TECHNISCHE
UNIVERSITÄT
WIEN
Vienna | Austria

DISSERTATION

Formation Control of Multi-Agent-Systems based on Continuum Models

Ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften (Dr.techn.)

unter der Leitung von
Prof. Dr.-Ing. habil. Thomas Meurer
E376
Institut für Automatisierungs- und Regelungstechnik

eingereicht an der
Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von

Dipl.-Ing. Gerhard Freudenthaler, BSc
Matrikelnummer: 0426049
[redacted]

Wien, im April 2021

In Kooperation mit dem
Lehrstuhl für Regelungstechnik
der
Christian-Albrechts-Universität zu Kiel

C|A|U

Christian-Albrechts-Universität zu Kiel
Technische Fakultät

Betreuer:	Prof. Dr.-Ing. habil. Thomas Meurer
Erster Gutachter:	Prof. Dr. Christophe Prieur
Zweiter Gutachter:	Univ.-Prof. Dr.techn. Andreas Kugi

Für Bea und Jakob

Vorwort

Das vorliegende Manuskript verfasste ich weitestgehend in den Jahren 2018 bis 2020 nach meinen Forschungstätigkeiten am Institut für Elektrotechnik und Informationstechnik der Christian-Albrechts-Universität zu Kiel. Die dort durchgeführten Untersuchungen wurden u.a. von der Deutschen Forschungsgemeinschaft (DFG) im Rahmen des Projekts ME3231/2-1 gefördert.

Mein besonderer Dank gilt Herrn Prof. Dr.–Ing. habil. Thomas Meurer sowohl für die hervorragende und engagierte Betreuung bei der Durchführung der Forschungstätigkeiten an seinem Lehrstuhl für Regelungstechnik in Kiel, als auch für die fachliche Unterstützung beim Verfassen dieser Arbeit. Ihm gebührt auch mein herzlicher Dank für die kontinuierliche und langjährige Förderung meiner akademischen Bildung, für seine motivierenden Impulse in schwierigen Phasen, sowie für seine Offenheit und seinen Elan für eine differenzierte Sichtweise bei neuen Fachthemen. Sehr gefreut habe ich mich über die Mitwirkung von Herrn Prof. Dr. Christophe Prieur, als Erstgutachter. Ihm danke ich für sein Interesse, die schnelle Durchsicht der Abhandlung sowie seine konstruktiven Hinweise und Anmerkungen. Im Zuge dessen möchte ich mich auch bei Herrn Univ.-Prof. Dr.techn. Andreas Kugi, Vorstand des Instituts für Automatisierungs- und Regelungstechnik und Professor für komplexe dynamische Systeme der Technischen Universität Wien, bedanken. Herr Prof. Kugi ist Zweitgutachter dieser Doktorarbeit und er hat mich seit meinem Masterstudium bei meiner akademischen Ausbildung stets gefördert und begleitet. Auch Ihm gebürt die Anerkennung für seinen fachlichen Kommentar zu dieser Arbeit und für seine administrativen und professionellen Hilfestellungen während meiner akademischen Tätigkeiten an der Technischischen Universität Wien.

Außerdem bedanke ich mich bei den Kolleginnen und Kollegen am Institut in Kiel für das gute Arbeitsklima und die stete Hilfsbereitschaft. Besonders danke ich Henning Weißbarth, Nico Engel, Kai Hilmer und Sönke Reese, die durch ihre Tätigkeiten am Institut bzw. durch Studien– und Diplomarbeiten wertvolle Beiträge zu der vorliegenden Thematik und beim Aufbau des Versuchsfeldes geleistet haben. Darüber hinaus danke ich Max Lutz und Pascal Jerono für die sehr gute Zusammenarbeit in unserem Büro.

Weiteres, möchte ich meiner Lebensgefährtin und Verlobten Bea für ihre Ausdauer und ihr

Verständnis meinen ganz besonderen Dank aussprechen. Sie war vor allem in den schwierigen Zeiten stets mein persönlicher Rückhalt und meine emotionale Stütze. Ich kann nicht genug betonen, wie viel Energie sie mir während der letzten sehr prägenden Jahre gegeben hat. Schließlich, aber nicht zuletzt, danke ich meinen Eltern für ihre Unterstützung während dieser Zeit.

Ravensburg, April 2021 Gerhard Freudenthaler

Contents

List of Symbols

Coordinates and Units

$\mathtt{e}_{ij}$ Edge between the vertices $\mathtt{v}_i$ and $\mathtt{v}_j$

$\imath$ Imaginary unit $\imath^2 = -1$

i Index for $i-$th vertex or the $i-$th associated agent

s Second spatial coordinate used for integral kernels

t Time coordinate

$\mathtt{v}_i$ Vertex of a graph with integer index i

z Spatial coordinate

Dimensions and Cardinalities

$d\left(\mathtt{v}_i\right)$ Cardinality of the neighbourhood of the vertex $\mathtt{v}_i$, or the degree of the vertex $\mathtt{v}_i$

E Cardinality of the edges of a graph

N Cardinality of the vertices of a graph and number of agents

ℓ Arbitrary length of the spatial domain

ℓ_{N} Length of the spatial domain linked to the number of agents according to $\ell_{\mathrm{N}} = \mathrm{N} - 1$

n Dimension of the state (vector) $\boldsymbol{x}(z,t)$

Sets

$\mathbb{N}$ Set of natural numbers

$\mathbb{N}_0$ Set of natural numbers including 0

$\Omega\left(\bar{\ell}_{\mathrm{N}}, \bar{t}_0\right)$ Closed spatial and closed temporal domain $[0, \ell_{\mathrm{N}}] \times \overline{\mathbb{R}}^{+}_{t_0}$

$\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right)$ Closed spatial and open temporal domain $[0, \ell_{\mathrm{N}}] \times \mathbb{R}^{+}_{t_0}$

$\Omega(\ell_{\mathrm{N}}, t_0)$ Open spatial and open temporal domain $(0, \ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0}$

$\mathbb{R}$ Set of real numbers

$\mathbb{R}^+_{t_0}$ Set of real numbers $t \in \{\mathbb{R} \,|\, t > t_0\}$

$\overline{\mathbb{R}}^+_{t_0}$ Set of real numbers $t \in \{\mathbb{R} \,|\, t \geq t_0\}$

$\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})$ Triangular spatial domain of the backstepping kernel for control design

$\mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}})$ Triangular spatial domain of the backstepping kernel for observer design

$\mathtt{Nbr}(i)$ Neighbourhood of the vertex $\mathtt{v}_i$

$\mathtt{E}$ Set of edges of a graph

$\mathtt{V}$ Set of vertices of a graph

Variables

$\boldsymbol{y}(t)$ Trajectory of the flat output (vector) $\boldsymbol{y}$ with $t \in \mathbb{R}^+_{t_0}$

$\boldsymbol{y}^*(t)$ Desired trajectory of the flat output (vector) $\boldsymbol{y}$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\Delta u_{\ell_{\mathrm{N}}}(t)$ (Scalar) Feedback control input at $z = \ell_{\mathrm{N}}$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\Delta u_0(t)$ (Scalar) Feedback control input at $z = 0$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\Delta \boldsymbol{u}_{\ell_{\mathrm{N}}}(t)$ Feedback control input (vector) at $z = \ell_{\mathrm{N}}$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\Delta \boldsymbol{u}_0(t)$ Feedback control input (vector) at $z = 0$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$u^*_{\ell_{\mathrm{N}}}(t)$ (Scalar) Feedforward control input at $z = \ell_{\mathrm{N}}$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$u^*_0(t)$ (Scalar) Feedforward control input at $z = 0$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\boldsymbol{u}^*_{\ell_{\mathrm{N}}}(t)$ Feedforward control input (vector) at $z = \ell_{\mathrm{N}}$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\boldsymbol{u}^*_0(t)$ Feedforward control input (vector) at $z = 0$ with $t \in \overline{\mathbb{R}}^+_{t_0}$

$\boldsymbol{u}_1(t)$ Input (vector) with $\mathtt{v}_1$ and $t \in \mathbb{R}^+_{t_0}$

$\boldsymbol{u}_{\mathrm{N}}(t)$ Input (vector) with $\mathtt{v}_{\mathrm{N}}$ and $t \in \mathbb{R}^+_{t_0}$

$x_i(t)$ State associated to agent i and $t \in \mathbb{R}^+_{t_0}$

$\boldsymbol{x}(\mathtt{v}_i, t)$ State (vector) with $(\mathtt{v}_i, t) \in \mathtt{V} \times \mathbb{R}^+_{t_0}$

$\boldsymbol{x}^*(\mathtt{v}_i, t)$ Desired state (vector) with $(\mathtt{v}_i, t) \in \mathtt{V} \times \mathbb{R}^+_{t_0}$

$\hat{\boldsymbol{x}}(\mathtt{v}_i, t)$ Observer state (vector) with $(\mathtt{v}_i, t) \in \mathtt{V} \times \mathbb{R}^+_{t_0}$

$u_{\ell_{\mathrm{N}}}(t)$ (Scalar) Input at $z = \ell_{\mathrm{N}}$ with $t \in \mathbb{R}^+_{t_0}$

$u_0(t)$ (Scalar) Input at $z = 0$ with $t \in \mathbb{R}_{t_0}^{+}$

$\boldsymbol{u}_{\ell_\mathrm{N}}(t)$ Input (vector) at $z = \ell_\mathrm{N}$ with $t \in \mathbb{R}_{t_0}^{+}$

$\boldsymbol{u}_0(t)$ Input (vector) at $z = 0$with $t \in \mathbb{R}_{t_0}^{+}$

$\boldsymbol{v}(z,t)$ Target state (vector) for observer design with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\boldsymbol{w}(z,t)$ Target state (vector) for control design with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\bar{x}(z)$ (Scalar) Steady state with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\boldsymbol{x}(z,t)$ State (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\bar{\boldsymbol{x}}(z)$ Steady state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\bar{\boldsymbol{x}}_\mathtt{a}$ Configured steady state (vector) at $z = 0$ with $t \geq \bar{t}$ and $\bar{t} \in \mathbb{R}_{t_0}^{+}$

$\bar{\boldsymbol{x}}_\mathtt{1}$ Configured steady state (vector) at $z = \ell_\mathrm{N}$ with $t \geq \bar{t}$ and $\bar{t} \in \mathbb{R}_{t_0}^{+}$

$\boldsymbol{x}^{*}(z,t)$ Desired state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\tilde{\boldsymbol{x}}_\mathrm{c}(z,t)$ Tracking error state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\tilde{\boldsymbol{x}}_\mathrm{o}(z,t)$ Observer error state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\hat{\boldsymbol{x}}(z,t)$ Observer state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\boldsymbol{x}_\mathrm{e}(z,t)$ State (vector) of an *Exogenous System* with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

$\boldsymbol{x}_\mathrm{s}(z,t)$ Shifted state (vector) with $(z,t) \in \Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right)$

Vectors, Matrices and Operators

$A(\mathcal{G})$ Adjacency matrix of the graph $\mathcal{G}$

$\Delta(\mathcal{G})$ Degree matrix of the graph $\mathcal{G}$

$D(\mathcal{G}^o)$ Incidence matrix of the arbitrarily oriented graph $\mathcal{G}^o$

$L(\mathcal{G})$ Graph Laplacian matrix of the graph $\mathcal{G}$

$L_w(\mathcal{G}, t)$ Weighted graph Laplacian matrix of the graph $\mathcal{G}$

$M(z,t)$ Compensation matrix for control design with $(z,t) \in \Omega(\ell_\mathrm{N}, t_0)$

$N(z,t)$ Compensation matrix for observer design with $(z,t) \in \Omega(\ell_\mathrm{N}, t_0)$

$W(\mathtt{E}, t)$ Diagonal weight matrix of the edge set $\mathtt{E}$

$0_{n,n}$ Zero matrix of dimension $n \times n$

$\mathcal{O}$ Error order in terms of the approximation a of smooth function

$\frac{\mathrm{d}}{\mathrm{d}z}$ Total derivative with respect to z for functions with $2-$dimensional spatial domains

$\mathcal{L}(\boldsymbol{x})$ (Graph) Laplacian control applied to the state (vector) $\boldsymbol{x}$

$\mathcal{L}_w(\boldsymbol{x})$ Weighted (Graph) Laplacian control applied to the state (vector) $\boldsymbol{x}$

$\boldsymbol{\nabla}$ Differential operator on the $m-$dimensional Euclidean space

A Parameter matrix of the diffusion term

$B(z,t)$ Parameter matrix of the convection term with $(z,t) \in \Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right)$

$C(z,t)$ Parameter matrix of the reaction term with $(z,t) \in \Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right)$

$D(t)$ Diagonal tuning matrix for control design with $t \in \mathbb{R}^+_{t_0}$

$E(t)$ Diagonal tuning matrix for oberver design with $t \in \mathbb{R}^+_{t_0}$

M^{a} Anti-symmetric part of the matrix M

M^{c} The off-diagonal part (coupling part) of the matrix M

M^{d} The (main) diagonal part of the matrix M

M^{s} Symmetric part of the matrix M

∂_t^β Temporal (partial) derivative of order β

∂_z^α Spatial (partial) derivative of order α

$\mathbf{0}_n$ Zero vector of dimension n

Special Graphs

$\mathcal{C}_{\mathrm{H}}$ Hamiltonian Cycle

$\mathcal{C}_{\mathrm{N}}$ N$-$Cycle Graph

$\mathcal{F}_{\mathrm{N}}$ Forest Graph with N vertices

$\mathcal{G}_{\mathrm{H}}$ Hamiltonian Graph with $|\mathtt{V}|$ vertices

$\mathcal{G}_t$ Traceable graph or multigraph

$\mathcal{G}$ A finite, directed or undirected, simple graph or multigraph

$\mathcal{K}_{\mathrm{N}}$ Complete Graph with N vertices

$\mathcal{P}_{\mathrm{H}}$ Hamiltonian Path with $|\mathtt{V}|$ vertices

$\mathcal{P}_{\mathrm{N}}$ Path Graph with N vertices

$\mathcal{S}_{\mathrm{N}}$ Star Graph with N vertices

Function spaces and Classes

X Subset of $\mathbb{R}$

$\mathbf{CG}^{k,\alpha}(\Omega \times \Lambda; X)$ Class of functions in two variables from $\Omega \times \Lambda$ to X which are $k-$times continuously differentiable in the first and are of Gevrey class of order α on domain Λ in the second variable

$\mathbf{CG}^{k,l,\alpha}(\Omega_1 \times \Omega_2 \times \Lambda; X)$ Class of functions in three variables from $\Omega_1 \times \Omega_2 \times \Lambda$ to X which are $k-$times continuously differentiable in the first, $l-$times continuously differentiable in the second, and are of Gevrey class of order α on domain Λ in the third variable

$\mathbf{C}^{k,l}(\Omega; X)$ Class of functions in two variables from Ω to X which are $k-$times continuously differentiable in the first and $l-$times continuously differentiable in the second variable

$\mathbf{C}^{k}(\Omega; X)$ Class of $k-$times continuously differentiable functions from Ω to X

$\mathbf{G}^{\delta}_{\alpha}(\Lambda; X)$ Gevrey class of order α on domain Λ

Integral Kernels

$G(z,s,t)$ Inverse backstepping kernel for control design with $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$K(z,s,t)$ Backstepping kernel for control design with $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$L(z,s,t)$ Backstepping kernel for observer design with $(z,s,t) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$S(z,s,t)$ Inverse backstepping kernel for observer design with $(z,s,t) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$H(z,s,t)$ Compensation kernel for the kernel PDE of K with $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$J(z,s,t)$ Compensation kernel for the kernel PDE of G with $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$P(z,s,t)$ Compensation kernel for the kernel PDE of L with $(z,s,t) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

$Q(z,s,t)$ Compensation kernel for the kernel PDE of S with $(z,s,t) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}^{+}_{t_0}$

Abbreviations

2DOF two degrees of freedom

3DOF three degrees of freedom

APPL application

ArUco Augmented Reality, University of Córdoba

ArUco Code Augmented Reality Code, University of Córdoba

ASW application software

AUTOSAR Automotive Open System Architecture

BC boundary condition

BE Burgers' equation

BSW basic software

cf. confer

CPU central processing unit

DAPP demo application

DCRS diffusion-convection-reaction system

DCS diffusion-convection system

DPS distributed-parameter system

DRS diffusion-reaction system

DS diffusion system

FB feedback

FBC feedback control

FDM finite difference method

FEM finite element method

FF feedforward

FFC feedforward control

FVM finite volume method

IC initial condition

KF Kalman filter

LTI linear, time-invariant

LTV linear, time-variant

MA multi-agent

MAS multi-agent system

MCU microcontroller unit

MTX matrix

MVBE modified, viscous Burgers' equation

NL non-linear

ODE ordinary differential equation

PDE partial differential equation

PIDE partial integro-differential equation

RTE runtime environment

s.t. such that

TASK task

TCC transversal compensation condition

TI time-invariant

TV time-variant

Abstract

The dynamics of swarms or networks is conventionally modelled using Graph-theoretical concepts. In this regard the $n-$dimensional state dynamics of each agent is described by a set of n ordinary differential equations (ODEs). Consequently, assuming the network consists of N agents, this directly leads to a nN$-$dimensional overall dynamic system where the dimension alters with the number of agents. In other words, describing swarm dynamics by ODEs, especially with a huge number of participants, may lead to unmanageable system representations and unclear properties. Now, it can easily be shown that there is, e.g., a structural equivalence between the fundamental consensus protocol of a multi-agent system (MAS) and the semi-discretised heat equation. In particular this is valid if, on the one hand, the topology of the network is represented by a so-called path graph, and on the other hand, the partial differential equation (PDE) is defined on an 1-dimensional spatial domain. Now, the present work generally deals with the formation control of MASs where the underlying dynamic flocking model is based on continuous problem formulations. In this context descriptions of distributed-parameter system (DPS) governed by the PDEs of the coupled diffusion-convection-reaction system (DCRS) or the coupled modified, viscous Burgers' equation (MVBE) are used to model the overall system dynamics.

The first part of the thesis addresses a theoretical discussion about the continuous modelling methodology. In this context, it systematically works out necessary steps which allow the transformation of the swarm model from a parabolic PDE to a discrete representation described by graph-theoretical considerations, and vice versa. From this, it is important to point out that the introduction of a variable length for the spatial domain of the PDE system plays a crucial role for both, the overall flocking dynamics as well as the single agent dynamics. Furthermore, it is shown that the length of the real-valued domain can directly be linked to the discrete number of agents forming the MAS.

The second part of this work develops a two degrees of freedom (2DOF) control synthesis consisting of a feedforward (FF) term and an observer-based tracking controller. Both are utilised based on the continuous formulation and the controller synthesis uses fundamental ideas in terms of controller design but the presented work extends and re-thinks them especially for coupled PDEs. While the FF design and the linked motion planning process take advantage of the flatness properties of parabolic PDEs, the error feedback controller and the associated state observer design are based on the so-called *Backstepping* approach. However, in this work the basic backstepping methods are extended in the manner that formulations established for PDEs of uncoupled DPS may still apply for PDEs of coupled systems. Subsequently an appropriate 2-step transformation process, including discretisation methods such as the finite difference method (FDM), is introduced which ensures that the protocols and control algorithms can be employed on discrete systems with Graph-theoretical background.

For this, the last part shows results from various and extensive simulation studies. They are conducted, first, to verify that the developed modelling approach is really applicable for discrete dynamic network systems, and second, the simulations results enhance the presented 2DOF controller design techniques for coupled PDE. Apart from that, the final sections of the thesis describe the set-up and the implementation of a test bed. This was installed from scratch in terms of hardware as well as in software during the author's research activities at the Chair of Automatic Control, Kiel University. The test rig allows to perform real-time experiments with a small swarm of robots. Furthermore, in view of the presented modelling approach it can serve as simulator for experiments with DPSs. Conducted tests with a swarm of 11 small caterpillar robots support and validate the claims of the presented theoretical achievements. Both, the simulation studies and the experimental results, show that the formulated goals are achieved and substantiate the robustness of the derived control architecture.

Kurzzusammenfassung

Die Dynamik von Schwärmen oder Netzwerke wird meist mithilfe von graphentheoretischen Konzepten modelliert. In diesem Zusammenhang wird die Zustandsdynamik eines jeden einzelnen Agenten durch ein System von n gewöhnlichen Differentialgleichungen beschrieben. Angenommen das Netzwerk besteht aus N Agenten, dann folgt daraus ein nN$-$dimensionales dynamisches Gesamtsystem, wobei sich die Dimension mit der Anzahl der Agenten ändert. Mit anderen Worten, diese Art der Beschreibung von Netzwerken kann, speziell für eine große Anzahl von Agenten, sehr schnell zu unhandlichen Systemrepräsentationen führen deren Eigenschaften nur sehr aufwendig überprüfbar sind. Jedoch kann sehr einfach gezeigt werden, dass eine strukturelle Äquivalenz zwischen dem fundamentalen Konsensprotokoll eines Multiagenten-Systems und der semi-diskretisierten Wärmeleitungsgleichung besteht, wenn einerseits die Topologie des Netzwerks durch einen Pfad-Graph repräsentiert werden kann und andererseits die partielle Differentialgleichung auf einer eindimensionalen örtlichen Domäne definiert ist. Die vorliegende Arbeit beschäftigt sich mit der Formationsregelung von Multiagenten-Systemen, wobei das zugrundeliegende dynamische Schwarmmodell auf einer kontinuierlichen Formulierung basiert. In diesem Zusammenhang werden Beschreibungen von verteilt-parametrischen Systemen, im speziellen die von gekoppelten Diffusion-Konvektion-Reaktions-Systemen oder die der gekoppelten modifizierten, viskosen Burgers-Gleichung, verwendet um die Dynamik des Gesamtsystems zu modellieren.

