_0^N) + \epsilon_0 B\big) \\
&\overset{5.4.7}{\subset}\; \Psi^{\tilde{n}}(\Phi(\xi_0^N)) + \Big(KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 + \epsilon_0(1 + L_{\tilde{F}}h_N)\Big)\, B \\
&=\; \Psi^{\tilde{n}}(\Phi(\xi_0^N)) + \underbrace{KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2}_{\epsilon_1 :=}\, B
\end{aligned}$$

So there exsits $\xi_1^N \in \Phi(\xi_0^N)$ such that

$$\eta_{\tilde{n}+1}^N \in \Psi^{\tilde{n}}(\xi_1^N) + \epsilon_1 B$$

This serves as the initial step. Next will be the induction step.

Induction Step

Suppose that

$$\eta^N_{\tilde{n}+k} \in \Psi^{\tilde{n}}(\xi^N_k) + \epsilon_k B \qquad\qquad (\epsilon_k \geq 0,\ \xi^N_k \in \mathbb{R}^{\tilde{n}})$$

Then it follows from Theorem 5.4.7 as in the case $k = 0$ above that

$$\begin{aligned}\eta^N_{\tilde{n}+k+1} \in \Psi(\eta^N_{\tilde{n}+k+1}) \ &\subset\ \Psi\big(\Psi^{\tilde{n}}(\xi^N_k) + \epsilon_k B\big) \overset{\xi^N_0 = \xi^N_0}{=} \Psi\big(\Psi^{\tilde{n}}(\xi^N_0) + \epsilon_0 B\big) \\ &\overset{5.4.7}{\subset}\ \Psi^{\tilde{n}}(\Phi(\xi^N_k)) + \underbrace{\Big(KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 + \epsilon_k(1 + L_{\tilde{F}}h_N)\Big)}_{\epsilon_{k+1}:=} B\end{aligned}$$

So there exsits $\xi^N_{k+1} \in \Phi(\xi^N_k)$ such that

$$\eta^N_{\tilde{n}+k+1} \in \Psi^{\tilde{n}}(\xi^N_{k+1}) + \epsilon_{k+1} B$$

Combining the initial step and the induction step we obtain that there exists $\xi^N \in Z^N(T, t_0, \{z_0\})$ such that for $k \in \{0, \dots, N - \tilde{n}\}$ it holds

①
$$\eta^N_{\tilde{n}+k} \in \Psi^{\tilde{n}}(\xi^N_k) + \epsilon_k B$$

with

$$\begin{aligned}\epsilon_0 &= 0 \\ \epsilon_{k+1} &= KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 + \epsilon_k(1 + L_{\tilde{F}}h_N)\end{aligned}$$

With the equations for $(\epsilon_k)_{k \in \{0,1,\dots,N-\tilde{n}\}}$ above one gets directly by analyzing the recursion that

②
$$\begin{aligned}\epsilon_k &= KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2 \sum_{i=0}^{k-1}(1 + L_{\tilde{F}}h_N)^i = KL_{\tilde{F}}\tilde{n}(\tilde{n}+1)h_N^2\,\frac{(1 + L_{\tilde{F}}h_N)^k - 1}{L_{\tilde{F}}h_N} \\ &\leq K\tilde{n}(\tilde{n}+1)h_N \left(e^{L_{\tilde{F}}h_N}\right)^k \overset{\text{Def. } h_N}{=} K\tilde{n}(\tilde{n}+1)e^{L_{\tilde{F}}k(T-t_0)/N}h_N \overset{k \leq N}{\leq} \underbrace{K\tilde{n}(\tilde{n}+1)\,e^{L_{\tilde{F}}(T-t_0)}h_N}_{\epsilon_\infty :=}\end{aligned}$$

Next we are going to estimate $d_{H,2}(\Psi^{\tilde{n}}(\xi^N_k),\, \xi^N_k)$ to obtain an estimation of $\|\eta^N_{\tilde{n}+k} - \xi^N_k\|_2$ from ①.

As already shown at the beginning of the proof of Lemma 5.4.6 it follows from the boundedness of $co\tilde{F}$ respectively $\tilde{F}$ that

③
$$\begin{aligned}d_{H,2}(\Psi^l(z), z) &\leq \sum_{i=1}^{l} d_{H,2}(\Psi^i(z), \Psi^{i-1}(z)) \leq lKh_N \qquad (l \in \{1, \dots, N\},\ z \in \mathbb{R}^{\tilde{n}}) \\ d_{H,2}(\Phi^l(z), z) &\leq \sum_{i=1}^{l} d_{H,2}(\Phi^i(z), \Phi^{i-1}(z)) \leq lKh_N \qquad (l \in \{1, \dots, N\},\ z \in \mathbb{R}^{\tilde{n}})\end{aligned}$$

Combining ③ and ② with ① by choosing $z = \xi_k^N$ and $l = \tilde{n}$ in ③ delivers

④
$$\begin{aligned}
&\eta^N_{\tilde{n}+k} \overset{\textcircled{1}}{\in} \Psi^{\tilde{n}}(\xi_k^N) + \epsilon_k B \overset{\textcircled{3}}{\subset} \xi_k^N + \tilde{n} K h_N B + \epsilon_k B \overset{\textcircled{2}}{\subset} \xi_k^N + (\tilde{n} K h_N + \epsilon_\infty)\, B \quad && (k \in \{0, \ldots, N-\tilde{n}\})\\
&\Rightarrow \|\eta^N_{\tilde{n}+k} - \xi_k^N\|_2 \le \tilde{n} K h_N + \epsilon_\infty && (k \in \{0, \ldots, N-\tilde{n}\})
\end{aligned}$$

As a direct consequence of ② one also gets by setting $z = \xi_k^N$ and considering that $\xi^N_{k+\tilde{n}} \in \Phi^{\tilde{n}}(\xi_k^N)$

⑤
$$\|\xi^N_{k+\tilde{n}} - \xi_k^N\|_2 \overset{\textcircled{2}}{\le} \tilde{n} K h_N \qquad (k \in \{0, \ldots, N-\tilde{n}\})$$

Combining ④ and ⑤ and using the definition of ϵ_∞ then yields

$$\begin{aligned}
\|\eta^N_{k+\tilde{n}} - \xi^N_{k+\tilde{n}}\|_2 &\le \|\eta^N_{k+\tilde{n}} - \xi_k^N\|_2 + \|\xi^N_{k+\tilde{n}} - \xi_k^N\|_2\\
&\overset{\textcircled{4},\textcircled{5}}{\le} \tilde{n} K h_N + K\tilde{n}(\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)} h_N + \tilde{n} K h_N\\
&= \left(2 + (\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)}\right) \tilde{n} K h_N \qquad (k \in \{0, \ldots, N-\tilde{n}\})
\end{aligned}$$

So we have the indices $j = \tilde{n}, \ldots, N$ covered and are only missing $j = 0, \ldots, \tilde{n}-1$. For these cases one easily obtains by using ② that

$$\begin{aligned}
\|\eta_j^N - \xi_j^N\|_2 &\le \|\eta_j^N - \eta_0^N\|_2 + \|\xi_j^N - \xi_0^N\|_2 \overset{\eta_0^N=\xi_0^N=z_0}{=} \|\eta_j^N - z_0\|_2 + \|\xi_j^N - z_0\|_2\\
&\overset{\textcircled{2}}{\le} jKh_N + jKh_N \overset{j<\tilde{n}}{<} 2\tilde{n} K h_N \le \left(2 + (\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)}\right) \tilde{n} K h_N \qquad (j \in \{0, \ldots, \tilde{n}-1\})
\end{aligned}$$

Combining the results for $j = \tilde{n}, \ldots, N$ and $j = 0, \ldots, \tilde{n}-1$ we get

⑥
$$\begin{aligned}
&\|\eta_j^N - \xi_j^N\|_2 \le \left(2 + (\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)}\right) \tilde{n} K h_N \qquad (j \in \{0, \ldots, N\})\\
\Leftrightarrow &\sup_{j=0,\ldots,N} \|\eta_j^N - \xi_j^N\|_2 \le \left(2 + (\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)}\right) \tilde{n} K h_N
\end{aligned}$$

From the notations section 5.4.1 we know that any element $\eta^N \in Z^N_{co}(T, t_0, \{z_0\})$ can be identified with an element $x^N \in X^N_{co}(T, t_0, \{z_0\})$ by

$$\eta_j^N = (t_j, x_j^N) \qquad (j = 0, \ldots, N)$$

Also any element $\xi^N \in Z^N(T, t_0, \{z_0\})$ can be identified with an element $y^N \in X^N(T, t_0, \{z_0\})$ by

$$\xi_j^N = (t_j, y_j^N) \qquad (j = 0, \ldots, N)$$

This leads to the final statement:
For any $x^N \in X^N_{co}(T, t_0, \{z_0\})$ there exists $y^N \in X^N(T, t_0, \{z_0\})$ such that

$$\sup_{j=0,\ldots,N} \|x^N_j - y^N_j\|_2 \leq \sup_{j=0,\ldots,N} \|\eta^N_j - \xi^N_j\|_2 \overset{\textcircled{6}}{\leq} \underbrace{\left(2 + (\tilde{n}+1)\, e^{L_{\tilde{F}}(T-t_0)}\right) \tilde{n} K}_{C_{S:=}} \, h_N$$

■(Proof of Theorem 5.2.3)

Due to its length, the proof of Theorem 5.4.5 has been postponed to the end of this section.

5.4.9 Proof of Theorem 5.4.5

The proof is heavily based on Carathéodorys Theorem 5.4.2. From this theorem we obtain that

$$coA = \bigcup_{\{A_d\}} coA_d$$

Where $\{A_d\} \subset A$ denotes a set with at most $d+1$ elements.
It is then sufficient to show that $(d+1)\; coA_d = d\; coA_d + A_d$, because with that result and the representation of coA from above we get:

$$(d+1)\; coA = \bigcup_{\{A_d\}} (d+1)\; coA_d = \bigcup_{\{A_d\}} (d\; coA_d + A_d) = d \bigcup_{\{A_d\}} coA_d + \bigcup_{\{A_d\}} A_d = d\; coA + A$$

Where $(d+1)\; coA = \bigcup_{\{A_d\}}(d+1)\; coA_d$ has to do with the fact that $\alpha\; coA + \beta\; coA = (\alpha+\beta)\; coA$, which has been proven in 5.4.2. From this point on the proof is pretty straightforward, but quite technical and lengthy.

So let us consider an arbitrary set $A_d = \{z_1, \ldots, z_{d+1}\} \subset A$ consisting of $d+1$ elements. It then holds:

$$(d+1)\; coA_d = d\; coA_d + A_d \iff \left(\begin{array}{l} \text{For } (\lambda_{i,j})^{i=1,\ldots,d+1}_{j=1,\ldots,d+1} \text{ with } \sum\limits_{j=1}^{d+1} \lambda_{i,j} = 1 \; (i = 1, \ldots, d+1) \\ \text{and } \lambda_{i,j} \geq 0 \; (i, j = 1, \ldots, d+1) \\ \text{there exists} \\ (\tilde{\lambda}_{i,j})^{i=1,\ldots,d}_{j=1,\ldots,d+1} \text{ with } \sum\limits_{j=1}^{d+1} \tilde{\lambda}_{i,j} = 1 \; (i = 1, \ldots, d) \\ \text{and } \tilde{\lambda}_{i,j} \geq 0 \; (i = 1, \ldots, d;\; j = 1, \ldots, d+1) \\ \text{such that} \\ \sum\limits_{i=1}^{d+1} \left(\sum\limits_{j=1}^{d+1} \lambda_{i,j}\, z_j \right) = \sum\limits_{i=1}^{d} \left(\sum\limits_{j=1}^{d+1} \tilde{\lambda}_{i,j}\, z_j \right) + z_{d+1} \end{array} \right)$$

Remarks:

- Without loss of generality the index $d+1$ is chosen in such a way that

  $$\textcircled{1}\;\Bigg| \qquad \sum_{i=1}^{d+1} \lambda_{i,d+1} \geq 1$$

  otherwise reorder the elements of A_d. **This is only possible because at least d+1 elements are summed up. This is the reason, why this theorem just holds for $(d+1)$ *coA* or the same with any higher scalar than $d+1$.** The fact that such an index exists can be easily shown:
  Suppose such an index does not exist, i.e.:

  $$\sum_{i=1}^{d+1} \lambda_{i,j} < 1 \; (j = 1, \ldots, d+1)$$

  This leads to

  $$\sum_{j=1}^{d+1} \left(\sum_{i=1}^{d+1} \lambda_{i,j} \right) < d+1$$

  which is obviously a contradiction to

  $$\sum_{j=1}^{d+1} \lambda_{i,j} = 1 \; (i = 1, \ldots, d+1) \Rightarrow \sum_{i=1}^{d+1} \sum_{j=1}^{d+1} \lambda_{i,j} = d+1$$

- The case that A_d consists of $k < d+1$ elements does not have to be considered separately, because it is represented by setting $\lambda_{i,j} = 0$ ($i = 1, \ldots, d+1$; $j = k+1, \ldots, d+1$).

A more optically appealing way of the new represenation introduced above is:

$$\begin{pmatrix} \lambda_{1,1} \\ + \\ \vdots \\ + \\ \lambda_{d,1} \\ + \\ \lambda_{d+1,1} \end{pmatrix} z_1 + \cdots + \begin{pmatrix} \lambda_{1,d} \\ + \\ \vdots \\ + \\ \lambda_{d,d} \\ + \\ \lambda_{d+1,d} \end{pmatrix} z_d + \begin{pmatrix} \lambda_{1,d+1} \\ + \\ \vdots \\ + \\ \lambda_{d,d+1} \\ + \\ \lambda_{d+1,d+1} \end{pmatrix} z_{d+1} =$$

$$\begin{pmatrix} \tilde{\lambda}_{1,1} \\ + \\ \vdots \\ + \\ \tilde{\lambda}_{d,1} \end{pmatrix} z_1 + \cdots + \begin{pmatrix} \tilde{\lambda}_{1,d} \\ + \\ \vdots \\ + \\ \tilde{\lambda}_{d,d} \end{pmatrix} z_d + \begin{pmatrix} \tilde{\lambda}_{1,d+1} \\ + \\ \vdots \\ + \\ \tilde{\lambda}_{d,d+1} \end{pmatrix} z_{d+1} + z_{d+1}$$

From this we can directly derive the central equations that have to be fulfilled with $(\tilde{\lambda}_{i,j})$ by comparing both sides for each vector z_j $(j = 1, \ldots, d+1)$:

②
$$\sum_{i=1}^{d+1} \lambda_{i,j} = \sum_{i=1}^{d} \tilde{\lambda}_{i,j} \qquad (j = 1, \ldots, d)$$

and for the index $d+1$, which should be handled separately

③
$$\sum_{i=1}^{d+1} \lambda_{i,d+1} = \sum_{i=1}^{d} \tilde{\lambda}_{i,d+1} + 1$$

All that is left to do now is to change the weights $(\lambda_{i,j})$, while keeping balance, i.e. fulfilling the equaitons in ② and ③.

Let's start with ③ first. The idea is to look at a slightly different form of ③, i.e.:

$$\sum_{i=1}^{d} \lambda_{i,d+1} = \sum_{i=1}^{d} \tilde{\lambda}_{i,d+1} + (1 - \lambda_{d+1.d+1})$$

Then "transfer" the weights from the left to the right side:
Start with $\lambda_{1,d+1}$ and look if it is smaller or equal than $1 - \lambda_{d+1,d+1}$. If so, set $\tilde{\lambda}_{1,d+1} := 0$. Then take a look at $\lambda_{2,d+1}$. If it is smaller or equal than $1 - \lambda_{d+1,d+1} - \lambda_{1,d+1}$ set $\tilde{\lambda}_{2,d+1} := 0$. For the next one, one has to compare to $1 - \lambda_{d+1,d+1} - \sum_{i=1}^{2} \lambda_{i,d+1}$. This procedure shall be carried on till the index $\tilde{i}$ is reached at which $\lambda_{\tilde{i},d+1} > 1 - \lambda_{d+1,d+1} - \sum_{i=1}^{\tilde{i}-1} \lambda_{i,d+1}$. At that point set $\tilde{\lambda}_{\tilde{i},d+1} := \lambda_{\tilde{i},d+1} - \left(1 - \lambda_{d+1,d+1} - \sum_{i=1}^{\tilde{i}-1} \lambda_{i,d+1}\right)$. That is all that was to "transfer". From $\tilde{i}+1$ on leave $\lambda_{i,d+1}$ as is, which means $\tilde{\lambda}_{i,d+1} := \lambda_{i,d+1}$. The whole process is described via the following definition:

④
$$\boxed{\tilde{\lambda}_{i,d+1} := \lambda_{i,d+1} - \min\left(\lambda_{i,d+1},\ 1 - \lambda_{d+1,d+1} + \sum_{k=1}^{i-1} \mu_{k,d+1}\right) \qquad (i = 1, \ldots, d)}$$

Where

$$\mu_{i,j} := \tilde{\lambda}_{i,j} - \lambda_{i,j} \qquad (i, j = 1, \ldots, d+1)$$

What is left to do is choose $\tilde{\lambda}_{i,j}$ $(i, j = 1, \ldots, d)$ in such a way, that ② is fulfilled and $\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} = 1$ $(i = 1, \ldots, d)$ holds. That this is possible should be more clearly when looking at the following coherence, which follows directly from the properties of the weights:

$$\sum_{i=1}^{d} \lambda_{i,d+1} - \sum_{i=1}^{d} \tilde{\lambda}_{i,d+1} \overset{\text{③}}{=} 1 - \lambda_{d+1,d+1} = \sum_{j=1}^{d} \lambda_{d+1,j}$$

This means that the difference of the weights, that occurred when achieving ③, can be compensated by adding $\lambda_{d+1,j}$ to $\lambda_{i,j}$ $(i = 1, \ldots, d)$ for arbitrary but fixed $j \in \{1, \ldots, d\}$.

That way the goal of fulfilling ② and $\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} = 1$ $(i = 1, \dots, d)$ can be achieved.
The idea is going through the whole system line by line $(i = 1, \dots, d)$ starting each line at $j = 1$. While going through the lines we add as much as we can to each weight, obeying the following rules: In each line i, the total amount added to the weights has to be smaller than $\lambda_{i,d+1} - \tilde{\lambda}_{i,d+1} = -\mu_{i,d+1}$ and the total amount added in each row j has to be smaller than $\lambda_{d+1,j}$. These rules are applied successively to each of the weights $\lambda_{i,j}$ $(i, j = 1, \dots, d)$ which then gives the desired $\tilde{\lambda}_{i,j}$ $(i, j = 1, \dots, d)$. As the procedure is somehow similar to the one we already used for obtaining the weights for ③, the result looks a bit like ④. But instead of subtracting, which has been done before, we this time add to the weights. Doing so we get the following formula for successively obtaining $\tilde{\lambda}_{i,j}$ $(i, j = 1, \dots, d)$:

$$\text{⑤} \quad \boxed{\tilde{\lambda}_{i,j} = \lambda_{i.j} + \min\left(-\mu_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k},\ \lambda_{d+1,j} - \sum_{k=1}^{i-1} \mu_{k,j}\right) \qquad (i, j = 1, \dots, d)}$$

With ④ and ⑤ we have definitions for $\tilde{\lambda}_{i,j}$ $(i = 1, \dots, d;\ j = 1, \dots, d+1)$. But we have yet to prove that these definitions indeed fulfill ②, ③ and that we are indeed dealing with convex combinations. The latter one means that $\tilde{\lambda}_{i,j} \geq 0$ $(i = 1, \dots, d;\ j = 1, \dots, d+1)$ and $\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} = 1$ $(i = 1, \dots, d)$ have to be shown. This will be the starting point for the quite lenghty, but always straightforward proof.

Assertion:

$$\tilde{\lambda}_{i,j} \geq 0 \qquad\qquad (i = 1, \dots, d;\ j = 1, \dots, d+1)$$

Proof :
From ⑤ it directly follows that

$$\sum_{k=1}^{i} \mu_{k,j} \overset{⑤}{=} \sum_{k=1}^{i-1} \mu_{k,j} + \min\left(\lambda_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k},\ \lambda_{d+1,j} - \sum_{k=1}^{i-1} \mu_{k,j}\right) \leq \lambda_{d+1,j}$$

$$\Leftrightarrow \lambda_{d+1,j} - \sum_{k=1}^{i} \mu_{k,j} \geq 0 \qquad\qquad (i, j = 1, \dots, d)$$

and

$$\sum_{k=1}^{j} \mu_{i,k} \overset{⑤}{=} \sum_{k=1}^{j-1} \mu_{i,k} + \min\left(\lambda_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k},\ \lambda_{d+1,j} - \sum_{k=1}^{i-1} \mu_{k,j}\right) \leq \lambda_{i,d+1}$$

$$\Leftrightarrow \lambda_{i,d+1} - \sum_{k=1}^{j} \mu_{i,k} \geq 0 \qquad\qquad (i, j = 1, \dots, d)$$

Combining those two results with ⑤ then delivers:

$$\tilde{\lambda_{i,j}} = \lambda_{i.j} + \min\Bigg(\underbrace{\lambda_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k},}_{\geq 0}\ \underbrace{\lambda_{d+1,j} - \sum_{k=1}^{i-1} \mu_{k,j}}_{\geq 0}\Bigg) \geq \lambda_{i,j} \geq 0 \qquad (i, j = 1, \dots, d)$$

The desired result for $j = d+1$ can be easily obtained from ④:

$$\tilde{\lambda}_{i,d+1} = \lambda_{i,d+1} - \min\left(\lambda_{i,d+1},\ 1 - \lambda_{d+1,d+1} + \sum_{k=1}^{i-1} \mu_{k,d+1}\right) \geq \lambda_{i,d+1} - \lambda_{i,d+1} = 0 \quad (i = 1, \dots, d)$$

Combining both results delivers the desired statement for $i = 1, \dots, d$ and $j = 1, \dots, d+1$.

■(Proof of Assertion)

As a byproduct of this proof we got that

⑥
$$\lambda_{d+1,j} - \sum_{k=1}^{i} \mu_{k,j} \geq 0 \text{ and } \lambda_{i,d+1} - \sum_{k=1}^{j} \mu_{i,k} \geq 0 \qquad (i, j = 1, \dots, d)$$

When proving properties of ⑤ this will turn out to be quite useful.
As already mentioned the next step of the proof is to show that $\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} = 1$ $(i = 1, \dots, d)$. The idea is to show that

⑦
$$\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} = \sum_{j=1}^{d+1} \lambda_{i,j} = 1 \qquad (i = 1, \dots, d)$$

This will be done using induction starting with $i = 1$ as the initial step.

Assertion:

⑧
For $i = 1, \dots, d$ there exists $\tilde{j} \in \{1, \dots, d\}$ such that

$$\lambda_{i,d+1} - \sum_{k=1}^{\tilde{j}-1} \mu_{i,k} \leq \lambda_{d+1,\tilde{j}} - \sum_{k=1}^{i-1} \mu_{k,\tilde{j}}$$

which leads directly to ⑦ being fulfilled with $\tilde{\lambda}_{i,j}$ $(i = 1, \dots, d;\ j = 1, \dots, d+1)$
Proof :
The proof uses induction. So we start with the case $i = 1$.
Initial Step: The statement holds for $i = 1$.
Let's suppose the opposite, i.e.

$$-\mu_{1,d+1} - \sum_{k=1}^{j-1} \mu_{1,k} > \lambda_{d+1,j} \qquad (j = 1, \dots, d)$$

Then we get from ⑤ and ④, that

$$\sum_{k=1}^{d} \mu_{1,k} \overset{⑤}{=} \sum_{k=1}^{d} \lambda_{d+1,k} = 1 - \lambda_{d+1,d+1} \overset{④}{\geq} -\mu_{1,d+1} \Rightarrow -\mu_{1,d+1} - \sum_{k=1}^{d-1} \mu_{1,k} \leq \mu_{1,d}$$

⑤ also delivers:

$$\mu_{1,d} \overset{⑤}{=} \min\left(-\mu_{1,d+1} - \sum_{k=1}^{d-1} \mu_{1,k},\ \lambda_{d+1,d}\right) \leq \lambda_{d+1,d}$$

Combining those results we get:

$$-\mu_{1,d+1} - \sum_{k=1}^{d-1} \mu_{1,k} \leq \lambda_{d+1,d}$$

So for $j = d$ we get a contradiction to the assumption not being true, which means it has to be true.
With ⑧ being true for $i = 1$ it is possible to prove that ⑦ holds for $i = 1$:
Let's choose $\tilde{j}$ as in ⑧ for $i = 1$. Combining ⑤ and ⑧ for $i = 1$ then delivers

$$\mu_{1,\tilde{j}} \overset{⑤}{=} \min\left(-\mu_{1,d+1} - \sum_{k=1}^{\tilde{j}-1} \mu_{1,k},\ \lambda_{d+1,\tilde{j}}\right) \overset{⑧}{=} -\mu_{1,d+1} - \sum_{k=1}^{\tilde{j}-1} \mu_{1,k} \Leftrightarrow -\mu_{1,d+1} = \sum_{k=1}^{\tilde{j}} \mu_{1,k}$$

If $\tilde{j} = d$, this is ⑦ for $i = 1$, so we are done. For $\tilde{j} < d$ one gets $\mu_{1,\tilde{j}+1} = 0$ and so forth by successively applying ⑤. This leads to

$$-\mu_{1,d+1} = \sum_{k=1}^{d} \mu_{1,k}$$

So overall we get that $\tilde{\lambda}_{1,j}$ $(j = 1, \dots, d+1)$ fulfills ⑦ for $i = 1$.

■($i = 1$, initial step)

Next comes the induction step:
Induction Step: Let ⑦ be fulfilled for $\lambda_{i,j}$ $(i = 1, \dots, \tilde{i} - 1;\ j = 1, \dots, d+1)$. Then the statement holds for $\tilde{i}$.
The way to go is almost the same as for the case $i = 1$, it just involves the inductive assumption:
Let's suppose that the inductive assumption is true, but the statement does not hold for $i = \tilde{i}$, i.e.

$$-\mu_{\tilde{i},d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k} > \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} \qquad (j = 1, \dots, d)$$

It then follows directly from ⑤ that

$$\mu_{\tilde{i},j} = \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} \qquad (j = 1, \dots, d)$$

The inductive assumption can be written as

$$\sum_{j=1}^{d} \mu_{\tilde{i},j} = -\mu_{i,d+1} \qquad (i = 1, \dots, \tilde{i} - 1)$$

These two results and ④ deliver

$$\sum_{j=1}^{d} \mu_{\tilde{i},j} \overset{⑤}{=} \sum_{j=1}^{d} \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \sum_{j=1}^{d} \mu_{k,j} \underset{\text{assumption}}{\overset{\text{inductive}}{=}} \sum_{j=1}^{d} \lambda_{d+1,j} + \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1} =$$

$$1 - \lambda_{d+1,d+1} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1} \overset{④}{\geq} -\mu_{\tilde{i},d+1} \Rightarrow -\mu_{\tilde{i},d+1} - \sum_{j=1}^{d-1} \mu_{\tilde{i},j} \leq \mu_{\tilde{i},d}$$

⑤ delivers:

$$\mu_{\tilde{i},d} \overset{⑤}{=} \min\left(\lambda_{\tilde{i},d+1} - \sum_{k=1}^{d-1}\mu_{\tilde{i},k},\ \lambda_{d+1,d} - \sum_{k=1}^{\tilde{i}-1}\mu_{k,d}\right) \le \lambda_{d+1,d} - \sum_{k=1}^{\tilde{i}-1}\mu_{k,d}$$

Combining those results we get:

$$-\mu_{\tilde{i},d+1} - \sum_{j=1}^{d-1}\mu_{\tilde{i},d} \le \lambda_{d+1,d} - \sum_{k=1}^{\tilde{i}-1}\mu_{k,d}$$

So for $j = d$ we get a contradiction to the statement not being true for $i = \tilde{i}$, supposed that the inductive assumption holds. This means it has to be true. So if the inductive assumption holds we get ⑧ for $i = \tilde{i}$.