Der erste Teil dieser Doktorarbeit thematisiert eine theoretische Auseinandersetzung mit Bezug auf die kontinuierliche Modellierungsmethodik. In diesem Kontext werden systematisch notwendige Schritte ausgearbeitet, die die Hin- und Rücktransformation des Schwarmmodells zwischen einer parabolischen partiellen Differentialgleichung und einer diskreten Repräsentation basierend auf graphentheoretischen Aspekten ermöglicht. Dabei ist es wichtig hervorzuheben, dass die Einführung einer variablen Länge der örtlichen Domäne der partiellen Differentialgleichung eine wichtige Rolle, sowohl für die übergeordnete Schwarmdynamik als auch die (Eigen-)Dynamik der einzelnen Agenten, spielt. Weiteres wird gezeigt, dass man die variable Länge der reellwertigen Domäne direkt mit der Anzahl der Agenten, die gemeinsam das Multiagenten-System bilden, verknüpfen kann.

Im zweiten Teil der Arbeit geht es um die Synthese einer 2-Freiheitsgrade-Regelung, bestehend aus einer Vorsteuerung und einer Beobachter-basierten Folgeregelung. Beide Regelungsanteile nutzen Konzepte basierend auf der kontinuierlichen Formulierung und verwenden fundamentale Strategien in Hinblick auf den Regelungsentwurf. Jedoch werden in dieser Arbeit diese Themen im Speziellen für gekoppelte partielle Differentialgleichungen erweitert bzw. neu überdacht. Während sich der Entwurfsprozess der Vorsteuerung und die dazugehörende Bewegungsplanung die Eigenschaften der Flachheit zu Nutze macht, basiert die Fehlerregelung und der damit verknüpfte Entwurf des Zustandsbeobachters auf der sogenannten Backstepping-Methodik. Es werden die grundlegenden Backstepping-Techniken in dem Sinne erweitert, sodass Formulierungen, die für partielle Differentialgleichungen nicht gekoppelter verteilt-parametrischer Systeme etabliert sind, auch auf Gleichungen gekoppelter Systeme angewendet werden können. In weiterer Folge wird ein entsprechender 2-Schritt-Prozess eingeführt, der unter anderem Diskretisierungsverfahren wie die Finite Differenzen Methode beinhaltet, und damit sicherstellt, dass die Protokolle und Regelalgorithmen auf die diskreten Systeme mit graphentheoretischen Hintergrund angewendet werden können.

Der letzte Teil zeigt die Resultate verschiedener ausgiebiger Simulationsstudien. Diese wurden durchgeführt um unter anderem die Anwendbarkeit des entwickelten Modellierungsansatzes auf diskrete Netzwerksysteme zu verifizieren. Weiteres unterstützen die Simulationsresultate die präsentierten Techniken für den Entwurf der 2-Freiheitsgrade-Regelung gekoppelter verteilt-parametrischer Systeme. Außerdem beschreiben die finalen Abschnitte der Arbeit den Aufbau und die Implementierung eines Versuchsfeldes. Dieses wurde im Rahmen der Forschungstätigkeiten am Lehrstuhl für Regelungstechnik an der Christian-Albrechts-Universität von Grund auf, sowohl in Hardware als auch in Software, installiert. Das Testfeld ermöglicht die Ausführung von Echtzeit-Experimenten mit Roboterschwärmen. In Hinblick auf die vorgestellte Modellierungsmethodik kann es als Simulator für Experimente mit verteilt-parametrischen Systemen dienen. Die mit einem Schwarm bestehend aus 11 Miniatur-Raupenrobotern durchgeführten Tests untermauern und validieren die Aussagen der präsentierten theoretischen Ergebnisse. Beide Praktiken, sowohl die Simulationsstudien und als auch die experimentellen Tests, zeigen, dass die formulierten Ziele erreicht werden und stellen die Robustheit der abgeleiteten Reglerarchitektur unter Beweis.

Chapter 1

Introduction

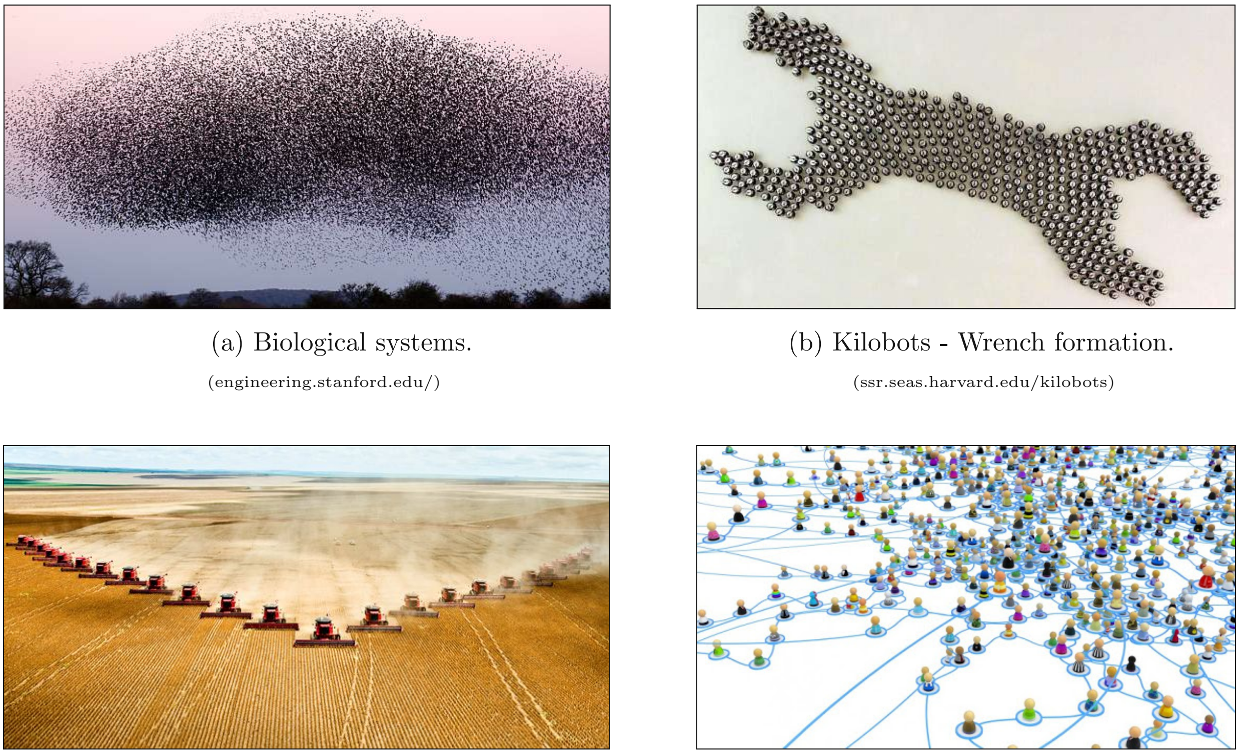
This work addresses the two major research areas *multi-agent systems (MASs)* and so-called *distributed-parameter systems (DPSs)*. More precisely, it combines the continuous modelling description and control theory based on *partial differential equations (PDEs)* with the inherently discrete aspects of swarm and flocking theory. For this, the chapter moves forward step by step carefully, starting with preliminary discussions on some nomenclature, modelling concepts, and the goals and objectives of this work.

1.1 Swarms, Flocks, and Multi-Agent Systems

Generally, no matter what term is used, i.e., either swarms, flocks, networks, or MASs, they have something some common features. All consist of interconnected dynamic subsystems which share information. Therefore the expressions may be used synonymously in certain idioms. However, MASs are characterised by the following statements [68, 93].

- *Multi* means many – therefore a system, consisting of many participants;
- the participants are called *Agents*;
- each agent has particular properties and is autonomous;
- between the agents a connection, communication, or another kind of exchange *may* exist;
- the agents can solve tasks in a collective manner;
- the agents have only a local focus;
- one main goal is decentralisation – there exists no central instance.

During the exertion of the collective task the individual agents can occupy different roles [46]. Roughly speaking they can be called



(a) Biological systems.

(engineering.stanford.edu/)

(b) Kilobots - Wrench formation.

(ssr.seas.harvard.edu/kilobots)

(c) Technical systems.

(www.pinterest.co.uk)

(d) Social network.

(www.cloud467.com)

Figure 1.1: Some hands-on examples of MASs.

- *passive agents*, if they have no real goal or task;
- *active agents*, if they peruse elementary goals or perform simple algorithms;
- *cognitive agents*, if they solve complex problems.

This elementary characteristics can be found in various areas in nature as well as in the human being's daily life of the 21st century. A few examples are pictured in Figure 1.1. As for many other technical aspects *biology* serves here as a role model for robotics or other technical systems and opens up a wide field of applications. Among others, the most common are consensus and synchronisation problems, decision making, crowd dynamics, formation control, cooperative multi-vehicle control, machine learning or complex oscillators networks [68, 67, 65, 53, 19, 12, 20, 11].

1.2 Modelling Approaches for Multi-Agent Systems

In literature many concepts and approaches are documented how MASs can be modelled or described in a formal language, e.g., in robotics potential theory is a common concept to coordinate groups [51]. However, instead of going through individual modelling approaches subsequently a more abstract point of view is taken, which allows to formulate three different modelling concepts.

(i). Behaviour models:
In 1987 a paper which deals with artificial life simulations was published in the proceedings of a conference for Computer Graphics by Craig Reynolds [73]. He named the application *Boids* referring to a bird-like object and it was fundamental for the simulation of flocking behaviour. The *Boids* in their simplest version follow three fundamental rules:

- *cohesion*: move towards the mean position of your local neighbours
- *separation*: avoid crowds, keep distance from your local neighbours
- *alignment*: head along the average direction of your local neighbours.

The program was successfully used in many practical applications reaching from movies and computer games to information visualisation and task optimisation [63, 5, 16, 78]. In [7] behaviour-based formation control enables a team of multi-robots to reach navigational goals.

(ii). Discrete models:
This modelling approach is most common and widespread concept to abstract the flocking behaviour. One of the main reasons for the broad acceptance is that these models are inherently supported by Graph theory, see a.o., [66, 67, 53]. As an example, let us assume that an agent can identified by a so-called *vertex*, which shall be abbreviated by $\mathtt{v}_i$. Moreover, each agent i shall be assigned with a state x for each coordinate. Assume each state may evolve with time, then this leads to the state vector $\boldsymbol{x}^m(t) = [x^m(\mathtt{v}_1, t),\, x^m(\mathtt{v}_2, t),\, \cdots\, x^m(\mathtt{v}_\mathrm{N}, t)]^T \in \mathbb{R}^\mathrm{N}$ where m stands for the mth coordinate. Considering n coordinates, i.e., $m = \{1, \ldots, n\}$, then the $n\mathrm{N}-$dimensional state can be written as $\boldsymbol{x}(t) = [\boldsymbol{x}^1(t),\, \boldsymbol{x}^2(t),\, \cdots\, \boldsymbol{x}^n(t)]^T$ and its autonomous, linear, time-variant (LTV) swarm dynamics is represented by

$$\frac{\mathrm{d}}{\mathrm{dt}}\begin{bmatrix}\boldsymbol{x}^1(t)\\ \boldsymbol{x}^2(t)\\ \vdots\\ \boldsymbol{x}^n(t)\end{bmatrix} = \begin{bmatrix} M^{11}(t) & M^{12}(t) & \cdots & M^{1n}(t)\\ M^{21}(t) & M^{22}(t) & \cdots & M^{2n}(t)\\ \vdots & \vdots & \ddots & \vdots\\ M^{n1}(t) & M^{n2}(t) & \cdots & M^{nn}(t)\end{bmatrix}\begin{bmatrix}\boldsymbol{x}^1(t)\\ \boldsymbol{x}^2(t)\\ \vdots\\ \boldsymbol{x}^n(t)\end{bmatrix},$$

for $t > t_0$ and with the initial condition (IC) $\boldsymbol{x}(t_0) = \boldsymbol{x}_0 \in \mathbb{R}^{(n\mathrm{N})}$. If $M^{jk}(t) = 0_{n,n}$ for $j \neq k$ the system is called uncoupled. In this case the swarm dynamics of each coordinate m is independent of each other and can be treated separately. In all others cases the swarm dynamics is called a coupled system. From a Graph theoretical point of view the sub-matrices $M^{jk}(t)$ are defined by the set of edges which defines the topology of the connection between the agents.

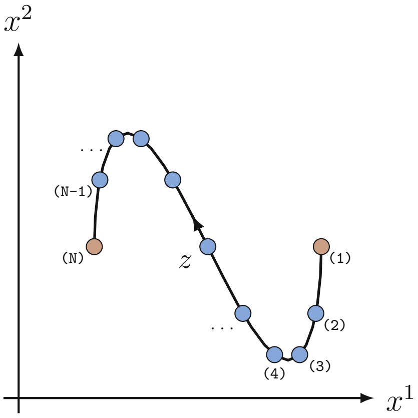


Figure 1.2: Formation profile of 11 agents in the (x^1, x^2)-plane; ● refer to the leader agents, ● denote the followers.

(iii). Continuous models

An alternative formalism to discrete models are mathematical descriptions of MASs in terms of DPSs governed by PDEs. Over the last years this approach gathers the interest and attention of various research groups and is topic in many publications, a.o., [21, 29, 30, 59, 55, 71, 69, 23, 13, 25]. The fundamental idea is schematically illustrated in Figure 1.2 where 11 agents in a so-called leader-follower configuration are deployed in the $\{x^1, x^2\}$ −plane. There are two types of agents and both refer to active and collaborative roles. However, leader agents may have to fulfil more sophisticated tasks than follower agents. By moving from the discrete set of agents denoted by coloured dots to an agent continuum the formation is visualized by the black line. Thereby, the spatial coordinate z is introduced and may being interpreted as a virtual communication path. With this, let us assume the swarm dynamics can be modelled by a so-called diffusion-reaction system (DRS) system with two states which, can be written as

$$\begin{aligned}
&\partial_t \boldsymbol{x}(z,t) = A\partial_z^2 \boldsymbol{x}(z,t) + C(z,t)\boldsymbol{x}(z,t)\,, \quad z \in (0,\ell)\\
&\boldsymbol{x}(0,t) = \mathbf{0}_n\,,\\
&\boldsymbol{x}(\ell,t) = \mathbf{0}_n\,,
\end{aligned}$$

for $t > t_0$ and the IC $\boldsymbol{x}(z,t_0) = \boldsymbol{x}_0(z) \in \mathbb{R}^2$. Here the first equation, representing the system dynamics, consists of a diffusion part with constant diagonal matrix A and a reaction part with the parameter matrix C which may evolve with space z and time t. The latter two equations impose Dirichlet boundary conditions (BCs). The formation for the agent continuum is then obtained by the superposition of the solution $\boldsymbol{x}(z,t) = [x^1(z,t), x^2(z,t)]^T$ of the PDE in each coordinate 1 and 2, respectively. Now, with an appropriate procedure it is possible to transform the $z-$distributed state $\boldsymbol{x}(z,t)$ into a set of discrete states $\{\boldsymbol{x}(\mathrm{v}_1,t), \boldsymbol{x}(\mathrm{v}_2,t), \ldots, \boldsymbol{x}(\mathrm{v}_\mathrm{N},t)\}$ attached to the vertices v_i with

$i = \{1, 2 \ldots, \mathrm{N}\}$, or formally written as

$$\boldsymbol{x}(z,t) \quad \xRightarrow{\texttt{Transformation}} \quad \boldsymbol{x}(\mathrm{v}_i, t)\,.$$

In general, the state space or the solution space for a PDEs are vector spaces of functions, e.g. Sobolev spaces, which are infinite-dimensional. In contrast to that ordinary differential equations (ODEs) are generally finite dimensional. Consequently, when applying the so-called *Early-Lumping* approach to a PDE the system is approximated by a high-dimensional ODE [58]. However when dealing with swarms this is the other way round. Dealing with swarms the continuous models may be interpreted as an approximation for the discrete modelling approach. It is fundamental for the further understanding that the entire motivation and way of proceeding is inverted for the work with swarm dynamics. Here the basis is the dynamic model of a MAS, which consists of a discrete number of participants, and it is approximated by a (set of) PDE(s) which refer(s) to an infinite-dimensional system representation. The consequences of this characteristic is worked out step by step in the upcoming chapters.

1.3 Goals, Objectives and Outline of the Thesis

As discussed above, modelling swarm dynamics of MASs by using continuous descriptions is recognised as a potential alternative formalism over discrete concepts for many years. However, to the best knowledge of the author of this thesis it is difficult to find a real application which proofs evidence of this theoretical idea. As a consequence the overall goal of the thesis is to show that DPSs governed by PDEs may be appropriate models for MASs, especially in case of a big number of participants. On top of that, it shall be demonstrated that DPSs theory in terms of controller and observer synthesis can be utilised in order to establish formation control algorithms. The claim is proven and validated in practice by establishing a real-time experiment with a swarm of caterpillar robots.

Apart from that the thesis sets certain objectives to reach the overall goal. Thus, consider the following targets.

- The continuous modelling approach shall be generally independent of the number of agents.
- Moreover, it shall grant individual dynamic properties of each agent. More precisely speaking, the requirements on the dynamics of the agents shall not increase as the number of agents increases.
- The so-called backstepping technique for DPS shall be applied for error tracking control and observer design. So far only a few scientific work groups have implemented the algorithms on real-time experiments [8, 42]. This thesis shall enhance the well-researched theory for practical use.

- Backstepping for coupled systems is currently one of the top research topics within the PDE and control community [89, 41]. Therefore in the associated chapter the author suggests a very different approach compared to the latest theoretical work in this area. The approach shall be verified by simulation studies.

The ambitious goal and the set objectives directly correlate with the structure of the thesis. Thus, Chapter 2 starts with some aspects on Graph theory which is the elementary tool framework for the abstraction of swarm behaviour and subsequently modelling flocking dynamics in discrete states. From this, the following sections motivate the model description in a continuous formalism and discuss prospects to transit from the discrete approach to the continuous one and vice-versa. In this context the chapter introduces the so-called *inverse design approach*.

Chapter 3 elaborates the continuous problem formulation of multi-agent (MA) dynamics using parabolic PDE from a very general view to the specific description of a diffusion-convection-reaction system (DCRS) and the modified, viscous Burgers' equation (MVBE). For the rest of the thesis these two models represent the investigated swarm dynamics in terms of the previously mentioned inverse design approach. This includes the concept of boundary control as well as the evaluation of formation profiles in case of constant model parameters. Furthermore they are discussed in the upcoming chapters in the context of control and observer design.

Consequently, Chapter 4 deals with the feedforward (FF) part of the so-called *two degrees of freedom (2DOF)* controller concept. For this, the flatness properties of the investigated PDEs are used to derive motion planning algorithms by formal state and input parametrisation. Subsequently the approach is formally proven by rigorous convergence analysis of the governed recursive formalism. Comments on appropriate trajectory assignment close this chapter.

Connecting to the previous ideas Chapter 5 addresses the feedback (FB) component of the controller as well as associated observer techniques. Thus, the so-called backstepping method is utilised for DPSs governed by both, uncoupled and coupled PDEs. Especially for a huge family of coupled PDEs the developed scheme allows a simplified design procedure. This is a real novelty in this field and generalises results from previous research, e.g. published in [3]. Each claim is followed by a theoretical analysis which is thoroughly conducted by making use of Lyapunov methods and ensures exponential stability properties.

After closing the theoretical parts Chapter 6 presents an extensive simulation study with four different scenarios as well as a real-time experiment where a swarm of caterpillar robots are deployed. In particular, the first two simulation cases deal with swarm dynamics governed by uncoupled PDEs, whereas the second scenario includes non-linear (NL) behaviour. The other two simulation studies verify the developed 2DOFs control approach for coupled PDEs. Eventually the real-time implementation is able to validate both, the continuous modelling approach as well as the flatness-based motion planning combined with the backstepping-based tracking controller.

The final chapter provides a summary and conclusion of the thesis. Apart from that it proposes and indicates aspects for future work and further research topics on formation control of MASs.

1.4 Remarks on Notation

It is of high interest to the author to keep the notation compact, consistent, and unique throughout the entire thesis. This shall support the readability for a broad audience and readership. Therefore, the *List of Symbols* starting at page XIII describes all used expressions and abbreviations. Appendix C clarifies the mathematical background and formal constructs. Apart from that some notational conventions are summarised in the following for the reader's convenience.

- The italic subscript is exclusively reserved to identify an agent quantity. Thus, $(\cdot)_i$ is used to describe a symbol for the $i-$th agent. The number of agents is denoted by N. The superscript index $(\cdot)^m$ refers to quantities of the $m-$th coordinate. The number of coordinates is n.

- Bold symbols indicate vectors or vector-valued functions, e.g. $\boldsymbol{v}$, $\boldsymbol{x}(z,t)$. For matrices or matrix-valued functions capital letters are used. Elements of these multi-dimensional constructs are indicated by a single or double superscripts, e.g. v^j is an element of a vector, and m^{jk} is an element of a matrix, respectively.

- The partial derivative of order α and β of a function $f(z,t)$ with respect to z and t is denoted by $\partial_z^\alpha \partial_t^\beta f(z,t) = \partial^{\alpha+\beta} f(z,t) / \left(\partial z^\alpha \partial t^\beta\right)$. The notation $\partial_t^\alpha f(t) = \frac{\mathrm{d}^\alpha f(t)}{\mathrm{d}t^\alpha}$ stands for the derivative of order α of a function $f(t)$ with respect to t. Moreover assume a function $g(z,s,t)$ with two independent variables in space and the last in time, then the abbreviation $\frac{\mathrm{d}}{\mathrm{d}z} g(z,z,t) = \partial_z g(z,s,t)|_{z=s} + \partial_s g(z,s,t)_{z=s}$ stands for the total derivative with respect to z.

- Symbols with a top bar are referred to steady state situations, a hat symbolises estimations, and symbols attached with a superscripted asterisk are desired values. Furthermore, symbols with the non-italic subscript $(\cdot)_\mathrm{c}$ or $(\cdot)_\mathrm{o}$ belong to control- or observer-related quantities, respectively.

Chapter 2

Graph Theory and Continuum Models for Multi-Agent Systems

The following sections discuss the abstraction process of a networked system and motivate its discrete and continuous modelling concepts. This deliberately guides the readership to the intended *inverse design approach*.

2.1 Abstraction and Discrete Models for Multi-Agent Systems

A natural and classic approach to abstract the complexity of a network consisting of a number of participants is to sketch them as labelled circles and draw lines between them. The lines are a vivid but simple way to visualise the explicit connections among the participants. With the resulting mesh of circles and lines it is possible to study the overall picture from a more conceptual perspective. Following this path the abstraction process leads to a discrete setup and thus a discrete problem formulation which is well supported by the mathematical concept of the so-called *graph theory*. Its scope offers broad and established tools to deal with the discrete nature of MASs. The unambiguous approach allows to abstract the interaction between the agents and copes with a rich family of explicit and specific problems such as formation control. The following discussion focusses on elementary constructs of graph theory and is highly oriented towards the statements in [53] and [6].

2.1.1 Graph Theory Fundamentals

Let $\mathcal{G}$ be a *finite, undirected, simple graph*, or short *graph*, which is built upon a pair of sets $\mathtt{V}$ and $\mathtt{E}$. First, the set $\mathtt{V}$ is assumed to consist of a finite number of elements. Each element i is called a vertex and assigned with an arbitrary *identification*, denoted as $\mathtt{v}_i \in \mathtt{V}$ to distinguish the elements from each other[1]. Hence, a set $\mathtt{V}$ with N elements is represented by

$$\mathtt{V} = \{\mathtt{v}_1, \mathtt{v}_2, \dots, \mathtt{v}_\mathrm{N}\}\,. \tag{2.1}$$

From this, the deduced set $\mathtt{E}$ is introduced, which is synthesised by *2-elements subsets* of $\mathtt{V}$. Then $\mathtt{E} \subseteq \mathtt{V}^2$ consists of elements of the form $\{\mathtt{v}_i, \mathtt{v}_j\} \in \mathtt{E}$ such that $i = 1, 2, \dots, \mathrm{N}$ and $i \neq j$. The elements $\{\mathtt{v}_i, \mathtt{v}_j\}$ are called edges, short $\mathtt{e}_{ij} = \{\mathtt{v}_i, \mathtt{v}_j\}$, of the graph and they establish a sort of connection between $\mathtt{v}_i$ and $\mathtt{v}_j$, which is formally expressed by $\mathtt{v}_i \sim \mathtt{v}_j$. The cardinality of $\mathtt{E}$ is abbreviated as $|\mathtt{E}| = \mathrm{E}$. *Simple graphs* only allow one connection, either directed or undirected, between two nodes in contrary to *multigraphs* which permit multiple edges and loops. A *loop* is an edge which connects a vertex to itself. In case the edges are associated with a certain direction, then $\mathtt{e}_{ij}$ is defined as the ordered pair $(\mathtt{v}_i, \mathtt{v}_j) \subseteq \mathtt{V} \times \mathtt{V}$ and the graph is labelled as a *directed graph*. Consequently a finite, directed or undirected, simple graph or multigraph $\mathcal{G}$ may be formally defined as the pair $\mathcal{G} = (\mathtt{V}, \mathtt{E})$, or often denoted conveniently as $\mathcal{G}(\mathtt{V}, \mathtt{E})$.

Remark 2.1. *In the following the discussion mainly concentrates on finite, undirected, simple graphs since in this work graph theory is mainly used to motivate a modelling approach for MASs based on an agent continuum model. Even though the presented concepts and notions can be extended in the context of finite, directed multigraphs as well. The interested reader is referred to [53].*

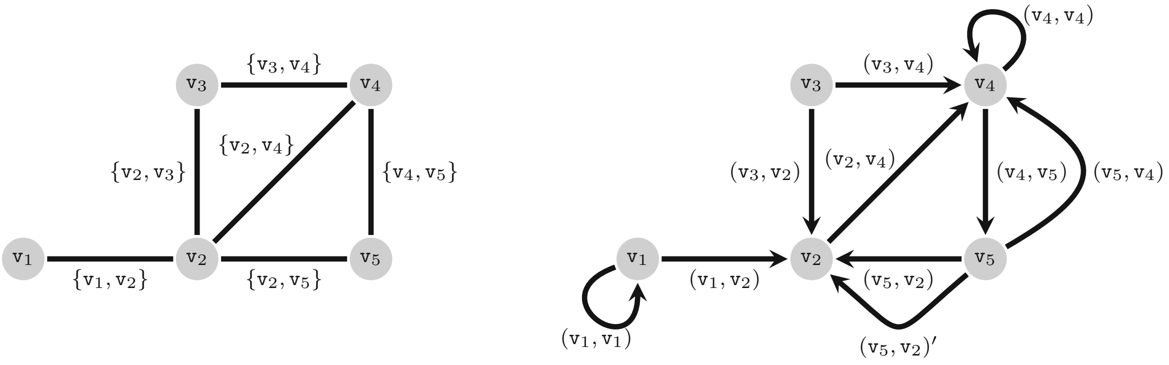
Since the definition of a graph is derived from a set-theoretical perspective it qualifies for vivid graphical interpretations. Vertices are commonly drawn as *dots* (•) and edges as *curves* (—), which connect two vertices. For example the graph $\mathcal{G}_1(\mathtt{V}_1, \mathtt{E}_1)$ sketched in Fig. 2.1a is built upon the set $\mathtt{V}_1 = \{\mathtt{v}_1, \mathtt{v}_2, \mathtt{v}_3, \mathtt{v}_4, \mathtt{v}_5\}$ of five vertices and six edges $\mathtt{E}_1 = \{\{\mathtt{v}_1, \mathtt{v}_2\}, \{\mathtt{v}_2, \mathtt{v}_3\}, \{\mathtt{v}_3, \mathtt{v}_4\}, \{\mathtt{v}_4, \mathtt{v}_5\}, \{\mathtt{v}_2, \mathtt{v}_4\}, \{\mathtt{v}_2, \mathtt{v}_5\}\}$. The other graph $\hat{\mathcal{G}}_1(\mathtt{V}_1, \hat{\mathtt{E}}_1)$ drawn in (b) of Figure 2.1 shows a directed multigraph with set of edges $\hat{\mathtt{E}}_1$ as ordered pairs according to the drawing. Here it has to be said that the edges $(\mathtt{v}_5, \mathtt{v}_2)$ and $(\mathtt{v}_5, \mathtt{v}_2)'$ have distinct labels. Therefore are appointed with their own identity and are considered as different connections. Without their own label they could have been treated as one single connection. The illustration motivates to derive further attributes and properties of graphs. Obviously, from the point of view of vertex $\mathtt{v}_1$ there is a connection to vertex $\mathtt{v}_2$, but it has no more other edges. However, vertex $\mathtt{v}_1$ is connected to all other vertices. Therefore, if a relationship between two vertices $\mathtt{v}_i \sim \mathtt{v}_j$ exists, they are called *adjacent* or they are 'neighbours' in some sense. At the same time the edge $\mathtt{e}_{ij}$ is called *incident* with the two involved vertices $\mathtt{v}_i$ and $\mathtt{v}_j$. From this, it is convenient to consolidate all neighbours of a vertex to the subset

$$\mathtt{Nbr}(i) = \{\mathtt{v}_j \in \mathtt{V} \,|\, \mathtt{e}_{ij} \in \mathtt{E}\} \subseteq \mathtt{V} \tag{2.2}$$

which is denoted as the *neighbourhood* of vertex $\mathtt{v}_i$. It is obvious that for undirected graphs the neighbourhood is symmetric in the sense that

$$\mathtt{v}_i \in \mathtt{Nbr}(j) \Leftrightarrow \mathtt{v}_j \in \mathtt{Nbr}(i) \tag{2.3}$$

[1] In some literature an element is called a node and consequently indicated by the symbol $\mathtt{n}_i$, e.g. in [55].



(a) Simple graph $\mathcal{G}_1(\mathtt{V}_1, \mathtt{E}_1)$.

(b) Multigraph $\hat{\mathcal{G}}_1(\mathtt{V}_1, \hat{\mathtt{E}}_1)$.

Figure 2.1: Graphical illustrations of finite graphs with five vertices; an undirected, simple graph with six edges in (a) and a directed multigraph in (b). The multigraph contains two loops for vertices $\mathtt{v}_1$ and $\mathtt{v}_4$ and multiple edges connect the pairs of vertices $\mathtt{v}_2 \sim \mathtt{v}_5$ and $\mathtt{v}_4 \sim \mathtt{v}_5$.

is applied. Moreover, the cardinality of the neighbourhood

$$|\mathtt{Nbr}(i)| = d\,(\mathtt{v}_i) \tag{2.4}$$

is referred as the degree d of the given vertex $\mathtt{v}_i$. In this context a node with degree zero, i.e., $d\,(\mathtt{v}_i) = 0$, is called *isolated*. The introduced properties inspire to have a closer look on the edges and ask, if there exists an indirect connection between two vertices $\mathtt{v}_i$ and $\mathtt{v}_j$ via other edges when they are not adjacent or equivalently $\mathtt{v}_j \notin \mathtt{Nbr}(i)$. Therefore, a sequence of distinct vertices and edges is called a path $\mathcal{P}$ of finite length m, e.g., assigned as

$$\mathtt{v}_0^{\mathcal{P}}, \mathtt{e}_{01}^{\mathcal{P}}, \mathtt{v}_1^{\mathcal{P}}, \mathtt{e}_{12}^{\mathcal{P}}, \mathtt{v}_2^{\mathcal{P}}, \ldots, \mathtt{v}_{m-1}^{\mathcal{P}}, \mathtt{e}_{m-1,m}^{\mathcal{P}}, \mathtt{v}_m^{\mathcal{P}}, \tag{2.5}$$

when the vertices $\mathtt{v}_k^{\mathcal{P}}$ and $\mathtt{v}_{k+1}^{\mathcal{P}}$ are adjacent for each $k \in \{0, 1, \ldots m-1\}$. Moreover, note the following definition.

Definition 2.1 (Graph isomorphism [6])**.** *An isomorphism between two graphs $\mathcal{G}(\mathtt{V}, \mathtt{E})$ and $\hat{\mathcal{G}}(\hat{\mathtt{V}}, \hat{\mathtt{E}})$ is a bijective function $f : \mathtt{V} \to \hat{\mathtt{V}}$ mapping the vertex sets, such that*

$$\mathtt{e}_{ij} = \{\mathtt{v}_i, \mathtt{v}_j\} \in \mathtt{E} \quad \Leftrightarrow \quad \hat{\mathtt{e}}_{ij} = \{f(\mathtt{v}_i), f(\mathtt{v}_j)\} \in \hat{\mathtt{E}} \tag{2.6}$$

applies. If an isomorphism between two graphs exists, then they are called isomorphic *and in the following this is denoted as $\mathcal{G} \simeq \hat{\mathcal{G}}$.*

Isomorphism between graphs can be described as an *edge-preserving* bijection. From an graph theoretical point of view isomorphic graphs can be treated equally, only the vertices may have a different identification or indexing order.

With this the following attributes for pairs of vertices and some classifications of graphs are deduced [53]:

- The vertices $\mathtt{v}_0^{\mathcal{P}}$ and $\mathtt{v}_m^{\mathcal{P}}$ are referred as *end vertices* of the path $\mathcal{P}$. Contrary, $\mathtt{v}_i = \mathtt{v}_k^{\mathcal{P}}$ with $k \in \{1, \ldots m-1\}$ are named as *inner vertices*.
- Two vertices $\mathtt{v}_i$ and $\mathtt{v}_j$ are called *connected*, if there exists a finite path (2.5) with the end vertices $\mathtt{v}_0^{\mathcal{P}} = \mathtt{v}_i$ and $\mathtt{v}_m^{\mathcal{P}} = \mathtt{v}_j$.
- A *cycle* is a closed path. It has equal end vertices $\mathtt{v}_0^{\mathcal{P}} = \mathtt{v}_m^{\mathcal{P}}$, but distinct inner vertices.
- A graph $\mathcal{G}(\mathtt{V}, \mathtt{E})$ is called *connected* when each pair of vertices in $\mathtt{V}$ is connected; a graph is *disconnected*, if the assertion is not applicable.
- A graph with $\mathtt{V} = \{\mathtt{v}_1, \mathtt{v}_2, \ldots, \mathtt{v}_{\mathrm{N}}\}$ is called a
  - *Complete graph* $\mathcal{K}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{K}})$ if a vertex is neighbour of every other vertex, or $\mathtt{E}_{\mathcal{K}} = \mathtt{V}^2$.
  - *Path graph*, if and only if it is *isomorphic* to the graph $\mathcal{P}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{P}})$ with $\mathtt{e}_{ij} \in \mathtt{E}_{\mathcal{P}}$ and $j = i+1$ for $i = 1, 2, \ldots, \mathrm{N}-1$.
  - N-*Cycle graph* $\mathcal{C}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{C}})$ if it consists of a single cycle.
  - *Forest graph* $\mathcal{F}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{F}})$ if it has no cycles.
  - *Star graph* $\mathcal{S}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{S}})$ with $\mathtt{e}_{ij} \in \mathtt{E}_{\mathcal{S}}$ if and only if $i = 1$ or $j = 1$.

Some of these common graphs are illustrated in Figure 2.2 with seven vertices. In view of MASs consisting of agents with different roles it is necessary to analyse subsets of graphs and introduce operations among them. For this reason let $\mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}})$ with $\mathtt{S} \subseteq \mathtt{V}$ and $\mathtt{E}_{\mathrm{s}} = \{\{\mathtt{v}_i, \mathtt{v}_j\} \in \mathtt{E} \,|\, \mathtt{v}_i,\, \mathtt{v}_j \in \mathtt{S}\} \subseteq \mathtt{E}$ be a *subgraph* of $\mathcal{G}(\mathtt{V}, \mathtt{E})$. If $\mathtt{S} = \mathtt{V}$ is applied it is called a *spanning subgraph* [6]. Hence the boundary of a subgraph $\partial\mathcal{G}_{\mathrm{s}}$ is defined as

$$\partial\mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}}) = \mathcal{G}_{\partial\mathrm{s}}(\partial\mathtt{S}, \mathtt{E}_{\mathrm{s}}) = (\partial\mathtt{S}, \{\{\mathtt{v}_i, \mathtt{v}_j\} \in \mathtt{E} \,|\, \mathtt{v}_i, \mathtt{v}_j \in \partial\mathtt{S}\}) \tag{2.7}$$

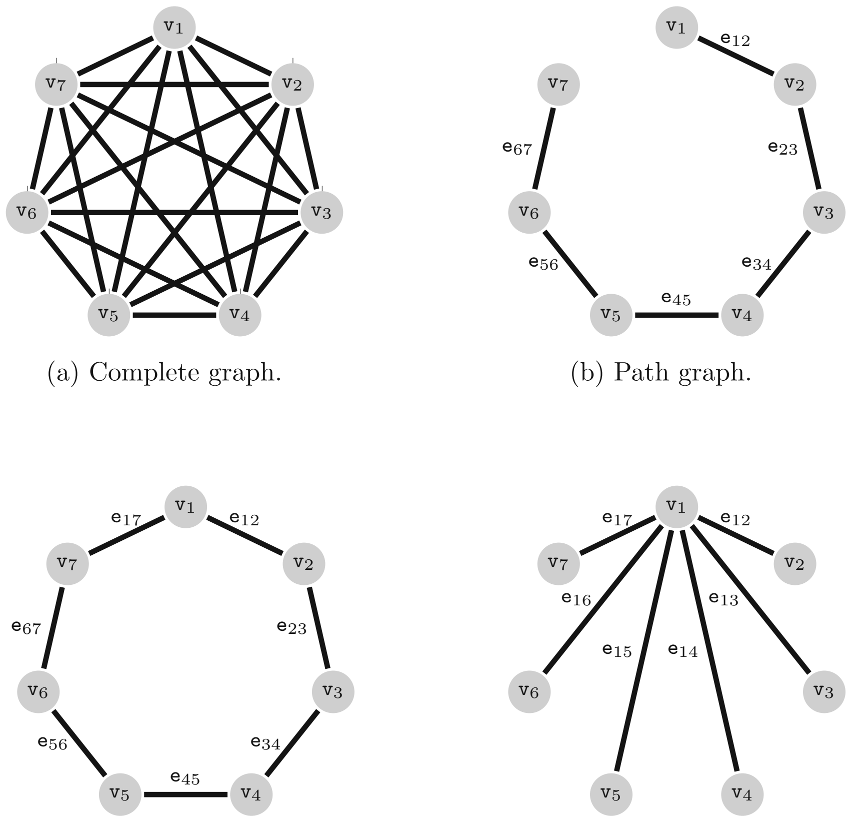
with the set of boundary vertices $\partial\mathtt{S} = \{\mathtt{v}_i \in \mathtt{V} \setminus \mathtt{S} \,:\, \exists \mathtt{v}_j \in \mathtt{S} \; s.t. \; \{\mathtt{v}_i, \mathtt{v}_j\} \in \mathtt{E}\}$. The union of the subgraph and its boundary, i.e.,

$$\overline{\mathcal{G}}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}}) = \partial\mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}}) \cup \mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}}) \tag{2.8}$$

is the closure of the subgraph $\mathcal{G}_{\mathrm{s}}$.

Remark 2.2 (Weighted graphs [53])**.** *For networks it is essential to describe the relationship between its members mathematically or to assign properties to its connections. This leads to so-called* weighted graphs*. In accordance to the notions above the definition of a graph $\mathcal{G}(\mathtt{V}, \mathtt{E})$ is extended by a function $w : \mathtt{E} \times \mathbb{R}_{t_0}^{+} \to \mathbb{R}$, which assigns a numerical value to each edge $\mathtt{e}_{ij}$ for $t \geq t_0$. Consequently, in the following the ordered triple $(\mathtt{V}, \mathtt{E}, w)$ stands for a weighted graph, labelled as $\mathcal{G}(\mathtt{V}, \mathtt{E}, w)$.*

Remark 2.3 (Time-invariant graphs)**.** *A graph $\mathcal{G}(\mathtt{V}, \mathtt{E})$ is called time-invariant (TI), if the set of edges $\mathtt{E}$ does not change with time. Additionally in case of weighted graphs $\mathcal{G}(\mathtt{V}, \mathtt{E}, w)$, the weight w assigned to an edge $\mathtt{e}_{ij} \in \mathtt{E}$ is TI, i.e., $w(\mathtt{e}_{ij}, t) = w(\mathtt{e}_{ij})$.*



(a) Complete graph.

(b) Path graph.

(c) 7−Cycle graph.

(d) Star graph.

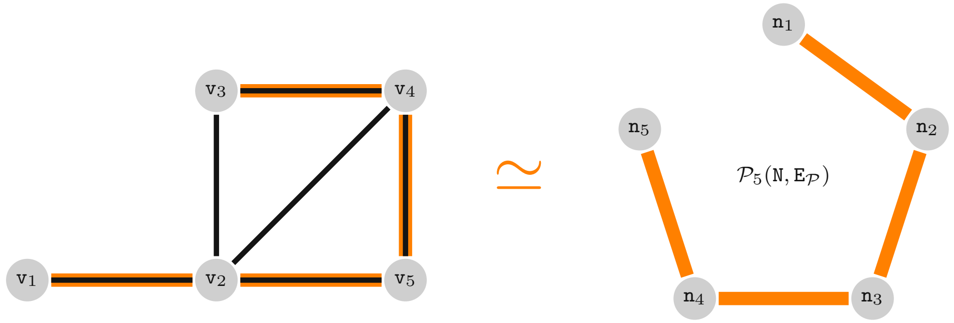
Figure 2.2: Examples of some standard graphs with seven vertices each edges according to the class of graph.

Generally, graphs can be classified in many ways. In this work paths defined as subgraphs $\mathcal{P}(\mathtt{S}, \mathtt{E}_{\mathcal{P}})$ for some host graph $\mathcal{G}(\mathtt{V}, \mathtt{E})$ play a crucial role. Among others, paths or path graphs benefit from the following features:

- a path graph with at least two vertices is connected, has no cycles and has $\mathrm{N} - 1$ edges;
- the end vertices $\mathtt{v}_i \in \partial\mathtt{S}$ have degree 1, the inner vertices $\mathtt{v}_i \in \mathtt{S} \setminus \partial\mathtt{S}$ have degree 2;
- a path is a tree with $d(\mathtt{v}_i) \leq 2$, and hence inherits the properties of trees,
- a linear forest is the disjoint union of path graphs.

Furthermore it is worth to have a closer look at paths, where all vertices of the host graph are involved, meaning $\mathtt{S} = \mathtt{V}$.

Definition 2.2 (Hamiltonian path and Hamiltonian cycle [6])**.** *A* Hamiltonian path $\mathcal{P}_{\mathrm{H}}(\mathtt{V}, \mathtt{E}_{\mathcal{P}})$ *is a path in an undirected or directed graph that visits each vertex* $\mathtt{v}_i \in \mathtt{V}$, $i = 1, 2 \ldots |\mathtt{V}|$ *exactly once, i.e., a spanning path. In case the end vertices of the Hamiltonian path are adjacent, then the Hamiltonian path is a spanning cycle and it is called a* Hamiltonian cycle $\mathcal{C}_{\mathrm{H}}(\mathtt{V}, \mathtt{E}_{\mathcal{C}})$.



$f : \mathtt{V}_1 \to \mathtt{N}$
$\mathtt{v}_1 \Leftrightarrow \mathtt{n}_1$
$\mathtt{v}_2 \Leftrightarrow \mathtt{n}_2$
$\mathtt{v}_3 \Leftrightarrow \mathtt{n}_5$
$\mathtt{v}_4 \Leftrightarrow \mathtt{n}_4$
$\mathtt{v}_5 \Leftrightarrow \mathtt{n}_3$

Figure 2.3: A spanning subgraph $\mathcal{G}_{s1}(\mathtt{V}_1,\mathtt{E}_{s1})$ of the host graph $\mathcal{G}_1(\mathtt{V}_1,\mathtt{E}_1)$ with edges (▬) drawn in orange; the spanning subgraph is a Hamiltonian path and hence isomorphic to the path graph $\mathcal{P}_5(\mathtt{N},\mathtt{E}_{\mathcal{P}}) \simeq \mathcal{G}_{s1}(\mathtt{V}_1,\mathtt{E}_{s1})$ with five vertices. The associated isomorphism is given as the bijection f.