With ⑧ for $i = \tilde{i}$ it is possible to prove that ⑦ holds for $i = \tilde{i}$:
Let's choose $\tilde{j}$ as in ⑧ for $i = \tilde{i}$. Combining ⑤ and ⑧ with $j = \tilde{j}$ then delivers

$$\mu_{\tilde{i},\tilde{j}} \overset{⑤}{=} \min\left(-\mu_{\tilde{i},d+1} - \sum_{k=1}^{\tilde{j}-1}\mu_{\tilde{i},k},\ \lambda_{d+1,\tilde{j}} - \sum_{k=1}^{\tilde{i}-1}\mu_{k,\tilde{j}}\right) \overset{⑧}{=} -\mu_{\tilde{i},d+1} - \sum_{k=1}^{\tilde{j}-1}\mu_{\tilde{i},k} \Leftrightarrow -\mu_{\tilde{i},d+1} = \sum_{k=1}^{\tilde{j}}\mu_{\tilde{i},k}$$

If $\tilde{j} = d$, this is ⑦ for $i = \tilde{i}$, so we are done. For $\tilde{j} < d$ one gets $\mu_{\tilde{i},\tilde{j}+1} = 0$ and so forth by successively applying ⑤. This leads to

$$-\mu_{\tilde{i},d+1} = \sum_{k=1}^{d}\mu_{\tilde{i},k}$$

So overall we get that $\tilde{\lambda}_{\tilde{i},j}$ $(j = 1, \ldots, d+1)$ fulfills ⑦.

■($\tilde{i} - 1$ to $\tilde{i}$, induction step)

The initial step together with the induction step deliver the assertion.

■(Proof of the Assertion)

As a byproduct of the above assertion we have proven, that ⑦ indeed holds with the definitions of $\tilde{\lambda}_{i,j}$ $(i = 1, \ldots, d;\ j = 1, \ldots, d+1)$. Of course ⑦ is the result we wanted to prove, but ⑧ is the central idea for reaching that goal.

After having shown that $\tilde{\lambda}_{i,j} \ge 0$ $(i = 1, \ldots, d;\ j = 1, \ldots, d+1)$ and $\sum_{j=1}^{d+1}\tilde{\lambda}_{i,j} = 1(i = 1, \ldots, d)$ the "only" thing left to show is that ② and ③ hold for $\tilde{\lambda}_{i,j}$ $(i = 1, \ldots, d;\ j = 1, \ldots, d+1)$. Let's start with ③, which can be shown by taking a look at column $d+1$. The reason for starting with ③ instead of ② is that on the right-hand side of definition ④ $\tilde{\lambda}_{i,j}$ does not appear. In ⑤ this is not the case. And indeed we need the following result to go on with proving ②.

Assertion

⑨ *$\exists \tilde{i} \in \{1, \ldots, d\}$ such that*

$$1 - \lambda_{d+1,d+1} + \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1} \leq \lambda_{\tilde{i},d+1}$$

Proof :
Let's suppose the opposite, i.e.

$$1 - \lambda_{d+1,d+1} + \sum_{k=1}^{i-1} \mu_{k,d+1} > \lambda_{i,d+1} \qquad\qquad (i = 1, \ldots, d)$$

Then we get from ④ and ①, that

$$\sum_{k=1}^{d-1} \mu_{k,d+1} \overset{④}{=} -\sum_{k=1}^{d-1} \lambda_{k,d+1} \overset{①}{\leq} -1 + \lambda_{d+1,d+1} + \lambda_{d,d+1} \Rightarrow 1 - \lambda_{d+1,d+1} + \sum_{k=1}^{d-1} \mu_{k,d+1} \leq \lambda_{d,d+1}$$

So for $i = d$ we get a contradiction to the assumption not being true, which means it has to be true. ■*(proof of ⑨)*

Choosing $\tilde{i}$ like in ⑨ and combining ⑨ with ④ delivers:

$$\mu_{\tilde{i},d+1} \overset{④}{=} -\min\left(\lambda_{\tilde{i},d+1},\ 1 - \lambda_{d+1,d+1} + \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1}\right) \overset{⑨}{=} -1 + \lambda_{d+1,d+1} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1}$$

$$\Leftrightarrow \lambda_{d+1,d+1} - \sum_{k=1}^{\tilde{i}} \mu_{k,d+1} = 1$$

If $\tilde{i} = d$ this is ③. For $\tilde{i} < d$ we get $\mu_{\tilde{i}+1,d+1} = 0$ and so forth, so again we get

$$\lambda_{d+1,d+1} - \sum_{k=1}^{d} \mu_{k,d+1} = 1$$

So $\tilde{\lambda}_{i,d+1}$ $(i = 1, \ldots, d)$ fulfills ③.

We are now ready to prove that ② is fulfilled. The central idea is the same as for proving ⑦, but this time we look at the definitions column by column (j fixed) instead of line by line (i fixed).

Assertion:

⑩ For $j \in \{1, \ldots, d\}$ there exists $\tilde{i} \in \{1, \ldots, d\}$ such that

$$\lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} \leq -\mu_{\tilde{i},d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k}$$

Proof :
Assume that the assertion does not hold, i.e.

$$\lambda_{d+1,j} - \sum_{k=1}^{i-1} \mu_{k,j} > -\mu_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k} \qquad (i = 1, \dots, d)$$

From this inequality one directly obtains with ⑤

$$\mu_{i,j} \overset{\textcircled{5}}{=} -\mu_{i,d+1} - \sum_{k=1}^{j-1} \mu_{i,k} \Leftrightarrow -\mu_{i,d+1} = \mu_{i,j} + \sum_{k=1}^{j-1} \mu_{i,k} \qquad (i = 1, \dots, d)$$

Let's choose $\tilde{i}$ as in ⑨. From the equation above and ⑨ we then get

$$\begin{aligned}
&- \mu_{\tilde{i},d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k} \overset{\textcircled{4}}{=} \min\left(\lambda_{\tilde{i},d+1},\ 1 - \lambda_{d+1,d+1} + \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1}\right) - \sum_{k=1}^{j-1} \mu_{\tilde{i},k} \overset{\textcircled{9}}{=} \\
&1 - \lambda_{d+1,d+1} + \sum_{k=1}^{\tilde{i}-1} \mu_{k,d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k} = 1 - \lambda_{d+1,d+1} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} - \sum_{k=1}^{\tilde{i}-1} \sum_{l=1}^{j-1} \mu_{k,l} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k} = \\
&\sum_{k=1}^{d} \lambda_{d+1,k} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} - \sum_{l=1}^{j-1} \sum_{k=1}^{\tilde{i}} \mu_{k,l} \geq \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} + \sum_{k=1}^{j-1} \lambda_{d+1,k} - \sum_{l=1}^{j-1} \sum_{k=1}^{\tilde{i}} \mu_{k,l} \overset{\textcircled{6}}{\geq} \\
&\lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} + \sum_{k=1}^{j-1} \lambda_{d+1,k} - \sum_{l=1}^{j-1} \lambda_{d+1,l} = \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j}
\end{aligned}$$

So overall we have

$$\lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} \leq -\mu_{\tilde{i},d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k}$$

So the case $i = \tilde{i}$ delivers a contradiction to the assumption of the assertion not being true. So it has to be true. ■(proof of ⑩)

---

⑩ leads directly to ② being fulfilled with $\tilde{\lambda}_{i,j}$ $(i, j = 1, \dots, d)$. The idea to show that should be familiar by now:
Choose $\tilde{i}$ as in ⑩, then it follows from ⑤ that

$$\mu_{\tilde{i},j} \overset{\textcircled{5}}{=} \min\left(-\mu_{\tilde{i},d+1} - \sum_{k=1}^{j-1} \mu_{\tilde{i},k},\ \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j}\right) \overset{\textcircled{10}}{=} \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}-1} \mu_{k,j} \Leftrightarrow \lambda_{d+1,j} - \sum_{k=1}^{\tilde{i}} \mu_{k,j} = 0$$

If $\tilde{i} = d$ this is ②. For $\tilde{i} < d$ we get $\mu_{\tilde{i}+1,j} = 0$ and so forth and finally obtain

$$\lambda_{d+1,j} - \sum_{k=1}^{d} \mu_{k,j} = 0$$

The fact that for any $j \in \{1, \dots, d\}$ there exists such an index $\tilde{i}$ leads to $\tilde{\lambda}_{i,j}$ $(i, j = 1, \dots, d)$ fulfilling ② for $j = 1, \dots, d$.

So overall with ④ and ⑤ we found a solution to

$$\begin{aligned}
\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j} &= 1 && (i = 1, \dots, d)\\
\tilde{\lambda}_{i,j} &\geq 0 && (i = 1, \dots, d;\ j = 1, \dots, d+1)\\
\sum_{i=1}^{d+1} \left(\sum_{j=1}^{d+1} \lambda_{i,j}\, z_j \right) &= \sum_{i=1}^{d} \left(\sum_{j=1}^{d+1} \tilde{\lambda}_{i,j}\, z_j \right) + z_{d+1}
\end{aligned}$$

■(Proof of Theorem 5.4.5)

6 Examples

6.1 Overview

In this section some examples are presented, which either substantiate the results from chapter 3 or explore theoretically only partially covered cases of this article. This involves simple cases of multidimensional state constraints and some investigations on the assumption (C2) from 5.3.

6.1.1 details on calculations and notations

Form of the optimization problem
All problems are presented in Bolza form (see 2.3.1). The discrete solution is obtained using numerical algorithms to solve the Discrete Mayer-Problem 2.4.2. The error that occurs when solving the Discrete Mayer-Problem via a certain optimization algorithm is not taken into account, but is of course present. This should be kept in mind when considering the following numerical results.
The software used solves exactly the Discrete Mayer-Problem described in this paper. The ODE is also solved using the forward Euler Method, so despite some numerical errors, the computed results should exactly represent the solution $(\hat{x}^N, \hat{u}^N)$ from this paper. Nevertheless there are some minor differences. First, because of Euler's method, it does not make sense to calculate the optimal control on the last point of the grid. So all control vectors are missing the last m components. The second thing is that the form of the constraints is a bit altered to compensate for the lack of the last component of the control vectors. But these are just minor differences and should not have any real impact on the final result.
The reader of this chapter should also be aware that $\hat{x}^N$ splits up into $\hat{\tilde{x}}^N$ and $\hat{z}^N$ (see Notations in 2.4.2), where $\hat{\tilde{x}}^N$ is the optimal discrete state function to the discrete Bolza-Problem. The directly discretized Bolza-Problem has not been introduced in this thesis, because it is almost the same as the Discrete Mayer-Problem presented in 2.4.2 and the theoretical focus lies on the Mayer form of the optimization problem. The only thing to do to obtain the Discrete Bolza-Problem from the Discrete Mayer-Problem is to write down what z^N exactly represents. That way one directly sees that usage of Euler's Method in the Discrete Mayer-Problem to obtain z^N directly transfers into a Riemann sum for the integral term in the objective function, when considering the Discrete Bolza-Problem.
All convergence results in chapter 3 are given for the extended state variable for the Mayer-Problem. But as $x(.) = (\tilde{x}(.), z(.))$, convergence of the state of the Mayer-Problem leads to convergence of the state of the Bolza-Problem. In addtion it deliveres convergence of the objective function integral terms. This can be easily seen by looking at the following inequality with appropriate norm $\|.\|$ and constants $C, p \in \mathbb{R}$:

$$\|\hat{x}^N - \rho_N(\hat{x}(.))\| = \| \begin{pmatrix} \hat{\tilde{x}}^N \\ \hat{z}^N \end{pmatrix} - \rho_N \left(\begin{pmatrix} \hat{\tilde{x}}(.) \\ \hat{z}(.) \end{pmatrix} \right) \| \leq C\, h_N^p$$

$$\Rightarrow \|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\| \leq C\, h_N^p \text{ and } \|\hat{z}^N - \rho_N(\hat{z}(.))\| \leq C\, h_N^p$$

The convergence of the integral term of the objective function follows from the fact that the difference of the integral terms of the discrete and the continuous case corresponds to $|\hat{z}_N^N - \hat{z}(T)|$.

obtaining the exact solution

It is in general preferable to obtain the exact solution analytically, so that an exact comparison can be done. The alternative would be to "trust" the numerical results and take the solution obtained for the maximum number of steps used in the calculation process as the exact solution. To obtain the solution analytically the so called maximum principle is used. This is usually not possible for complex problems. There is a lot of literature on that basic topic in infinite dimensional optimization. This article refers to [1].

norms used

For the convergence analysis done it is essential which norm to use. In most cases this will be the discrete L_∞-norm for the state and the discrete L_2-norm or L_∞-norm for the control. Calculation of these norms is straightforward.
In some special cases the continuous L_2-norm or the continuous L_∞-norm will be used. In that case the discrete solution is interpolated using a linear spline. The linear spline for the discrete optimal state data shall be called $\hat{\tilde{x}}^N(.)$, the one for the discrete optimal control data $\hat{u}^N(.)$. For calculating the L_2-norm numerical integration is used to integrate the difference between the optimal function of the continuous problem and the spline created from the discrete data. This process, of course, involves some numerical errors. The L_∞ is calculated using numerical maximization routines.
Taking a look at $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$, when actually trying to gather information about the convergence rate of $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$ makes sense because of the following cosiderations: From Theorem 2.6.4 we know that $\hat{\tilde{x}}^N(.)$ is Lipschitz-continuous with Lipschitz constant $\tilde{L}$. Like in the proof of Theorem 3.2.7 we get that $\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)$ is Lipschitz-continuous with Lipschitz constant $\tilde{L} + L$ with respect to the supremum norm. From the Compatibility Property 3.2.6 we then get that

$$\begin{aligned} &\left| \|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty - \|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty \right| \le (\tilde{L} + L)h_N \\ \Rightarrow\ &\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty \le (\tilde{L} + L)h_N + \|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty \end{aligned}$$

So if we get an estimation for the convergence rate $\tilde{p}$ of $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$, i.e we suppose that $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty \le C h_N^{\tilde{p}}$, the above inequality delivers:

$$\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty \le (\tilde{L} + L)h_N + C h_N^{\tilde{p}} \overset{h_N \le 1}{\le} (\tilde{L} + L + C)\, h_N^{\min(1,\tilde{p})}$$

So we get an estimation of the convergence rate p of $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$ by setting $p = \min(1, \tilde{p})$.

verifying sufficient optimality conditions

For verifying sufficient optimality conditions it might be advantageous to stick with the Bolza-Problem and show that the sufficient conditions hold. If the integrand of the objective function $f(.,.,.)$ is Lipschitz-continuous in all of its arguments on the feasible set the desired inverse stability property for the extended state can be obtained from the inverse stability property delivered by analyzing the Bolza-Problem. Note that that the

Lipschitz-continuity of $f(.,.,.)$ follows directly from the Lipschitz-continuity of $\psi(.,.,.)$, which is needed for Theorem 3.2.4 anyway. So needing the Lipschitz-continuity of $f(.,.,.)$ is no additional restriction.

As an example, which shows the idea on how to transfer inverse stability properties for the Bolza-Problem to inverse stability properties for the Mayer-Problem, once again second order sufficient optimality condtions and the resulting inverse stability property will be used.

Let's suppose that second order sufficient optimality condtitions hold for the Bolza-Problem. From optimization theorey we then get that the following inverse stability property holds:

①
$$\gamma \left\| (\tilde{x}(.), u(.)) - (\hat{\tilde{x}}(.), \hat{u}(.)) \right\|_2^2 \leq \tilde{J}(\tilde{x}(.), u(.)) - \tilde{J}(\hat{\tilde{x}}(.), \hat{u}(.))$$

The goal is to obtain:

$$\tilde{\alpha} \left\| (x(.), u(.)) - (\hat{x}(.), \hat{u}(.)) \right\|_2^2 \leq J(x(t_0), x(T)) - J(\hat{x}(t_0), \hat{x}(T))$$

We start with estimating the left side of the above inequality.

②
$$\begin{aligned} \left\| (x(.), u(.)) - (\hat{x}(.), \hat{u}(.)) \right\|_2^2 = \left\| x(.) - \hat{x}(.) \right\|_2^2 + \left\| u(.) - \hat{u}(.) \right\|_2^2 &= \left\| \begin{pmatrix} \tilde{x}(.) \\ z(.) \end{pmatrix} - \begin{pmatrix} \hat{\tilde{x}}(.) \\ \tilde{z}(.) \end{pmatrix} \right\|_2^2 + \left\| u(.) - \hat{u}(.) \right\|_2^2 \\ &= \left\| \tilde{x}(.) - \hat{\tilde{x}}(.) \right\|_2^2 + \left\| z(.) - \hat{z}(.) \right\|_2^2 + \left\| u(.) - \hat{u}(.) \right\|_2^2 \end{aligned}$$

So we have "extracted" $\left\| z(.) - \hat{z}(.) \right\|_2^2$, which we will now estimate using the Lipschitz-continuity of f:

③
$$\begin{aligned} \left\| z(.) - \hat{z}(.) \right\|_2^2 \quad &= \quad \int_{t_0}^{T} \left(\int_{t_0}^{t} f(\tau, \tilde{x}(\tau), u(\tau)) - f(\tau, \hat{\tilde{x}}(\tau), \hat{u}(\tau)) \, d\tau \right)^2 dt \\ &\leq \quad \int_{t_0}^{T} \left(\int_{t_0}^{t} \left| f(\tau, \tilde{x}(\tau), u(\tau)) - f(\tau, \hat{\tilde{x}}(\tau), \hat{u}(\tau)) \right| d\tau \right)^2 dt \\ &\overset{f \text{ Lipschitz}}{\leq} \quad \int_{t_0}^{T} \left(\int_{t_0}^{t} \left\| (\tilde{x}(\tau), u(\tau)) - (\hat{\tilde{x}}(\tau), \hat{u}(\tau)) \right\|_\infty d\tau \right)^2 dt \\ &\leq \quad \int_{t_0}^{T} \left(\int_{t_0}^{T} \left\| (\tilde{x}(\tau), u(\tau)) - (\hat{\tilde{x}}(\tau), \hat{u}(\tau)) \right\|_2 d\tau \right)^2 dt \\ &\overset{\text{Hölder}}{\leq} \quad \int_{t_0}^{T} (T - t_0) \left\| (\tilde{x}(.), u(.)) - (\hat{\tilde{x}}(.), \hat{u}(.)) \right\|_2^2 dt \\ &= \quad (T - t_0)^2 \left\| (\tilde{x}(.), u(.)) - (\hat{\tilde{x}}(.), \hat{u}(.)) \right\|_2^2 \end{aligned}$$

Combining ③ and ① delivers

$$\begin{aligned}\|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2^2 &\leq \|(\tilde{x}(.),u(.))-(\hat{\tilde{x}}(.),\hat{u}(.))\|_2^2+(T-t_0)^2\ \|(\tilde{x}(.),u(.))-(\hat{\tilde{x}}(.),\hat{u}(.))\|_2^2\\ &=\Big(1+(T-t_0)^2\Big)\ \|(\tilde{x}(.),u(.))-(\hat{\tilde{x}}(.),\hat{u}(.))\|_2^2\end{aligned}$$

This leads directly to

$$\textcircled{4}\quad \frac{\gamma}{1+(T-t_0)^2}\|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2^2\leq\gamma\,\|(\tilde{x}(.),u(.))-(\hat{\tilde{x}}(.),\hat{u}(.))\|_2^2$$

As $\tilde{J}(\tilde{x}(.),u(.))-\tilde{J}(\hat{\tilde{x}}(.),\hat{u}(.))=J(x(t_0),x(T))-J(\hat{x}(t_0),\hat{x}(T))$ placing ④ in ① yields the final result

$$\underbrace{\frac{\gamma}{1+(T-t_0)^2}}_{\tilde{\alpha}}\|(x(.),u(.))-(\hat{x}(.),\hat{u}(.))\|_2^2\leq J(x(t_0),x(T))-J(\hat{x}(t_0),\hat{x}(T))$$

estimation of the convergence rate

Estimation of the convergence rate is done using logarithmic analysis of the data in conjunction with calculation of the corresponding regression line. Let's consider the convergence analysis for the optimal state in a certain norm $\|.\|$ as an example. It is assumed that the following inequality holds with the constants C and p:

$$\|\hat{x}^N-\rho_N(\hat{x}(.))\|\leq C\ h_N^p$$

With the data calculated for certain numbers of steps $N_1,\ldots,N_k$ we know the left-hand side for those number of steps. The goal is to obtain C and above all the convergence rate p. Therefore we set

$$\|\hat{x}^N-\rho_N(\hat{x}(.))\|\approx C\ h_N^p$$

Applying the natural logarithm then yields

$$\ln\|\hat{x}^N-\rho_N(\hat{x}(.))\|\approx\ln\left(C\ h_N^p\right)=\ln C+p\ \ln h_N$$

The line that fulfills the above equation best is the regression line corresponding to the points $\left(\ln h_{N_i},\ln\|\hat{x}^{N_i}-\rho_{N_i}(\hat{x}(.))\|\right)$ $(i=1,\ldots,k)$. The slope of that line is then considered to be an approximation to p.

To obtain an estimation for C the regression line should be moved up till all the points $\left(\ln h_{N_i},\ln\|\hat{x}^{N_i}-\rho_{N_i}(\hat{x}(.))\|\right)$ $(i=1,\ldots,k)$ lie below it. Evaluating the moved line at 0 then should give a rough approximation of $\ln C$. There are also some other tweaks that could be applied to the line to obtain supposedly better results for C and p. But these are all heuristics that are not needed in this article. Unless otherwise noted, only unmodified regression lines are used for the convergence analysis.

explanation of the plots

The plots were done using mathematica. In the state and control plots the blue line represents the exact solution. The read dots depict the discrete solution, which is interpolated by a linear spline, which in turn is represented by green lines. An orange line marks the maximum distance from the discrete to the exact solution. In the logarithmic plots the blue line represents the regression line and the red dots represent the distance values, which depend on the norm used. The values shown at the axes in the double logarithmic plot do not have the logarithm applied, the logarithmic plotting just affects the scaling.

6.1.2 optimization routines

The author of this thesis has done quite some research on different optimizers and quite a number adjustments in terms of optimizer settings have been made to get good results. The most significant impacts come from the optimizers accuracy settings and the way derivatives of the objective function are calculated. The preferred way of course is to pass exact derivatives to the optimizer. The software used for computation has been extended by the author to work with exact derivatives. There are lots of details to care about, when programming the software to deal with optimization problems. Those are not part of this article, because it would be kind of off-topic. The main optimizer for solving finite dimensional problems used in this thesis is from Björn Sachsenberger from the University of Bayreuth and is called NLPIP. It essentially combines Interior Point mechanics with an SQP method. For literature on that Optimizer see [10]. Another optimizer used is from Klaus Schittkowski, also from the University of Bayreuth, and is called NLPQLP. For further information see [11]. The software package, that encapsulates these optimizers has been developed by Jürgen Pannek. It has been adjusted by the author to work with Optimal Control Problems, whereas its native domain is Model Predictive Control. This whole package serves as a framework in C++ and provides classes for defining the optimization problem itself (in the continuous form), solving the ODE, calculating approximated and exact derivatives, discretizing the continuous problem and wrapping the optimizers (which have been written in Fortran). Literature on and the software package itself can be found taking a look at [12].