Definition 2.3 (Traceable graph and Hamiltonian graph [6])**.** *A* traceable graph $\mathcal{G}_t(\mathtt{V},\mathtt{E})$ *is a graph that contains a Hamiltonian path* $\mathcal{P}_H(\mathtt{V},\mathtt{E}_{\mathcal{P}})$*. A* Hamiltonian graph $\mathcal{G}_H(\mathtt{V},\mathtt{E})$ *is a graph that possesses a Hamiltonian cycle* $\mathcal{C}_H(\mathtt{V},\mathtt{E}_{\mathcal{C}})$*. Consequently a Hamiltonian graph is traceable, but the converse does not hold in general.*

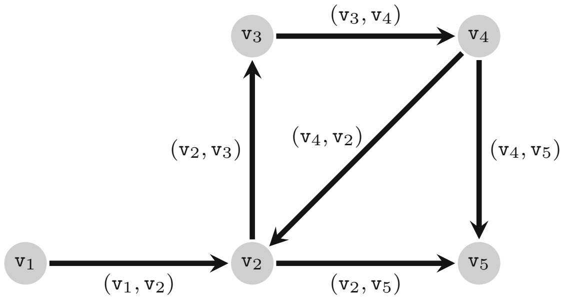
In Figure 2.3 a Hamiltonian path is sketched as a spanning subgraph $\mathcal{G}_{s1}(\mathtt{V}_1,\mathtt{E}_{s1})$ of the host graph $\mathcal{G}_1(\mathtt{V}_1,\mathtt{E}_1)$ initially depicted in Figure 2.1. The edge set of the Hamiltonian path is drawn in orange. Moreover, the illustration highlights the isomorphism between path graphs and states the bijective function for the given example. The graphs drawn in Figures 2.2a to 2.2c are obviously traceable, additionally Figure 2.2a and Figure 2.2c show Hamiltonian graphs. As a consequence for a traceable graph $\mathcal{G}_t(\mathtt{V},\mathtt{E})$ the following useful corollary can be verified.

Corollary 2.1. *If a graph* $\mathcal{G}_t(\mathtt{V},\mathtt{E})$ *is traceable then there exists a spanning subgraph* $\mathcal{G}_s(\mathtt{V},\mathtt{E}_s) \subseteq \mathcal{G}_t(\mathtt{V},\mathtt{E})$ *which is a path graph. This means the path is a Hamiltonian path.*

This completes a very brief introduction to graph theory fundamentals. As a summary, graphs can be characterised as constructions composed of a finite number of abstract objects represented as vertices, which might interact with each other. Interactions are referred as edges between the objects. Even though this representation comes very natural, graphs can be discussed from a linear algebra point of view as well. The latter approach gives powerful tools to handle graphs in a very compact and systematic way. Therefore the following section introduces important matrix representations of graphs.

2.1.2 Matrix Representations of Graphs

For the attributes *degree*, *adjacency*, and *incidency* corresponding matrix associations can be derived intuitively. The latter is significant for the discussion of (arbitrarily) oriented graphs. First, the *degree matrix* is the listing of the degree of each vertex $\mathtt{v}_i$ in a diagonal matrix of the



$(\mathtt{v}_3, \mathtt{v}_4)$
$\mathtt{v}_3$
$\mathtt{v}_4$
$(\mathtt{v}_2, \mathtt{v}_3)$
$(\mathtt{v}_4, \mathtt{v}_2)$
$(\mathtt{v}_4, \mathtt{v}_5)$
$\mathtt{v}_1$
$\mathtt{v}_2$
$\mathtt{v}_5$
$(\mathtt{v}_1, \mathtt{v}_2)$
$(\mathtt{v}_2, \mathtt{v}_5)$

Figure 2.4: Graph $\mathcal{G}_1$ that has been oriented as $\mathcal{G}_1^o$.

form

$$\Delta(\mathcal{G}) = \begin{bmatrix} d(\mathtt{v}_1) & 0 & \cdots & 0 \\ 0 & d(\mathtt{v}_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d(\mathtt{v}_N) \end{bmatrix} . \tag{2.9}$$

Secondly, the neighbourhood of each vertex $\mathtt{v}_i$ is summarized in the row i of the *adjacency matrix*. The entries of the matrix are obtained according to

$$A(\mathcal{G})_{ij} = \begin{cases} 1 & \text{if } \mathtt{v}_j \in \mathtt{Nbr}(i), \text{ or equivalently } \mathtt{e}_{ij} \in \mathtt{E}, \\ 0 & \text{otherwise.} \end{cases} \tag{2.10}$$

This means if vertex i is adjacent to vertex j, i.e., $\mathtt{v}_i \sim \mathtt{v}_j$, the entry $A(\mathcal{G})_{ij}$ is 1. From (2.3) it can be determined that the adjacency matrix is symmetric and is of dimension $\mathrm{N} \times \mathrm{N}$, with $\mathrm{N} = |\mathtt{V}|$. Moreover it has zeros at the main diagonal. As an example the degree matrix and adjacency matrix for the graph given in Figure 2.3 can be written as

$$\Delta(\mathcal{G}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} \quad \text{and} \quad A(\mathcal{G}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} . \tag{2.11}$$

The last of the three initially named matrices, the *incidence matrix*, is similar to the adjacency matrix but is constructed from the perspective of the edges of a graph. For this, it is necessary to arbitrarily assign an orientation to the edges in case the graph is undirected. The orientation turns the edges into a set of ordered pairs $(\mathtt{v}_i, \mathtt{v}_j) \in \mathtt{E}^o \subset \mathtt{V} \times \mathtt{V}$ where $\mathtt{v}_i$ is named as tail and $\mathtt{v}_j$ is known as the head of the edge. Then the $\mathrm{N} \times \mathrm{E}$−incidence matrix D of the arbitrarily orientated graph $\mathcal{G}^o$ with N vertices and E edges is defined as

$$D(\mathcal{G}^o) = [d_{ij}] , \quad \text{where} \quad d_{ij} = \begin{cases} -1 & \text{if } \mathtt{v}_i \text{ is the tail of edge } \mathtt{e}_{ij} \in \mathtt{E}^o , \\ 1 & \text{if } \mathtt{v}_i \text{ is the head of edge } \mathtt{e}_{ij} \in \mathtt{E}^o , \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

Fig. 2.4 basically shows the graph $\mathcal{G}_1(\mathtt{V}_1, \mathtt{E}_1)$, previously depicted in Fig. 2.1, but now orientated as $\mathcal{G}_1^o(\mathtt{V}_1, \mathtt{E}_1^o)$ and its incidence matrix is given by

$$D(\mathcal{G}_1^o) = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} . \tag{2.13}$$

With this basic matrix representations it is possible to construct the so-called *graph Laplacian*. This matrix plays an crucial role in formation control of MASs, especially when agreement shall be reached on a certain property within a network, e.g., a joint value of a state variable. Depending on the properties of the graph it can be derived in different ways. For undirected graphs the graph Laplacian can be deduced from the first two matrices (2.9) and (2.10) discussed above by simply computing

$$L(\mathcal{G}) = \Delta(\mathcal{G}) - A(\mathcal{G}) . \tag{2.14}$$

As an example the graph Laplacian of the undirected graph illustrated in Fig. 2.1 can be determined as

$$L(\mathcal{G}) = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 4 & -1 & -1 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & -1 & 0 & -1 & 2 \end{bmatrix} . \tag{2.15}$$

An alternative way allows to calculate the graph Laplacian for undirected, but arbitrarily oriented, graphs by making use of the incidence matrix (2.12) and

$$L(\mathcal{G}) = D(\mathcal{G}^o)D(\mathcal{G}^o)^T . \tag{2.16}$$

At this state it has to be pointed out that $L(\mathcal{G})$ is obviously independent from the orientation of the edges since (2.14) does not require any related information. Therefore and for convenience the notation $D(\mathcal{G}) = D(\mathcal{G}^o)$ may be applied for arbitrarily oriented graphs. The second definition (2.16) allows to extend the framework with the *weighted graph Laplacian* denoted as

$$L_w(\mathcal{G}, t) = D(\mathcal{G})W(\mathtt{E}, t)D(\mathcal{G})^T . \tag{2.17}$$

Here, the E × E diagonal matrix $W(\mathtt{E}, t)$ holds the weights $w_{ij} = w(\mathtt{e}_{ij}, t)$ of each edge $\mathtt{e}_{ij}$ of the underlying graph $\mathcal{G}(\mathtt{V}, \mathtt{E})$ with the cardinality of the set of edges $|\mathtt{E}| = \mathrm{E}$.

2.1.3 Modelling Multi-Agent Dynamics based on Graph Theory

The graph theoretical fundamentals discussed above qualify to abstract a MAS from its complex coherence to its elementary structure and properties. In this context an agent i is assumed to be represented subsequently by a vertex $\mathtt{v}_i$ of a graph $\mathcal{G}(\mathtt{V},\mathtt{E})$. An agent may hold properties which vary with time and thus is assigned with the state vector $\boldsymbol{x}(\mathtt{v}_i,t)$. This is formally expressed by the mapping

$$\boldsymbol{x} : \mathtt{V} \times [t_0, \infty] \to \mathbb{R}^n \tag{2.18}$$

for $t \geq t_0$ and the initial state value is denoted as $\boldsymbol{x}(\mathtt{v}_i,t_0) = \boldsymbol{x}_0(\mathtt{v}_i)$. Moreover, the agent $\mathtt{v}_i$ may share properties with its neighbouring agents represented by $\mathtt{v}_{j_k} \in \mathtt{Nbr}(i)$ with $k = 1,\dots,d(\mathtt{v}_i)$. Consequently, the edges $\mathtt{e}_{ij_k}$ define a mesh of interaction among the agents. Assuming that the properties of the agent change with time the dynamics of state vector $\boldsymbol{x}(\mathtt{v}_i,t)$ is modelled as a simple integrator

$$\partial_t \boldsymbol{x}(\mathtt{v}_i,t) = \boldsymbol{u}_i(t,\boldsymbol{x}(\mathtt{v}_i,t),\boldsymbol{x}(\mathtt{v}_{j_1},t),\dots,\boldsymbol{x}(\mathtt{v}_{j_r},t))\,, \qquad \mathtt{v}_{\{j_1,\dots,j_r\}} \in \mathtt{Nbr}(i)\,, \tag{2.19}$$

with the input $\boldsymbol{u}_i(t,\boldsymbol{x}(\mathtt{v}_i,t),\boldsymbol{x}(\mathtt{v}_{j_1},t),\dots,\boldsymbol{x}(\mathtt{v}_{j_r},t))$, whereas r is an abbreviation for the degree $d(\mathtt{v}_i)$. In the following the inputs of the agents are often denoted as *input protocol* or simply *protocol*. For instance, if the components of the state variable $\boldsymbol{x}(\mathtt{v}_i,t)$ refer to a physical position of the agent in the $n-$dimensional plane, the protocol $\boldsymbol{u}$ defines the velocity of the agent in each dimension.

Remark 2.4. *Generally the $\boldsymbol{x}-$dynamics can be freely assigned or can be even set up independently for each agent. In the context of a swarm of mobile robots it stands to reason that integrator dynamics as introduced in* (2.19) *or a double integrator*

$$\partial_t^2 \boldsymbol{x}(\mathtt{v}_i,t) = \boldsymbol{u}_i(t,\boldsymbol{x}(\mathtt{v}_i,t),\boldsymbol{x}(\mathtt{v}_{j_1},t),\dots,\boldsymbol{x}(\mathtt{v}_{j_r},t))\,, \qquad \mathtt{v}_{\{j_1,\dots,j_r\}} \in \mathtt{Nbr}(i)\,, \tag{2.20}$$

where the input protocol $\boldsymbol{u}_i$ defines the acceleration of the agent, give appropriate dynamic models.

Moreover, assume the operator

$$\mathcal{L}(\boldsymbol{x})(\mathtt{v}_i,t) := -L_i(\mathcal{G}) \circ [\boldsymbol{x}(\mathtt{v}_1,t),\boldsymbol{x}(\mathtt{v}_2,t),\dots \boldsymbol{x}(\mathtt{v}_\mathrm{N},t)]^T \tag{2.21}$$

wherein the operation $\circ$ stands for the Hadamard product and the notation $L_i(\mathcal{G})$ denotes the $i-$th row of the graph Laplacian matrix of the underlying graph $\mathcal{G}$ and $i = 1,2,\dots,\mathrm{N}$. Then the choice of the input protocol $\boldsymbol{u}_i(t,\boldsymbol{x}(\mathtt{v}_i,t),\boldsymbol{x}(\mathtt{v}_{j_1},t),\dots,\boldsymbol{x}(\mathtt{v}_{j_r},t)) = \mathcal{L}(\boldsymbol{x})(\mathtt{v}_i,t)$ leads to the so-called (graph) Laplacian control

$$\partial_t \boldsymbol{x}(\mathtt{v}_i,t) = \mathcal{L}(\boldsymbol{x})(\mathtt{v}_i,t) = \sum_{\mathtt{v}_j \in \mathtt{Nbr}(i)} \left(\boldsymbol{x}(\mathtt{v}_j,t) - \boldsymbol{x}(\mathtt{v}_i,t)\right)\,, \qquad \mathtt{v}_i \in \mathtt{V} \tag{2.22}$$

with proper initial condition $\boldsymbol{x}(\mathtt{v}_i,t_0) = \boldsymbol{x}_0(\mathtt{v}_i) \in \mathbb{R}^n$. As a descriptive example the Laplacian control evaluated for the 7$-$cycle graph $\mathcal{C}_7(\mathtt{V},\mathtt{E}_\mathcal{C})$ given in Figure 2.2c and the derived graph

Laplacian matrix

$$L(\mathcal{G}_c) = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & -1 & 2 \end{bmatrix} \tag{2.23}$$

gives

$$\begin{aligned} \partial_t \boldsymbol{x}(\mathtt{v}_1, t) &= \boldsymbol{x}(\mathtt{v}_2, t) - 2\boldsymbol{x}(\mathtt{v}_1, t) + \boldsymbol{x}(\mathtt{v}_7, t)\,, \\ \partial_t \boldsymbol{x}(\mathtt{v}_2, t) &= \boldsymbol{x}(\mathtt{v}_3, t) - 2\boldsymbol{x}(\mathtt{v}_2, t) + \boldsymbol{x}(\mathtt{v}_1, t)\,, \\ \vdots \qquad & \quad \vdots \qquad\qquad\qquad \vdots \\ \partial_t \boldsymbol{x}(\mathtt{v}_6, t) &= \boldsymbol{x}(\mathtt{v}_7, t) - 2\boldsymbol{x}(\mathtt{v}_6, t) + \boldsymbol{x}(\mathtt{v}_5, t)\,, \\ \partial_t \boldsymbol{x}(\mathtt{v}_7, t) &= \boldsymbol{x}(\mathtt{v}_1, t) - 2\boldsymbol{x}(\mathtt{v}_7, t) + \boldsymbol{x}(\mathtt{v}_6, t)\,. \end{aligned} \qquad \mathtt{v}_i \in \mathtt{V} \tag{2.24}$$

This set of expressions forms a system of $\mathrm{N}n = 7n$ ODEs. Hence it is evident that every further agent adds an additional set of n equations to the system.

Now let $\mathcal{S} = \mathcal{G}_\mathrm{s}(\mathtt{S}, \mathtt{E}_\mathrm{s})$ be a non-empty, connected subgraph of $\mathcal{G}(\mathtt{V}, \mathtt{E})$ as introduced in Sec. 2.1.1 and $\partial\mathcal{S} = \partial\mathcal{G}_\mathrm{s}(\mathtt{S}, \mathtt{E}_\mathrm{s})$ its boundary graph. With this, it is assumed that the subgraphs $\mathcal{S}_\mathrm{l}$ and $\mathcal{S}_\mathrm{f}$ and their associated set of vertices $\mathtt{S}_\mathrm{l}$ and $\mathtt{S}_\mathrm{f}$ are constructed in a way to meet the characteristics $\mathcal{S}_\mathrm{l} = \partial\mathcal{S}_\mathrm{f}$ and $\mathcal{S}_\mathrm{l} \cup \mathcal{S}_\mathrm{f} = \mathcal{G}$, respectively. This introduces a graph-theoretical representation of the so-called *leader-follower* concept, which is a common approach to design control algorithms for MASs. Therefore, the subscripts l and f stand for *leader* agent and *follower* agent. When the concept is applied to the path graph $\mathcal{G}_\mathrm{p}(\mathtt{V}, \mathtt{E}_\mathrm{p})$ illustrated in Fig. 2.2b Laplacian control leads to

$$\partial_t \boldsymbol{x}(\mathtt{v}_i, t) = \boldsymbol{x}(\mathtt{v}_{i+1}, t) - 2\boldsymbol{x}(\mathtt{v}_i, t) + \boldsymbol{x}(\mathtt{v}_{i-1}, t)\,, \qquad \forall \mathtt{v}_i \in \mathtt{S}_\mathrm{f} \tag{2.25}$$

$$\begin{aligned} \partial_t \boldsymbol{x}(\mathtt{v}_1, t) &= \boldsymbol{x}(\mathtt{v}_2, t) - \boldsymbol{x}(\mathtt{v}_1, t)\,, \\ \partial_t \boldsymbol{x}(\mathtt{v}_7, t) &= \boldsymbol{x}(\mathtt{v}_6, t) - \boldsymbol{x}(\mathtt{v}_7, t)\,. \end{aligned} \qquad \{\mathtt{v}_1, \mathtt{v}_7\} \in \mathtt{S}_\mathrm{l} \tag{2.26}$$

Focusing on the expressions in (2.24) and (2.25) the analogy to discretisation methods such as the finite difference method (FDM) seems natural. The right-hand side of each expression (2.24) and (2.25) directly implements the pattern of the approximation of the second derivative when using second-order central differences. The structure of the right-hand side of (2.26) resembles the forward and backward finite difference method to approximate first order derivatives. This, combined with the self-induced integrator dynamics on the left-hand side of the equations describes a similar scheme compared to the approximation of PDEs using appropriate discretisation algorithms. Especially when PDEs of so-called parabolic type are semi-discretised by the FDM they render a system of ODEs of equivalent format. In this context the structure of

(2.26) does not match, but only resembles comparable Neumann boundary conditions. For this reason it could be convenient to refrain from the integrator for the leader agents but provide them with external input signals $\boldsymbol{u}_1(t)$ and $\boldsymbol{u}_7(t)$, respectively. This leads to

$$\begin{aligned} \boldsymbol{x}(\mathtt{v}_2,t)-\boldsymbol{x}(\mathtt{v}_1,t) &= \boldsymbol{u}_1(t)\,, \\ \boldsymbol{x}(\mathtt{v}_6,t)-\boldsymbol{x}(\mathtt{v}_7,t) &= \boldsymbol{u}_7(t)\,. \end{aligned} \qquad \{\mathtt{v}_1,\mathtt{v}_7\}\in\mathtt{S}_\mathrm{l}\,,\ \{\mathtt{v}_2,\mathtt{v}_6\}\in\mathtt{S}_\mathrm{f} \tag{2.27}$$

which is a possible discrete pattern of non-homogeneous Neumann boundary conditions in the framework of semi-discretised PDEs. In this case the set of equations in (2.27) directly determines the trajectories of the leader agents $\boldsymbol{x}(\mathtt{v}_1,t)$ and $\boldsymbol{x}(\mathtt{v}_7,t)$. Alternatively, it might be preferable to keep up the integrator dynamics amongst all agents. In this case the external input signals directly determine the velocity of the leader agents according to the dynamic (boundary) conditions

$$\begin{aligned} \partial_t\boldsymbol{x}(\mathtt{v}_1,t) &= \boldsymbol{u}_1(t)\,, \\ \partial_t\boldsymbol{x}(\mathtt{v}_7,t) &= \boldsymbol{u}_7(t)\,. \end{aligned} \qquad \{\mathtt{v}_1,\mathtt{v}_7\}\in\mathtt{S}_\mathrm{l} \tag{2.28}$$

Remark 2.5. *The independent and unrestricted setup of conditional equations for the leader agents in* (2.27) *and* (2.28) *invites to formulate more general conditions. In fact establish the expressions*

$$\begin{aligned} p_1\partial_t^k\boldsymbol{x}(\mathtt{v}_1,t) &= q_1\boldsymbol{x}(\mathtt{v}_2,t)-r_1\boldsymbol{x}(\mathtt{v}_1,t)+\boldsymbol{u}_1(t)\,, \\ p_\mathrm{N}\partial_t^k\boldsymbol{x}(\mathtt{v}_N,t) &= q_\mathrm{N}\boldsymbol{x}(\mathtt{v}_{\mathrm{N}-1},t)-r_\mathrm{N}\boldsymbol{x}(\mathtt{v}_\mathrm{N},t)+\boldsymbol{u}_\mathrm{N}(t)\,, \end{aligned} \qquad \{\mathtt{v}_1,\mathtt{v}_\mathrm{N}\}\in\mathtt{S}_\mathrm{l}\,,\ \{\mathtt{v}_2,\mathtt{v}_{\mathrm{N}-1}\}\in\mathtt{S}_\mathrm{f} \tag{2.29}$$

with appropriate parameters $k\in\mathbb{N}$, $\{p_j,q_j,r_j\}\in\mathbb{R}$, and the external inputs $\boldsymbol{u}_j(t)\,,j\in\{1,\mathrm{N}\}$, respectively, a discrete arrangement for classic boundary conditions of PDEs including the extension of dynamic conditions. By way of example $p_1=q_1=0$ and $r_1=1$ renders for leader agent 1 *an inhomogeneous Dirichlet boundary condition, while $p_\mathrm{N}=0$, $q_\mathrm{N}=1$ and $r_\mathrm{N}\neq 1$, $|r_\mathrm{N}|<\infty$ defines for leader agent* N *a discrete match of the so-called Robin boundary condition in the classical sense of distributed parameter systems.*

Obviously the free design of the input $\boldsymbol{u}_i$ applied to the agent dynamics (2.19) and the in principle independent assignment of the boundary conditions (see Remark 2.5) motivates to choose other analogous discrete arrangements of established distributed parameter systems. For instance the equivalent feedback of a diffusion-reaction system can be written as

$$\partial_t\boldsymbol{x}(\mathtt{v}_i,t)=a\mathcal{L}(\boldsymbol{x})(\mathtt{v}_i,t)+c(\mathtt{v}_i,t)\boldsymbol{x}(\mathtt{v}_i,t)\,, \qquad \forall\mathtt{v}_i\in\mathtt{S}_\mathrm{f} \tag{2.30}$$

with constant diffusion coefficient a and time-variant, agent-dependent reaction coefficient $c(\mathtt{v}_i,t)$. Otherwise, if a double integrator dynamics is preferred over a single one, the combination with Laplacian control leads to

$$\partial_t^2\boldsymbol{x}(\mathtt{v}_i,t)=-v^2\mathcal{L}(\boldsymbol{x})(\mathtt{v}_i,t)\,, \qquad \forall\mathtt{v}_i\in\mathtt{S}_\mathrm{f}. \tag{2.31}$$

This set of ODEs builds an analogue to the semi-discretised PDE of the homogeneous wave equation. Then the constant parameter v is related to the so-called wave velocity.

Retrospectively, the initially mentioned proposition was the abstraction and simplification of the nature of complex networks and systems with various participants. Graph theory is a chapter within mathematics which strongly supports the abstraction process of networks and offers tools and concepts for many arising problems and applications. Though, the analogies between the mathematical representations in form of ODEs and the structure of certain semi-discretised distributed parameter systems invite to pursue the abstraction process even further, especially for networks with large numbers of agents. With this observations it is worth to have a closer look at the relationship between discrete models based on graph-theoretical concepts and continuous models represented by PDEs. Following this objective it is essential to study and derive preliminaries, which allow the transition from continuous to discrete models or vice versa, and consequently embed the discrete set-up into an appropriate framework. A comprehensive conceptional treatise will introduce theories and methods to MA control, which are actually established for PDE-based controller and observer design. This further inspires a so-called inverse design approach and its motivation is discussed in the following paragraphs.

2.2 Motivating Continuum Models for Multi-Agent Systems

Following the previous section the agent dynamics can be described by a system of ODEs, where the order of the system is inherently defined by the number of agents N times the number of states n for each agent. As a consequence, the modelling of the dynamical behaviour of large scale networks results in high order representations in form of coupled ordinary differential equations. Analysis, scalability, controller, and observer design may suffer from the increasing order and the resulting mathematical complexity. The theoretical transfer to an infinite number of agents $\mathrm{N} \to \infty$ would render hypothetically a system of coupled ODEs of infinite order and thus motivates a continuous reformulation of certain classes of problems, e.g. [71, 72]. Even though this thought experiment will be modified in this section the main idea remains. The perspective leads to a modelling process using distributed parameter systems governed by PDEs. With this, it would reveal the possibility to follow controller and observer concepts and conduct stability analysis based on control theory for PDEs. Since for real applications any computation hardware, e.g. a central processing unit (CPU), can only deal with finite dimensional systems and real time applications require finite computation time a suitable implementation strategy, which sufficiently approximates the spatial continuous expressions, is necessary by all means. Here, model-based control theory of distributed parameter systems distinguishes between the so-called *Early lumping* and *Late lumping* approach. While the former already approximates the underlying model within the spatial domain before the actual controller or observer design, the latter stalls this action to the very last step of the development process. Especially the late lumping concept involves some handy consequences for the controller and observer design for networks. Among others, algorithms designed in a spatially continuous way bring along the following upsides:

- The designed algorithms are independent of the number of participants since they are constructed for an infinite dimensional system.
- In principle, the designed algorithms are independent of the underlying communication topology, i.e., the adjacency of the agents.

The choice of the spatial approximation method defines a finite number of states and consequently their neighbourhood within the discretised spatial domain. Referring the states to properties of participants within a network, a swarm, or a MAS, the approximation step induces a (pseudo-)graph-theoretical interpretation. Though, the approximation of the spatial domain or appropriate discretisation methods, receptively, do not close the gap completely between control theory based on graphs and based on DPSs governed by PDEs. As derived in Section 2.1.1, in principle, a graph is defined as a pair of sets and a PDE is valid on some continuous domain with all its associated features. Moreover, in most cases the spatial domain completely looses its physical meaning when put in conjunction with swarms or networks, and hence requires some reinterpretation. The other way round is not satisfying either. As already mentioned above, the journey from control concepts based on graphs to control theory for PDEs can not be accomplished by just introducing a transfer from a finite number to infinite number of agents. Here a key argument is that this process does not maintain the dynamic properties of the single agent. Consequently, this drives the motivation to rethink the linking between the two theories in order to overcome some inconsistencies and to discuss preliminaries and conditions, which allow to formulate related representations.

For this purpose first the initial situation and requirements for both theories are discussed before connection both approaches. The objective of this process is to find missing ingredients which can patch discrepancies between the theories. In the following, statements for discrete models based on graph-theoretical concepts make use of the preliminary notions derived in section 2.1. The required properties to perform control theory for distributed parameter system are introduced below by studying the family of DCRSs briefly.

2.2.1 Comparison between Discrete and Continuum Models

First, let us recall the construction of graphs. Generally speaking a graph consists of a set of vertices $\mathtt{V}$ and a set of edges $\mathtt{E}$. Among other additional definitions, a vertex can be assigned with properties, e.g., similarly to (2.18), but here with only one single, time-varying state $x_i(t) = x(\mathtt{v}_i, t)$ for each vertex $\mathtt{v}_i \in \mathtt{V}$, defined as the mapping

$$x_i : \mathtt{v}_i \times [t_0, \infty] \to \mathbb{R} \tag{2.32}$$

This means, in case of N agents and each of them is assigned with a state, the entire vertex set is mapped to a suitable Vector space, e.g., $(\mathbb{R}^{\mathrm{N}}, \|\cdot\|)$. In order to describe the temporal evolution of the states x_i an input protocol u_i is applied to define a dynamical system for $t > t_0$ by means of

$$\frac{\partial x_i(t)}{\partial t} = u_i(t, x_i, x_{j_1}, \ldots, x_{j_r})\,, \qquad \mathtt{v}_{\{j_1,\ldots,j_r\}} \in \mathtt{Nbr}(i)\,, \tag{2.33}$$

with IC $x_{i,0} = x_i(\mathtt{v}_i, t_0)$. Moreover the protocol u_i and its derivatives $\partial u_i/\partial x_k\,, k = i, j_1 \ldots j_r$, are assumed continuous on $[t_0, t_0 + \tau] \times \mathbf{I}$ with $\mathbf{I} \subseteq \mathbb{R}^{r+1}$. From this it gets clear, that the set of edges $\mathtt{E}$ influences the possible structure of the protocol u_i. If it depends on other vertices, then only adjacent vertices $\mathtt{v}_i \sim \mathtt{v}_j$ are involved and r can be identified as the degree of the vertex $\mathtt{v}_i$, i.e., $r = d[\mathtt{v}_i]$. In general external inputs can be introduced for any vertex, though in practice this devalues the nature of the favoured distributed intelligence of swarms, its autonomous dynamics and the objective of decentralised control algorithms. With this, control theory for linear systems or even non-linear systems can be applied to this sort of problems. Moreover, the study of the underlying graph and the usage of sub-disciplines, such as algebraic and spectral graph theory, enhances analysis and gives access to a more structured approach in terms of eigenvalues, stability and all other associated algebraic objects to graphs [53, 6].

Second, in this work a state $x = x(\boldsymbol{z}, t)$ in a spatially continuous formulation and $t > t_0$ is assumed as a mapping from subsets of an $m-$dimensional, real coordinate space $\mathbb{R}^m$ with Euclidean structure in form of

$$x : [0, \ell_1] \times [0, \ell_2] \times \cdots \times [0, \ell_m] \times [t_0, \infty] \to \mathbf{X}\,, \tag{2.34}$$

whereas $[0, \ell_k]\ , k = 1, 2 \ldots m$ define some finite intervals on $\mathbb{R}$ and $\mathbf{X}$ stands for an appropriate function space or Hilbert space, respectively, e.g., the Lebesgue space $(\mathbf{L}^2, \|\cdot\|_{\mathbf{L}^2})$ or the Sobolev spaces $(\mathbf{H}^k, \|\cdot\|_{\mathbf{H}^k})$. Analogous to above the dynamic behaviour of the state $x(\boldsymbol{z}, t)$ is described by the definition of a dynamical system. In this case one might draw upon a system with some physical meaning, e.g., the DCRS written as

$$\frac{\partial x(\boldsymbol{z}, t)}{\partial t} = \underbrace{\boldsymbol{\nabla} \cdot (a \boldsymbol{\nabla} x(\boldsymbol{z}, t))}_{\text{diffusion}} - \underbrace{\boldsymbol{\nabla} \cdot (\boldsymbol{v} x(\boldsymbol{z}, t))}_{\text{convection}} + \underbrace{r(t, \boldsymbol{z}, x(\boldsymbol{z}, t))}_{\text{reaction}} \tag{2.35}$$

with IC $x(\boldsymbol{z}, t_0) = x_0(\boldsymbol{z})$. The spatial coordinates shall have equal length ℓ. Then (2.35) is defined on $(\boldsymbol{z}, t) \in (0, \ell)^m \times \mathbb{R}^+_{t_0}$, where the notation $(0, \ell) \subset \mathbb{R}$ stands for an open interval in space. Furthermore,

$$\boldsymbol{\nabla} = \left[\frac{\partial}{\partial z_1}, \frac{\partial}{\partial z_2}, \ldots, \frac{\partial}{\partial z_m}\right]^T \tag{2.36}$$

denotes the gradient on the $m-$dimensional Euclidean space and a stands for the diffusion coefficient. The vector $\boldsymbol{v} \in \mathbb{R}^m$ describes the vector field of the convection and the function r expresses a reaction term in a general way. For convenience it is presumed that a and $\boldsymbol{v}$ are independent of $\boldsymbol{z}$ and the reaction term follows the LTV ansatz $r(t, \boldsymbol{z}, , x(\boldsymbol{z}, t)) = r(\boldsymbol{z}, t)x(\boldsymbol{z}, t)$. With this, the introduction of the Laplace-Operator

$$\boldsymbol{\nabla} \cdot \boldsymbol{\nabla} = \nabla^2 = \left(\frac{\partial^2}{\partial {z_1}^2} + \frac{\partial^2}{\partial {z_2}^2} + \cdots + \frac{\partial^2}{\partial {z_m}^2} \right) = \sum_{i=1}^{m} \frac{\partial^2}{\partial {z_i}^2} \tag{2.37}$$

and additionally setting $m = 1$ simplifies (2.35) to the PDE of a 1−dimensional DCRS

$$\frac{\partial x(z,t)}{\partial t} = a \frac{\partial^2 x(z,t)}{\partial z^2} - v \frac{\partial x(z,t)}{\partial z} + r(z,t)x(z,t) \tag{2.38a}$$

for $(z, t) = (0, \ell) \times (t_0, \infty]$ and with the initial state $x(z, t_0) = x_0(z)$. For a unique solution of the PDE it is necessary to define boundary conditions which include equations for the state at $z = 0$ and $z = \ell$. Having in mind that (2.38a) serves as a model for a network it seems natural to preserve the time derivative over the entire spatial domain (cf. Remark 2.5 in Section 2.1.3). In combination with so-called Neumann boundaries it leads to

$$\begin{aligned} \frac{\partial x(0,t)}{\partial t} &= \nu_0 \frac{\partial x(0,t)}{\partial z} \,, \\ \frac{\partial x(\ell,t)}{\partial t} &= -\nu_\ell \frac{\partial x(\ell,t)}{\partial z} \,, \end{aligned} \tag{2.38b}$$

with some arbitrary parameters ν_0, $\nu_\ell \in \mathbb{R}$.

Finally, comparing the set-up of (2.32) and (2.33) with the problem formulation of (2.34) and (2.38) reveals the following picture. Obviously any attempt to reconcile these two representations demands to put attention to the domain of definition regarding the state mappings (2.32) and (2.34), and as a consequence it requires the focus on graph-theoretical (differential) operators, e.g. Laplacian control (2.22), as well as linear partial differential operators of DPSs. Neglecting derivatives of the time domain and concentrating on the spatial domain, the latter can be formally defined as

$$D^{(\boldsymbol{\alpha})}(x)(\boldsymbol{z}, t) := \sum_{|\boldsymbol{\alpha}| \leq l} a_{\boldsymbol{\alpha}}(\boldsymbol{z}, t) \frac{\partial^{\boldsymbol{\alpha}} x(\boldsymbol{z}, t)}{\partial \boldsymbol{z}^{\boldsymbol{\alpha}}} \tag{2.39}$$

with the order l of the operator, the vector $\boldsymbol{z}$ defined on the open domain $\boldsymbol{z} \in \mathcal{Z} \subset \mathbb{R}^m$, and the multi-index $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_m)$. The functions $a_{\boldsymbol{\alpha}}(\boldsymbol{z}, t)$ are assumed to be continuous in space and time. The length of the multi-index is given by $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2 + \cdots + \alpha_m$.

Remark 2.6. *In general (partial) differential operators, such as gradients, divergence, or curl can be generalised and naturally extended to formulations independent from any coordinate system. This leads to definitions on differentiable manifolds or even more generalised mathematical objects [43]. Although this is not an objective for the discussion in this thesis, it could be an interesting approach to study the relationship between differentiable manifolds and graphs, since the latter are constructed from a set-theoretical angle only. This idea is supported by the fact that in the vast majority the spatial domains are not linked to any physical meaning when continuous models get associated with MASs. In this work the formulation of differential operators for the spatial domain requires at least an Euclidean space of dimension m. However, for simplicity but without loss of generality it shall be set to $m = 1$ for the following.*

In the same manner (differential) operators for discrete configurations can be embedded in an equivalent formalism expressed as

$$\mathcal{D}(x)(\mathtt{v}_i,t) := \sum_{\mathtt{v}_i \sim \mathtt{v}_j} \mathtt{a}(\mathtt{v}_j,t)\mathtt{d}(x(\mathtt{v}_i,t),x(\mathtt{v}_j,t)) \tag{2.40}$$

with $\mathtt{v}_i \in \mathtt{V}(\mathcal{G}_\mathrm{s})$ and $\mathtt{v}_j \in \mathtt{V}(\overline{\mathcal{G}_\mathrm{s}})$ as a vertex within the closure of the underlying subgraph $\mathcal{G}_\mathrm{s}$ and the smooth functions $\mathtt{d} \in C^0(\mathbb{R}^2)$. The coefficients $\mathtt{a}(\mathtt{v}_j,t)$ stand for continuous functions in time for every adjacent vertex $\mathtt{v}_j$. Obviously $\mathtt{v}_i$ and consequently $\mathtt{v}_j$ must have degree greater than zero, i.e., $d(\mathtt{v}_i) > 0\,, d(\mathtt{v}_j) > 0$ with (2.3) and (2.4). Thus both are non-isolated vertices and consequently isolated vertices are ignored. This explains the definition of the operator on states assigned to a subgraph $\mathcal{G}_\mathrm{s}$ of a given host graph $\mathcal{G}$. It is comparable to the definition of differential operators for sufficiently smooth functions on open intervals as in (2.39).

As hinted at the beginning of this section there are stated strategies to transit between the formulations [71, 55]. On the one hand the proposal arises to tend the number of agents to infinity in order to converge from a discrete formulation to a continuous model. On the other hand, obviously appropriate approximation methods applied on PDEs render a system of ODEs of finite order. The following two paragraphs shall discuss these strategies in more detail and extend or adjust them if appropriate. The goal is to close some gaps between the representations by defining missing preliminaries and adding or modifying conditions and methods to these strategies . In this context going from continuous to discrete seems trivial. Defining rules for extracting a finite set $\mathtt{V}$ from $[0,\ell]^m$ and the approximation of operators which induces neighbourhood and adjacencies seems a promising path without too many barriers. Therefore in the following clause the discussion advances with semi-discretisation methods of PDEs.

2.2.2 Semi-discretisation – Transition from Continuum to Discrete Models

As discussed above, in this thesis models of distributed parameters systems are mainly used to design controllers based on PDEs and then apply them to a swarm with a finite number of agents by appropriate approximation of the differential operators on the spatial domain. Typically discretisation methods are used to compute numerical solutions of PDEs. In this context it is distinguished between semi- and full-discretisation. While the former approximates the spatial domain only, the latter applies a discrete estimation scheme to time derivatives as well. Among others, widespread discretisation methods are the following [35].

- *Finite element method (FEM)*: The method splits the domain into a finite number of parts of simpler structure, called elements. This renders a system of equations which are easier to handle than the original, infinite dimensional formulation. The solution of these equations approximates the solution of the initial problem.

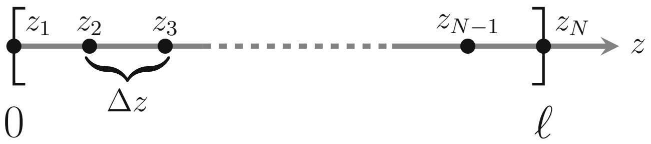


Figure 2.5: Equidistant discretisation points for the FDM.

- *Finite volume method (FVM)*: Here the domain is subdivided into a finite number of small cells or volumes. Methods of first order derive equations which allow to compute the temporal evolution of mean values within the cells.
- *Finite difference method (FDM)*: The main purpose of this method is the approximation of derivatives by means of finite differences. For this the discretisation of a function is performed mostly at equidistant points within the underlying domain. It is distinguished between the so-called central FDM and forward or backward FDM, respectively.

In this work only the FDM is investigated since from a graph-theoretical point of view agents are referred to vertices within the network model. This fits to the nature of the FDM since it deals with points within the spatial domain. Besides that, the domain has no real physical meaning when it comes to continuous models for swarms or MASs.

Therefore, suppose the DCRS introduced in (2.38) shall be approximated on N points, each point representing a vertex of an induced MAS. Then the FDM performed for the spatial domain $[0,\ell] \subset \mathbb{R}$ leads to the system of ODEs

$$\begin{aligned}\frac{\partial x(z_i,t)}{\partial t} &= a\frac{x(z_{i+1},t)-2x(z_i,t)+x(z_{i-1},t)}{\Delta z^2} - v\frac{x(z_{i+1},t)-x(z_{i-1},t)}{2\Delta z}\\ &\quad + r(z_i,t)x(z_i,t)\,, \quad i=2,3\ldots,\mathrm{N}-1\\ \frac{\partial x(z_1,t)}{\partial t} &= \nu_0\frac{x(z_2,t)-x(z_1,t)}{\Delta z}\,,\\ \frac{\partial x(z_\mathrm{N},t)}{\partial t} &= -\nu_\ell\frac{x(z_\mathrm{N},t)-x(z_{\mathrm{N}-1},t)}{\Delta z}\,,\end{aligned} \tag{2.41}$$

with the equidistant discretisation points z_i and the discretisation step size Δz defined as $\Delta z = \ell/(\mathrm{N}-1) > 0$. The structure of the domain is briefly sketched in Figure 2.5. With this, the infinite dimensional DCRS defined on $[0,\ell]\times\mathbb{R}^+_{t_0}$ with $\mathbb{R}^+_{t_0} = \{t \in \mathbb{R} : t > t_0\}$ is approximated by a system of ODEs of order N at the points

$$\mathrm{Z} = \{z_i \,|\, z_i = (i-1)\Delta z \,\wedge\, i = 1,\ldots,\mathrm{N}\} \subset [0,\ell] \subset \mathbb{R}\,. \tag{2.42}$$

Basically this set represents coordinates of a coordinate system which models the 1-dimensional Euclidean space $\mathbb{R}$. Recalling Laplacian control it is obvious that (2.41) with the particular parameter setting $a/\Delta z^2 = 1$ and $v = r(\mathtt{v}_i,t) = 0$ emulates the structure of (2.25) and (2.26).