6.2 Simple OCP without state constraints

This example is based on Example 8.2 in [?]. It is an example that involves box constraints for the control, but no state constraints. The optimal solution will be obtained analytically using the maximum principle (see 5.1.2 in [1]). Having the exact solution makes a precise error analysis possible. It is also pretty simple to verify that all necessary assumptions for the convergence proposition in chapter 3 are fulfilled. The example looks like this:

Problem

$$\text{Minimize :} \qquad \tilde{J}(\tilde{x}(.), u(.)) = -\int_0^2 2\tilde{x}(t) - 3u(t) - u^2(t)dt$$

with respect to :

$$\dot{\tilde{x}}(t) = \tilde{x}(t) + u(t)$$

$$\tilde{x}(0) = 5$$

$$u(t) \in [0, 2] \qquad a.e.$$

with $\tilde{x}(.) \in AC([0, 2])$ and $u(.) \in L_\infty([0, 2])$

Note: *As this is the Bolza form of the OCP, the state has been named $\tilde{x}$ to be consistent with the notations of chapter 2.*

6.2.1 obtaining the solution analytically

The initial condition can be written as $r(\tilde{x}(0), \tilde{x}(2)) := (x(0) - 5) = 0$.
From Theorem 5.1.2 from [1], i.e. the global maximum principle, it then follows with the notations from [1] that:

① $$H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t))(\hat{u}(t) - u) \geq 0 \quad \text{(for all } u \in [0, 2] \text{ and almost all } t \in [0, 2])$$

where

$$\begin{aligned} H(t, \tilde{x}(t), u(t), p(t)) &= p^\star(t)\, \psi(t, \tilde{x}(t), u(t)) - f(t, \tilde{x}(t), u(t)) \\ &= p(t)(\tilde{x}(t) + u(t)) + 2\tilde{x}(t) - 3u(t) - u^2(t) \end{aligned}$$

$$p(t) = \int_t^2 H_x(\tau, \hat{\tilde{x}}(\tau), \hat{u}(\tau), p(\tau))\, d\tau$$

$$p(0) = l_R\, r_{x_0}(\tilde{x}(0), \tilde{x}(2)) = l_R$$

$$p(2) = 0$$

With $H_x = (t, \tilde{x}(t), u(t), p(t)) = p(t) + 2$ we get for $p(t)$:

② $$p(t) = \int_t^2 p(\tau) + 2\, d\tau \Rightarrow \dot{p}(t) = -p(t) - 2 \Leftrightarrow p(t) = -2 + Ce^{-t}$$

$$p(2) = 0 \Rightarrow C = 2e^2 \Rightarrow p(t) = 2e^{2-t} - 2$$

With $H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t)) = p(t) - 3 - 2\hat{u}(t)$ we get by analyzing ①:

$$H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t)) > 0 \overset{\textcircled{1}}{\Rightarrow} \hat{u}(t) = 2 \Rightarrow p(t) - 3 - 2\hat{u}(t) = -9 + 2e^{2-t} > 0 \Rightarrow t < 2 - \ln(\tfrac{9}{2})$$

$$H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t)) < 0 \overset{\textcircled{1}}{\Rightarrow} \hat{u}(t) = 0 \Rightarrow p(t) - 3 - 2\hat{u}(t) = -5 + 2e^{2-t} < 0 \Rightarrow t > 2 - \ln(\tfrac{5}{2})$$

$$H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t)) = 0 \Rightarrow p(t) - 3 - 2\hat{u}(t) = 2 \Rightarrow \hat{u}(t) = \frac{1}{2}(p(t) - 3)$$

$$\overset{\hat{u}(t)\in[0,2]}{\Rightarrow} 0 \leq \frac{1}{2}(2e^{2-t} - 5) \leq 2 \Rightarrow 2 - \ln(\tfrac{9}{2}) \leq t \leq 2 - \ln(\tfrac{5}{2})$$

So overall we got for $\hat{u}(t)$:

③

$$\hat{u}(t) = \begin{cases} 2 & 0 \leq t < 2 - \ln(\frac{9}{2}) \\ e^{2-t} - \frac{5}{2} & 2 - \ln(\frac{9}{2}) \leq t \leq 2 - \ln(\frac{5}{2}) \\ 0 & 2 - \ln(\frac{5}{2}) < t \leq 2 \end{cases}$$

From the ODE $\dot{\hat{\tilde{x}}}(t) = \tilde{x}(t) + \hat{u}(t)$ we are now able to obtain the optimal state $\hat{\tilde{x}}(.)$ with ③. Obtaining one specific solution for $0 \leq t < 2 - \ln(\frac{9}{2})$ and $2 - \ln(\frac{5}{2}) < t \leq 2$ is pretty easy, because $\hat{u}(.)$ is constant there. So setting $\tilde{x}(t) = -\hat{u}(t)$ solves the ODE on $[0, 2 - \ln(\frac{9}{2})) \cup (2 - \ln(\frac{5}{2}), 2]$. For $2 - \ln(\frac{9}{2}) \leq t \leq 2 - \ln(\frac{5}{2})$ we set $\tilde{x}(t) = Ce^{2-t} + \frac{5}{2}$. Inserting the expression in the ODE delivers $-Ce^{2-t} = Ce^{2-t} + e^{2-t}$, which leads to $C = -\frac{1}{2}$. So the specific solution we were looking for is $\tilde{x}(t) = -\frac{1}{2}e^{2-t} + \frac{5}{2}$. So overall we have for $\hat{\tilde{x}}(.)$:

④

$$\hat{\tilde{x}}(t) = \begin{cases} C_1 e^t - 2 & 0 \leq t < 2 - \ln(\frac{9}{2}) \\ C_2 e^t - \frac{1}{2}e^{2-t} + \frac{5}{2} & 2 - \ln(\frac{9}{2}) \leq t \leq 2 - \ln(\frac{5}{2}) \\ C_3 e^t & 2 - \ln(\frac{5}{2}) < t \leq 2 \end{cases}$$

The constants C_1, C_2 and C_3 are determined by the initial value $\hat{\tilde{x}}(0) = 5$ and the fact that $\hat{\tilde{x}}(.)$ is continuous. From ④ and $\hat{\tilde{x}}(0) = 5$ we get

$$5 = C_1 - 2 \Leftrightarrow C_1 = 7$$

From $\hat{\tilde{x}}(.)$ being continuous and ④ it then follows that

$$\hat{\tilde{x}}(2 - \ln(\tfrac{9}{2})) = 7e^{2-\ln(\frac{9}{2})} - 2 = \frac{14}{9}e^2 - 2$$

$$\Rightarrow \frac{14}{9}e^2 - 2 = \frac{2}{9}e^2 C_2 - \frac{1}{2}e^{\ln(\frac{9}{2})} + \frac{5}{2} \Leftrightarrow -\frac{9}{4} = (C_2 - 7)\frac{2}{9}e^2 \Leftrightarrow C_2 = -\frac{81}{8}e^{-2} + 7$$

$$\Rightarrow \hat{\tilde{x}}(2 - \ln(\tfrac{5}{2})) = (7 - \frac{81}{8}e^{-2})e^{2-\ln(\frac{5}{2})} - \frac{1}{2}e^{\ln(\frac{5}{2})} + \frac{5}{2} = \frac{14}{5}e^2 - \frac{81}{20} + \frac{5}{4} = \frac{14}{5}e^2 - \frac{14}{5} = \frac{14}{5}(e^2 - 1)$$

$$\Rightarrow \frac{14}{5}(e^2 - 1) = C_3\, e^{2-\ln(\frac{5}{2})} = \frac{2}{5}e^2 C_3 \Leftrightarrow C_3 = 7(1 - e^{-2})$$

So overall we have the following result for the optimal state:

$$\hat{\tilde{x}}(t) = \begin{cases} 7e^t - 2 & 0 \leq t < 2 - \ln(\frac{9}{2}) \\ (7 - \frac{81}{8}e^{-2})\, e^t - \frac{1}{2}e^{2-t} + \frac{5}{2} & 2 - \ln(\frac{9}{2}) \leq t \leq 2 - \ln(\frac{5}{2}) \\ 7(1 - e^{-2})\, e^t & 2 - \ln(\frac{5}{2}) < t \leq 2 \end{cases}$$

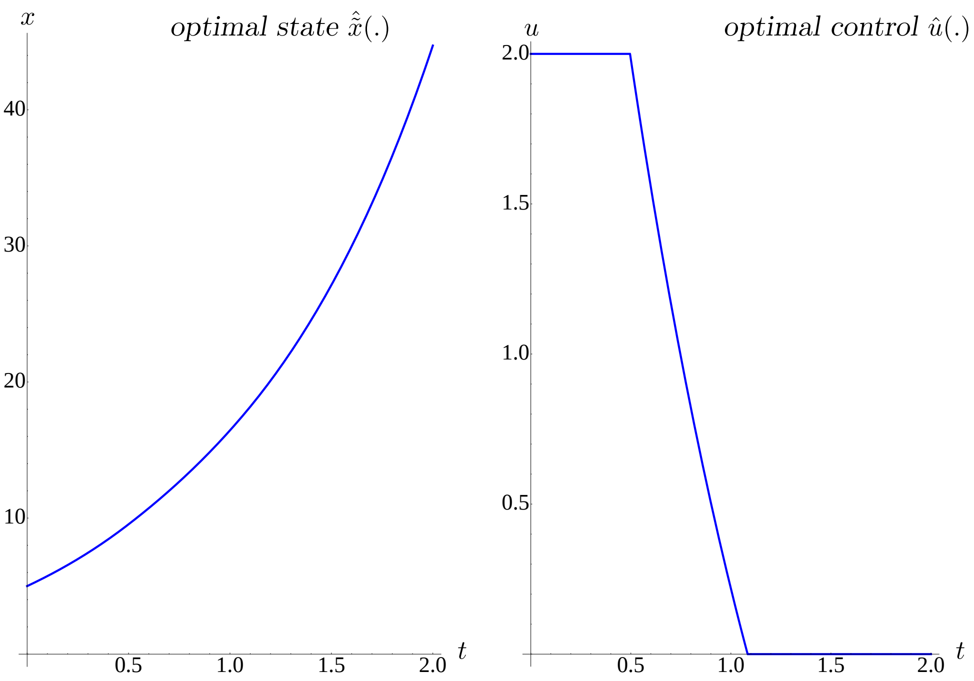


Figure 1: optimal state and control (Example 6.2)

Knowing $\hat{\tilde{x}}$ and $\hat{u}(.)$ delivers the minimal objective function value:

$$\tilde{J}(\hat{\tilde{x}}(.), \hat{u}(.)) = -\int_0^2 2\hat{\tilde{x}}(t) - 3\hat{u}(t) - \hat{u}^2(t)dt = \frac{1}{4}\left(236 - 56e^2 - 25\log\left(\frac{9}{5}\right) - 56\log\left(\frac{9}{2}\right)\right)$$
$$\approx -69.1775356$$

6.2.2 applying the convergence theorem

For this example it is pretty easy to check if the premises of the Convergence Theorem 3.2.8 are fulfilled. As there are no pure state constraints present, the assumptions (C1) and (C2) from chapter 5 do not have to be considered. Showing that (A1), (A2) and (A3) apply implies considering the set-valued form of the corresponding Mayer-Problem, i.e.

Problem (set-valued Mayer form)

$$\begin{aligned}
\text{Minimize :}\quad & J(x(0),x(2)) = z(2)\\
\text{with respect to :}\quad &
\end{aligned}$$

$$\dot{x}(t) \in F(t,x(t)) = \Big\{ \begin{pmatrix} \tilde{x}(t)+u \\ -2\tilde{x}(t)+3u+u^2 \end{pmatrix} \mid u \in [0,2] \Big\}$$

$$x(0) = \begin{pmatrix} 5 \\ 0 \end{pmatrix}$$

with $x(.) \in AC([0,2])^2$ and $u(.) \in L_\infty([0,2])$

We can now examine F from above:

- The following shows that (A1) is fulfilled.

$$\begin{aligned}
\|F(t,x)\|_2 &= \max_{u\in[0,2]} \left\| \begin{pmatrix} \tilde{x}+u \\ -2\tilde{x}+3u+u^2 \end{pmatrix} \right\|_2 \le \max_{u\in[0,2]} \left(|\tilde{x}+u| + |-2\tilde{x}+3u+u^2| \right)\\
&\le \max_{u\in[0,2]} \left(|\tilde{x}| + |u| + |2\tilde{x}| + |3u+u^2| \right) \le 3|\tilde{x}| + 12 \le 12(|\tilde{x}|+1) \le 12(\|x\|_2+1)
\end{aligned}$$

- (A2) is also fulfilled, which follows from the following considerations:
  It should be obvious, that the images of F are not empty. The mapping

$$G(u) := \begin{pmatrix} \tilde{x}+u \\ -2\tilde{x}+3u+u^2 \end{pmatrix}$$

  is continuous. As $[0,2]$ is compact it follows that $F(t,x) = G([0,2])$ is compact.

- (A3) follows from:
  Let $v \in F(\tilde{t},y)$. Then there exists $\tilde{u} \in [0,2]$ such that $v = \begin{pmatrix} \tilde{y}+\tilde{u} \\ -2\tilde{y}+3\tilde{u}+\tilde{u}^2 \end{pmatrix}$ with $\tilde{y}$ representing the first component of y. We have to prove now that there exists $w \in F(t,x)$ and L_F independent of t, $\tilde{t}$, x, y and $\tilde{u}$ such that

$$\|w-v\|_2 \le L_F(|t-\tilde{t}| + \|x-y\|_2) \qquad (t,\tilde{t} \in I,\ x,y \in \mathbb{R}^n)$$

  As $w = \begin{pmatrix} \tilde{x}+u \\ -2\tilde{x}+3u+u^2 \end{pmatrix}$ with $u \in [0,2]$ it holds:

$$\begin{aligned}
\|w-v\|_2 &= \left\| \begin{pmatrix} \tilde{x}+u \\ -2\tilde{x}+3u+u^2 \end{pmatrix} - \begin{pmatrix} \tilde{y}+\tilde{u} \\ -2\tilde{y}+3\tilde{u}+\tilde{u}^2 \end{pmatrix} \right\|_2 \overset{\text{set } u=\tilde{u}}{=} \left\| \begin{pmatrix} \tilde{x}-\tilde{y} \\ -2(\tilde{x}-\tilde{y}) \end{pmatrix} \right\|_2\\
&\le |\tilde{x}-\tilde{y}| + 2|\tilde{x}-\tilde{y}| = 3|\tilde{x}-\tilde{y}| \overset{|\tilde{x}-\tilde{y}|\le\|x-y\|_2}{\le} 3\left(|t-\tilde{t}| + \|x-y\|_2\right)
\end{aligned}$$

> The other way round, i.e. finding $v \in F(\tilde{t}, y)$ for given $w \in F(t, x)$ such that a similar result to the one above holds, involves exactly the same calculation as done above and delivers the same result. So overall it has been shown that (A3) holds with $L_F = 3$.

The next thing is to show that $J(.,.)$ is Lipschitz-continuous in both arguments. This is very easy due to the simple form of J: Let $x, y, a, b \in \mathbb{R}^2$ with

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, a = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Then

$$|J(x,y) - J(a,b)| = |y_2 - b_2| \leq \|y - b\|_\infty \leq \|(x,y) - (a,b)\|_\infty$$

Which means that $J(.,.)$ is Lipschitz-continuous with Lipschitz constant $L_J = 1$ with respect to the supremum norm. With (A1), (A2) and (A3) being fulfilled the Approximation Property 3.2.1 holds. Together with J being Lipschitz-continuous the premises of the Value Convergence Theorem 3.2.2 are met. This means that the objective function values converge with at least rate one.
$\psi(.,.,.)$ is Lipschitz-continuous in all arguments on the feasible set. Indeed, the Lipschitz-continuity is only needed there (for details see 3.2.4). The Lipschitz-continuity can be shown the following way: First, $\psi(.,.,.)$ does not explicitly depend on the time t. Second, $\psi(.,.,.)$ is linear with respect to x. Finally u is bounded.

The last thing needed to apply the Convergence Theorem 3.2.8 is some sort of inverse stability property. We will make use of second order sufficient optimality conditions here, which have to be verified. This is consistent with the Inverse Stability Property 3.2.3. Another way would be to go for first order sufficient optimality conditions, which would deliver the better convergence rate of 1 instead of 1/2 in the Convergence Theorem 3.2.8.
We have to show that $L''(\hat{\tilde{x}}(.), \hat{u}(.))(v(.), w(.))(v(.), w(.)) \geq \gamma \|(v(.), w(.))\|_2^2$ for all $(v(.), w(.)) \in L\big(\Sigma, (\hat{\tilde{x}}(.), \hat{u}(.))\big)$, where L'' is the second Fréchet derivative of of the Lagrange function and $L\big(\Sigma, (\hat{\tilde{x}}(.), \hat{u}(.))\big)$ is the linearizing cone of the feasible set Σ in $(\hat{\tilde{x}}(.), \hat{u}(.))$. For details, see [1]. For this example we won't need the restriction to the linearizing cone, because of the simple structure of the problem. As all constraints are linear, the second derivative of the Lagrange function is the second derivative of the objective function:

①

$$L''\big((\hat{\tilde{x}}(.), \hat{u}(.))\big)(v(.), w(.))(v(.), w(.)) = \int_0^2 2\, w^2(t)\, dt = 2\|w(.)\|_2^2$$

Because of the linear structure of the ODE it is easy to estimate $\|v(.)\|_2 = \|\tilde{x}(.) - \hat{\tilde{x}}(.)\|_2$ by $\|w(.)\|_2 = \|u(.) - \hat{u}(.)\|_2$. The general solution to the ODE is:

$$\tilde{x}(t) = 5\, e^t + e^t \int_0^t e^{-\tau}\, u(\tau)\, d\tau$$

So we have

$$\begin{aligned}\|\tilde{x}(.) - \hat{\tilde{x}}(.)\|_2 \le \sqrt{2}\|\tilde{x}(.) - \hat{\tilde{x}}(.)\|_\infty &\le \sqrt{2}\, e^2 \int\limits_0^2 \underbrace{e^{-\tau}}_{\le 1} \|u(\tau) - \hat{u}(\tau)\|_\infty \, d\tau \\ &\le \sqrt{2}\, e^2 \int\limits_0^2 \|u(\tau) - \hat{u}(\tau)\|_2 \, d\tau \overset{\text{Hölder}}{\le} 2\, e^2 \, \|u(.) - \hat{u}(.)\|_2\end{aligned}$$

Combining this result with ① yields

$$\begin{aligned}L''\big((\hat{\tilde{x}}(.), \hat{u}(.))\big)(v(.), w(.))(v(.), w(.)) &= 2\|w(.)\|_2^2 \ge \|w(.)\|_2^2 + \frac{1}{4}e^{-4}\|v(.)\|_2^2 \\ &\ge \frac{1}{4}e^{-4}\left(\|w(.)\|^2 + \|v(.)\|^2\right) = \underbrace{\frac{1}{4}e^{-4}}_{\gamma :=} \|(v(.), w(.))\|_2^2\end{aligned}$$

So overall we have shown, that the Value Convergence Theorem holds, which leads to convergence of the objective function values with at least rate one. Furthermore the Convergence Theorem 3.2.8 can be applied with convergence rate 1/2. So from the numerical results at least convergence rate 1 for the objective function values and convergence rate 1/2 for the optimal state with respect to the discrete L_∞-norm can be expected.

As a note it shall be mentioned, that for this simple example there exist results, that deliver at least convergence rate 1 for the optimal discrete state and convergence rate 1 for the optimal discrete controls, both with respect to the discrete supremum norm. For details, see [13]. But the assumptions needed to obtain those results are way more restrictive, than the ones presented in this article. Furthermore the concept shown here is far more flexible, which can be seen by taking a look at chapter 4.

6.2.3 convergence analysis

The last section about applying the convergence theorem delivers that we can at least expect convergence rate one for the optimal state and the optimal control. The results for this section have all been calculated using the NLPIP optimizer and exact derivatives, which provided the best results out of several scenarios. The term best in this case means that the calculated discrete solutions were closest to the exact optimal solutions derived in 6.2.1. As computing power does not really matter for this example it is convenient to use a power function for increasing the number of steps. For this example powers of 2 were used. That way the stepsizes are equidistant in a logarithmic plot.

state convergence
As this thesis is for the most part about convergence of the discrete optimal state, we take a look at some plots containing the optimal state of the continuous problem (blue) and the optimal state of the directly discretized problem (red points). The discrete solution is presented by red points and interpolated using a linear spline, which is shown in green color. The orange line shows where the maximum distance of the discrete solution to the exact solution appears.

So for the state the length of the orange bar corresponds to $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$.

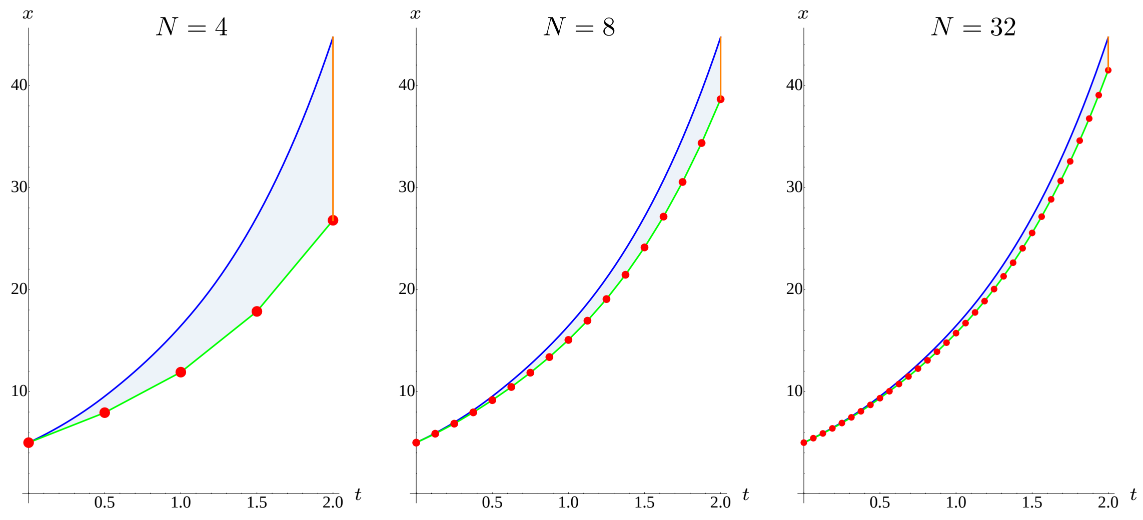


Figure 2: discrete optimal state $\hat{\tilde{x}}^N$ (Example 6.2)

For $N \geq 256$ there is not really an optical difference between the two curves left.

Applying the logarithmic analysis explained in the overview section 6.1 yields the following LogLogPlot (logarithmic logarithmic plot). Note that the values shown on the axis, which correspond to the calculated points, have no logarithm applied. They just represent the distance $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$ for a given steplength $h_N = 1/N$.

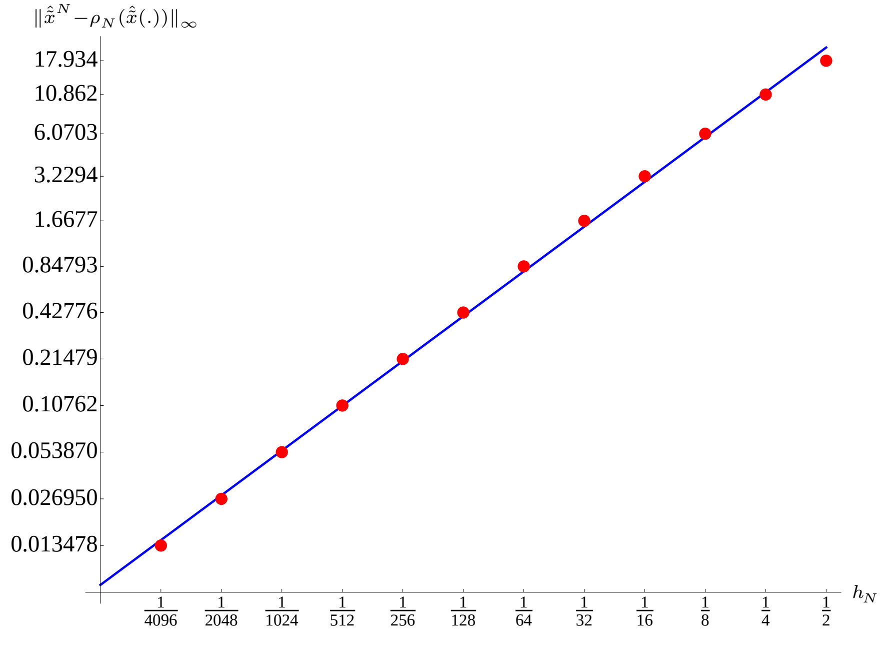


Figure 3: double logarithmic plot of $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 13)$ (Example 6.2)

The regressionline $r(.)$ (blue) is defined by the following expression:

$$r(t) = 3.83364607 + 0.958591079\, t$$

This means that the estimation of the convergence rate is 0.958591079. This is pretty close to the expected convergence rate, i.e. 1.

control convergence

For the controls results look pretty much as good as for the state, but due to the slightly more complex appearance of the control the result for $N = 4$ is way worse. In that case the optimizer generated a warning, that the line search could not be terminated successfully after the maximum number of iterations. This has to be expected for such a small number of steps N. As already mentioned in the overview of this chapter the calculated discrete control vector lacks the vector corresponding to the optimal control at the last grid point, which in this case is $t_N = 2$.

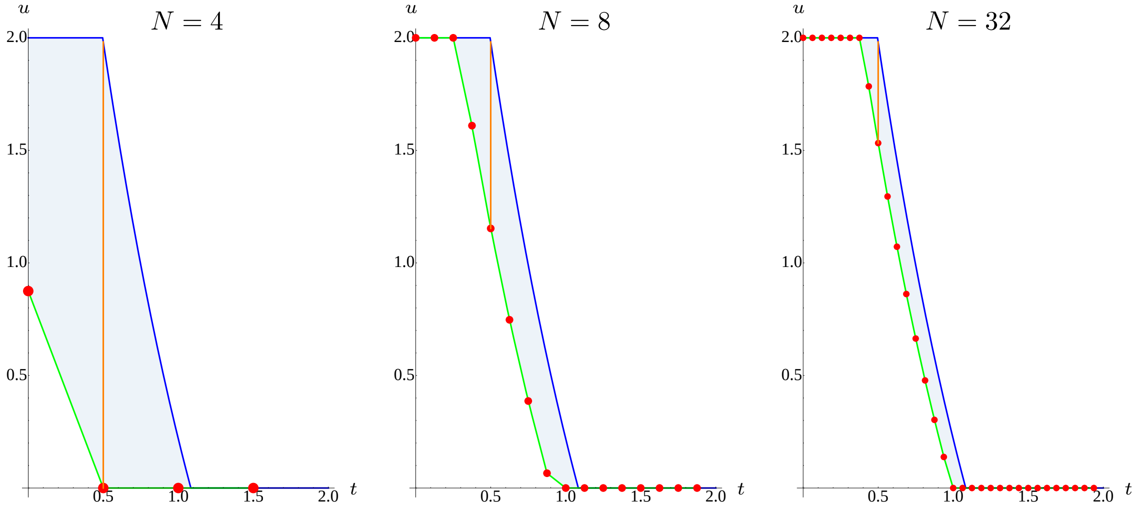


Figure 4: discrete optimal control $\hat{u}^N$ (Example 6.2)

As for the state for $N \geq 256$ there is hardly any viewable difference between the two curves left.

Applying the logarithmic analysis explained in the overview section 6.1 yields the following LogLogPlot (logarithmic logarithmic plot). Note that the values shown on the axis, which correspond to the calculated points, have no logarithm applied. They just represent the distance $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$ for a given steplength $h_N = 1/N$.

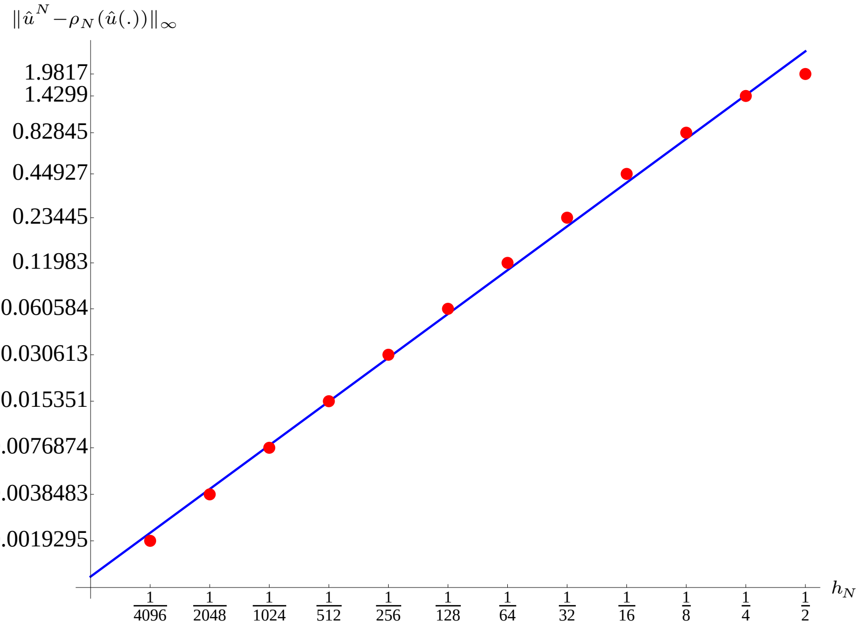


Figure 5: double logarithmic plot of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 13)$ (Example 6.2)

The regressionline $r(.)$ appearing in this plot (blue) is defined by the following expression:

$$r(t) = 1.80085114 + 0.93810558\, t$$

This means that the estimation of the convergence rate for the controls is 0.93810558. This is not quite as good as the estimation for the state convergence rate, but still pretty close to the expected convergence rate, i.e. 1.

objective function value convergence

The objective funciton values are also expected to be converging with at least rate 1. The corresponding double logarithmic plot for this example is

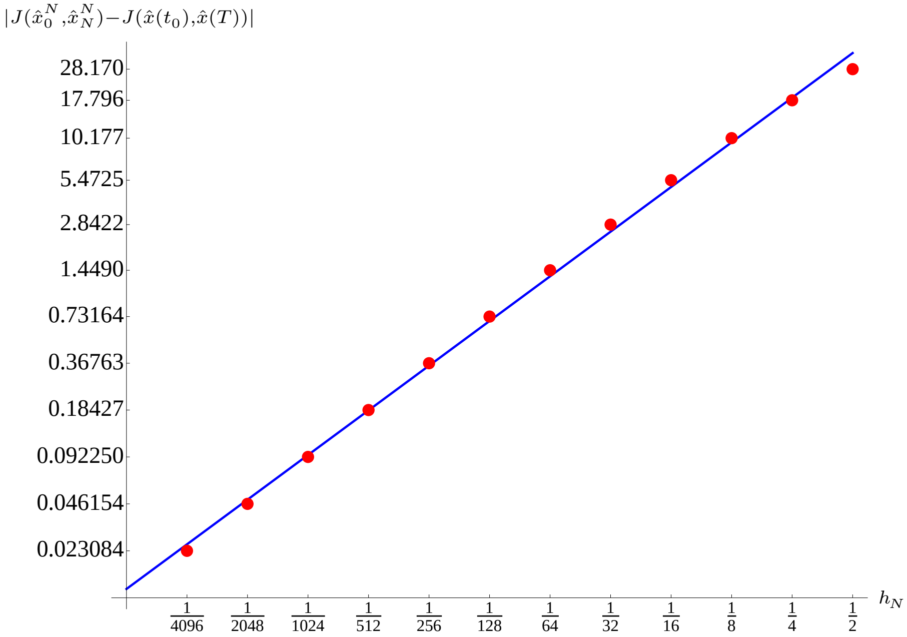


Figure 6: double logarithmic plot of $|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))|$ *for* $N = 2^k$ $(k = 2, \ldots, 13)$ *(Example 6.2)*

The regressionline $r(.)$ appearing in this plot (blue) is:

$$4.33944057 + 0.950699286\,t$$

This means that the estimation of the convergence rate for the controls is 0.950699286. Again, this is pretty close to the expected convergence rate, i.e. 1.

Summary

Although the estimations of the convergence values are all slightly below one, this is a really good result. Using the regression line approach just delivers a guess for the convergence rate. Also a very good sign is, that all the double logarithmic plots show nearly straight lines. Overall these are the computed values:

stepsize	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$	$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$	$|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))|$
1/2	1.618698	1.981689	17.934330	28.169723
1/4	0.941112	1.429931	10.862134	17.796303
1/8	0.502039	0.828453	6.070301	10.177303
1/16	0.260061	0.449271	3.229420	5.472450
1/32	0.132335	0.234455	1.667662	2.842170
1/64	0.066804	0.119832	0.847926	1.448962
1/128	0.033577	0.060584	0.427759	0.731635
1/256	0.016860	0.030613	0.214792	0.367630
1/512	0.008439	0.015351	0.107624	0.184271
1/1024	0.004222	0.007687	0.053870	0.092250
1/2048	0.002112	0.003848	0.026950	0.046154
1/4096	0.001056	0.001930	0.013478	0.023084

6.3 Simple multidimensional state constraints

This example is based on Problem 6.4.7 in [14]. It's about the flow of water involving two water boxes, but this shall not be the concern here. The minimization problem is the following:

<u>Problem</u>

$$\text{Minimize :} \qquad \tilde{J}(\tilde{x}(.), u(.)) = -\int_0^{10} (10-t)\, u_1(t) + t\, u_2(t) dt$$

with respect to :

$$\dot{\tilde{x}}(t) = \begin{pmatrix} -u_1(t) \\ u_1(t) - u_2(t) \end{pmatrix} \qquad a.e.$$

$$\tilde{x}(0) = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

$$s_1(t, \tilde{x}(t)) = -x_1(t) \leq 0$$
$$s_2(t, \tilde{x}(t)) = -x_2(t) \leq 0$$

$$u(t) \in [0,1]^2 \qquad a.e.$$

$$\text{with } \tilde{x}(.) = \begin{pmatrix} x_1(.) \\ x_2(.) \end{pmatrix} \in AC([0,10])^2 \text{ and } u(.) = \begin{pmatrix} u_1(.) \\ u_2(.) \end{pmatrix} \in L_\infty([0,10])^2$$

Note: *As this is the Bolza form of the OCP, the state has been named $\tilde{x}$ to be consistent with the notations of chapter 2.*

6.3.1 obtaining the solution analytically

The initial conditions can be written as $r(\tilde{x}(0), \tilde{x}(10)) := (x_1(0) - 4, x_2(0) - 4) = 0_{\mathbb{R}^2}$. From Theorem 5.1.2 from [1], i.e. the global maximum principle, it then follows with the notations from [1] that:

① $H_u(t, \hat{\tilde{x}}, \hat{u}(t), p(t))(\hat{u}(t) - u) \geq 0$ (for all $u \in [0,1]^2$ and almost all $t \in [0,10]$)

where

$$
\begin{aligned}
H(t,\tilde{x}(t),u(t),p(t)) &= p^{\star}(t)\,\psi(t,\tilde{x}(t),u(t)) - f(t,\tilde{x}(t),u(t)) \\
p(t) = \begin{pmatrix} p_1(t) \\ p_2(t) \end{pmatrix} &= \int\limits_t^{10} H_x(\tau,\hat{\tilde{x}}(\tau),\hat{u}(\tau),p(\tau))\,d\tau + \int\limits_t^{10} s_x(\tau,\hat{\tilde{x}}(\tau))\,d\mu_s(\tau) \\
p(0) &= l_R\, r_{x_0}(\tilde{x}(0),\tilde{x}(10)) = l_R \\
p(10) &= 0
\end{aligned}
$$

and

$$\hat{\tilde{x}}(.) = \begin{pmatrix} \hat{x}_1(.) \\ \hat{x}_2(.) \end{pmatrix} \text{ and } \hat{u}(.) = \begin{pmatrix} \hat{u}_1(.) \\ \hat{u}_2(.) \end{pmatrix}$$

are the optimal state respectively control variables.

From the fact that

$$
\begin{aligned}
\psi(t,\tilde{x}(t),u(t)) &= \begin{pmatrix} -u_1(t) \\ u_1(t) - u_2(t) \end{pmatrix} \\
f(t,\tilde{x}(t),u(t)) &= (t-10)\,u_1(t) - t\,u_2(t)
\end{aligned}
$$

we get that $H(t,\tilde{x}(t),u(t),p(t))$ does not depend on $\tilde{x}$, so we get $H_x(t,\tilde{x}(t),u(t),p(t)) = 0$. In addition we get from the definition of $s(.,.,.)$ that

$$s_x(t,\tilde{x}(t)) = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

Using those results and the equation for $p(t)$ we get:

②
$$p(t) = \int\limits_t^{10} d\mu_s(\tau) = \mu_s(10) - \mu_s(t) = \begin{pmatrix} \mu_{s_1}(10) - \mu_{s_1}(t) \\ \mu_{s_2}(10) - \mu_{s_2}(t) \end{pmatrix}$$

Analyzing ① further and using

$$\psi_u(t,\tilde{x},u(t)) = \begin{pmatrix} -1 & 0 \\ 1 & -1 \end{pmatrix} \text{ and } f_u(t,\tilde{x},u(t)) = -(10-t,t)$$

yields

①
$$
\begin{aligned}
&H_u(t,\hat{\tilde{x}},\hat{u}(t),p(t))(\hat{u}(t)-u) = \left(p^{\star}(t) \begin{pmatrix} -1 & 0 \\ 1 & -1 \end{pmatrix} + (10-t,t) \right)(\hat{u}(t)-u) = \\
&(p_2(t) - p_1(t) + 10 - t, -p_2(t) + t)\,(\hat{u}(t)-u) \geq 0 \quad \text{(for all } u \in [0,1]^2 \text{ and almost all } t \in [0,10])
\end{aligned}
$$

From this we can conclude the following:
Set $u_2 = \hat{u}_2(t)$. Then ① is

$$(p_2(t) - p_1(t) + 10 - t)(\hat{u}_1(t) - u_1) \geq 0 \qquad \text{(for all } u_1 \in [0,1] \text{ and almost all } t \in [0,10])$$

So we get that for $p_2(t) - p_1(t) + 10 - t > 0$ it holds $\hat{u}_1(t) = 1$ and for $p_2(t) - p_1(t) + 10 - t < 0$ it holds $\hat{u}_1(t) = 0$. So we have

③
$$\hat{u}_1(t) = \begin{cases} 0 & \text{for } p_2(t) - p_1(t) + 10 - t < 0 \\ 1 & \text{for } p_2(t) - p_1(t) + 10 - t > 0 \end{cases} \qquad a.e.$$

Arguing the same way for $\hat{u}_2$ delivers

③
$$\hat{u}_2(t) = \begin{cases} 0 & \text{for } -p_2(t) + t < 0 \\ 1 & \text{for } -p_2(t) + t > 0 \end{cases} \qquad a.e.$$

Considering $\hat{u}_1$

μ_{s_i} $(i = 1, 2)$ is monotonously increasing and the starting point may be chosen arbitrarily. So we set $\mu_s = 0_{\mathbb{R}^2}$. From the complementary slackness condition, i. e. $\int_0^{10} -\hat{x}^\star(t)\, d\mu_s(t) = 0$ it follows that for $\hat{x}_i > 0$ on $[\tau_1, \tau_2]$ it holds $\mu_{s_i} = const.$ on $[\tau_1, \tau_2]$ $(i = 1, 2)$. So it holds $\int_{\tau_1}^{\tau_2} -x^\star(t)\, d\mu_s(t) = 0$ for all $x(.) \in AC([0, 10])^2$.

So let's suppose that $\hat{x}_1(t) > 0$ for all $t \in [0, 10]$. Then $\mu_{s_1}(t) \equiv 0$ on $[0, 10]$. From ② we then get $p_1(t) = \mu_{s_1}(10) - \mu_{s_1}(t) \equiv 0$ on $[0, 10]$. Thus it holds:

$$p_2(t) - p_1(t) = p_2(t) = \mu_{s_2}(10) - \mu_{s_2}(t) \underset{\text{increasing}}{\overset{\mu_{s_2} \text{ mon.}}{\geq}} 0 > t - 10 \qquad (t \in [0, 10))$$

So with ③ we have $\hat{u}_1(t) = 1$ for almost all $t \in [0, 10)$. Using the connection of $\hat{u}_1(.)$ and $\hat{x}_1(.)$ via the ODE we get

$$\hat{x}_1(t) = 4 - \int\limits_0^t \hat{u}_1(\tau)\, d\tau \leq 0 \qquad (t \in [4, 10])$$

which is a contradiction to the assumption that $\hat{x}_1(t) > 0$ for all $t \in [0, 10]$. So considering that $\hat{x}_1(0) = 4 > 0$ we have that there exists $t \in [0, 10]$ with $\hat{x}_1(t) = 0$.

Let's set

$$\tilde{t} := \min\{t \in [0, 10] \mid \hat{x}_1(t) = 0\}$$

Because of $\hat{x}_1(0) = 4 > 0$ and $\hat{x}_1(.)$ being continuous it holds by the definition of $\tilde{t}$ that $\hat{x}_1(t) > 0$ for $t \in [0, \tilde{t})$. So we have $\mu_{s_1}(t) \equiv 0$ on $[0, \tilde{t})$, but this time $\mu_{s_1}(10)$ must not be 0. Using ② this leads to $p_2(t) - p_1(t) = \mu_{s_2}(10) - \mu_{s_2}(t) - \mu_{s_1}(10)$, which is monotonously decreasing. Because of $t - 10$ being strictly monotonously increasing there is at most one point $t_1 \in [0, \tilde{t})$ with $p_2(t_1) - p_1(t_1) = t_1 - 10$. If this point exists it then holds

$$p_2(t) - p_1(t) \begin{cases} > t - 10 & \text{for } t \in [0, t_1) \\ < t - 10 & \text{for } t \in (t_1, \tilde{t}) \end{cases}$$

So we have $\hat{u}_1(t) \equiv 1$ almost everywhere on $[0, t_1)$. Using the ODE then leads to

$$\hat{x}_1(t_1) = 4 - \int\limits_0^{t_1} \hat{u}_1(\tau)\, d\tau = 0$$

With $t_1 < \tilde{t}$ this is a contradiction to the definition of $\tilde{t}$.
As $p_2(t) - p_1(t) < t - 10$ for all $t \in [0, 10]$ would lead to $\hat{x}_1(t) \equiv 4 > 0$ on $[0, 10]$, this cannot be the case. So with the monotonicity considerations made before we know that $p_2(t) - p_1(t) > t - 10$ on $[0, \tilde{t})$. So we have $\hat{u}_1(t) = 1$ for almost all $t \in [0, \tilde{t})$. This leads directly to $\tilde{t} = 4$ via applying the ODE. Furthermore we get

$$0 \leq \hat{x}_1(t) = 4 - \int\limits_0^4 \hat{u}_1(\tau)\, d\tau - \int\limits_4^t \hat{u}_1(\tau)\, d\tau = -\int\limits_4^t \hat{u}_1(\tau)\, d\tau \qquad (t \in [4, 10])$$

Together with the fact that $u_1(t) \geq 0$ we have $\hat{u}_1(t) = 0$ for almost all $t \in [\tilde{t}, 10]$. So overall the optimal control $\hat{u}_1(.)$ is:

④ $$\hat{u}_1(t) = \begin{cases} 1 & \text{for } t \in [0, 4) \\ 0 & \text{for } t \in [4, 10] \end{cases} \qquad a.e.$$

The fact that $u_1(t) \geq 0$ for all $t \in [0, 10]$, $\hat{x}_1(\tilde{t}) = 0$ and that $\hat{x}(t) \geq 0$ on $[0, 10]$ together with the ODE then delivers that $\hat{x}_1(t) = 0$ on $[\tilde{t}, 10]$ and $\hat{u}_1(t) = 0$ for almost all $t \in [\tilde{t}, 10]$.

Considering $\hat{u}_2$
Pretty much the same considerations that were made for obtaining $\hat{u}_1(.)$ will be used to obtain $\hat{u}_2(.)$. From what we already know from ③ this should be easier than for $\hat{u}_1(.)$ because only $p_2(.)$ will be involved. Although the second component of the ODE looks a bit more complicated at first sight this is not the case because we already know $\hat{u}_1(.)$, hence it has been deduced before analyzing $\hat{u}_2(.)$.
$p_2(t) = \mu_{s_2}(10) - \mu_{s_2}(t)$ is monotonously decreasing. t is strictly monotonously increasing. So there exists at most one point $\hat{t} \in [0, 10]$ with $p_2(\hat{t}) = \hat{t}$. Because $p_2(10) = 0$ there is exactly one such point. From $p_2(t) > t$ for $t \in [0, \hat{t})$ and $p_2(t) < t$ for $t \in (\hat{t}, 10]$ we get from ③ that

⑤ $$\hat{u}_2(t) = \begin{cases} 0 & \text{for } t \in [0, \hat{t}) \\ 1 & \text{for } t \in (\hat{t}, 10] \end{cases} \qquad a.e.$$

All that is left to do now is find $\hat{t}$.
Combining the ODE with ④ delivers

⑥ $$\hat{x}_2(t) = 4 + \int\limits_0^t \hat{u}_1(\tau) - \hat{u}_2(\tau)\, d\tau \overset{④}{=} \begin{cases} 4 + t - \int_0^t \hat{u}_2(\tau) d\tau & \text{for } t \in [0, 4) \\ 8 - \int_0^t \hat{u}_2(\tau) d\tau & \text{for } t \in [4, 10] \end{cases}$$

Let's suppose that $\hat{x}_2(t) > 0$ for all $t \in [0, 10]$. Then from the complementary slackness condition it follows that $p_2(t) \equiv 0$ on $[0, 10]$. This means that $\hat{t} = 0$. With ⑤ it then holds that $\hat{u}_2(t) = 1$ for almost all $t \in [0, 10]$. ⑥ then delivers that $\hat{x}_2(t) < 0$ for $t \in (8, 10]$, which violates the state constraints. So $\hat{x}_2(t) > 0$ for all $t \in [0, 10]$ cannot hold, which means that there exists $t_1 \in [0, 10]$ such that $\hat{x}_2(t) = 0$.
From ⑥ we know that $t_1 \in [8, 10]$. Let's suppose that $t_1 \in (8, 10)$. Due to $\hat{x}_2(t) \geq 0$ for all $t \in [0, 10]$ it then holds that $\hat{u}_2(t) = 0$ for almost all $t \in (t_1, 10]$. This is a contradiction to

⑤. So $t_1 = 10$ which means $\hat{x}_2(10) = 0$.
Combining $\hat{x}_2(10) = 0$, ⑤ and ⑥ delivers

$$0 = \hat{x}_2(10) \overset{⑤,⑥}{=} 8 - \int\limits_{\hat{t}}^{10} 1(\tau)\, d\tau = -2 + \hat{t} \Leftrightarrow \hat{t} = 2$$

So from ⑤ we get

⑦
$$\hat{u}_2(t) = \begin{cases} 0 & \text{for } t \in [0,2) \\ 1 & \text{for } t \in (2,10] \end{cases} \qquad a.e.$$

The corresponding optimal state can be easily obtained from the ODE by using ④ and ⑦. So overall we have

⑧
$$\hat{u}_1(t) = \begin{cases} 1 & \text{for } t \in [0,4) \\ 0 & \text{for } t \in [4,10] \end{cases} \qquad a.e.$$

$$\hat{u}_2(t) = \begin{cases} 0 & \text{for } t \in [0,2) \\ 1 & \text{for } t \in (2,10] \end{cases} \qquad a.e.$$

$$\hat{x}_1(t) = \begin{cases} 4-t & \text{for } t \in [0,4) \\ 0 & \text{for } t \in [4,10] \end{cases}$$

$$\hat{x}_2(t) = \begin{cases} 4+t & \text{for } t \in [0,2) \\ 6 & \text{for } t \in [2,4) \\ 10-t & \text{for } t \in [4,10] \end{cases}$$

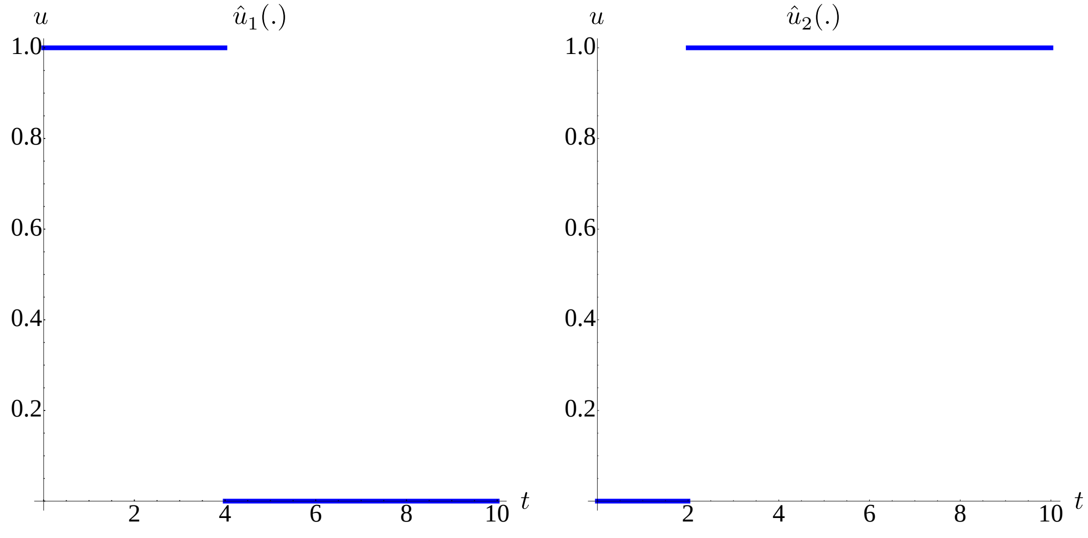


Figure 7: optimal control (Example 6.3)

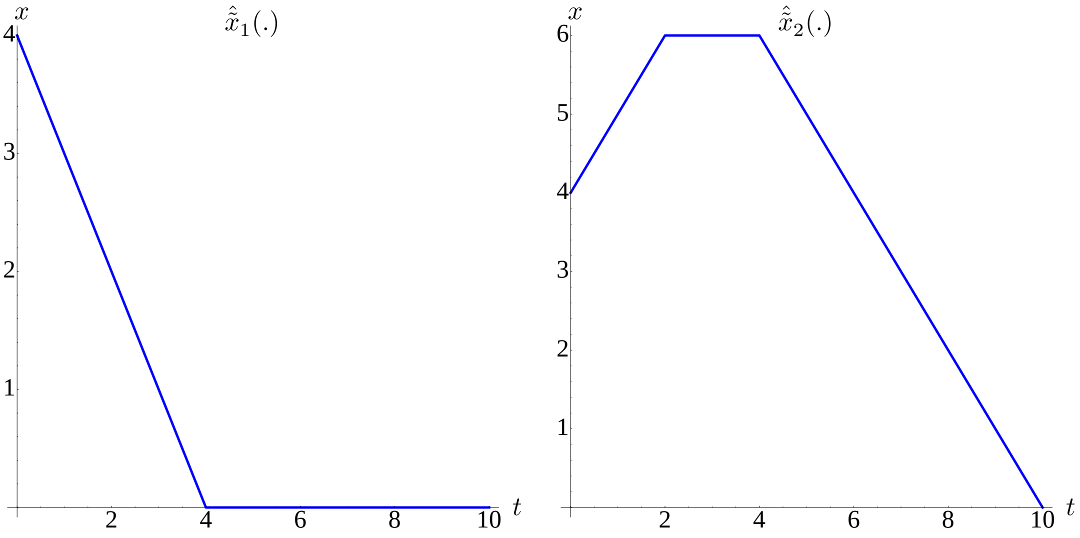


Figure 8: optimal state (Example 6.3)

Knowing $\hat{u}(.)$ delivers the minimal objective function value:

$$\tilde{J}(\hat{\hat{x}}(.), \hat{u}) = \tilde{J}(\hat{u}(.)) = -\int_{0}^{10} (10-t)\, \hat{u}_1(t) + t\, \hat{u}_2(t) dt = -80$$

6.3.2 applying the convergence theorem

Checking if the premises of the Convergence Theorem 3.2.8 are fulfilled has been done in detail for example 6.2 in section 6.2.2. The major difference to that example is, that this time there are simple state constraints present. Although these constraints are simple they are not fully covered by the theory of this thesis. This is because (C1) and (C2) from 5.3 only allow a single scalar state constraint. So this section will argue with (C1E) and (C2E). which are the assumptions for the case including multidimensional state constraints (see 5.3.4).

For investigations we need the Mayer form of the problem, which is:

Problem (set-valued Mayer form)

$$
\begin{aligned}
\text{Minimize :}\quad & J(x(0),x(10)) = z(10)\\
\text{with respect to :}\quad & \\
& \dot{x}(t) \in F(t,x(t)) = \Big\{ \begin{pmatrix} -u_1 \\ u_1 - u_2 \\ -(10-t)\,u_1 - t\,u_2 \end{pmatrix} \;\Big|\; u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in [0,1]^2 \Big\} \quad a.e\\
& s_1(t,\tilde{x}(t)) = -x_1(t) \le 0\\
& s_2(t,\tilde{x}(t)) = -x_2(t) \le 0\\
& x(0) = \begin{pmatrix} 4 \\ 4 \\ 0 \end{pmatrix}\\
& u(t) \in [0,1]^2 \qquad\qquad a.e.\\
& \text{with } x(.) \in AC([0,10])^2 \text{ and } u(.) \in L_\infty([0,10])
\end{aligned}
$$

Showing that (A1), (A2) and (A3) apply is pretty easy: First, $F(t,x)$ does not depend on x here. Second, $t \in [0,10]$ and $u(t) \in [0,1]^2$, which means they are bounded. Those two facts directly lead to the desired result.
Showing that $J(.,.)$ is Lipschitz-continuous is also straightforward and works exactly the same way as in 6.2.2. It should also be clear that $\psi(.,.,.)$ is Lipschitz-continuous in all of it's arguments on $I \times S \times U$ (see Theorem 3.2.4) because the controls are bounded.

It is left to show that (C1E) and (C2E) are fulfilled. For details and notations see 5.3 and 5.3.4. Obviously $s(.,.) \in C^{1,L}([0,1] \times \mathbb{R}^3)^2$ and $x \in \partial\Theta_i(t) \Leftrightarrow s_i(t,x) = 0$ $(i = 1,2)$. So (C1E) holds. We will now show, that (C2E) holds for $s_2(.)$, but not for $s_1(.)$. So overall (C2E) won't be fulfilled for this example. Nevertheless taking a look at the exact optimal solution for the state reveals, why violating (C2E) still leads to pleasant convergence results. Those results are presented in detail in the next section.
Let $x = \binom{\tilde{x}}{z}$. As $s_i(.,.)$ $(i = 1,2)$ does not depend on z, the last component of $\nabla s_i(t,x)$ $(i = 1,2)$ equals 0. As $s_1(.)$ just depends on x_1 and $s_2(.)$ just depends on x_2 we have:

$$
\nabla s_1(t,x) = \begin{pmatrix} 0 \\ \frac{\partial}{\partial x_1} s_1(t,x) \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} \text{ and } \nabla s_2(t,x) = \begin{pmatrix} 0 \\ 0 \\ \frac{\partial}{\partial x_2} s_2(t,x) \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \end{pmatrix}
$$

With the first and the second component of $F(t,x)$ not depending on (t,x) we know that $\min\limits_{v \in F(t,x)} \langle \nabla s_i(t,x), \binom{1}{v} \rangle$ $(i = 1,2)$ is independent of (t,x). Both expressions just depend on $u \in [0,1]^2$.