From this it gets essential that semi-discretised distributed parameter systems, e.g., as given in (2.41), can only be applied to a given host graph $\mathcal{G}(\mathtt{V},\mathtt{E})$ when each coordinate $z_i \in \mathrm{Z}$ can get associated in an appropriate way with a vertex $\mathtt{v}_k \in \mathtt{S}$ of a subgraph $\mathcal{G}_\mathrm{s}(\mathtt{S},\mathtt{E}_\mathrm{s}) \subseteq \mathcal{G}$. Moreover, the edge set $\mathtt{E}_\mathrm{s}$ has to resemble the adjacencies induced by the FDM. These two conditions motivate the formal mapping

$$\phi : \mathrm{Z} \to \mathtt{S}(\mathcal{G}_\mathrm{s}) \qquad \phi(z_i) = \mathtt{v}_k \qquad\qquad i \in \{1,2,\dots\mathrm{N}\}\,,\, k \in \{1,2,\dots|\mathtt{S}|\}. \tag{2.43}$$

At the first sight this function looks trivial, though it is necessary since the coordinates $z_i \in \mathrm{Z}$ and the vertices $\mathtt{v}_k \in \mathtt{S}$ have completely different algebraic features. For instance the handy abbreviation $z_i = (i-1)\Delta z$ yields the recursion

$$z_i = z_{i-1} + \Delta z\,, \tag{2.44}$$

though summation as an arithmetic operation does not make sense for vertices which are abstract objects but not numbers. If the function (2.43) can be constructed to be bijective, which can be accomplished for a spanning subgraph $\mathcal{G}_\mathrm{s}$ with $\mathrm{N} = |\mathtt{V}| = |\mathtt{S}| < \infty$ by all means, then there exists a mapping from the vertex set $\mathtt{S}(\mathcal{G}_\mathrm{s})$ to the state space of x in equivalent fashion as stated in (2.32). From this follows

$$x(z_i,t) = x(\phi^{-1}(\mathtt{v}_k),t) = \check{x}(\mathtt{v}_k,t) \quad \text{and} \quad x(\mathtt{v}_k,t) := \check{x}(\mathtt{v}_k,t)\,. \tag{2.45}$$

For convenience the symbol for the state variable shall be preserved. Additionally to bijectivity ϕ needs to satisfy

$$\{\mathtt{v}_j,\mathtt{v}_k\} \quad \Leftrightarrow \quad z_i = \phi^{-1}(\mathtt{v}_j) \wedge z_{i+1} = \phi^{-1}(\mathtt{v}_k)\,, \quad \forall i = 1,2,\cdots,\mathrm{N}-1\,. \tag{2.46}$$

This basically ensures that neighbouring coordinates in Z are mapped to adjacent vertices of the spanning subgraph $\mathcal{G}_\mathrm{s}(\mathtt{N},\mathtt{E}_\mathrm{s})$. Generally speaking, this induces a edge set of a path graph. With $j = i$ and $k = i+1$ (2.46) can be rewritten as

$$\begin{aligned}
\frac{\partial x(\mathtt{v}_i,t)}{\partial t} &= \tilde{a}(\mathrm{N},\ell)\left[x(\mathtt{v}_{i+1},t) - 2x(\mathtt{v}_i,t) + x(\mathtt{v}_{i-1},t)\right] - \tilde{v}(\mathrm{N},\ell)\left[x(\mathtt{v}_{i+1},t) - x(\mathtt{v}_{i-1},t)\right]/2 \\
&\quad + r(\mathtt{v}_i,t)x(\mathtt{v}_i,t)\,, \\
\frac{\partial x(\mathtt{v}_1,t)}{\partial t} &= \tilde{\nu}_0(\mathrm{N},\ell)\left[x(\mathtt{v}_2,t) - x(\mathtt{v}_1,t)\right]\,, \\
\frac{\partial x(\mathtt{v}_\mathrm{N},t)}{\partial t} &= -\tilde{\nu}_\ell(\mathrm{N},\ell)\left[x(\mathtt{v}_\mathrm{N},t) - x(\mathtt{v}_{\mathrm{N}-1},t)\right]\,,
\end{aligned} \tag{2.47}$$

with the abbreviations

$$\tilde{a}(\mathrm{N},\ell) = a\frac{(\mathrm{N}-1)^2}{\ell^2}\,, \quad \tilde{v}(\mathrm{N},\ell) = v\frac{\mathrm{N}-1}{\ell}\,, \quad \text{and} \quad \tilde{\nu}_{\{0,\ell\}}(\mathrm{N},\ell) = \nu_{\{0,\ell\}}\frac{\mathrm{N}-1}{\ell}\,. \tag{2.48}$$

Comparing the semi-discretised equations (2.47) with the configuration in (2.25) and (2.26) it can be concluded that the central finite difference method for the second derivative applies the same scheme as Laplacian control does for a path graph. Though, a main remaining difference is that the leading coefficients in (2.47) depend on N as well as the length ℓ of the spatial domain of the underlying PDE. For this reason one may request a constant discretisation step size Δz, which is independent of the number of discretisation points N or number of agents, respectively. Following this path the request for the condition $\Delta z \overset{!}{=} 1$ implies

$$\ell = \ell_{\mathrm{N}} = \mathrm{N} - 1\,. \tag{2.49}$$

Therefore, taking (2.49) into account the length of the spatial domain varies with N instead of the discretisation step size Δz. Note, in case of a finite number of agents the length ℓ_{N} is still finite. Then the pattern simplifies to (2.47) to

$$\begin{aligned}
\frac{\partial x(\mathtt{v}_i,t)}{\partial t} &= a\left[x(\mathtt{v}_{i+1},t) - 2x(\mathtt{v}_i,t) + x(\mathtt{v}_{i-1},t)\right] - v\left[x(\mathtt{v}_{i+1},t) - x(\mathtt{v}_{i-1},t)\right]/2 \\
&\quad + r(\mathtt{v}_i,t)x(\mathtt{v}_i,t)\,, \\
\frac{\partial x(\mathtt{v}_1,t)}{\partial t} &= \nu_0\left[x(\mathtt{v}_2,t) - x(\mathtt{v}_1,t)\right]\,, \\
\frac{\partial x(\mathtt{v}_{\mathrm{N}},t)}{\partial t} &= -\nu_{\ell_{\mathrm{N}}}\left[x(\mathtt{v}_{\mathrm{N}},t) - x(\mathtt{v}_{\mathrm{N}-1},t)\right]\,.
\end{aligned} \tag{2.50}$$

From the discussion above it can be concluded that this 2-step process, FDM together with an appropriate mapping of the form (2.43), applied to the distributed parameter system (2.38) renders an input protocol. For instance the right hand side of (2.50) forms a protocol in the sense of (2.33) for the state variable $x(\mathtt{v}_i,t)$ which, e.g., can be assigned to an agent of a MAS. Since the number of states applied to an agent is irrelevant to the discretisation and mapping process this concept can be extended to DPS governed by coupled PDE. Consequently, note the following theorem which covers a whole family of DCRSs.

Theorem 2.1. *Consider the coupled diffusion-convection-reaction system*

$$\frac{\partial \boldsymbol{x}(z,t)}{\partial t} = A(z,t)\frac{\partial^2 \boldsymbol{x}(z,t)}{\partial z^2} - B(z,t)\frac{\partial \boldsymbol{x}(z,t)}{\partial z} + C(z,t)\boldsymbol{x}(z,t)\,, \tag{2.51a}$$

defined on the domain $(z,t) \in \mathfrak{D}_{\mathrm{N},t_0} = \left\{(0,\ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0}\right\}$ *and for the state vector* $\boldsymbol{x}(z,t) \in \mathbf{C}^{2,2}\left(\mathfrak{D}_{\mathrm{N},t_0};\mathbb{R}^n\right)$ *with the coefficient matrices* $\{A(z,t),B(z,t),C(z,t)\} \in \mathbf{C}^{0,0}\left(\mathfrak{D}_{\mathrm{N},t_0};\mathbb{R}^{n\times n}\right)$*. Furthermore, consider the boundary conditions*

$$\begin{aligned}
\frac{\partial \boldsymbol{x}(0,t)}{\partial t} &= P_0(t)\frac{\partial \boldsymbol{x}(0,t)}{\partial z} + Q_0(t)\boldsymbol{x}(0,t)\,, \\
\frac{\partial \boldsymbol{x}(\ell_{\mathrm{N}},t)}{\partial t} &= -P_{\ell_{\mathrm{N}}}(t)\frac{\partial \boldsymbol{x}(\ell_{\mathrm{N}},t)}{\partial z} + Q_{\ell_{\mathrm{N}}}(t)\boldsymbol{x}(0,t)\,,
\end{aligned} \tag{2.51b}$$

with $\{P_0(t),P_{\ell_{\mathrm{N}}}(t),Q_0(t),Q_{\ell_{\mathrm{N}}}(t)\} \in \mathbf{C}^0\left(\mathbb{R}^+_{t_0};\mathbb{R}^{n\times n}\right)$ *and the initial state* $\boldsymbol{x}(z,t_0) = \boldsymbol{x}_0(z)$*. Then*

- *the central finite difference method (FDM) with the order of accuracy of* $\mathcal{O}\left(\Delta z^2\right)$ *for the differential operators* $\frac{\partial^\alpha}{\partial z^\alpha}, \alpha \in \{1,2\}$ *on the open interval* $(0,\ell_{\mathrm{N}})$*, and*

- *the forward and backward finite difference method (FDM) with the order of accuracy of* $\mathcal{O}(\Delta z)$ *for the boundary conditions at* $z = 0$ *and* $z = \ell_{\mathrm{N}}$*;*

applied to the given system (2.51) *at the discretisation points* $z_i \in \mathrm{Z}$ *according to* (2.42) *and* $\Delta z := 1$*, i.e.,* $\ell_{\mathrm{N}} = \mathrm{N} - 1$*, together with the bijective mapping*

$$\phi : \mathrm{Z} \to \mathtt{V}(\mathcal{P}_{\mathrm{N}}) \qquad \Rightarrow \qquad \phi(z_i) = \mathtt{v}_i \qquad\qquad i \in \{1, 2, \dots |\mathtt{V}|\}\,, \tag{2.52}$$

render an input protocol of the form $\boldsymbol{u}(\mathtt{v}_i, t, \boldsymbol{x}(\mathtt{v}_{i-1}, t), \boldsymbol{x}(\mathtt{v}_i, t), \boldsymbol{x}(\mathtt{v}_{i+1}, t))$*, which is built upon the underlying path graph* $\mathcal{P}_{\mathrm{N}}(\mathtt{V}, \mathtt{E}_{\mathcal{P}})$ *with* $|\mathtt{V}| = \mathrm{N}$*. Thus, it holds*

$$z_i = \phi^{-1}(\mathtt{v}_i) \qquad \textit{and} \qquad \mathtt{e}_{i,i+1} = \{\mathtt{v}_i, \mathtt{v}_{i+1}\} \in \mathtt{E}_{\mathcal{P}}\,, \tag{2.53}$$

for $i = 1, 2, \dots, \mathrm{N} - 1$*.*

Note, obviously there exists an isomorphism between the path graph $\mathcal{P}_{\mathrm{N}}$ in (2.52) and the spanning subgraph $\mathcal{G}_{\mathrm{s}}$ in (2.43), (cf. Definition 2.1). This allows to set $k = i$ in (2.46) and subsequently motivates (2.53). Therefore, a comprehensive proof of Theorem 2.1 is omitted since it follows in principle in an identical way the exposition before. Basically, the theorem presumes that the stated model (2.51) is defined spatially on the real interval $[0, \ell_{\mathrm{N}}]$. For this, it explains the structure of the resulting graph through FDM discretisation. In detail, the function (2.52) defines the set of vertices $\mathtt{V}$ and FDM the structure of the set of edges $\mathtt{E}_{\mathcal{P}}$. However, sometimes it is the case that a graph is given and one might ask if a sequence of discretised state variables with an associated protocol can be assigned to the given host graph. So when the perspective is inverted, i.e. to originate from a given graph, the following corollary can be derived for DCRSs.

Theorem 2.2. *Let* $\mathcal{G}(\mathtt{V}, \mathtt{E})$ *be a given host graph and* $\boldsymbol{x}$ *a state vector which origins from a diffusion-convection-reaction system of the form* (2.51)*. Moreover, consider a subsequent discretisation of* $\boldsymbol{x}$ *by means of an appropriate FDM which renders an input protocol* $\boldsymbol{u}$ *respecting* (2.53)*. Then the state* $\boldsymbol{x}(\mathtt{v}_i, t)$ *can be assigned to the vertices* $\mathtt{v}_i \in \mathtt{V}$ *of a spanning subgraph* $\mathcal{P}(\mathtt{V}, \mathtt{E}_{\mathcal{P}}) \subseteq \mathcal{G}(\mathtt{V}, \mathtt{E})$*, if and only if* $\mathcal{G}$ *is traceable. The subgraph involved is a Hamiltonian path* $\mathcal{P}_{\mathrm{H}}(\mathtt{V}, \mathtt{E}_{\mathcal{P}})$ *of the host graph* $\mathcal{G}$*.*

Proof. The theorem is proven by contradiction. Therefore, assume $\mathcal{G}$ is not traceable. The application of the input protocol $\boldsymbol{u}$ which respects (2.53), demands that there exists at least one isomorphic spanning subgraph $\mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}}) \subseteq \mathcal{G}(\mathtt{V}, \mathtt{E})$ where every vertex $\mathtt{v}_i \in \mathtt{V}$, $i = 1, \dots, \mathrm{N} - 1$ is adjacent to $\mathtt{v}_{i+1}$. However, then $\mathcal{G}_{\mathrm{s}}(\mathtt{S}, \mathtt{E}_{\mathrm{s}})$ contains a spanning path which stands in contradiction to the assumption that the host graph $\mathcal{G}$ is not traceable. □

Influence of the Length of the Domain on the Eigenvalues

The correlation of the length of the spatial domain with the number of agents N makes it necessary to investigate its impact on the continuous formulation and compare it with the discrete set-up. In the following the focus is on both, the properties of the swarm dynamics as well as the dynamics of the single agent. Recalling (2.38) and the convenient assumptions $v = 0$, $r(\mathtt{v}_i, t) = r$ and $\nu = \nu_0 = \nu_{\ell_\mathrm{N}}$ allows the compact notation

$$\frac{\partial \boldsymbol{x}}{\partial t} = \mathcal{A}\boldsymbol{x} \tag{2.54a}$$

with the linear, time-invariant (LTI) operator

$$\mathcal{A} = \begin{bmatrix} a\frac{\partial^2}{\partial z^2} + r & 0 & 0 \\ 0 & \nu\frac{\partial}{\partial z} & 0 \\ 0 & 0 & -\nu\frac{\partial}{\partial z} \end{bmatrix}, \text{ and the state } \boldsymbol{x}(z,t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x(z,t) \\ x(0,t) \\ x(\ell_\mathrm{N},t) \end{bmatrix}. \tag{2.54b}$$

Using the state space $\mathbf{X} = \mathbf{L}^2(0, \ell_\mathrm{N}) \times \mathbb{R}^2$ equipped with the inner product

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle_\mathbf{X} = \langle f_1, g_1 \rangle_{\mathbf{L}^2} + f_2^T g_2 + f_3^T f_3 \tag{2.55}$$

then the domain of definition of the operator $\mathcal{A}$ can be written as

$$\mathfrak{D}(\mathcal{A}) = \left\{ \boldsymbol{x} \in \mathbf{X} \,|\, \partial_z^2 x_1 \in \mathbf{L}^2(0, \ell_\mathrm{N}),\, x_1, \partial_z \boldsymbol{x} \text{ abs. cont.},\, x_1(0) = x_2, x_1(\ell_\mathrm{N}) = x_3 \right\} \tag{2.56}$$

With this, the eigenvalue problem $(\lambda E - \mathcal{A})\boldsymbol{x}(z,t) = \mathbf{0}_3$ gives the set of equations

$$a\frac{\partial^2 x_1}{\partial z^2} + r x_1 = \lambda x_1 \quad \Rightarrow \quad \frac{\partial^2 x_1}{\partial z^2} = \frac{\lambda - r}{a} x_1 = -\mu^2 x_1\,, \tag{2.57}$$

$$\begin{aligned} \nu\frac{\partial x_2}{\partial z} &= \lambda x_2 \quad \Rightarrow \quad \frac{\partial x_2}{\partial z} = \frac{\lambda}{\nu} x_2 = -\frac{a\mu^2 - r}{\nu} x_2\,, \\ -\nu\frac{\partial x_3}{\partial z} &= \lambda x_3 \quad \Rightarrow \quad \frac{\partial x_3}{\partial z} = -\frac{\lambda}{\nu} x_3 = \frac{a\mu^2 - r}{\nu} x_3\,, \text{ and} \end{aligned} \tag{2.58}$$

$x_1(0) = x_2, x_1(\ell_\mathrm{N}) = x_3$, with the eigenvalues

$$\lambda = -a\mu^2 + r \tag{2.59}$$

to be determined. The first condition (2.57) for x_1 is obviously fulfilled by trigonometric functions and their derivatives according to

$$\begin{aligned} x_1 &= A\cos(\mu z) + B\sin(\mu z) \\ \frac{\partial x_1}{\partial z} &= \mu(-A\sin(\mu z) + B\cos(\mu z)) \\ \frac{\partial^2 x_1}{\partial z^2} &= -\mu^2(A\cos(\mu z) + B\sin(\mu z)) = -\mu^2 x_1\,. \end{aligned} \tag{2.60}$$

With this, the other two equations in (2.58) provide the conditions

$$\left.\begin{aligned} x_2 &= A \\ \frac{\partial x_2}{\partial z} &= \mu B \end{aligned}\right\} \quad \Rightarrow \quad \mu B + \frac{a\mu^2 - r}{\nu} A = 0$$

$$\left.\begin{aligned} x_3 &= A\cos(\mu\ell_\mathrm{N}) + B\sin(\mu\ell_\mathrm{N}) \\ \frac{\partial x_3}{\partial z} &= -\mu A\sin(\mu\ell_\mathrm{N}) + \mu B\cos(\mu\ell_\mathrm{N}) \end{aligned}\right\} \quad \Rightarrow$$

$$\left(\mu A + B\frac{a\mu^2 - r}{\nu}\right)\sin(\mu\ell_\mathrm{N}) + \left(A\frac{a\mu^2 - r}{\nu} - \mu B\right)\cos(\mu\ell_\mathrm{N}) = 0 \tag{2.61}$$

which can be written in matrix form as

$$\underbrace{\begin{bmatrix} \frac{a\mu^2-r}{\nu} & \mu \\ \frac{a\mu^2-r}{\nu}\cos(\mu\ell_\mathrm{N})+\mu\sin(\mu\ell_\mathrm{N}) & \frac{a\mu^2-r}{\nu}\sin(\mu\ell_\mathrm{N})-\mu\cos(\mu\ell_\mathrm{N}) \end{bmatrix}}_{M(\mu,\ell_\mathrm{N})} \begin{bmatrix} A \\ B \end{bmatrix} = \mathbf{0} \tag{2.62}$$

The existence of a non-trivial solution for A and B requires $\det(M(\mu,\ell_\mathrm{N}))=0$. This gives

$$\left(\frac{a\mu^2-r}{\nu}\right)^2\sin(\mu\ell_\mathrm{N})-2\mu\frac{a\mu^2-r}{\nu}\cos(\mu\ell_\mathrm{N})-\mu^2\sin(\mu\ell_\mathrm{N})\overset{!}{=}0 \tag{2.63}$$

and consequently

$$\tan(\mu\ell_\mathrm{N})=2\mu\nu\frac{a\mu^2-r}{(a\mu^2-r)^2-\mu^2\nu^2}=h(\mu) \tag{2.64}$$

for $\cos(\mu\ell_\mathrm{N})\neq 0$. Further arrangement yields the non-linear characteristic equation

$$\tan(\mu_k\ell_\mathrm{N})=2\frac{\nu}{a}\frac{\mu_k-\frac{r}{a\mu_k}}{(\mu_k-\frac{r}{a\mu_k})^2-\frac{\nu^2}{a^2}} \tag{2.65}$$

for μ_k, $k\in\mathbb{N}_0$, in dependence of the parameter setting a,ν, and r and the length of the domain ℓ_N.

This allows to compare the distribution of the eigenvalues for Laplacian control as defined in (2.22) and applied on a path graph with the eigenvalues of (2.54). More precisely, on the one hand, the investigated Laplacian control multiplied with the factor a gives

$$\begin{aligned} \frac{\partial \boldsymbol{x}(\mathrm{v}_i,t)}{\partial t} &= a(\boldsymbol{x}(\mathrm{v}_{i+1},t)-2\boldsymbol{x}(\mathrm{v}_i,t)+\boldsymbol{x}(\mathrm{v}_{i-1},t))\,, \quad \text{for} \quad i=2,\dots,\mathrm{N}-1\,,\\ \frac{\partial \boldsymbol{x}(\mathrm{v}_1,t)}{\partial t} &= a(\boldsymbol{x}(\mathrm{v}_2,t)-\boldsymbol{x}(\mathrm{v}_1,t))\,, \quad \text{and}\\ \frac{\partial \boldsymbol{x}(\mathrm{v}_\mathrm{N},t)}{\partial t} &= -a(\boldsymbol{x}(\mathrm{v}_\mathrm{N},t)-\boldsymbol{x}(\mathrm{v}_{\mathrm{N}-1},t))\,. \end{aligned} \tag{2.66a}$$

The eigenvalues can be computed as

$$\lambda_k^{\mathcal{L}}=2a(\cos(k\pi/\mathrm{N})-1)\,, \qquad k=0,1,\dots,\mathrm{N}-1\,. \tag{2.66b}$$

At this point the interested reader is referred to [53] and [15] where the derivation of eigenvalues for special graphs is discussed in detail. Then on the other hand, the special choice $a=\nu$ and $r=0$ simplifies (2.54) and subsequently (2.65), which results in a simple form of a diffusion system as

$$\begin{aligned} \frac{\partial x(z,t)}{\partial t} &= a\frac{\partial^2 x(z,t)}{\partial z^2}\\ \frac{\partial x(0,t)}{\partial t} &= a\frac{\partial x(0,t)}{\partial z}\,,\\ \frac{\partial x(\ell_\mathrm{N},t)}{\partial t} &= -a\frac{\partial x(\ell_\mathrm{N},t)}{\partial z}\,, \end{aligned} \tag{2.67a}$$

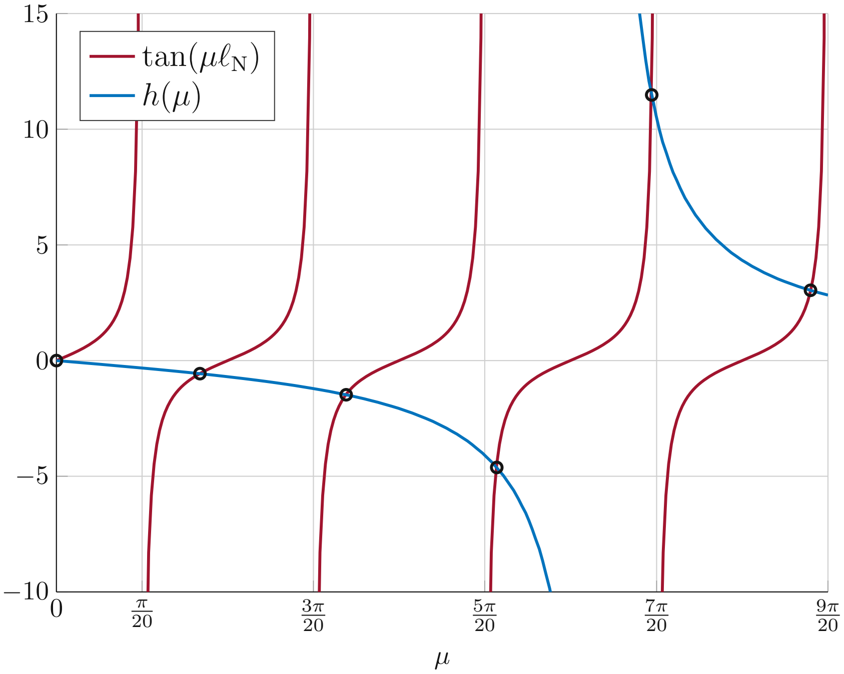


Figure 2.6: Graphical interpretation of the non-linear conditional equation (2.67b) with $\ell_\mathrm{N} = 10$ and $a = 1$. The intersections are possible solutions μ_k.

and its appertaining eigenvalues

$$\lambda_k^{\mathrm{DS}} = -a\mu_k^2\,, \qquad \text{with } \tan(\mu_k\ell_\mathrm{N}) = 2\frac{\mu_k}{\mu_k^2-1},\ k\in\mathbb{N}_0,\ \text{and}\ \ell_\mathrm{N} = \mathrm{N}-1. \tag{2.67b}$$

The two equations (2.66b) and (2.67b) provide $\lambda_0^{\mathcal{L},\mathrm{DS}} = 0$ and $\lambda_k^{\mathcal{L},\mathrm{DS}} < 0\,, k \in \mathbb{N}$. A graphical representation of the non-linear condition in (2.67b) is presented in Figure 2.6 for $\ell_\mathrm{N} = 10$ and $a = 1$. By means of the illustration it can be concluded that the largest eigenvalue, i.e., λ_1^{DS} tends to zero with increasing ℓ_N. This motivates the linearisation of the equations. On the one hand, the non-linear equation (2.67b) and subsequently λ_1^{DS} can be estimated by

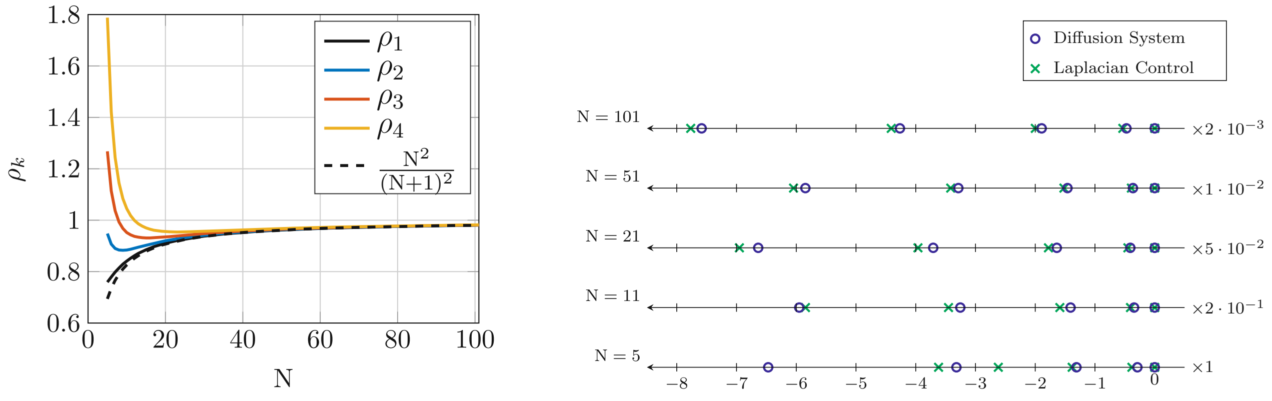
$$\mu_1\ell_\mathrm{N} - \pi \approx -2\mu_1 \quad\Rightarrow\quad \lambda_1^{\mathrm{DS}} \approx -a\pi^2/(\mathrm{N}+1)^2\,, \tag{2.68}$$

whereas $\tan(\mu\ell_\mathrm{N})$ is linearised around $\mu_1 = \pi/\ell_\mathrm{N}$ and the rational function[2] h around $\mu_1 = 0$. On the other hand, (2.66b) can be approximated around 0 by the quadratic term

$$\lambda_1^{\mathcal{L}} = 2a\left(\cos(\pi/\mathrm{N}) - 1\right) \approx -a\pi^2/\mathrm{N}^2\,, \tag{2.69}$$

This approximation confirms some properties of the diffusion system (DS) and allows further analysis.