So for the first constraint (i =1) we have for all $(t,x) \in [0,1] \times \mathbb{R}^3$:

$$\min_{v \in F(t,x)} \langle \nabla s_1(t,x), \left(\begin{smallmatrix}1\\v\end{smallmatrix}\right) \rangle = \min_{u_1 \in [0,1]} \frac{\partial}{\partial x_1} s_1(t,x)(-u_1) = \min_{u_1 \in [0,1]} u_1 \geq 0$$

This violates (C2E).
For the second constraint (i =2) we have for all $(t,x) \in [0,1] \times \mathbb{R}^3$:

$$\min_{v \in F(t,x)} \langle \nabla s_2(t,x), \left(\begin{smallmatrix}1\\v\end{smallmatrix}\right) \rangle = \min_{u \in [0,1]^2} \frac{\partial}{\partial x_2} s_2(t,x)(u_1 - u_2) = \min_{u \in [0,1]^2} u_2 - u_1 = -1 < 0$$

So the second constraint does not pose any problems for the "inward steering" process, used in the proof of Theorem 5.3.2.
The following plots show the problematic region (red) and the one that does not pose any difficulties (green). The gray arrows symbolize the gradients of $s_1(.)$ and $s_2(.)$. The blue curve is the exact solution and the red dots (interpolated with a green linear spline) show the optimal discrete state. Although (C2E) is clearly violated and the optimal solution stays in the critical (red) zone for about half the time, this does not seem to affect the optimal discrete solution negatively. This probably has to do with the simple structure of the problem, which leads to the fact, that the ODE solver does not have to do any redirection of the solution towards the feasible set $\Theta(.) \equiv \{x \in \mathbb{R}^3 \mid x_1 \geq 0,\ x_2 \geq 0\}$.

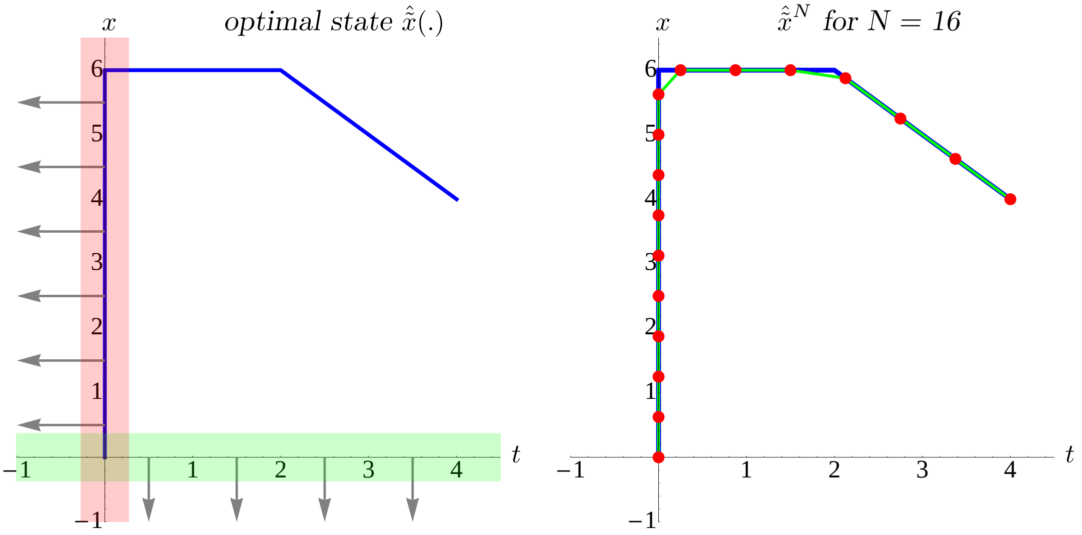


Figure 9: left: red shows the problematic area for (C2E), green the one where no problems occur; right: the optimal discrete solution shows no negative influence (Example 6.3)

The last thing to do is obtain an inverse stability property. Second order sufficient optimality conditions do not hold, because all second Fréchet derivatives vanish. Also first order optimality conditions do not hold. So we are leaving this as an unsolved issue.

6.3.3 convergence analysis

The results for this section have all been calculated using the NLPQLP optimizer (see [11]) and exact derivatives, which provided the best results out of several scenarios. The term best in this case means that the calculated discrete solutions were closest to the exact optimal solutions derived in 6.2.1.

6.3.3.1 first approach

As computations for this example are more memory and calculation power consuming than for example 6.2, the first approach to get an estimation for the convergence rage may be to use multiples of 20 for the number of steps. The following double logarithmic plot of the state distances shows that this is not such a good idea.

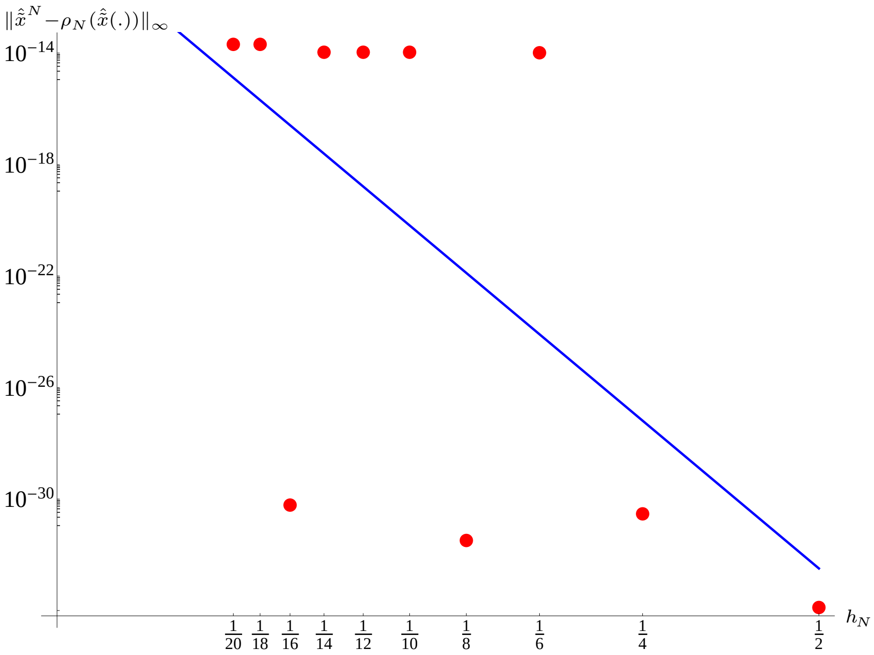


Figure 10: double logarithmic plot of $\|\hat{\hat{x}}^N - \rho_N(\hat{\hat{x}}(.))\|_\infty$ for $N = 20\,k$ $(k = 1, \ldots, 10)$ (Example 6.3)

To shed some light on that result we take a look at the discrete state for $N = 5$, i.e. a really small stepsize. Again, the blue line represents the exact optimal solution whereas the red dots depict the discrete solution. The green line is the interpolation of the discrete solution with a linear spline.

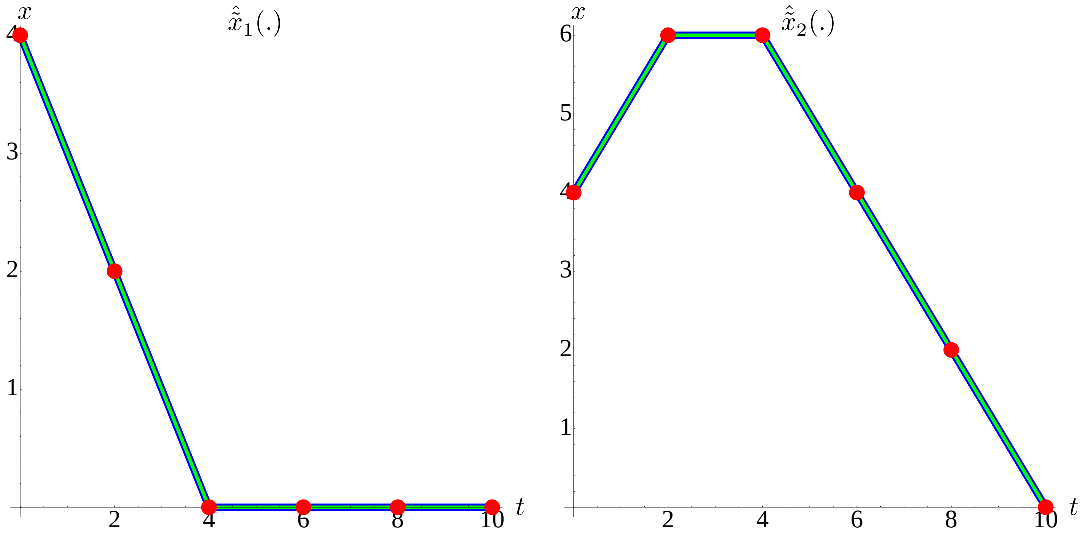


Figure 11: discrete optimal state for $N = 5$ (Example 6.3)

So even for such a small number of steps the discrete solution points lie pretty much exactly on the curve of the optimal state $\hat{\hat{x}}(.)$. This has to do with two things. First, the optimal solution is progressively linear. Second, the switching points of the optimal control $\hat{u}(.)$ are $t = 4$ for $\hat{u}_1(.)$ and $t = 2$ for $\hat{u}_2(.)$. So if the number of steps N is a multiple of 5, these switching points lie on the grid. This leads to Euler's Method being able to deliver an exact solution to the ODE. If the optimizer is now supposed to deliver an exact solution to the directly discretized problem (see 2.4.2), the numerical solution is exact for $N = 5\,k$ ($k \in \mathbb{N}$). Of course the numerical errors are not avoidable and the accuracy of the optimizer has been set to 10^{-8}. This should explain Figure 13. The consequence of this behavior is to use no multiple of 5 for the number of steps.

6.3.3.2 second approach

One way to avoid multiples of 5 for N, that will be used here, is to set $N = 2^k$ ($k \in \mathbb{N}$) like in example 6.2. This time we will just investigate $N = 2^k$ ($k = 2, \ldots, 9$).

state convergence

For $N = 8$ the discrete optimal state looks the following way:

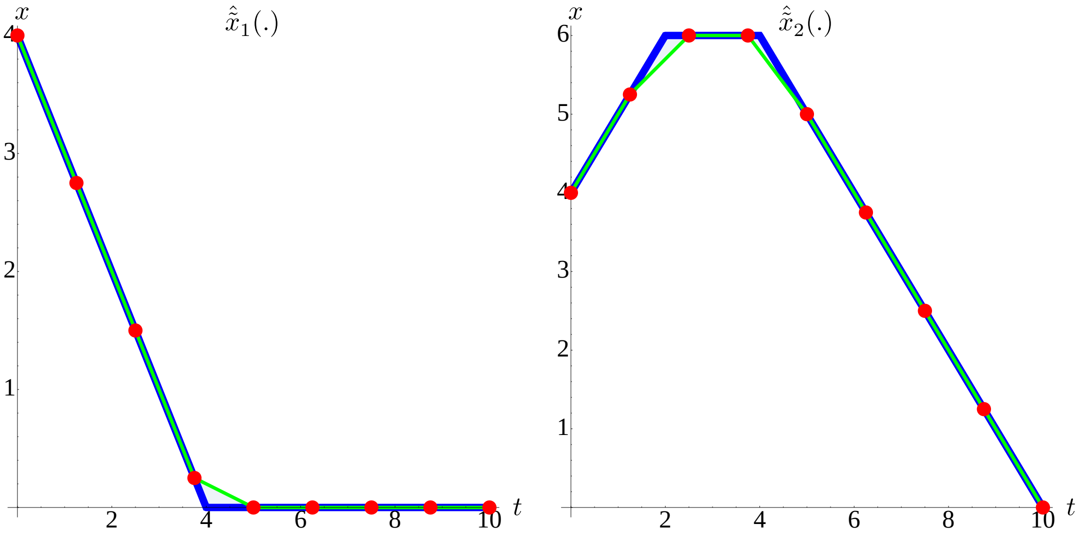


Figure 12: discrete optimal state for $N = 8$ (Example 6.3)

As one can see the discrete points pretty much lie on the exact curve again. This can be observed for the other cases ($N = 2^k$ $(k = 2, \ldots, 9)$), too. Again, this is a really good result, but is not that great for convergence analysis, because the results are kind of too good. Nevertheless, the double logarithmic plot of the state distances (calculated with respect to the discrete supremum norm) shows a better result now than for the first approach.

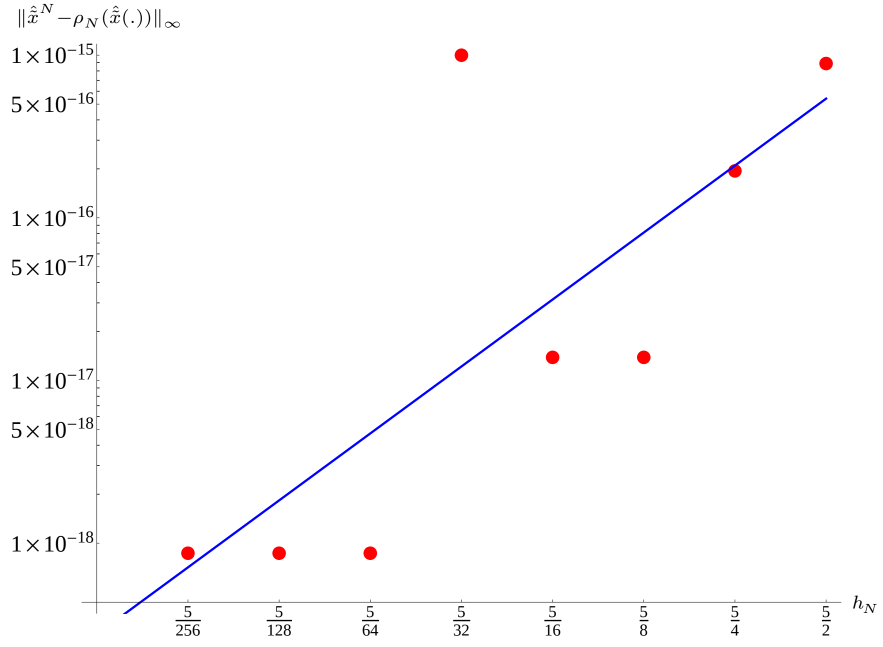


Figure 13: double logarithmic plot of $\|\hat{\hat{x}}^N - \rho_N(\hat{\hat{x}}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The regressionline $r(.)$ (blue) is defined by the following expression:

$$r(t) = -36.4071456 + 1.3674625\, t$$

Principally this is a perfect result. But as the distance values are that small and relatively seen wide spread this result can't be taken too serious.
To get a more convincing result, considering figure 12 leads to the idea of comparing the linear spline (green curve) to the exact curve using the L_∞-norm. The linear spline, connecting the discrete solution points, shall be named $\hat{\tilde{x}}^N(.)$. The L_∞-norm is calculated by $\max(\|\hat{\tilde{x}}^N(2) - \hat{\tilde{x}}(2)\|_\infty, \|\hat{\tilde{x}}^N(4) - \hat{\tilde{x}}(4)\|_\infty)$. The fact that $t = 2$ and $t = 4$ are the switching points for the control means they are the buckling points for the state. As already shown, numerical analysis delivers that $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|$ is almost zero. This means, that the discrete solution points pretty much lie on the exact curve. So the maximum distance has to occur at $t = 2$ or $t = 4$.
Taking a look at $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$, when actually trying to gather information about the convergence rate of $\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty$ makes sense because the Compatibility Property 3.2.6 delivers:

$$\|\hat{\tilde{x}}^N - \rho_N(\hat{\tilde{x}}(.))\|_\infty \leq C h_N + \|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$$

This is explained in more detail in 6.1.1 in the discussion about norms. And in fact, the double logarithmic plot associated with $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ shows a way more convincing result.

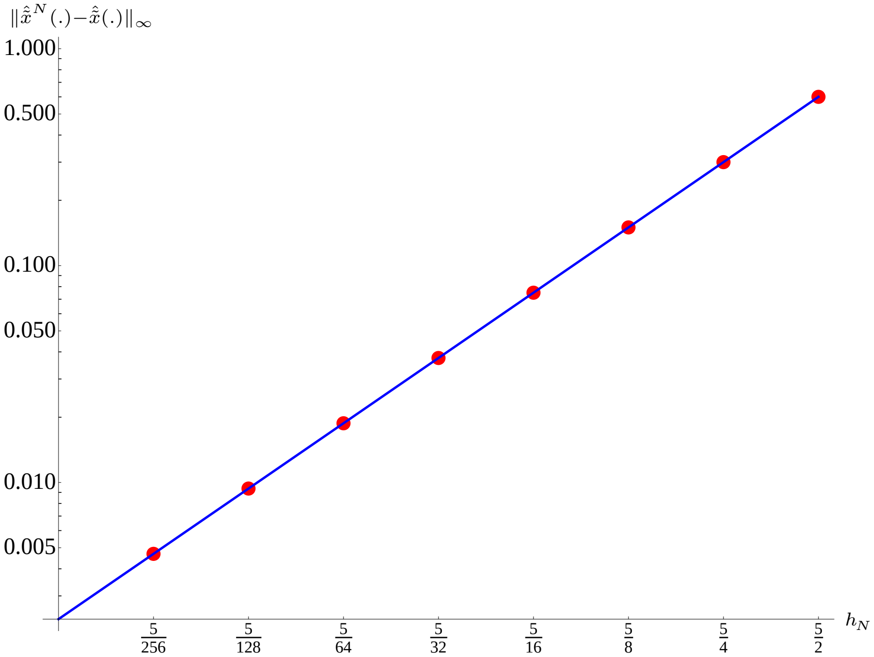


Figure 14: double logarithmic plot of $\|\hat{\tilde{x}}^N(.) - \hat{\tilde{x}}(.)\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The corresponding regressionline $r(.)$ (blue) is defined by the following expression:

$$r(t) = -1.42711636 + 1.00000000\, t$$

This really is a perfect result. So according to the norms discussion in 6.1.1, the estimation for the regression rate p would be $p = \min(1, 1.00000000) = 1$.

Overall the following values have been obtained for the state distances:

steps	stepsize	$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$	$\|\hat{x}^N(.) - \hat{x}(.)\|_\infty$
4	5/2	$8.881784 \cdot 10^{-16}$	0.600000
8	5/4	$1.942890 \cdot 10^{-16}$	0.300000
16	5/8	$1.387779 \cdot 10^{-17}$	0.150000
32	5/16	$1.387779 \cdot 10^{-17}$	0.075000
64	5/32	$9.992007 \cdot 10^{-16}$	0.037500
128	5/64	$8.673617 \cdot 10^{-19}$	0.018750
256	5/128	$8.673617 \cdot 10^{-19}$	0.009375
512	5/256	$8.673617 \cdot 10^{-19}$	0.004688

control convergence

As the optimal control jumps at $t = 2$ and $t = 4$, it is very likely that the discrete controls won't converge in the discrete supremum. This is a fact well known in optimal control theory. The following plots of the control substantiate that forecast.

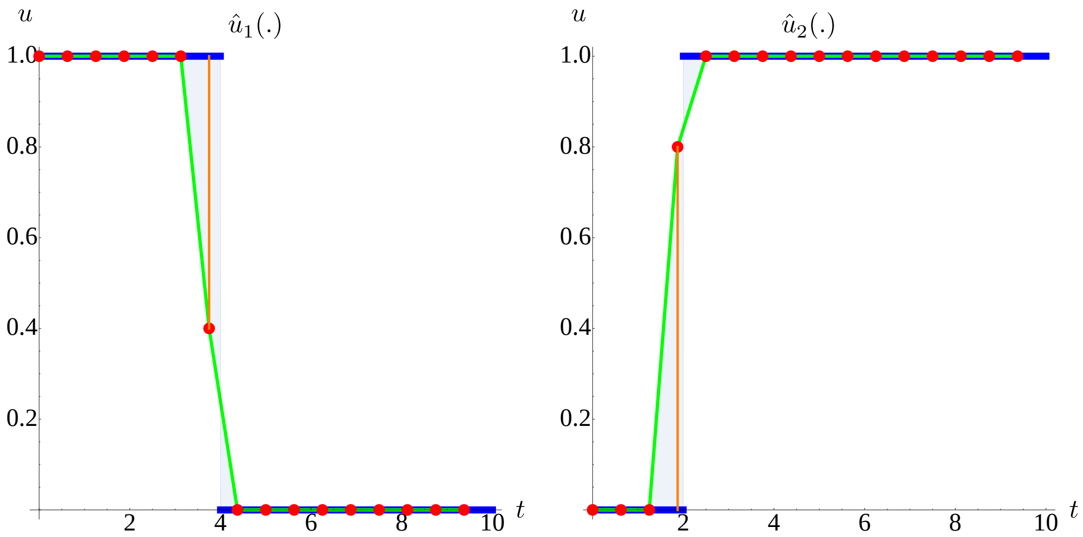


Figure 15: discrete optimal control for $N = 8$ (Example 6.3)

The single discrete point sitting near the switching point of the control does not vanish when using greater number of steps.

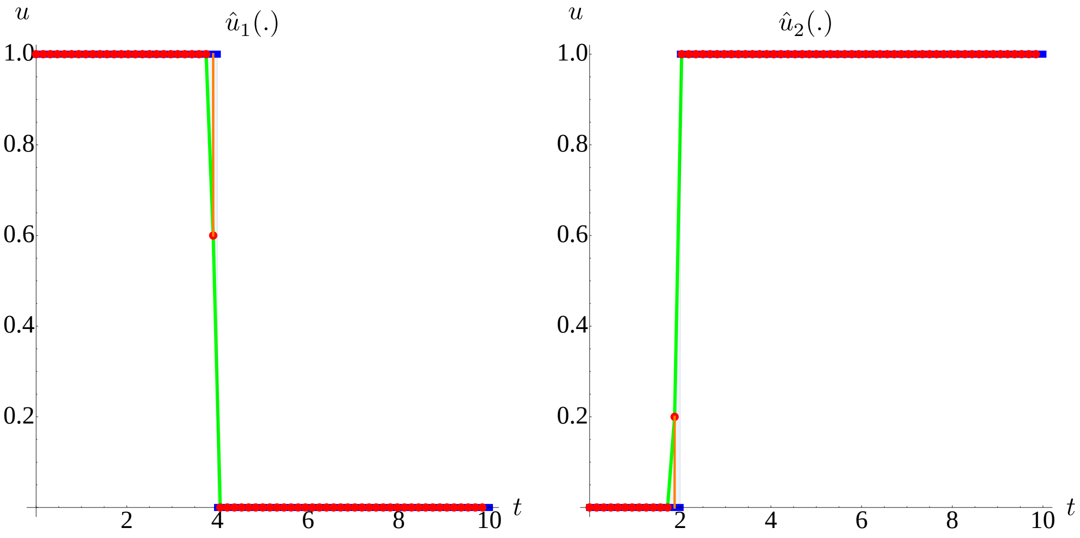


Figure 16: discrete optimal control for $N = 64$ (Example 6.3)

The double logarithmic plot of the discrete supremum norm shows that the distances are not decreasing. Instead they have somewhat random values in $[0, 1]$, which is expected from the control plots above.

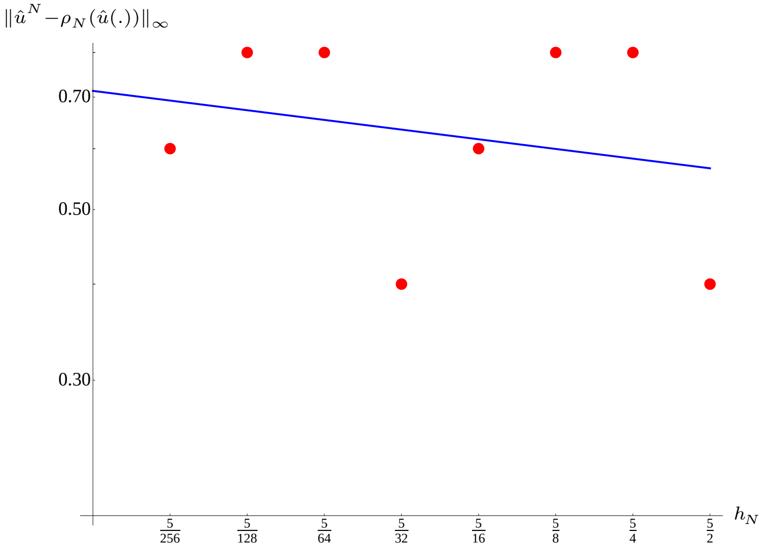


Figure 17: double logarithmic plot of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The discrete L_2-norm should deliver different results. This is because each point is weighted by the steplength. So the one point, that ruined it all in the case of applying the supremum norm, plays more and more less of a role when the stepsize decreases. This leads to the following result:

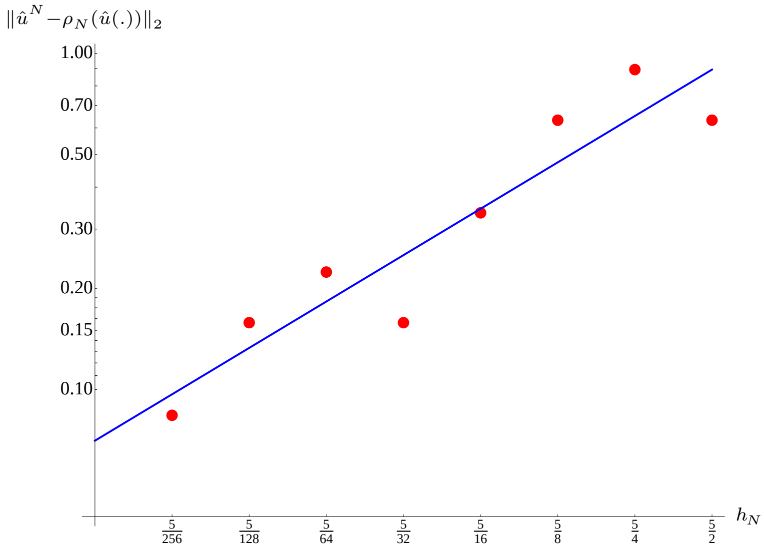


Figure 18: double logarithmic plot of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$-0.531431733 + 0.458216964\, t$$

This suggests a convergence rate for the controls with respect to the discrete L_2-norm with rate $p = 0.458216964 \approx 1/2$.
As the plots in Figure 15 and 16 suggest, comparing the exact curve $\hat{u}(.)$ and the green linear spline $\hat{u}^N(.)$ using the L_2-norm should deliver good results for the convergence analysis. In deed, comparing the calculated values delivers

$$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2 \leq \|\hat{u}^N(.) - \hat{u}(.)\|_2 \qquad (N = 2^k\ (k = 2, \ldots, 9))$$

So estimating the convergence rate of $\|\hat{u}^N(.) - \hat{u}(.)\|_2$ should deliver a good guess for the convergence rate of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$.