[2] The linearisation of h around $\mu_1 = 0$ allows a better analytic comparison between λ_1^{DS} and $\lambda_1^{\mathcal{L}}$ than the approximation at $\mu_1 = \pi/\ell_\mathrm{N}$. Moreover, the developments around 0 and π/ℓ_N approach each other for increasing ℓ_N.



(a) Ratios $\rho_k = \lambda_k^{\mathcal{L}}/\lambda_k^{\mathrm{DS}}$ of the eigenvalues.

(b) Absolute comparison of the eigenvalues $\lambda_k^{\mathcal{L}}$ and λ_k^{DS}.

Figure 2.7: Comparisons of the distribution of the biggest five eigenvalues between the diffusion system (2.67a) and the operator (2.21) applied on a path graph depending on the number of vertices N.

- First, diffusion or Laplacian control have a zero eigenvalue and are inherently stable. The steady state is defined by their initial condition, i.e., $\bar{x}(z) = \int_0^{\ell_\mathrm{N}} x(z, t_0)\mathrm{d}z/\ell_\mathrm{N}$ for the diffusion system and $\bar{x}(\mathtt{v}_i) = \sum_{i=1}^{\mathrm{N}} x(\mathtt{v}_i, t_0)/\mathrm{N}$ for Laplacian control.

- The stability properties are characterised by the largest eigenvalue $\lambda_1^{\mathcal{L}}$, $\lambda_1^{\mathrm{DS}} < 0$. The stability properties may be further adjusted by the coefficient a.

- However, the swarm dynamics for both, the discrete and continuous formulation, slows down proportionally with $\sim 1/\mathrm{N}^2$. Note, for the limit $\mathrm{N} \to \infty$ all eigenvalues $\lambda_k^{\mathcal{L}}$, λ_k^{DS} tend to zero. From a practical point of view this behaviour is obvious since the limit to infinity means infinite participants for the MAS and an infinite spatial domain for the DS, respectively.

- In contradiction to that the agent dynamics is invariant under the discretisation process since the discretisation step is set constant $\Delta z = 1$, i.e., the agent dynamics is independent of N.

Note, Figure 2.7a show the ratios $\rho_k = \lambda_k^{\mathcal{L}}/\lambda_k^{\mathrm{DS}}$, $k = 1, \ldots, k$ over the number of agents N. Apart from the points mentioned above two further aspects can be concluded from the presented plots. First, ρ_1 converges to the approximated ratio $\mathrm{N}^2/(\mathrm{N}+1)^2$ governed by (2.68) and (2.69), respectively, with increasing N. Second, the ratio ρ_1 of the largest eigenvalues, which define the overall dynamics of each system, approaches 1 from below. However the ratio of the higher modes alter in this matter. Subsequently this means that the fastest eigenmode of the discrete formulation is always faster than the fastest eigenmode of the continuous formulation. A plausible reason for this behaviour can give the discretisation scheme of the FDM, which renders the sequence of Laplacian control of an undirected, unweighted path graph. In the sense of the FDM the discretisation pattern is an approximation for the differential operator in (2.67a) of second order of accuracy but the approximation of the boundary conditions is of first error order. Thus, the complete approximation must be considered of first error order only which is lower than order of the spatial derivative of the DS. This results in a slightly faster first eigenmode for the discrete system than for the corresponding continuous formulation, but simultaneously to a more volatile behaviour at the boundaries.

An overview of the discretisation pattern for different orders of accuracy are given in Table A.1 for central FDM and in Table A.2 for forward and backward FDM. Moreover, Figure 2.7b gives an appropriate comparison between the five biggest eigenvalues λ_k^{DS} and $\lambda_k^{\mathcal{L}}$, $k = \{0, 1, \dots, 4\}$ for a various number of agents N. It shows that the continuous formulation (2.67a), with choice of only 11 agents, already gives a reasonable dynamic representation of the operator (2.21) when it is applied to an undirected, unweighted path graph. With this, the following proposition shall summarize the results of the discussion above.

Proposition 2.3. *Assume the differential operator $\mathcal{A}$ of a DS of the form*

$$\mathcal{A} = \begin{bmatrix} \frac{\partial^2}{\partial z^2} & 0 & 0 \\ 0 & \frac{\partial}{\partial z} & 0 \\ 0 & 0 & -\frac{\partial}{\partial z} \end{bmatrix} \quad \textit{with the state vector } \boldsymbol{x} = \begin{bmatrix} x(z,t) \\ x(0,t) \\ x(\ell_{\mathrm{N}},t) \end{bmatrix}, \tag{2.70}$$

and its domain of definition $\mathfrak{D}(\mathcal{A}) = \{\boldsymbol{x} \in \mathbf{X} \,|\, \partial_z^2 x_1 \in \mathbf{L}^2(0, \ell_{\mathrm{N}}),\, x_1, \partial_z \boldsymbol{x}$ abs. cont., $x_1(0) = x_2$, $x_1(\ell_{\mathrm{N}}) = x_3\}$ with $\ell_{\mathrm{N}} = \mathrm{N} - 1$. Then the FDM with properties according to Theorem 2.1 and an appropriate function ϕ equivalent to (2.52) *transforms $\mathcal{A}\boldsymbol{x}$ into the discrete operator*

$$\mathcal{L}(\boldsymbol{x}) := -L(\mathcal{P})\boldsymbol{x} \tag{2.71}$$

with the state vector $\boldsymbol{x} = [x_1(t), x_2(t), \dots, x_{\mathrm{N}}(t)]^T$ and where $L(\mathcal{P})$ stands for the graph Laplacian matrix (2.14) *of an underlying simple, undirected, and unweighted path graph $\mathcal{P}(\mathtt{V}, \mathtt{E}_{\mathcal{P}})$ with $|\mathtt{V}| = \mathrm{N}$. For increasing N the largest N eigenvalues $\lambda_i \leq 0$, $i = 0, 1, \cdots, \mathrm{N}-1$ of operator $\mathcal{A}$ converge to the eigenvalues of operator $\mathcal{L}$. Moreover all eigenvalues of both operators converge to* 0 *for* $\mathrm{N} \to \infty$.

Proof. The convergence claim between the two operators cannot be shown analytically since the computation of the eigenvalues of the DS require the solution of a transcendental equation (2.67b). Only numerical methods allow the verify the statement up to a certain number af agents. Therefore, Figure 2.8 shows that the normalised Euclidean norm of the relative error between the eigenvalues of the DS and Laplacian control satisfies

$$\frac{1}{\mathrm{N}+1} \left\| \boldsymbol{e}_\lambda(\mathrm{N}+1) \right\|_2 < \frac{1}{\mathrm{N}} \left\| \boldsymbol{e}_\lambda(\mathrm{N}) \right\|_2 \tag{2.72}$$

with $\left\| \boldsymbol{e}_\lambda(\mathrm{N}) \right\|_2 = \sqrt{\sum_{k=1}^{\mathrm{N}-1} (1 - \lambda_k^{\mathcal{L}} / \lambda_k^{\mathrm{DS}})^2}$. Here it is computed from $\mathrm{N} = 5$ up to 10^4 which is numerically quite challenging. However, in the following it is shown that all eigenvalues of $\mathcal{A}$ and $\mathcal{L}$ converge to 0 in case of the limit $\mathrm{N} \to \infty$. Note, considering (2.66b) for the eigenvalues for Laplacian this leads to

$$\lim_{\mathrm{N}\to\infty} \lambda_k^{\mathcal{L}} = 2a \lim_{\mathrm{N}\to\infty} (\cos(k\pi/\mathrm{N}) - 1) = 2a(1-1) = 0\,. \tag{2.73}$$

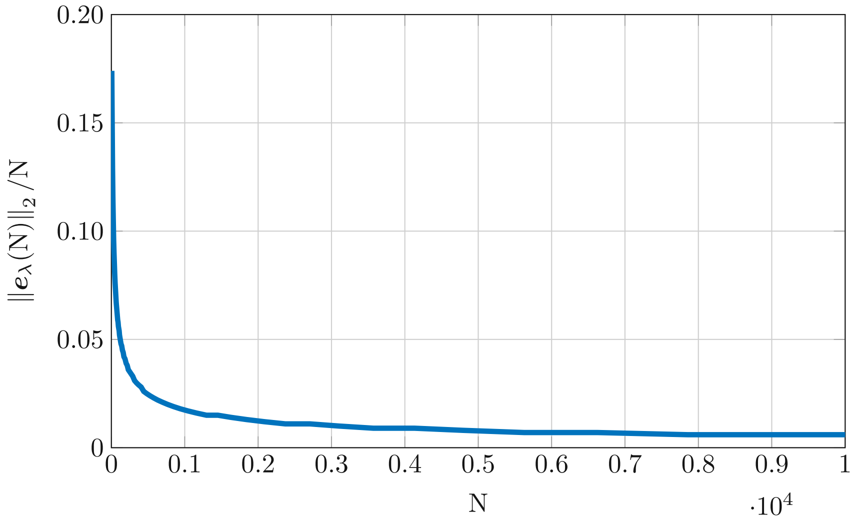


Figure 2.8: Normalised Euclidean norm of the relative error between λ_k^{DS} and $\lambda_k^{\mathcal{L}}$.

Moreover, respecting the image set of arctan, then obviously it follows

$$\lim_{\mathrm{N}\to\infty} \lambda_k^{\mathrm{DS}} = -a \lim_{\mathrm{N}\to\infty} \frac{\left(\arctan\left(\frac{2\mu_k}{\mu_k^2-1}\right)\right)^2}{(\mathrm{N}-1)^2} = 0\,. \tag{2.74}$$

□

In summary, Proposition 2.3 pictures the situation for a sole DS with the defined boundary conditions. In other words, it bridges the definition between the differential operator of a DS defined on real intervals in continuous space with a discrete operator, i.e., Laplacian control, for path graphs. If the requirement – mapping to a analogous graph theoretical object – is neglected, the transfer from derivatives in continuous space to discrete operators for a certain subclass of graphs can be generalised by the following definition.

Proposition 2.4. *Partial derivatives of the order of α, written as*

$$D^{(\alpha)}(x)(z,t) = \frac{\partial^\alpha x(z,t)}{\partial z^\alpha} \tag{2.75}$$

and defined on the open interval $(0, \mathrm{N}-1)$, are transformed into the discrete operators

$${}^{l}_{k}\mathcal{D}^{(\alpha)}(x)(\mathtt{v}_i,t) = \sum_{j=k}^{l} \mathtt{a}^{(\alpha)}(\mathtt{v}_{i+j})\left(x(\mathtt{v}_{i+j},t) - x(\mathtt{v}_i,t)\right) \tag{2.76}$$

by means of a proper vertex mapping ϕ according to (2.52) and appropriate central, forward, or backward FDMs which define the coefficients $\mathtt{a}^{(\alpha)}(\mathtt{v}_{i+j})$. With the discretisation step $\Delta z := 1$ the parameters $k, l \in \mathbb{Z}$ denote the $\{k,l\}-$nearest neighbour topology for the vertices $\mathtt{v}_i$ in the sense of

$$\phi^{-1}(\mathtt{v}_{i+j}) - \phi^{-1}(\mathtt{v}_i) = j \qquad\qquad j = k, k \mp 1, \dots, l \pm 1, l. \tag{2.77}$$

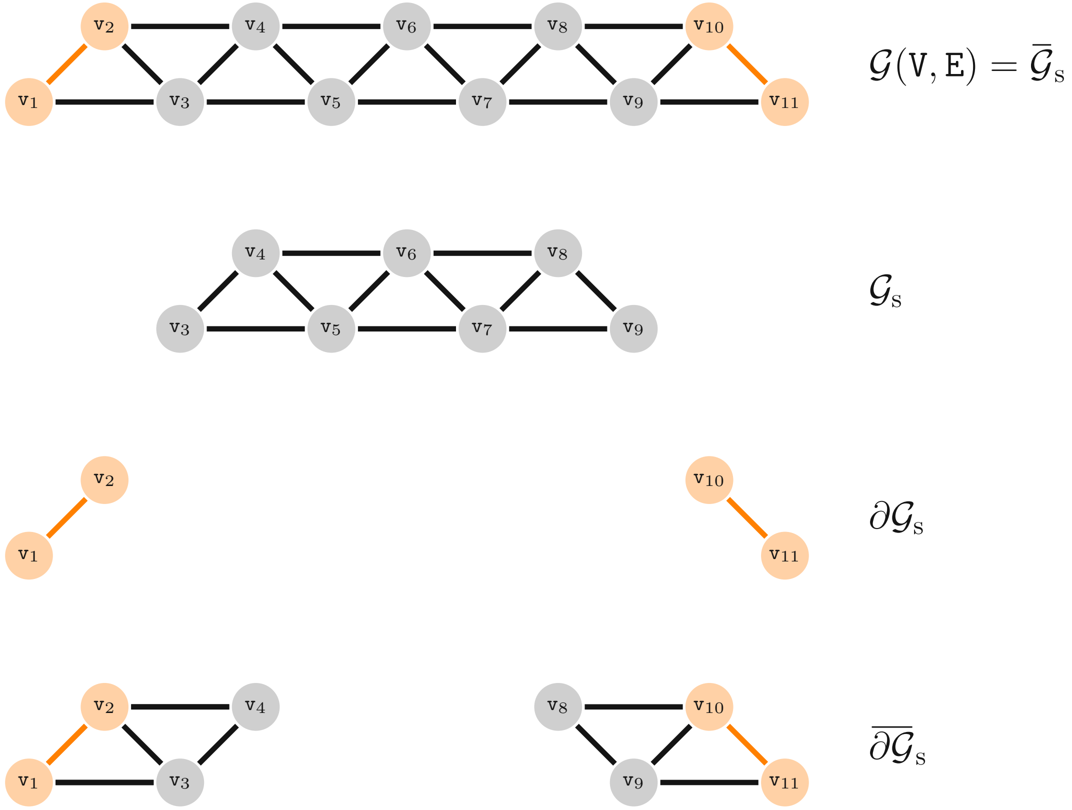

v2
v4
v6
v8
v10
v1
v3
v5
v7
v9
v11
$\mathcal{G}(\mathtt{V},\mathtt{E}) = \bar{\mathcal{G}}_\mathrm{s}$
$\mathcal{G}_\mathrm{s}$
$\partial\mathcal{G}_\mathrm{s}$
$\overline{\partial\mathcal{G}}_\mathrm{s}$

Figure 2.9: Graphical representation of a host graph $\mathcal{G}(\mathtt{V},\mathtt{E})$ which defines a 2−nearest neighbour topology for the inner vertices $\mathtt{v}_i \in \mathtt{S}_\mathrm{f}(\mathcal{G}_\mathrm{s}) \subset \mathtt{V}$.

Moreover the resulting underlying subgraph $\mathcal{G}_\mathrm{s}(\mathtt{S},\mathtt{E}_\mathrm{s})$ is traceable and has to satisfy

$$|l - k| = d(\mathtt{v}_i)\,. \tag{2.78}$$

with $d(\mathtt{v}_i)$ as the degree (2.4) *of the vertex $\mathtt{v}_i \in \mathtt{S}$.*

Here it has to be said that Proposition 2.4 directly follows from the definition of the FDM, the definition of $d(\mathtt{v}_i)$, and the bijective properties of the function ϕ for a traceable subgraph. The parameters $\mathtt{a}^{(\alpha)}(\mathtt{v}_{i+j})$ can be obtained from Table A.1 and Table A.2 up to the 4th derivative and 4th order of accuracy. By construction the coefficients fulfil

$$\sum_{j=k}^{l} \mathtt{a}^{(\alpha)}(\mathtt{v}_{i+j}) = 0\,. \tag{2.79}$$

Obviously in case of $l = -k$ it follows $d(\mathtt{v}_i) = 2l$ and the topology is simply denoted as $l-$*nearest neighbour*. As an example Figure 2.9 shows a host graph with $\mathcal{G}(\mathtt{V},\mathtt{E})$ with a corresponding subgraph $\mathcal{G}_\mathrm{s}$. For the configuration it holds that $\mathcal{G} = \mathcal{G}_\mathrm{s} \cup \partial\mathcal{G}_\mathrm{s} = \bar{\mathcal{G}}_\mathrm{s}$. The vertices which are element of the subgraph $\mathtt{v}_i \in \mathtt{S}_\mathrm{f} \subset \mathtt{V}$ form a 2−nearest neighbour topology. Thus, by means of central FDM and an appropriate formal, bijective mapping ϕ the operator

$$ {}_{-2}^{\;2}\mathcal{D}^{(4)}(x)(\mathtt{v}_i,t) := \sum_{j=-2}^{2} \mathtt{a}^{(4)}(\mathtt{v}_{i+j})\,\left(x(\mathtt{v}_{i+j},t)-x(\mathtt{v}_i,t)\right)\,, \qquad \forall \mathtt{v}_i \in \mathtt{S}_\mathrm{f}\,. \tag{2.80} $$

may be defined for the subgraph $\mathcal{G}_\mathrm{s}$ which approximates the 4th derivative in space of the state variable x with an accuracy order of $\mathcal{O}\left(\Delta z^2\right)$. The parameter $\mathtt{a}^{(4)}(\mathtt{v}_{i+j})$ are constructed by means of Table A.1. Here it is pointed out that the domain of definition includes the state variables $x(\mathtt{v}_{i+j},t)$ with $\mathtt{v}_{i+j} \in \mathtt{S}_\mathrm{f}(\mathcal{G}_\mathrm{s}) \cup \mathtt{S}_\mathrm{l}(\partial\mathcal{G}) = \mathtt{V}(\overline{\mathcal{G}}_\mathrm{s})$. With this the operator can be written as the mapping

$$ {}_{-2}^{\;2}\mathcal{D}^{(4)} : \mathbf{C}^{-1,m}(\{\mathtt{S}_\mathrm{f} \cup \mathtt{S}_\mathrm{l}\} \times \mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}) \mapsto \mathbf{C}^{-1,m}(\mathtt{S}_\mathrm{f} \times \mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}) \tag{2.81} $$

with a sufficiently smooth state variable $x(\mathtt{v}_i,t)$ with respect to time. The general notion of the classification of functions in this thesis, e.g. $\mathbf{C}^{-1,m}(\Omega \times \mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0})$, is discussed Appendix C.2. Similarly

$$ {}_{0}^{\pm 2}\mathcal{D}^{(\alpha)} : \mathbf{C}^{-1,m}(\mathtt{S}(\partial\overline{\mathcal{G}}_\mathrm{s}) \times \mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}) \mapsto \mathbf{C}^{-1,m}(\mathtt{S}_\mathrm{l} \times \mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}) \tag{2.82} $$

defines an operator for the states of the boundary vertices $\mathtt{v}_i \in \mathtt{S}_\mathrm{l}$ by using the forward the backward FDM. For the given example the subgraph $\partial\overline{\mathcal{G}}_\mathrm{s}$ is illustrated at the bottom of Figure 2.9. Depending on the choice of $\alpha \in \{1,2\}$ it either approximates the first derivative in space with second order accuracy or the second derivative with order of accuracy $\mathcal{O}\left(\Delta z\right)$. Again, appropriate parameters $\mathtt{a}^{(\alpha)}(\mathtt{v}_{i+j})$ can be constructed with Table A.2.