The double logarithmic plot is even more convincing:

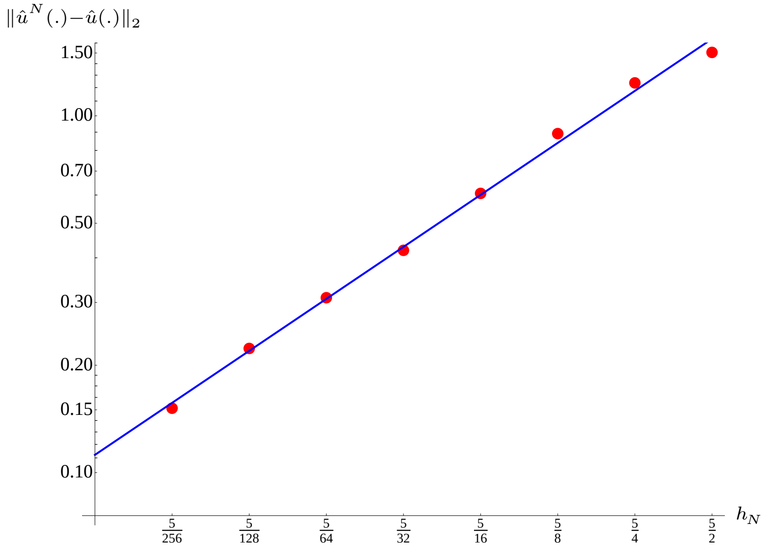


Figure 19: double logarithmic plot of $\|\hat{u}^N(.) - \hat{u}(.)\|_2$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$0.0529738111 + 0.484072793\, t$$

So the estimation of the convergence rate $p = 0.484072793 \approx 1/2$ is pretty much identical to the estimation we got from the regression line of the control distances with respect to the discrete L_2-norm.

Overall the following values have been obtained for the control distances:

steps	stepsize	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$	$\|\hat{u}^N(.) - \hat{u}(.)\|_2$
4	5/2	0.400000	0.632456	1.505680
8	5/4	0.800000	0.894427	1.236931
16	5/8	0.800000	0.632456	0.891628
32	5/16	0.600000	0.335410	0.606218
64	5/32	0.400000	0.158114	0.419821
128	5/64	0.800000	0.223607	0.309233
256	5/128	0.800000	0.158114	0.222907
512	5/256	0.600000	0.083853	0.151554

objective function value convergence

For comparison of the objective funciton values only the absolute value comes into play. No other norms need to be considered.

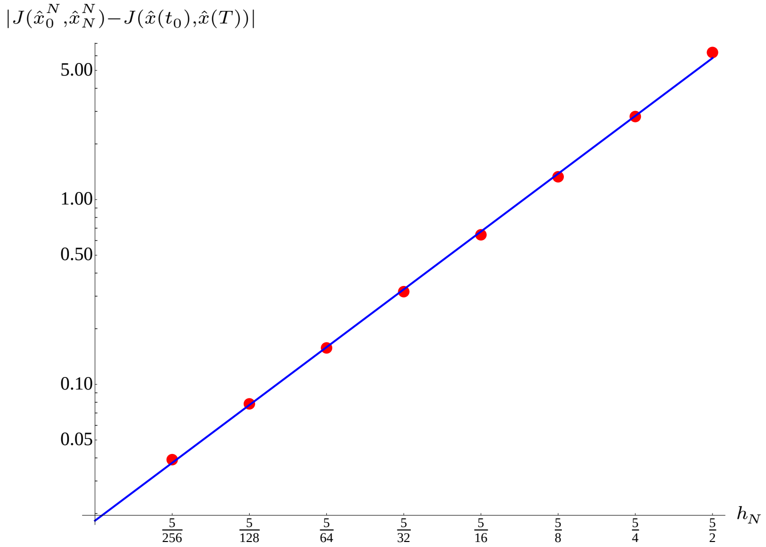


Figure 20: double logarithmic plot of $|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))|$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$0.810248484 + 1.03935746\, t$$

So the estimated convergence rate of the objective function values is $p = 1.03935746 \approx 1$.

Summary

The convergence rate for the state has been estimated to be $p = 1$. For the control we got $p = 0.484072793 \approx 1/2$ and for the objective function values $p = 1.03935746 \approx 1$. Even though (C2E) is violated and sufficient order optimality conditions do not hold, we get pretty good convergence results.

The decisive values for obtaining these results are:

steps	stepsize	$\|\hat{u}^N(.) - \hat{u}(.)\|_2$	$\|\hat{x}^N(.) - \hat{x}(.)\|_\infty$	$\lvert J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))\rvert$
4	5/2	1.505680	0.600000	6.250000
8	5/4	1.236931	0.300000	2.812500
16	5/8	0.891628	0.150000	1.328125
32	5/16	0.606218	0.075000	0.644531
64	5/32	0.419821	0.037500	0.317383
128	5/64	0.309233	0.018750	0.157471
256	5/128	0.222907	0.009375	0.078430
512	5/256	0.151554	0.004688	0.039139

6.4 High peak with multidimensional state constraints

This example has been constructed by the author to present another example with multidimensional state constraints, which is all about two pretty steep curves, that built a huge peek. This example is not very easy to handle for the optimizers and shows some interesting, yet unmentioned, results. Presentation of this example will be devided into two parts. Those parts use slightly different parameters for the following parametrized optimization problem.

Problem

$$\text{Minimize}: \qquad \tilde{J}(\tilde{x}(.), u(.)) = -\int_0^1 \tilde{x}(t)dt$$

with respect to :

$$\dot{\tilde{x}}(t) = u^3(t) \qquad a.e$$

$$\tilde{x}(0) = 1/2$$

$$s_1(t, \tilde{x}(t)) = \tilde{x}(t) - \frac{e^{10d} + 7e^{10t} - 8}{e^{10d} - 1} \leq 0$$

$$s_2(t, \tilde{x}(t)) = \tilde{x}(t) - \frac{7e^{\frac{10d(t-1)}{d-1}} + e^{10d} - 8}{e^{10d} - 1} \leq 0$$

$$u(t) \in [u_{\min}, u_{\max}] \qquad a.e.$$

$$\text{with } \tilde{x}(.) = \in AC([0,1]) \text{ and } u(.) \in L_\infty([0,1])$$

As can be seen the free parameters are d, $u_{\min}$ and $u_{\max}$. d determines the timepoint the peak appears, but more on that in the next section.

6.4.1 notes on constructing the example

constructing the state constraints Let's take a look at the constraints first, which make up the main part of constructing the problem. The idea is to construct $\tilde{s}_1(.)$ and $\tilde{s}_2(.)$ with $x(t) \leq \tilde{s}_1(t)$ and $x(t) \leq \tilde{s}_2(t)$ for $t \in [0,1]$. For $d = 1/2$ the relevant part of the plot of the constraints looks the following way (note the scaling of the axis, the curves are actually much steeper):

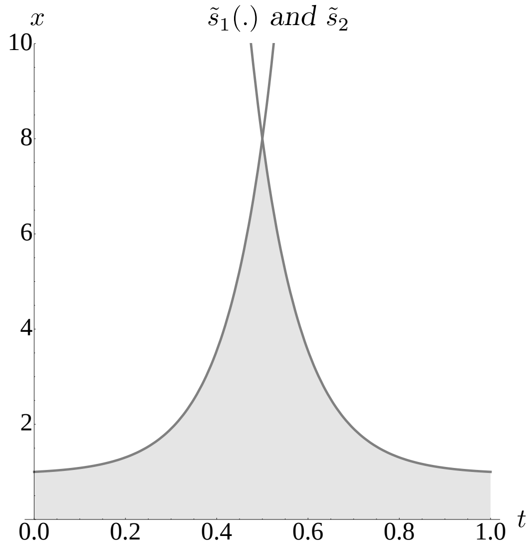


Figure 21: state constraints (gray) and feasible set (light gray) for $d = 1/2$ (Example 6.4)

Despite the fact of having two dimensional constraints it is desired to fulfill (C1), so $s(.,.) \in C^{1,L}([0,1] \times R^2)$ should hold. Therefore $\tilde{s}_1(.)$ and $\tilde{s}_2(.)$ should be in $C^{1,L}(I)$. The approach taken here is using exponential functions for $\tilde{s}_1(.)$ and $\tilde{s}_2(.)$. Starting point is setting

$$\begin{aligned}\tilde{s}_1(t) &= b_1 e^{a_1 t} + c_1 \\ \tilde{s}_2(t) &= b_2 e^{a_2(-t+1)} + c_2\end{aligned}$$

with free parameters b_1, b_2, a_1, a_2, c_1 and c_2.
Those functions should also fulfill the following conditions:

$$\begin{aligned}\tilde{s}_1(0) = \tilde{s}_2(1) = h_s \\ \tilde{s}_1(d) = \tilde{s}_2(d) = h_c\end{aligned}$$

So d is the timepoint at which the two curves intersect at h_c. The other new parameter is h_s. Setting $c_1 = c_2 =: c$ leads to $b_1 = b_2 =: b$ and $a_2 = \frac{d}{1-d} a_1$. Let's set $a := a_1$. Further calculation then leads to

$$b = \frac{h_c - h_s}{e^{ad} - 1} \quad \text{and} \quad c = \frac{h_s e^{ad} - h_c}{e^{ad} - 1}$$

The final state constraint is then represented by

$$s(t, x(t)) = \begin{pmatrix} s_1(t, x(t)) \\ s_2(t, x(t)) \end{pmatrix} = \begin{pmatrix} x(t) - \tilde{s}_1(t) \\ x(t) - \tilde{s}_2(t) \end{pmatrix} = \begin{pmatrix} x(t) - b\, e^{at} - c \\ x(t) - b\, e^{\frac{d}{1-d} a(-t+1)} - c \end{pmatrix}$$

Setting the starting height $h_s = 1$, the crossing height $h_c = 8$ and the steepness control $a = 10$ leaves only d as a free parameter. This is the form used for this example.

objective function and ODE

The objective function has been chosen in such a way, that the state will try to reach the highest values possible obeying the constraints. The slope of the state $\tilde{x}(.)$ is represented pretty much directly by the control $u(.)$, which can be seen by taking a look at the ODE. The exponent 3 for the control has just been chosen instead of an exponent 1 to make the discretized example nonlinear. As we will see later on this simple nonlinearity is enough to let the approach of using approximated derivatives in the computation process fail. Because of the simple structure of the ODE, u_{min} and u_{max} directly determine if the state will be able to reach the peak of the mountain (i.e. the point $h_c = 8$ for $t = d$) or not. If $\sqrt[3]{u_{\max}} \leq \max\limits_{t\in[0,d]} \dot{\tilde{s}}_1(t)$ or if $\sqrt[3]{u_{\min}} \leq \min\limits_{t\in[d,1]} \dot{\tilde{s}}_2(t)$, this will not be possible. In Problem 6.4.2 the state will reach the peak, in Problem 6.4.3 it will not. Instead of reaching the top, the state will detach from $\tilde{s}_1(.)$ and connect to $\tilde{s}_2(.)$ some time later on.

6.4.2 Problem 1: tracing the state constraint curves

In this subexample the free parameters are set to $\boldsymbol{d = 1/\sqrt{3}}$, $\boldsymbol{u_{\min} = -5}$ and $\boldsymbol{u_{\max} = 5}$. So the state constraints are:

$$s_1(t) = x(t) - \tilde{s}_1(t) = x(t) - \frac{7e^{10t} - 8 + e^{10/\sqrt{3}}}{e^{10/\sqrt{3}} - 1} \leq 0$$

$$s_2(t) = x(t) - \tilde{s}_2(t) = x(t) - \frac{7e^{\frac{10\sqrt{3}(t-1)}{\sqrt{3}-3}} - 8 + e^{10/\sqrt{3}}}{e^{10/\sqrt{3}} - 1} \leq 0$$

$d = 1/\sqrt{3}$ has been chosen, so that no point on the grid will ever coincide with the position of the peak. That way a better convergence analysis is possible.

6.4.2.1 deriving the optimal state and control

This time there is no detailed calculation needed to obtain the optimal solution for the control and the state. As already mentioned the objective function favors high values for the state $\tilde{x}(.)$. So the optimal state will be the one getting as high as possible while obeying the constraints. It starts at $\hat{\tilde{x}}(0) = 1/2$, because that is the starting value for the ODE. It will then try to rise up as fast as possible (this means $\hat{u} = u_{\max}$) till it reaches the first state constraint $\tilde{s}_1(.)$ at the time $\tilde{t}$. To obtain $\tilde{t}$ we have to solve the following equation:

$$\frac{1}{2} + u_{\max}^3\, t = \tilde{s}_1(t) \Leftrightarrow \frac{1}{2} + 125\, t = \frac{7e^{10t} - 8 + e^{10/\sqrt{3}}}{e^{10/\sqrt{3}} - 1}$$

The exact solution to this equation is pretty complicated. The approximated value is $\tilde{t} \approx 0.004$. All calculations in analyzing the data have been done using the exact value. From $\tilde{t}$ on the exact optimal solution for the state follows $\tilde{s}_1(.)$ till it reaches the peak at $t = d$. Reaching the peak is possible because the maximum of the derivative of $\tilde{s}_1(.)$ is smaller than $u_{\max}^3 = 125$ and the minimum of the derivative of $\tilde{s}_2(.)$ is bigger than $u_{\min}^3 = -125$. The ladder condition is important for not violating $s_2(t, x(t)) \leq 0$ after having arrived at the peak. As the derivatives of $\tilde{s}_1(.)$ and $\tilde{s}_2(.)$ are strictly monotonously increasing it holds:

$$\max_{t\in[\tilde{t},d]} \dot{\tilde{s}}_1(t) = \dot{\tilde{s}}_1(d) = -\frac{70\,e^{10/\sqrt{3}}}{1-e^{10/\sqrt{3}}} \approx 70.2182981 < 125$$

$$\min_{t\in[d,1]} \dot{\tilde{s}}_2(t) = \dot{\tilde{s}}_2(d) = \frac{70\,e^{10/\sqrt{3}}}{\sqrt{3}\left(1-1/\sqrt{3}\right)\left(1-e^{10/\sqrt{3}}\right)} \approx -95.919979 > -125$$

So overall we know that the optimal state $\hat{\tilde{x}}(.)$ rises up with maximum derivative $u^3_{\max} = 125$ on $[0,\tilde{t}]$, then traces $\tilde{s}_1(.)$ on $[\tilde{t},d]$ and finally traces $\tilde{s}_2(.)$ on $[d,1]$. So the optimal state is:

$$\hat{\tilde{x}}(t) = \begin{cases} \frac{1}{2} + 125\,t & \text{for } t \in [0,\tilde{t}) \\ \tilde{s}_1(t) & \text{for } t \in [\tilde{t},d) \\ \tilde{s}_2(t) & \text{for } t \in [d,1] \end{cases}$$

with

$$\tilde{s}_1(t) = \frac{7e^{10t} - 8 + e^{10/\sqrt{3}}}{e^{10/\sqrt{3}} - 1} \quad \text{and} \quad \tilde{s}_2(t) = \frac{7e^{\frac{10\sqrt{3}(t-1)}{\sqrt{3}-3}} - 8 + e^{10/\sqrt{3}}}{e^{10/\sqrt{3}} - 1}$$

As $\dot{\hat{\tilde{x}}}(t) = \hat{u}^3(t)$ *a.e.* the optimal control can be directly deduced from the optimal state

$$\hat{u}(t) = \begin{cases} 5 & \text{for } t \in [0,\tilde{t}) \\ \sqrt[3]{\dot{\tilde{s}}_1(t)} & \text{for } t \in [\tilde{t},d) \\ \sqrt[3]{\dot{\tilde{s}}_2(t)} & \text{for } t \in [d,1] \end{cases}$$

with

$$\dot{\tilde{s}}_1(t) = -\frac{70e^{10t}}{1-e^{10/\sqrt{3}}} \quad \text{and} \quad \dot{\tilde{s}}_2(t) = \frac{70e^{\frac{10(1-t)}{\sqrt{3}(1-1/\sqrt{3})}}}{\sqrt{3}\left(1-\frac{1}{\sqrt{3}}\right)\left(1-e^{10/\sqrt{3}}\right)}$$

Knowing $\hat{\tilde{x}}(.)$ delivers the minimal objective function value:

$$\tilde{J}(\hat{\tilde{x}}(.),\hat{u}(.)) = \tilde{J}(\hat{\tilde{x}}(.)) = -\int\limits_0^1 \hat{\tilde{x}}(t)\;dt \approx -2.18960398$$

The exact solution to the integral is too long to present it here.

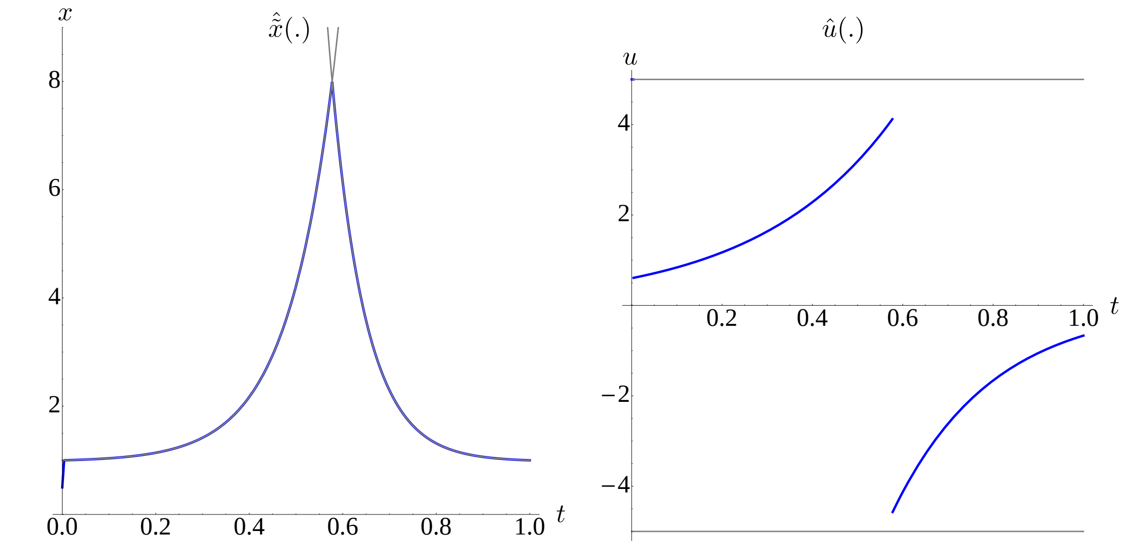


Figure 22: optimal state and control (blue) with constraints (gray) (Example 6.4.2)

6.4.2.2 applying the convergence theorem

Checking if the premises of the Convergence Theorem 3.2.8 are fulfilled has been done in detail for example 6.2 in section 6.2.2. The major difference to that example is, that this time there are multidimensional state constraints present. As in example 6.3 those are not completely covered by the theorey of chapter 5 about the Approximation Property. So again we are using the "experimental" theory for the multidimensional case. This means we need to verify (C1E) and (C2E). Verification of (C2E) depends on the values chosen for the free parameters of the problem. For this problem (Problem 1) we will see, that (C1E) and (C2E) hold. But let's take a look at the other assumptions needed to apply theorem 3.2.8 first.

For investigations we need the Mayer form of the problem, which is:

Problem (set-valued Mayer form)

$$\text{Minimize}: \qquad J(x(0),x(1)) = z(1)$$

$$\text{with respect to}:$$

$$\dot{x}(t) \in F(t,x(t)) = \Big\{ \begin{pmatrix} u^3 \\ -\tilde{x}(t) \end{pmatrix} \ \Big| \ u \in [u_{\min}, u_{\max}] \Big\} \qquad a.e$$

$$\tilde{x}(0) = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}$$

$$s_1(t,\tilde{x}(t)) = \tilde{x}(t) - \frac{e^{10d} + 7e^{10t} - 8}{e^{10d} - 1} \leq 0$$

$$s_2(t,\tilde{x}(t)) = \tilde{x}(t) - \frac{7e^{\frac{10d(t-1)}{d-1}} + e^{10d} - 8}{e^{10d} - 1} \leq 0$$

$$u(t) \in [u_{\min}, u_{\max}] \qquad a.e.$$

$$\text{with } x(.) \in AC([0,1])^2 \text{ and } u(.) \in L_\infty([0,1])$$

Taking a look at F should make clear, that (A1), (A2) and (A3) hold. As the objective function of the Bolza-Problem just consists of the integral term it is once again obvious, that J is Lipschitz-continuous as needed for the Value Convergence Theorem 3.2.2. It should also be clear that $\psi(.,.,.)$ is Lipschitz-continuous in all of it's arguments on $I \times S \times U$ (see Theorem 3.2.4) because the controls and the states are bounded.

It is left to show that (C1E) and (C2E) are fulfilled. For details and notations see 5.3 and 5.3.4. Obviously $s(.,.) \in C^{1,L}([0,1] \times \mathbb{R}^2)^2$ and $x \in \partial\Theta_i(t) \Leftrightarrow s_i(t,x) = 0$ $(i = 1,2)$. So (C1E) holds. Let $(t,x) \in [0,1] \times \mathbb{R}^2$. As $0 \in [u_{\min}, u_{\max}]$ we know that $\tilde{v} = \left(\begin{smallmatrix} 0 \\ \tilde{v}_2 \end{smallmatrix}\right) \in F(t,x)$. Let $x = \left(\begin{smallmatrix} \tilde{x} \\ z \end{smallmatrix}\right)$. As $s_i(.,.)$ $(i = 1,2)$ does not depend on z, the last component of $\nabla s_i(t,x)$ $(i = 1,2)$ equals 0.
So for the first constraint (i =1) we have for all $(t,x) \in [0,1] \times \mathbb{R}^2$:

$$\min_{v \in F(t,x)} \langle \nabla s_1(t,x), \left(\begin{smallmatrix} 1 \\ v \end{smallmatrix}\right) \rangle \leq \langle \nabla s_1(t,x), \left(\begin{smallmatrix} 1 \\ \tilde{v} \end{smallmatrix}\right) \rangle = \frac{\partial}{\partial t} s_1(t,x) = -\dot{\tilde{s}}_1(t) = \frac{70e^{10t}}{1 - e^{10/\sqrt{3}}} \leq \frac{70}{1 - e^{10/\sqrt{3}}} < 0$$

For the second constraint things are a bit more complicated, because setting $\tilde{v} = \left(\begin{smallmatrix} 0 \\ \tilde{v}_2 \end{smallmatrix}\right) \in F(t,x)$ won't do the trick any more. Instead we will use the smallest value for u available and use $\bar{v} = \left(\begin{smallmatrix} -125 \\ \tilde{v}_2 \end{smallmatrix}\right)$ for the estimation process. This time it is essential to do the estimation only for $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_2(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$, where $\mu > 0$ is yet to be determined. So for the second constraint (i =2) we have for all $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_2(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$:

$$\min_{v \in F(t,x)} \langle \nabla s_2(t,x), \left(\begin{smallmatrix} 1 \\ v \end{smallmatrix}\right) \rangle \leq \langle \nabla s_2(t,x), \left(\begin{smallmatrix} 1 \\ \bar{v} \end{smallmatrix}\right) \rangle = \frac{\partial}{\partial t} s_2(t,x) + \frac{\partial}{\partial \tilde{x}} s_2(t,x)(-125) = -\dot{\tilde{s}}_2(t) - 125$$

$$= -\frac{70 e^{\frac{10(1-t)}{\sqrt{3}(1-1/\sqrt{3})}}}{\sqrt{3}\left(1 - \frac{1}{\sqrt{3}}\right)\left(1 - e^{10/\sqrt{3}}\right)} - 125 \leq -\frac{70 e^{\frac{10(1-(d-\mu))}{\sqrt{3}(1-1/\sqrt{3})}}}{\sqrt{3}\left(1 - \frac{1}{\sqrt{3}}\right)\left(1 - e^{10/\sqrt{3}}\right)} - 125$$

The last inequality holds because $-\dot{\tilde{s}}_2(.)$ is monotonously decreasing and

$$\min\left\{ t \;\middle|\; (t,x) \in B_\mu(\operatorname{graph} \partial\Theta_2(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2) \right\} = d - \mu$$

So μ has to be chosen such that

$$d > d - \mu > \underbrace{\min\left\{ t \;\middle|\; -\frac{70 e^{\frac{10(1-t)}{\sqrt{3}(1-1/\sqrt{3})}}}{\sqrt{3}\left(1 - \frac{1}{\sqrt{3}}\right)\left(1 - e^{10/\sqrt{3}}\right)} - 125 \leq 0 \right\}}_{\beta :=} \Leftrightarrow 0 < \mu < d - \beta$$

Calculating β yields:

$$\beta = -\frac{-10\sqrt{3} - 3\log(2) + \sqrt{3}\log(2) + 6\log(5) - 2\sqrt{3}\log(5) - 3\log(7) + \sqrt{3}\log(7)}{10\sqrt{3}}$$
$$- \frac{3\log\left(\sqrt{3}-1\right) - \sqrt{3}\log\left(\sqrt{3}-1\right) + 3\log\left(e^{\frac{10}{\sqrt{3}}} - 1\right) - \sqrt{3}\log\left(e^{\frac{10}{\sqrt{3}}} - 1\right)}{10\sqrt{3}} \approx 0.557965604$$

As $d = 1/\sqrt{3} > \beta$ selecting an appropriate μ is possible. The smaller μ is chosen the bigger $\gamma > 0$ can be chosen such that for all $(t,x) \in B_\mu(\operatorname{graph} \partial\Theta_2(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$ the following inequality holds:

$$\min_{v \in F(t,x)} \langle \nabla s_2(t,x), \left(\begin{smallmatrix} 1 \\ v \end{smallmatrix}\right) \rangle \leq -\gamma$$

Finally setting $\alpha = \min\left(\frac{70}{1-e^{10/\sqrt{3}}}, \gamma\right)$ shows that (C2E) holds.

As (A1), (A2), (A3), (C1E) and (C2E) hold, the Approximation Property 3.2.1 fulfilled. As we additionaly have the Lipschitz-continuity of $J(.,.)$ we know that the Value Convergence Theorem 3.2.2 holds. This means that the objective function values converge with at least rate one.

The thing still missing is the Inverse Stability Property. It is the author's believe, that second order sufficient optimality conditions hold for this example. But this won't be discussed here. For an example on obtaining second order sufficient optimality conditions, see 6.2.2. However in this example, verifying second order optimality conditions involves dealing with the multiplicator $p(.)$, which corresponds to the ODE. This makes things more difficult.

6.4.2.3 convergence analysis

Numerical computations have been done using the NLPIP optimizer with exact derivatives. In this pretty demanding example for the optimizer using exact derivatives has proven to be essential for obtaining useful results. Of course the liability of the optimizer for errors in the derivatives plays a great role here. The authors investigation so far and the experience of the developer of the NLPIP optimizer Björn Sachsenberger show, that the NLPIP optimizer has quite some weaknesses on that area and is outperformed by the NLPQLP optimizer, when using approximated derivatives. But NLPIP was able to handle the calculations for this example pretty well when using exact derivatives, while NLPQLP was not.
Like in the examples before the solutions for the number of steps $N = 2^k$ $(k = 2, \ldots, 9)$ have been computed.

state convergence
To get an impression of what is going on some plots of the discrete state will be presented first. As already mentioned, no gridpoint can conincide with $d = 1/\sqrt{3}$, so the discrete solution will never fully reach the peak. Again, the blue line represents the exact solution, the gray lines the state constraints, the green curve the linear interpolating spline, the red points the computed data and the orange line the maximum distance. Note the scaling of the axes.

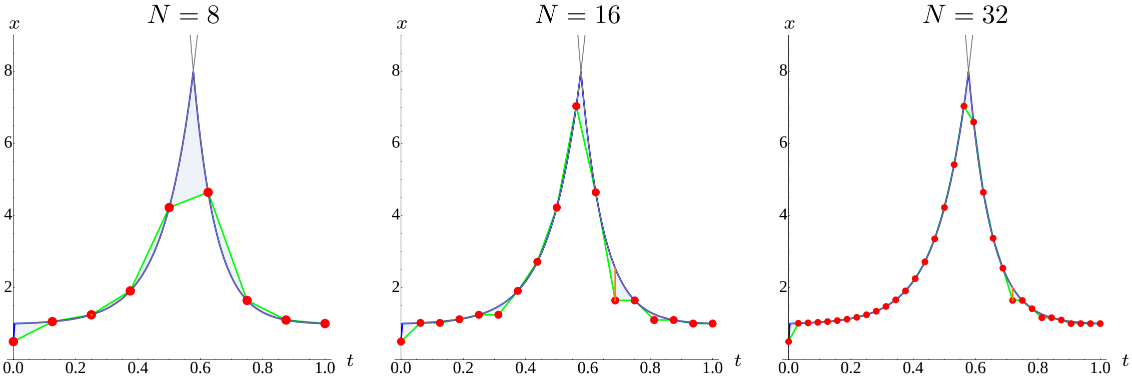


Figure 23: discrete optimal state (Example 6.4.2)

As one can see the discrete points pretty much lie on the exact curve. This can be observed for the other cases $(N = 2^k$ $(k = 2, \ldots, 9))$, too. Again, this is a really good result, but is not that great for convergence analysis, because the results are kind of too good.

Nevertheless, the double logarithmic plot of the state distances (calculated with respect to the discrete supremum norm) shows a result, that is not completely useless.

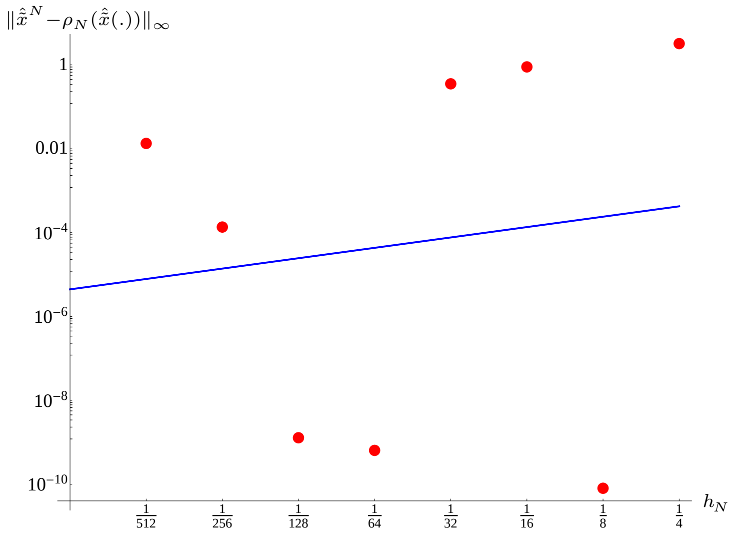


Figure 24: double logarithmic plot of $\|\hat{\hat{x}}^N - \rho_N(\hat{\hat{x}}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.4.2)

The regressionline $r(.)$ (blue) is defined by the following expression:

$$-6.62680774 + 0.822327355\, t$$

Principally this is a good result. But as the distance values are, relatively seen, wide spread this result can't be taken too serious.

To get a more convincing result, considering figure 23 leads to the idea of comparing the linear spline (green curve) to the exact curve using the L_∞-norm. The linear spline, connecting the discrete solution points, shall be named $\hat{\hat{x}}^N(.)$. The L_∞-norm is calculated using numerical maximization routines. Taking a look at $\|\hat{\hat{x}}^N(.) - \hat{\hat{x}}(.)\|_\infty$, when actually trying to gather information about the convergence rate of $\|\hat{\hat{x}}^N - \rho_N(\hat{\hat{x}}(.))\|_\infty$ makes sense because the Compatibility Property 3.2.6 delivers:

$$\|\hat{\hat{x}}^N - \rho_N(\hat{\hat{x}}(.))\|_\infty \leq Ch_N + \|\hat{\hat{x}}^N(.) - \hat{\hat{x}}(.)\|_\infty$$

This is explained in more detail in 6.1.1 in the discussion about norms.

And in fact, the double logarithmic plot associated with $\|\hat{\hat{x}}^N(.) - \hat{\hat{x}}(.)\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ shows a way more convincing result.

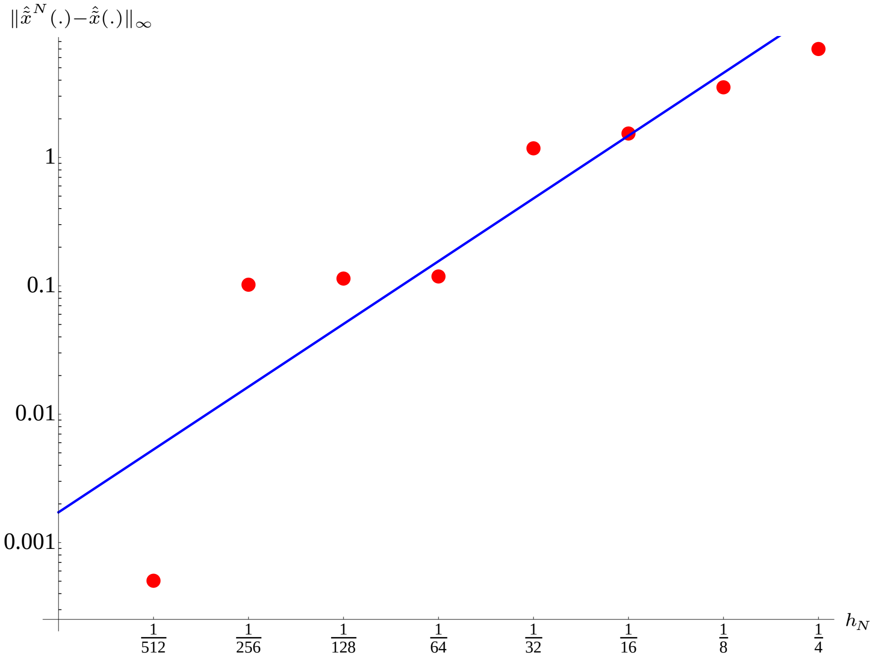


Figure 25: double logarithmic plot of $\|\hat{\hat{x}}^N(.) - \hat{\hat{x}}(.)\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.4.2)

The corresponding regressionline $r(.)$ (blue) is defined by the following expression:

$$r(t) = 4.89486298 + 1.62442965\, t$$

So according to the norms discussion in 6.1.1, the estimation for the regression rate p would be $p = \min(1, 1.62442965) = 1$.

Overall the following values have been obtained for the state distances:

steps	stepsize	$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$	$\|\hat{x}^N(.) - \hat{x}(.)\|_\infty$
4	1/4	3.218001	7.000000
8	1/8	$8.000089 \cdot 10^{-11}$	3.520517
16	1/16	0.895439	1.537000
32	1/32	0.353583	1.178255
64	1/64	$6.400049 \cdot 10^{-10}$	0.118302
128	1/128	$1.280009 \cdot 10^{-9}$	0.113911
256	1/256	0.000136	0.102034
512	1/512	0.013396	0.000504

control convergence
As the optimal control jumps at $t = \tilde{t} \approx 0.004$ and $t = d = 1/\sqrt{3}$, it is very likely that the discrete controls won't converge in the discrete supremum. This is a fact well known in optimal control theory. The following plots of the control substantiate that forecast.

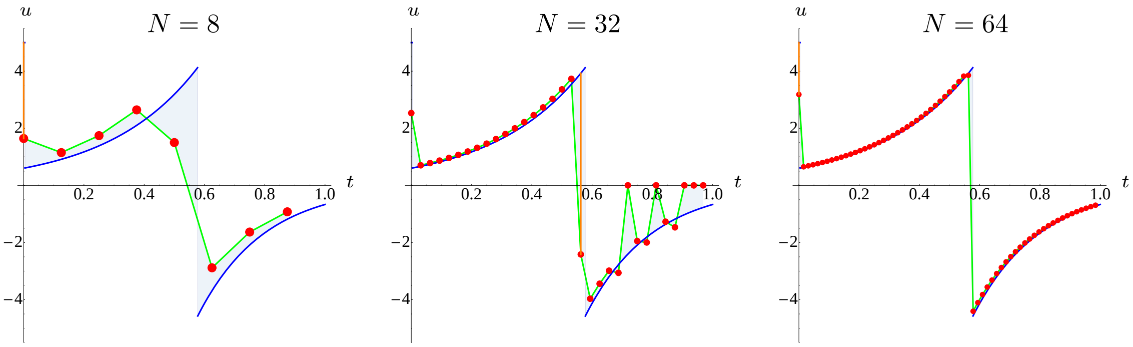


Figure 26: discrete optimal control (Example 6.4.2)

The discrete points sitting near the switching points and not lying on the exact curve of the control do not vanish when using greater number of steps.

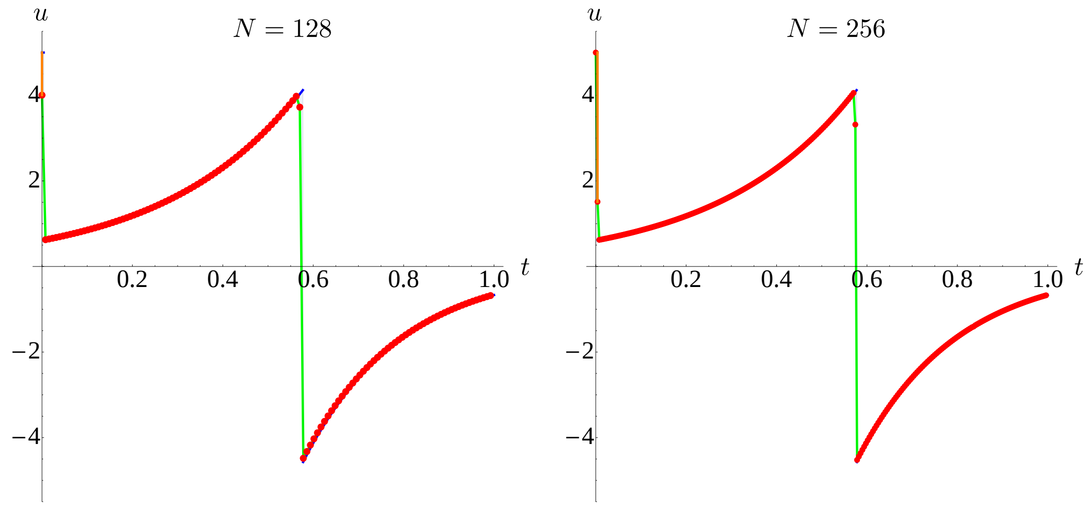


Figure 27: discrete optimal control for higher number of steps (Example 6.4.2)

The double logarithmic plot of the discrete supremum norm shows that the distances are not decreasing. Instead they have somewhat random values in $[0, 8]$, which is expected from the control plots above.

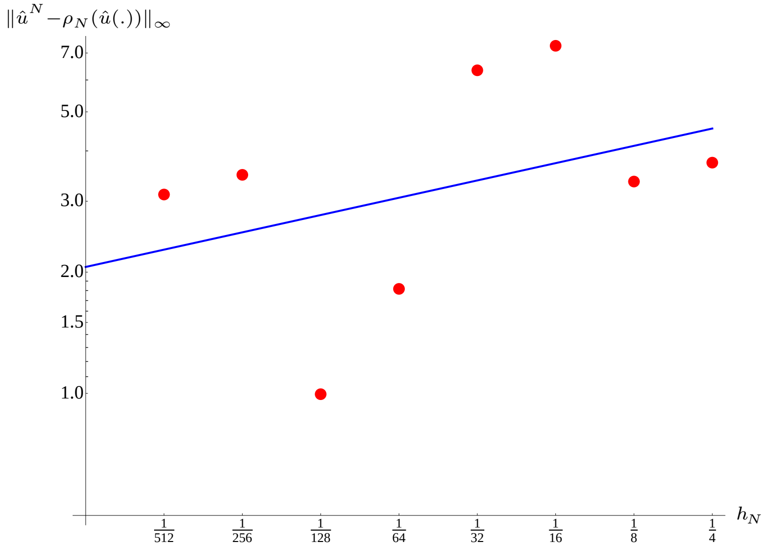


Figure 28: double logarithmic plot of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.3)

The discrete L_2-norm should deliver different results. This is because each point is weighted by the steplength. So the few points, that ruined it all in the case of applying the supremum norm, play more and more less of a role when the stepsize decreases. This leads to the following result:

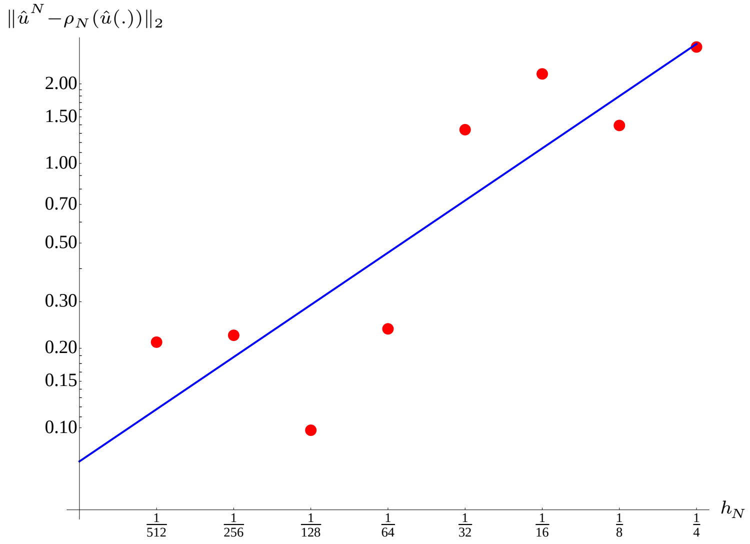


Figure 29: double logarithmic plot of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.4.2)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$1.95010309 + 0.65576042\, t$$

This suggests a convergence rate for the controls with respect to the discrete L_2-norm with rate $p = 0.65576042 \approx 1/2$.
As the plots in Figure 15 and 16 suggest, comparing the exact curve $\hat{u}(.)$ and the green linear spline $\hat{u}^N(.)$ using the L_2-norm should deliver good results for the convergence analysis. So estimating the convergence rate of $\|\hat{u}^N(.) - \hat{u}(.)\|_2$ should deliver a good guess for the convergence rate of $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$.
The double logarithmic plot is even more convincing:

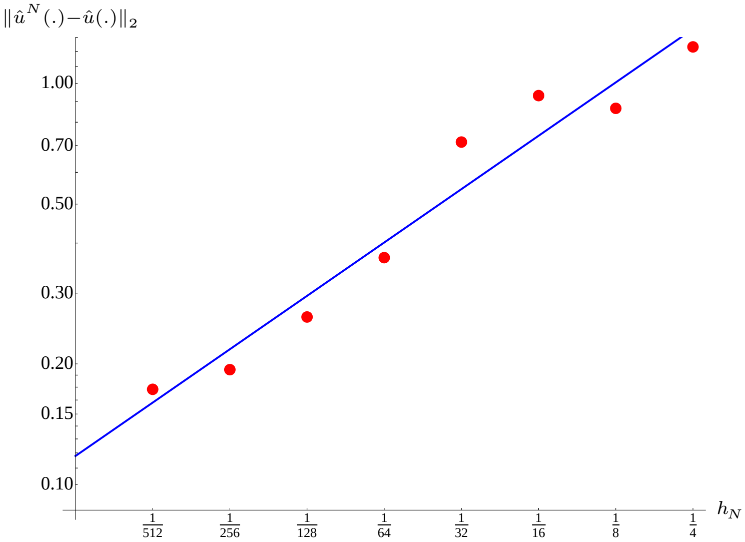


Figure 30: double logarithmic plot of $\|\hat{u}^N(.) - \hat{u}(.)\|_2$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.4.2)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$0.923803152 + 0.441718091\, t$$

So the estimation of the convergence rate $p = 0.441718091 \approx 1/2$ is consistent with the estimation we got from the regression line of the control distances with respect to the discrete L_2-norm.

Overall the following values have been obtained for the control distances:

steps	stepsize	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$	$\|\hat{u}^N(.) - \hat{u}(.)\|_2$
4	1/4	3.740079	2.757310	1.232240
8	1/8	3.357035	1.392250	0.866261
16	1/16	7.295549	2.181487	0.931825
32	1/32	6.341561	1.341855	0.713832
64	1/64	1.817403	0.236735	0.367844
128	1/128	0.995275	0.097895	0.261696
256	1/256	3.488371	0.223840	0.193459
512	1/512	3.117195	0.210850	0.172823

objective function value convergence
For comparison of the objective funciton values only the absolute value comes into play. No other norms need to be considered.

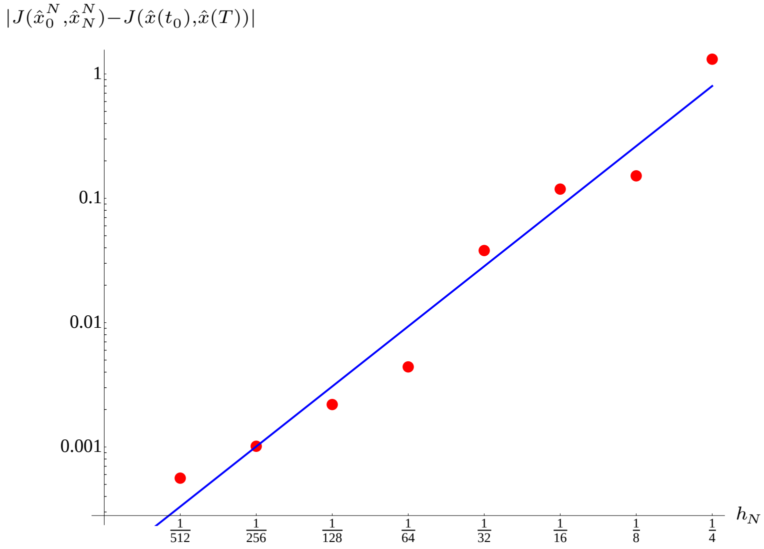


Figure 31: double logarithmic plot of $|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))|$ for $N = 2^k$ $(k = 2, \ldots, 9)$ (Example 6.4.2)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$2.0025268 + 1.60535163\, t$$

So the estimated convergence rate of the objective function values is $p = 1.60535163 \approx 3/2$.

Summary
The convergence rate for the state has been estimated to be $p = 1$. For the control we got $p \approx 1/2$ and for the objective function values $p = 1.60535163 \approx 3/2$.

The decisive values for obtaining these results are:

steps	stepsize	$\|\hat{u}^N(.)-\hat{u}(.)\|_2$	$\|\hat{x}^N(.)-\hat{x}(.)\|_\infty$	$\lvert J(\hat{x}_0^N,\hat{x}_N^N)-J(\hat{x}(t_0),\hat{x}(T))\rvert$
4	1/4	1.232240	7.000000	1.314604
8	1/8	0.866261	3.520517	0.151577
16	1/16	0.931825	1.537000	0.118575
32	1/32	0.713832	1.178255	0.038016
64	1/64	0.367844	0.118302	0.004407
128	1/128	0.261696	0.113911	0.002195
256	1/256	0.193459	0.102034	0.001013
512	1/512	0.172823	0.000504	0.000562

6.4.3 Problem 2: detaching state

When choosing $u_{\max}$ small and $u_{\min}$ high enough the optimal state $\hat{\tilde{x}}(.)$ won't be able to reach the peak any more. Instead, the optimal state will detach from the curve $\tilde{s}_1(.)$ at some point and reconnect some time later on to the curve $\tilde{s}_2(.)$. This will be the major difference to 6.4.2. For this problem we set $\boldsymbol{u_{\max} = 2}$, $\boldsymbol{u_{\min} = 2}$ and for the sake of simplicity $\boldsymbol{d = 1/2}$. The reason for setting $d = 1/\sqrt{3}$ in Problem 1 was to be able to do a better convergence analysis. It is not intended to do a detailed analysis for this problem again. With $d = 1/2$ the state constraints are:

$$s_1(t) = x(t) - \tilde{s}_1(t) = x(t) - \frac{7e^{10t} - 8 + e^5}{e^5 - 1} \leq 0$$

$$s_2(t) = x(t) - \tilde{s}_2(t) = x(t) - \frac{7e^{10-10t} - 8 + e^5}{e^5 - 1} \leq 0$$

6.4.3.1 deriving the optimal state and control

Obtaining the exact solution follows pretty much the way of 6.4.2.1. This time there is also no detailed calculation needed to obtain the optimal solution for the control and the state. The main difference to Problem 1 is, that the optimal solution detaches from $\tilde{s}_1(.)$ at a certain time $d - \tilde{d}$ and reconnects to $\tilde{s}_2(.)$ at $d + \tilde{d}$, with some $\tilde{d} \in [0, d)$. Because of $d = 1/2$ the functions $\tilde{s}_1(.)$ and $\tilde{s}_2(.)$ are symmetrical with respect to the axis $t = 1/2$. This leads to $\dot{\tilde{s}}_1(d - \tilde{d}) = -\dot{\tilde{s}}_2(d + \tilde{d})$ and explains why we only needed to introduce one new parameter $\tilde{d}$.
So let's start at the beginning: As already mentioned the objective function favors high values for the state $\tilde{x}(.)$. So the optimal state will be the one getting as high as possible while obeying the constraints. It starts at $\hat{\tilde{x}}(0) = 1/2$, because that is the starting value for the ODE. It will then try to rise up as fast as possible (this means $\hat{u} = u_{\max}$) till it reaches the first state constraint $\tilde{s}_1(.)$ at the time $\tilde{t}$. To obtain $\tilde{t}$ we have to solve the following equation:

$$\frac{1}{2} + u_{\max}^3\, t = \tilde{s}_1(t) \Leftrightarrow \frac{1}{2} + 125\, t = \frac{7e^{10t} - 8 + e^5}{e^5 - 1}$$

The exact solution to this equation is pretty complicated. The approximated value is $\tilde{t} \approx 0.0683179563$. All calculations in analyzing the data have been done using the exact

value.
From $\tilde{t}$ on the exact optimal solution for the state follows $\tilde{s}_1(.)$ till it the reaches the point $t = 1/2 - \tilde{d}$, with $\tilde{d} > 0$. This means that reaching the peak is not possible because the maximum of the derivative of $\tilde{s}_1(.)$ is bigger than $u^3_{\max} = 8$. As $\tilde{s}_1(.)$ is strictly monotonously increasing it follows that $\tilde{d}$ may be obtained by the following equation, which has exactly one solution:

$$\dot{\tilde{s}}_1(d - \tilde{d}) = \frac{70e^{10(d-\tilde{d})}}{e^5 - 1} = u^3_{\max} = 8 \Leftrightarrow d - \tilde{d} = \frac{1}{10}\left(2\log(2) - \log(5) - \log(7) + \log\left(e^5 - 1\right)\right)$$
$$\approx 0.282418555$$

The point of connecting to $\tilde{s}_2(.)$ (the point where the constraints get active again) is

$$d + \tilde{d} = 2d - (d - \tilde{d}) = 1 - \frac{1}{10}\left(2\log(2) - \log(5) - \log(7) + \log\left(e^5 - 1\right)\right) \approx 0.717581445$$

After connecting to $\tilde{s}_2(.)$ the constraint stays active, i.e. $\hat{x}(.)$ traces $\tilde{s}_2(.)$. The only thing left to find out is how the optimal solution behaves on $[d - \tilde{d}, d + \tilde{d}]$. To do so we once again make use of the maximum principle (for details and the notation of operators see Theorem 5.1.2 in [1]) for this problem without state constraints, because we already know, that they won't turn active on $[d - \tilde{d}, d + \tilde{d}]$:
For this problem the Hamiltonian H is

$$H(t, x(t), u(t), p(t)) = p(t)\, u^3(t) + x(t)$$

The maximum principle then reads

$$H_u(t, x(t), u(t), p(t))(u - \hat{u}(t)) = (3\, p(t)\, u^2(t))(u - \hat{u}(t)) \leq 0 \qquad a.e. \text{ for all } u \in [-2, 2]$$

Let's suppose it exists $[a, b] \subset [d - \tilde{d}, d + \tilde{d}]$ with $a < b$ and $\hat{u}(t) = 0$ on $[a, b]$. Then obviously

$$\tilde{u}(t) := \begin{cases} 2 & \text{for } t \in [a, \frac{a+b}{2}) \\ -2 & \text{for } t \in [\frac{a+b}{2}, b] \end{cases}$$

is a better feasible solution on $[a, b]$ than $\hat{u}(t) \equiv 0$, so $\hat{u}(t) \neq 0$ $a.e.$ on $[d - \tilde{d}, d + \tilde{d}]$.
So we have

$$\hat{u}(t) := \begin{cases} 2 & \text{for } p(t) > 0 \\ -2 & \text{for } p(t) < 0 \end{cases}$$

Considering the boundary conditions $x(d - \tilde{d}) = \tilde{s}_1(d - \tilde{d})$ and $x(d + \tilde{d}) = \tilde{s}_1(d - \tilde{d})$ and the fact that $H_x(t, x(t), u(t), p(t)) = 1$ delivers that $p(t) = -(l_R)_2 + b - t$ for $t \in (d - \tilde{d}, d + \tilde{d}]$. So there is at most one point $c \in (d - \tilde{d}, d + \tilde{d}]$ with $p(c) = 0$. For $x(d - \tilde{d}) = x(d + \tilde{d}) = \tilde{s}_1(d - \tilde{d})$ being fulfilled it must hold $c = d$. As $p(.)$ is monotonously decreasing it follows that:

$$\hat{u}(t) := \begin{cases} 2 & \text{for } t \in [d - \tilde{d}, d) \\ -2 & \text{for } t \in [d, d + \tilde{d}] \end{cases}$$

So overall we know that the optimal state $\hat{\tilde{x}}(.)$ rises up with maximum derivative $u^3_{\max} = 8$ on $[0, \tilde{t}]$, then traces $\tilde{s}_1(.)$ on $[\tilde{t}, d - \tilde{d}]$ then detaches with keeping maximum derivative 8 till $t = d$. Afterwards it decreases with minimum derivative -8 till $t = d + \tilde{d}$ and finally traces $\tilde{s}_2(.)$ on $[d + \tilde{d}, 1]$. So the optimal state is:

$$\hat{\tilde{x}}(t) = \begin{cases} \frac{1}{2} + 8\,t & \text{for } t \in [0, \tilde{t}) \\ \tilde{s}_1(t) & \text{for } t \in [\tilde{t}, d - \tilde{d}) \\ \tilde{s}_1(d - \tilde{d}) + 8\,(t - (d - \tilde{d})) & \text{for } t \in [d - \tilde{d}, d) \\ \tilde{s}_1(d - \tilde{d}) + 8\,\tilde{d} - 8\,t & \text{for } t \in [d, d + \tilde{d}) \\ \tilde{s}_2(t) & \text{for } t \in [d + \tilde{d}, 1] \end{cases}$$

with

$$\tilde{s}_1(t) = \frac{7e^{10t} - 8 + e^5}{e^5 - 1},\ \tilde{s}_2(t) = \frac{7e^{10-10t} - 8 + e^5}{e^5 - 1},\ \tilde{t} \approx 0.0683179563,\ d = 1/2$$
$$\text{and } \tilde{d} = \frac{1}{2} + \frac{1}{10}\left(-2\log(2) + \log(5) + \log(7) - \log\left(e^5 - 1\right)\right) \approx 0.217581445$$

As $\dot{\hat{\tilde{x}}}(t) = \hat{u}^3(t)$ *a.e.* the optimal control can be directly deduced from the optimal state

$$\hat{\tilde{x}}(t) = \begin{cases} 8 & \text{for } t \in [0, \tilde{t}) \\ \sqrt[3]{\dot{\tilde{s}}_1(t)} & \text{for } t \in [\tilde{t}, d - \tilde{d}) \\ 8 & \text{for } t \in [d - \tilde{d}, d) \\ -8 & \text{for } t \in [d, d + \tilde{d}) \\ \sqrt[3]{\dot{\tilde{s}}_2(t)} & \text{for } t \in [d + \tilde{d}, 1] \end{cases}$$

with

$$\dot{\tilde{s}}_1(t) = -\frac{70e^{10t}}{1 - e^5} \text{ and } \dot{\tilde{s}}_2(t) = \frac{70e^{10(1-t)}}{1 - e^5}$$

Knowing $\hat{\tilde{x}}(.)$ delivers the minimal objective function value:

$$\tilde{J}(\hat{\tilde{x}}(.), \hat{u}(.)) = \tilde{J}(\hat{\tilde{x}}(.)) = -\int_0^1 \hat{\tilde{x}}(t)\ dt \approx -1.81298124$$

The exact solution to the integral is too long to present it here.

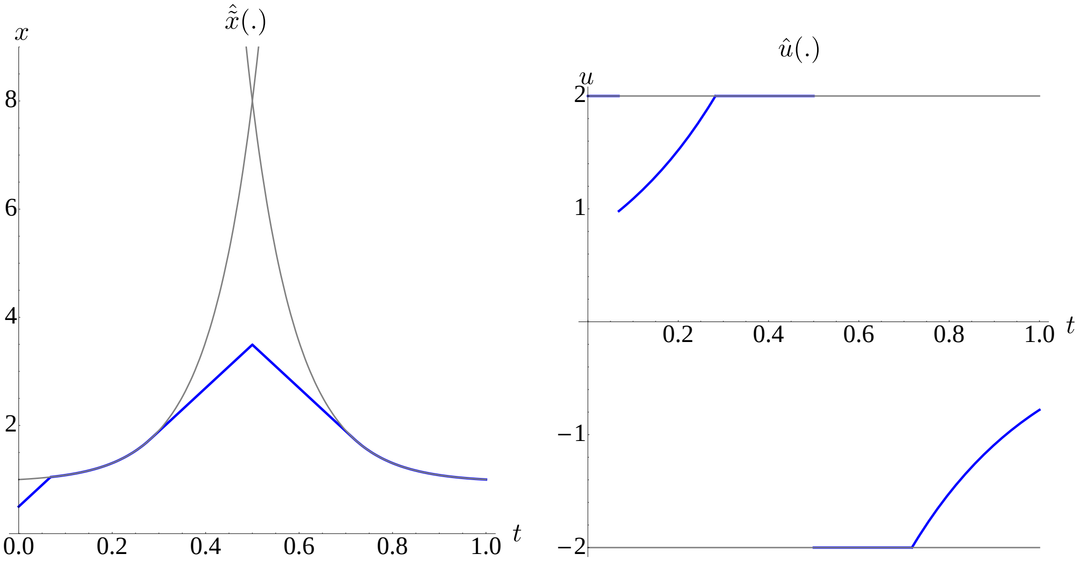


Figure 32: optimal state and control (blue) with constraints (gray) (Example 6.4.3)

6.4.3.2 applying the convergence theorem

The situation here is pretty much the same as for 6.4.2. So most statements made in 6.4.2.2 are true for this problem, too. But the Verification of (C2E) depends on the values chosen for the free parameters of the problem. For this problem (Problem 2) we will see, that (C1E) is fulfilled, but contrary to Problem 1 (C2E) does not hold.
The fact that (A1), (A2) and (A3) hold, that J is Lipschitz-continuous as needed for the Value Convergence Theorem 3.2.2 that $\psi(.,.,.)$ is Lipschitz-continuous in all of it's arguments and that (C1E) is fulfilled has been shown in 6.4.2.2.
It is left to show that (C2E) holds. As already mentioned, this part differs from 6.4.2.2. For details and notations see 5.3 and 5.3.4.
As As $0 \in [u_{\min}, u_{\max}]$ the first part of verifying (C2E) stays the same (except of the exact values appearing in the estimation process): Let $(t, x) \in [0, 1] \times \mathbb{R}^2$. As $0 \in [u_{\min}, u_{\max}]$ we know that $\tilde{v} = \left(\begin{smallmatrix} 0 \\ \tilde{v}_2 \end{smallmatrix}\right) \in F(t, x)$. Let $x = \left(\begin{smallmatrix} \tilde{x} \\ z \end{smallmatrix}\right)$. As $s_i(.,.)$ $(i = 1, 2)$ does not depend on z, the last component of $\nabla s_i(t, x)$ $(i = 1, 2)$ equals 0.
So for the first constraint (i =1) we have for all $(t, x) \in [0, 1] \times \mathbb{R}^2$:

$$\min_{v \in F(t,x)} \langle \nabla s_1(t, x), \left(\begin{smallmatrix} 1 \\ v \end{smallmatrix}\right) \rangle \leq \langle \nabla s_1(t, x), \left(\begin{smallmatrix} 1 \\ \tilde{v} \end{smallmatrix}\right) \rangle = \frac{\partial}{\partial t} s_1(t, x) = -\dot{\hat{s}}_1(t) = \frac{70e^{10t}}{1 - e^5} \leq \frac{70}{1 - e^5} < 0$$

For the second constraint things are more complicated, because setting $\tilde{v} = \left(\begin{smallmatrix} 0 \\ \tilde{v}_2 \end{smallmatrix}\right) \in F(t, x)$ won't do the trick any more. Instead we will use the smallest value for u available and so use $\bar{v} = \left(\begin{smallmatrix} -8 \\ \tilde{v}_2 \end{smallmatrix}\right)$ for the estimation process.

This time it is essential to do the estimation only for $(t, x) \in B_\mu(\operatorname{graph} \partial\Theta_2(.)) \cap B_\mu(\operatorname{graph} \partial\Theta(.)) \cap ([0, 1] \times \mathbb{R}^2)$, where $\mu > 0$ is yet to be determined. So for the second constraint (i =2) we have for all $(t,x) \in B_\mu(\text{graph}\,\partial\Theta_2(.)) \cap B_\mu(\text{graph}\,\partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$:

$$\begin{aligned}\min_{v\in F(t,x)} \langle \nabla s_2(t,x), \left(\begin{smallmatrix}1\\ v\end{smallmatrix}\right)\rangle &= \langle \nabla s_2(t,x), \left(\begin{smallmatrix}1\\ \bar{v}\end{smallmatrix}\right)\rangle = \frac{\partial}{\partial t} s_2(t,x) + \frac{\partial}{\partial \tilde{x}} s_2(t,x)(-8) = -\dot{\tilde{s}}_2(t) - 8 \\ &= -\frac{70e^{10(1-t)}}{1-e^5} - 8 \geq -\frac{70e^{10(1-d)}}{1-e^5} - 8 = -\frac{70e^5}{1-e^5} - 8 > 0\end{aligned}$$

So (C2E) does not hold for all $(t,x) \in B_\mu(\text{graph}\,\partial\Theta_2(.)) \cap B_\mu(\text{graph}\,\partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$. And that is exactly why the detaching case is presented. As the optimal solution coincides with the solution reaching maximum height while obeying the constraints it has to lie on the boundary of the feasible set of all solutions $X_\Theta(1,0,\{1/2\})$. Because of that, verifying (C2E) will always fail, if the optimal solution does not reach the peak. This is because in that case (C2E) does not hold for all $(t,x) \in B_\mu(\text{graph}\,\partial\Theta_2(.)) \cap B_\mu(\text{graph}\,\partial\Theta(.)) \cap ([0,1] \times \mathbb{R}^2)$, it just holds for $(t,x) \in ([0,1] \times \mathbb{R}^2) \cap [d - \tilde{d}, 1] \times \mathbb{R}^2$.

But as we know the feasible set $X_\Theta(1,0,\{1/2\})$, we can weaken (C2E) to (C2EW):

(C2EW) The boundary of $\Theta(.)$ fulfills the "strict inwardness condition". This means that there exist $\alpha, \mu > 0$ such that for each $i \in \{1, \dots, n_s\}$ it holds:
For all $(t,x) \in B_\mu(\text{graph}\,\partial\Theta_i(.)) \cap B_\mu(\text{graph}\,\partial\Theta(.)) \cap B_\epsilon\big(\text{graph}(X_\Theta(T,t_0,X_0))\big)$ the following inequality applies:

$$\min_{v\in F(t,x)} \langle \nabla s_i(t,x), \left(\begin{smallmatrix}1\\ v\end{smallmatrix}\right)\rangle \leq -\alpha$$

with $\text{graph}(X_\Theta(T,t_0,X_0)) := \big\{(t,x) \in \text{graph}(x(.)) \;\big|\; x(.) \in X_\Theta(T,t_0,X_0)\big\}$

For this example it would for example be sufficient, that (C2E) only holds on the violet areas:

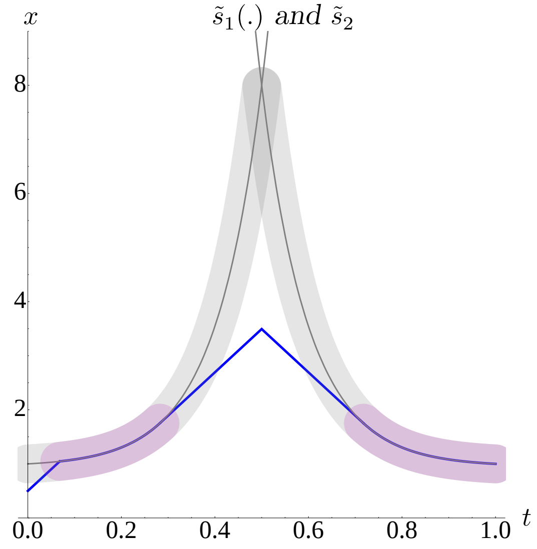


Figure 33: Verification of (C2E): area for (C2E) to be fulfilled on (violet), exact solution (blue), state constraints (gray), postulated area for (C2E) to hold (light gray) (Example 6.4.3)

But even though this way of proceeding may give an idea of why the computed solutions for this problem actually converge (see next section), even (C2EW) is not fulfilled for this problem. The problematic area is the environment of the reconnecting point $t = d + \tilde{d}$. Despite that checking a problem for (C2EW) is usually not possible in practice, because usually the set $X_\Theta(T, t_0, X_0)$ is unknown.
Nevertheless the reader should be aware, that all the assumptions for the Approximation Property are sufficient, but in general not necessary. So (C2E) being violated for this problem is no death sentence for any kind of convergence, which the next section suggests.

The thing still missing is the Inverse Stability Property. It is the author's believe, that second order sufficient optimality conditions hold for this example. But this won't be discussed here. For an example on obtaining second order sufficient optimality conditions, see 6.2.2. However in this example, verifying second order optimality conditions involves dealing with the multiplicator $p(.)$, which corresponds to the ODE. This makes things more difficult.

6.4.3.3 convergence analysis

Numerical computations have been done using the NLPIP optimizer with exact derivatives. In this pretty demanding example for the optimizer, using exact derivatives has proven to be essential for obtaining useful results as well. Of course the liability of the optimizer for errors in the derivatives plays a great role here. The authors investigation so far and the experience of the developer of the NLPIP optimizer Björn Sachsenberger show, that the NLPIP optimizer has quite some weaknesses on that area and is outperformed by the NLPQLP optimizer, when using approximated derivatives. But NLPIP was able to handle the calculations for this example pretty well when using exact derivatives, while NLPQLP was not.
Like in the examples before the solutions for the number of steps $N = 2^k$ $(k = 2, \ldots, 9)$ have been computed. As already mentioned, the results are pretty similar to the ones in the convergence analysis of Problem 1 in 6.4.2.3. The major difference is that for simplicity reasons (symmetry), $d = 1/2$ has been set for Problem 2 instead of $d = 1/\sqrt{3}$ like in Problem 1. When using powers of 2 for the number of steps, this leads to the fact, that this time $t = d$ lies on the grid for every computation done. So there won't be no additional error from the peak this time. As the optimal solution is linear in the "detached phase" and otherwise corresponds to the one from Problem 1, pretty small error values and a pretty rough convergence analysis have to be expected. But all this analysis should show here is, that despite the fact of (C2E) being violated the solutions seem to converge. This may have to do with the tiny area on which (C2E) does not hold, i.e. where no "inward steering" can take place.

state convergence
As the following plots show, the optimal solution to the discrete problem is pretty close to the exact solution even for a small number of steps. One major reason for that is the linear structure, when the constraints are not active (in the sense of the optimal solution delivers $s_1(t, \hat{\hat{x}}(t)) = 0$ or $s_2(t, \hat{\hat{x}}(t)) = 0$). Again, the blue line represents the exact solution, the gray lines the state constraints, the green curve the linear interpolating spline, the red points the computed data and the orange line the maximum distance

(which is hardly visible here, because it is pretty small). Note the scaling of the axes.

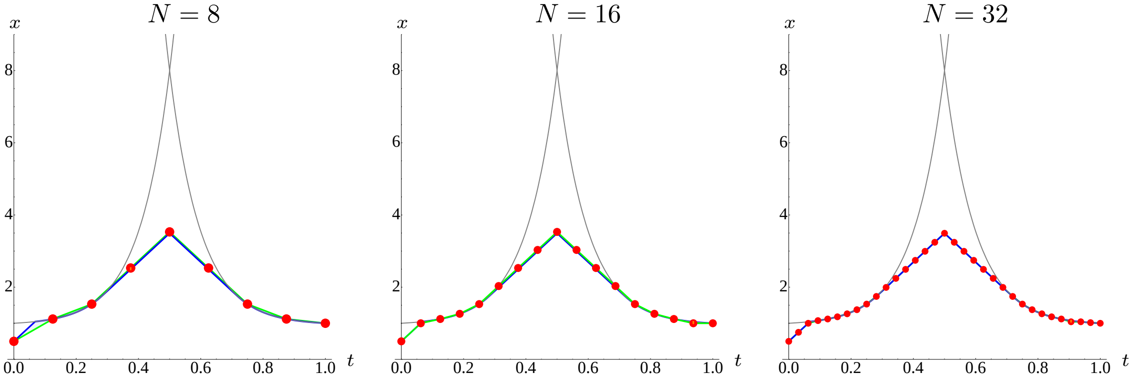


Figure 34: discrete optimal state (Example 6.4.3)

As one can see the discrete points pretty much lie on the exact curve. This can be observed for almost all other cases ($N = 2^k$ ($k = 3, \ldots, 9$)), too. The only exception is $N = 4$. Of course, for such a small number of steps a reasonable solution can't be expected. Instead the optimizer seems to deliver another local minimum:

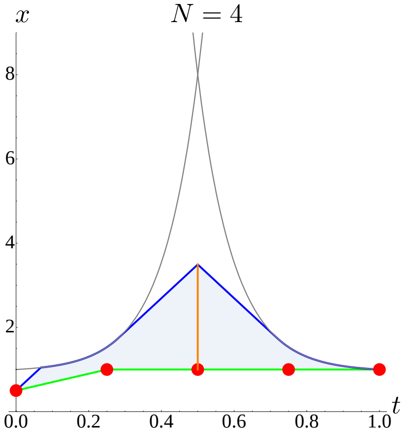


Figure 35: discrete optimal state (Example 6.4.3)

So the case N = 4 will be excluded from any further calculations. The results for the state with respect to the discrete supremum and the L_∞-norm (see 6.1.1) are the following.

steps	stepsize	$\|\hat{x}^N - \rho_N(\hat{x}(.))\|_\infty$	$\|\hat{x}^N(.) - \hat{x}(.)\|_\infty$
8	1/8	0.037841	0.208640
16	1/16	0.041229	0.041229
32	1/32	0.032544	0.032544
64	1/64	0.000054	0.025606
128	1/128	0.062446	0.062424
256	1/256	0.031196	0.031196
512	1/512	0.000025	0.000038

As one can see, there is no big difference between the discrete supremum and the L_∞-norm this time. This was expected, because the critical point $d = 1/2$ lies on the gird for all number of steps used. The values show that there is small tendency, which indicates convergence, present.

control convergence

As the optimal control jumps at $t = \tilde{t} \approx 0.0683179563$ and $t = d = 1/2$, it is very likely that the discrete controls won't converge in the discrete supremum. This is a fact well known in optimal control theory. Despite that, results are pretty close to the exact optimal control.

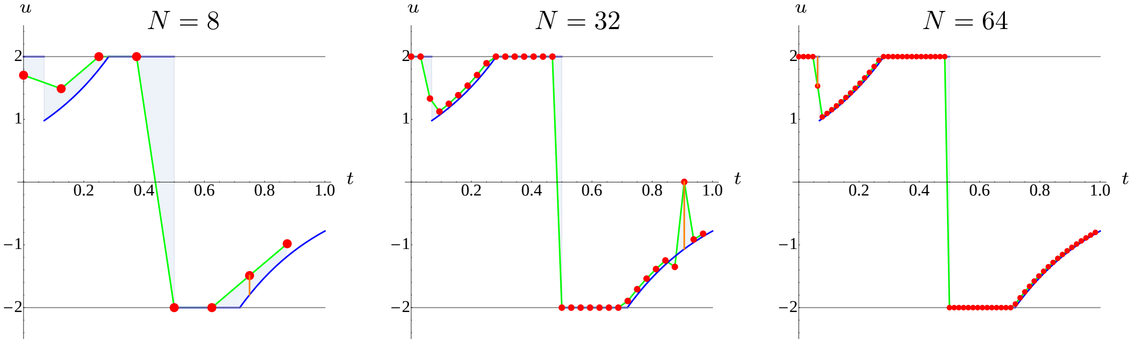


Figure 36: discrete optimal control (Example 6.4.3)

The discrete points sitting near the switching points and not lying on the exact curve of the control do in general not vanish when using greater number of steps.

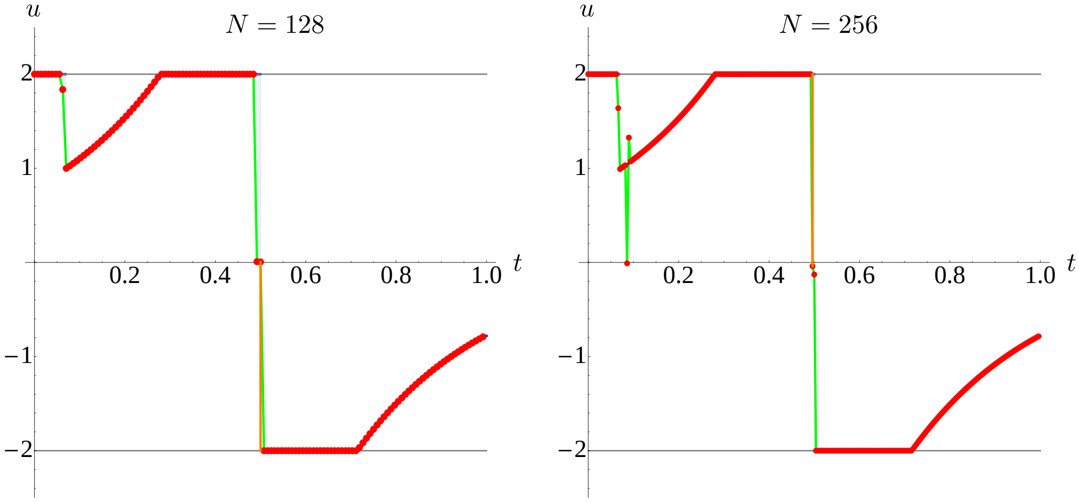


Figure 37: discrete optimal control for higher number of steps (Example 6.4.2)

So there will be definitely no convergence of the controls with respect to the discrete supremum norm. Nevertheless the distance values are included in the following chart. As in the other examples treated so far it contains the discrete L_2-norm and the L_2-norm for the interpolated case (for details see the control convergence section of 6.3.3).

steps	stepsize	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_\infty$	$\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$	$\|\hat{u}^N(.) - \hat{u}(.)\|_2$
8	1/8	0.306036	0.211394	0.629175
16	1/16	0.961777	0.320819	0.489191
32	1/32	1.070153	0.230116	0.378115
64	1/64	0.467540	0.063700	0.238663
128	1/128	2.005366	0.250524	0.208463
256	1/256	2.039708	0.187273	0.162490
512	1/512	0.012554	0.003240	0.084031

As one can see only the values for $\|\hat{u}^N(.) - \hat{u}(.)\|_2$ show a real tendency towards convergence. The values for $\|\hat{u}^N - \rho_N(\hat{u}(.))\|_2$ just a very slight one.

objective function value convergence

For comparison of the objective funciton values only the absolute value comes into play. No other norms need to be considered.

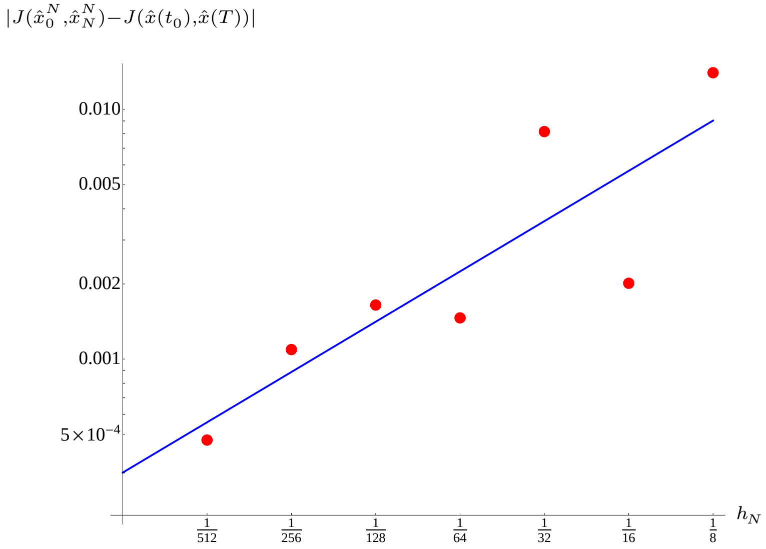


Figure 38: double logarithmic plot of $|J(\hat{x}_0^N, \hat{x}_N^N) - J(\hat{x}(t_0), \hat{x}(T))|$ for $N = 2^k$ $(k = 3, \ldots, 9)$ (Example 6.4.3)

The regressionline $r(.)$ appearing in this plot (blue) is:

$$-3.31580587 + 0.669307853\, t$$

So the estimated convergence rate of the objective function values is $p = 0.669307853 \approx 1/2$. This value is not equal or higher as 1, which has to be expected from the value convergence theorem. But nevertheless this might be an indicator, that the Approximation Property (see 3.2.1) and with it the Value Convergence Theorem 3.2.2 itself still do apply.

Summary
Despite the fact that (C2E) does not hold, we still got some indicators for properly convergent solutions.
The decisive values for this statement are:

steps	stepsize	$\|\hat{u}^N(.)-\hat{u}(.)\|_2$	$\|\hat{x}^N(.)-\hat{x}(.)\|_\infty$	$\lvert J(\hat{x}_0^N,\hat{x}_N^N)-J(\hat{x}(t_0),\hat{x}(T))\rvert$
8	1/8	0.629175	0.208640	0.014038
16	1/16	0.489191	0.041229	0.002012
32	1/32	0.378115	0.032544	0.008156
64	1/64	0.238663	0.025606	0.001462
128	1/128	0.208463	0.062424	0.001646
256	1/256	0.162490	0.031196	0.001092
512	1/512	0.084031	0.000038	0.000474

acknowledgement

Literature

[1] Frank Lempio, *DYNAMIC OPTIMIZATION*, lecture notes, university of Bayreuth

[2] R.Baier, I.Chahma and F. Lempio, *Stability and convergence of Euler's method for state-constrained differential inclusions*, SIAM J. OPTIM. Vol. 18 (2007), No. 3, pp. 1004-1026

[3] A. L. Dontchev and E. M. Farkhi, *Error estimates for discretized differential inclusions*, Computing, 41 (1989), pp. 349358

[4] Mattias Sandberg, *Convergence of the forward Euler method for nonconvex differential inclusions*, SIAM J. NUMER. ANAL. Vol. 47 (2008), No. 1, pp. 308-320

[5] Mattias Sandberg, *The forward Euler Scheme for nonconvex Lipschitz Differential Inclusions converges with rate one*, (2009)

[6] J.-P. Aubin and A. Cellina, *Differential Inclusions*, Grundlehren Math. Wiss. 264 (1984), Springer, Berlin

[7] C. D. Aliprantis, K. C. Border, *Infinite Dimensional Analysis*, Springer (2006), Berlin

[8] R. Webster, *Convexity*, Oxford Sci. Publ. (1994), Clarendon Press, Oxford University Press, New York

[9] S. Lenhart and J. T. Workman, *Optimal Control Applied to Biological Methods*, Mathematical and Computational Biology Series No. 15 (2007), Chapman&Hall/CRC, Boca Raton Florida

[10] B. Sachsenberg, *An sqp interior point algorithm for solving large scale nonlinear optimization problems*, Department of Computer Science University of Bayreuth (2008)

[11] K. Schittkowski, *NLPQLP: A Fortran Implementation of a Sequential Quadratic Programming Algorithm with Distributed and Non-Monotone Line Search*, Department of Computer Science University of Bayreuth (2007)

[12] J. Pannek, *Receding Horizon Control: A Suboptimality–based Approach*, University of Bayreuth (2009), www.nonlinearmpc.com

[13] A. L. Dontchev, *Error Estimates For a Discrete Approximation to Constrained Control Problems*, SIAM J. NUMER. ANAL. Vol. 18 No.3 (1981)

[14] Matthias Gerdts, *Optimal Control of Ordinary Differential Equations and Differential-Algebraic Equations*, Habilitation Thesis (2006), Department of Mathematics at the University of Bayreuth

I hereby declare that I wrote the thesis at hand by myself and with the sole help of the specified sources and auxiliaries. This thesis has never been presented to any other examination board in neither the present nor any similar form.