In order to sum up the latest achievements one may recall the starting point and objective of this section. In general the section discusses the transfer from continuum models to discrete models for the application on MASs. This scenario starts with a given distributed parameter system defined on some finite interval $[0,\ell_\mathrm{N}]$. Related theories of infinite dimensional systems allow to design algorithms for system stabilisation or state observation and conduct associated stability analysis. Discretisation methods, such as FDM, approximate differential operators and lead to (semi-)discretised PDEs. Generally, with this the systems are transformed to a set of ODEs which estimate the continuous formulation on a predefined finite number of points. Here, they form a subset Z of the interval $[0,\ell_\mathrm{N}]$. Moreover, in this paragraph it was shown the introduction of a formal bijective mapping ϕ is necessary to continue the rigorous transfer from discrete coordinates $z_i \in \mathrm{Z}$ to vertices $\mathtt{v}_i \in \mathtt{V}$ of the rendered graph $\mathcal{G}(\mathtt{V},\mathtt{E})$. In this context it was worked out that the mapping characterises the vertex set $\mathtt{V}$ and the discretisation method formally defines the edge set $\mathtt{E}$. When particularly using a real interval for the spatial domain, this render graphs which contain a Hamiltonian path, when the 2-step discretisation process is applied to the family of DCRSs. Viewed from the opposite perspective the vertices of a given graph can only be assigned with the discretised state of a DCRS when the graph is traceable. In order to establish structural and consequently dynamic equivalence between the continuum and discrete models the condition $\Delta z = 1$ for the discretisation step must hold. This leads to the preliminary that the given distributed parameters system has to be defined on the interval

$[0, \mathrm{N}-1]$ with N as the number of agents. For the special case of a DS it was shown that the largest eigenvalues of the discretised operator converge to those of the differential operator defined on the interval $[0, \mathrm{N}-1]$ as N increases. Interestingly for the DS the discrete operator is built upon the so-called Laplacian matrix, an elementary graph theoretical construct.

Finally it can be concluded that discretisation methods, such as FDM, together with an appropriate formal mapping ϕ forms a consistent path for the transition from continuum models to discrete models. In this context the discrete states are associated with properties of vertices and again, a vertex stands for an agent of a MAS. For the objective of the thesis this is a sufficient concept to design control algorithms in continuous formulation and apply them to a discrete set of wheeled robots. However, so far this approach does not cover the linkage between the continuous and the discrete setup from a modelling perspective. Furthermore, the discussion still left behind graph theoretical concepts such as the weight of an edge $w(\mathsf{e}_{ij}, t)$ and it is limited to a special class of graphs. In other words, following this strategy for distributed parameter systems with boundary conditions defined on an finite interval within the Euclidean space always renders traceable or similar graphs, but others, e.g. trees, are not covered with this concept. Here the definition of PDEs on very abstract mathematical objects as hinted in Remark 2.6 seems an interesting approach. Yet, these thoughts are no objective of the thesis. Consequently, the next section only gives an brief overview of approaches which try to mimic operations in the field of vector analysis such as gradient, divergence, and curl on the discrete nature of graphs. The main discussion of the next section leads to the alternative approach, which deals with the approximation of discrete swarm dynamics by continuous formulations. This leads to the complete opposite point of view as pictured in the current section. Therefore, the following paragraphs especially shall link some general properties of graphs to the continuous formulation of DCRSs.

2.2.3 Series Expansion – Transition from Discrete to Continuum Models

Establishing calculus for and on graphs is an approach which is researched mainly by institutes of mathematics, see an overview in [83]. Generally there are different methods to embed differential operators on graphs. The interested reader is referred to the literature discussing the following concepts

- PDEs defined on the edges [47, 48, 49],
- PDEs defined on the vertices [84],
- PDEs defined on the vertices and edges [27, 28].

However, this section mainly deals with the approximation of the (discrete) system dynamics of a MAS by using a continuous formulation. For this, the starting point shall be a generic host graph $\mathcal{G}(\mathtt{V},\mathtt{E})$ with the vertex set $\mathtt{V}$ and the edge set $\mathtt{E}$. Moreover, let us assume the time-invariant weights $w(\mathtt{e}_{ij},t)$ are connected to each edge. Now, for the transition from the discrete to the continuous formulation it is necessary to define a function which introduces a virtual variable along a path. This is followed by the definition of the function $\zeta : \mathtt{V}\times\mathtt{V} \to \{-\ell, -\Delta z(\mathrm{N}-2), \dots, -\Delta z, 0, \Delta z, \dots, \Delta z(\mathrm{N}-2), \ell\}$

$$\zeta(\mathtt{v}_i,\mathtt{v}_j) = (j-i)\Delta z\,, \qquad \mathtt{v}_i,\ \mathtt{v}_j \in \mathtt{V} \tag{2.83}$$

with $1 \leq j,\, i \leq \mathrm{N}$, and $\Delta z = \ell/(\mathrm{N}-1) > 0$. This introduces a relative signed coordinate to the pair of vertices $\mathtt{v}_i$ and $\mathtt{v}_j$. Obviously, applying the absolute value to (2.83) features the properties

$$\begin{aligned} &|\zeta(\mathtt{v}_i,\mathtt{v}_j)| = 0 \Leftrightarrow \mathtt{v}_i = \mathtt{v}_j\,,\\ &|\zeta(\mathtt{v}_i,\mathtt{v}_j)| = |\zeta(\mathtt{v}_j,\mathtt{v}_i)|\ ,\ \text{and}\\ &|\zeta(\mathtt{v}_i,\mathtt{v}_k)| \leq |\zeta(\mathtt{v}_i,\mathtt{v}_j)| + |\zeta(\mathtt{v}_j,\mathtt{v}_k)| \end{aligned} \tag{2.84}$$

and therefore $|\zeta(\mathtt{v}_i,\mathtt{v}_j)|$ introduces a metric. Here, the proof is omitted since it is trivial for the preliminaries of Δz and the indices i, j. With this, note the bijective function $z : \mathtt{V} \to \{0, \Delta z, 2\Delta z, \dots, \ell\}$

$$z(\mathtt{v}_i) = \zeta(\mathtt{v}_1,\mathtt{v}_i) = (i-1)\Delta z = z_i\,, \qquad \mathtt{v}_1,\ \mathtt{v}_i \in \mathtt{V} \tag{2.85}$$

which sorts the vertices in a monotonously increasing order dependent on its index i. This motivates the change of the state variables into the form

$$\begin{aligned}\left(\mathbf{C}^{-1,m}\left(\mathtt{V}\times\mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}\right)\right)^n \ni \boldsymbol{x}(\mathtt{v}_i,t) = \boldsymbol{x}(z^{-1}((i-1)\Delta z),t) = &\\ \check{\boldsymbol{x}}((i-1)\Delta z,t) = \check{\boldsymbol{x}}(z_i,t) \in (\mathbf{C}^{-1,m}(\mathrm{Z}\times\mathbb{R}^+_{t_0};\mathbb{R}^+_{t_0}))^n\,. &\end{aligned} \tag{2.86}$$

with Z defined by (2.42). Since this process does not modify the state vector itself the redefinition $\boldsymbol{x}(z_i,t) := \check{\boldsymbol{x}}(z_i,t)$ preserves the labelling of the state. Comparing the discussion with the function definition (2.43) the connection

$$z(\mathtt{v}_i) := \phi^{-1}(\mathtt{v}_i) \quad \text{and} \quad \phi(z(\mathtt{v}_i)) = \phi(z_i) = \mathtt{v}_i := z^{-1}((i-1)\Delta z)\,, \tag{2.87}$$

respectively, is obvious. As an example the change is graphically illustrated in Figure 2.10 for a path graph with seven vertices,

Now, let us omit one step in the transfer to the continuous formulation. Assume that there exists a classical smooth solution for the associated continuous formulation which allows to derive the Taylor series expansion around z_i. In particularly, this allows the second order approximation of the state vector $\boldsymbol{x}$ for every $z \in [0,\ell]$. Formally this can be written as

$$\boldsymbol{x}(z,t) = \boldsymbol{x}(z_i,t) + \frac{\partial \boldsymbol{x}(z_i,t)}{\partial z}(z-z_i) + \frac{\partial^2 \boldsymbol{x}(z_i,t)}{\partial z^2}\frac{(z-z_i)^2}{2} + \mathcal{O}\left((z-z_i)^3\right) \tag{2.88}$$

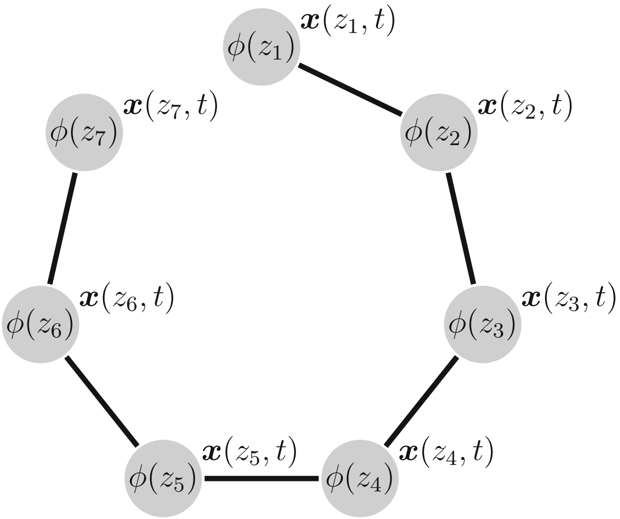


Figure 2.10: A path graph $\mathcal{P}_7$ where the state association is transformed by (2.86) from the vertices $\mathtt{v}_i = \phi(z_i)$ to the virtual communication path variable $z_i = z(\mathtt{v}_i)$.

Applying the transformation (2.86) to the protocol (2.30) with the Laplacian operator (2.22) and consequently utilising the series expansion (2.88), then this gives the approximation

$$\partial_t \boldsymbol{x}(z_i,t) \approx \sum_{\phi(z_j)\in\mathtt{Nbr}(i)} a\left(\frac{\partial^2\boldsymbol{x}(z_i,t)}{\partial z^2}\frac{(z_j-z_i)^2}{2} + \frac{\partial\boldsymbol{x}(z_i,t)}{\partial z}(z_j-z_i)\right) + c(z_i,t)\boldsymbol{x}(z_i,t)\,. \quad (2.89)$$

Now, assuming the host graph $\mathcal{G}(\mathtt{V},\mathtt{E})$ is traceable, then with Definition 2.3 there exists a Hamiltonian path and consequently an isomorphic sub graph $\mathcal{P}_{\mathrm{N}}$ which is a spanning path. Finally for $\mathcal{P}_{\mathrm{N}}$ the neighbouring vertices are $\phi(z_j) = \mathtt{v}_j \in \mathtt{Nbr}(i) = \{\mathtt{v}_{i+1}, \mathtt{v}_{i-1}\}$ and the approximation (2.89) leads to

$$\partial_t \boldsymbol{x}(z_i,t) \approx a\frac{\partial^2\boldsymbol{x}(z_i,t)}{\partial z^2}(\Delta z)^2 + c(z_i,t)\boldsymbol{x}(z_i,t)\,. \quad (2.90)$$

With this it is obvious to interpret z as the virtual communication path variable with N discrete and equidistant values in $[0,\ell]$ and Δz as the discretisation step. As discussed in Section 2.2.2, linking the length of the spatial domain to the number of agents N, see (2.49) and subsequently $\Delta z := 1$, allows to compare (2.90) with the PDE of a DRS[3]. This illustrates, that the discrete and continuum model can be formally exchanged at any $z_i = (i-1)\Delta z$ with an error $\mathcal{O}\left(\Delta z^4\right)$. This motivates further comments on the discrete-continuous modelling relationship.

Remark 2.7. *The introduction of the Taylor series formally allows to connect continuous formulations with the discrete description of the agent dynamics when the expansion is evaluated at the specific spatial coordinates z_i. Then the linkage explicitly appoints the discrete states by approximated continuous states, which are sufficiently smooth along the communication path variable z. A subsequent transformation back from the discrete points z_i, $i = \{1,\dots,\mathrm{N}\}$ to the*

[3] Assume $v = 0$ in (2.38a)

continuous domain $z \in [0, \ell]$ can be pictured as a kind of blurring process *from discrete points $z_i \to z$. For this reason the concept of continuous models is particularly suited for networks, swarms, or MASs with a high number* N *of participants since the* blurring distance $\Delta z = 1$ *gets relatively small compared to the length of the spatial domain $\ell = \ell_\mathrm{N} = \mathrm{N} - 1$.*

The same procedure can be applied for the boundary conditions. In this case it is convenient to consider the generalised conditions summarised in (2.29) of Remark 2.5. Exercising a proper redefinition of the coefficients p_j, q_j and r_j , $j \in \{0, \ell_\mathrm{N}\}$ and performing a first order series expansion one obtains

$$\begin{aligned} p_0 \partial_t^k \boldsymbol{x}(0,t) &\approx q_0 \boldsymbol{x}(0,t) + r_0 \partial_z \boldsymbol{x}(0,t) + \boldsymbol{u}_0(t)\,, \\ p_{\ell_\mathrm{N}} \partial_t^k \boldsymbol{x}(\ell_\mathrm{N},t) &\approx q_{\ell_\mathrm{N}} \boldsymbol{x}(\ell_\mathrm{N},t) + r_{\ell_\mathrm{N}} \partial_z \boldsymbol{x}(\ell_\mathrm{N},t) + \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,. \end{aligned} \tag{2.91}$$

Again, with the appropriate choice of the parameters $k \in \mathbb{N}$, $\{p_j, q_j, r_j\} \in \mathbb{R}$, and the external inputs $\boldsymbol{u}_j(t)$ it is possible to render the classic boundary conditions for distributed parameter system, i.e., Dirichlet, Neumann and mixed conditions as well as the extension to dynamic conditions,

With the results from above it is possible to derive a continuous model, e.g., a PDE formulation, which approximates the initial discrete setup modelled by means of graph theoretical reflections. The latter context mainly involves the vertex and edge sets. In comparison the following shall include weights w_{ij} assigned to the edges as introduced in Remark 2.2 as well as weights c_i associated with the vertices similarly introduced in [27, 28] as vertex measures. Consequently let us recall integrator dynamics applied to graph Laplacian control with a reaction component again. For the follower agents this gives

$$\partial_t \boldsymbol{x}(\mathtt{v}_i,t) = \mathcal{L}_w(\boldsymbol{x})(\mathtt{v}_i,t) + c(\mathtt{v}_i,t)\boldsymbol{x}(\mathtt{v}_i,t)\,, \qquad \forall \mathtt{v}_i \in \mathtt{S}_\mathrm{f} \tag{2.92}$$

which is similar to (2.30) but an edge weighted graph Laplacian $\mathcal{L}_w(\boldsymbol{x})$ instead of $\mathcal{L}(\boldsymbol{x})$ with the leading coefficient a. Now, let us recall the definition of the incidence matrix (2.12) and the the evaluation formula of the weighted graph Laplacian matrix $L_w(\mathcal{P})$ (2.17). With this, consider the arbitrarily oriented $\mathrm{N} \times (\mathrm{N}-1)$ incidence matrix of a path graph $(\mathcal{P}_\mathrm{N}^o)$ and the $(\mathrm{N}-1) \times (\mathrm{N}-1)$ diagonal weight matrix

$$D(\mathcal{P}_\mathrm{N}^o) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \ddots & \vdots \\ 0 & 0 & -1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 1 \\ 0 & 0 & \cdots & 0 & -1 \end{bmatrix}, \quad W(\mathtt{E}_\mathcal{P}) = \begin{bmatrix} w_{12} & 0 & \cdots & 0 \\ 0 & w_{23} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & w_{\mathrm{N}-1,\mathrm{N}} \end{bmatrix} \tag{2.93}$$

with $w_{ij} = w(\mathtt{e}_{ij})$. Then the weighted graph Laplacian matrix $L_w(\mathcal{P})$ is evaluated to

$$L(\mathcal{P}) = \begin{bmatrix} w_{12} & -w_{12} & 0 & 0 & \cdots & 0 \\ -w_{12} & w_{12}+w_{23} & -w_{23} & 0 & \cdots & 0 \\ 0 & -w_{23} & w_{23}+w_{34} & -w_{34} & \ddots & \vdots \\ 0 & 0 & -w_{34} & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & w_{\mathrm{N-2,N-1}}+w_{\mathrm{N-1,N}} & -w_{\mathrm{N-1,N}} \\ 0 & 0 & \cdots & 0 & -w_{\mathrm{N-1,N}} & w_{\mathrm{N-1,N}} \end{bmatrix} \tag{2.94}$$

and protocol (2.92) leads to

$$\begin{aligned} \partial_t x(\mathtt{v}_1, t) &= w_{12}(x(\mathtt{v}_2, t) - x(\mathtt{v}_1, t)) \\ \partial_t x(\mathtt{v}_i, t) &= w_{i,i+1} x(\mathtt{v}_{i+1}, t) - (w_{i,i+1} + w_{i-1,i} - c(\mathtt{v}_i, t))\, x(\mathtt{v}_i, t) + w_{i-1,i} x(\mathtt{v}_{i-1}, t) \\ \partial_t x(\mathtt{v}_{\ell_\mathrm{N}}, t) &= w_{\mathrm{N-1,N}}(x(\mathtt{v}_{\mathrm{N-1}}, t) - x(\mathtt{v}_\mathrm{N}, t))\,. \end{aligned} \tag{2.95}$$

Comparing the pattern with the equations of the semi-discretised DCRS (2.50) by second error order FDM then coefficients can be mapped by

$$\begin{aligned} & a \overset{!}{=} \frac{w_{i,i+1} + w_{i-1,i}}{2} = \bar{w}_{i-1,i+1}\,, \\ & v \overset{!}{=} -\left(w_{i,i+1} - w_{i-1,i}\right) = -\Delta w_{i-1,i+1}\,, \\ & r(\mathtt{v}_i, t) \overset{!}{=} c(\mathtt{v}_i, t) \end{aligned} \tag{2.96}$$

Obviously the reaction term coefficient r is directly mapped to the vertex weight $c(\mathtt{v}_i, t)$. Note, if $c(\mathtt{v}_i, t) > 0$ it can be labelled as a source and for $c(\mathtt{v}_i, t) < 0$ it shall be named as a sink vertex. Moreover, since a and v are formulated as constants in (2.50) for this case the conditions above are only fulfilled for constant weights, meaning $w_{i,i+1} = w = a$, $i = 1, \ldots \mathrm{N} - 1$, and $v = 0$. However, allowing vertex dependent coefficients the diffusion coefficient $a(\mathtt{v}_i)$ is mapped to the mean value of the involved weights and the convection coefficient $v(\mathtt{v}_i)$ to the difference of the weights. This coincides pretty well with the definition of the so-called *directional derivative* $D_{w,\mathtt{v}_j}$ of a function and the related $w-$Laplacian operator ∇_w in [83, 84]. There these operators are defined as

$$D_{w,\mathtt{v}_j}(\boldsymbol{x})(\mathtt{v}_i, t) := (\boldsymbol{x}(\mathtt{v}_j, t) - \boldsymbol{x}(\mathtt{v}_i, t)) \sqrt{\frac{w(\mathtt{e}_{ij}, t)}{d_w(\mathtt{v}_i, t)}} \tag{2.97}$$

$$\nabla_w(\boldsymbol{x})(\mathtt{v}_i, t) := \sum_{\mathtt{v}_j \in \mathtt{Nbr}(i)} (\boldsymbol{x}(\mathtt{v}_j, t) - \boldsymbol{x}(\mathtt{v}_i, t)) \frac{w(\mathtt{e}_{ij}, t)}{d_w(\mathtt{v}_i, t)} \tag{2.98}$$

The degree $d_w(\mathtt{v}_i, t)$ is defined as the summed weights $\sum_{\mathtt{v}_j \in \mathtt{Nbr}(i)} w(\mathtt{e}_{ij}, t)$. Assuming a path graph and inserting the series approximation (2.88) into the $(\mathtt{v}_i \to z_i)-$ transformed operator

(2.98) then it can be written as

$$
\begin{aligned}
\nabla_w(\boldsymbol{x})(z_i,t) \approx{}& \left(\frac{1}{2}\frac{\partial^2 \boldsymbol{x}(z_i,t)}{\partial z^2} + \frac{\partial \boldsymbol{x}(z_i,t)}{\partial z}\right)\frac{w(\mathsf{e}_{i,i+1},t)}{d_w(\mathtt{v}_i,t)} \\
&+ \left(\frac{1}{2}\frac{\partial^2 \boldsymbol{x}(z_i,t)}{\partial z^2} - \frac{\partial \boldsymbol{x}(z_i,t)}{\partial z}\right)\frac{w(\mathsf{e}_{i-1,i},t)}{d_w(\mathtt{v}_i,t)} \\
&= \frac{1}{2}\frac{\partial^2 \boldsymbol{x}(z_i,t)}{\partial z^2} + \frac{w(\mathsf{e}_{ii+1},t) - w(\mathsf{e}_{i-1,i},t)}{w(\mathsf{e}_{ii+1},t) + w(\mathsf{e}_{i-1,i},t)}\frac{\partial \boldsymbol{x}(z_i,t)}{\partial z}
\end{aligned}
\tag{2.99}
$$

where $\Delta z = 1$ is considered. With this $\nabla_w(\boldsymbol{x})$ can be interpreted as the *weight normalised* graph Laplacian operator.

These expositions finalise the discussion for the transfer from a discrete setup to the a continuous modelling approach. Here, the first key element is the transformation from vertices $\mathtt{v}_i$ to discrete spatial coordinates z_i as introduced in (2.86). Moreover, the introduction of the Taylor series expansion allows to approximate the discrete setup by means of a continuous formulation which coincides at $z = z_i$ with certain error order accuracy. Last but not least, in this work a rigorous transfer to a continuous spatial domain is described a thought experiment without any mathematical considerations. One has to undergo a kind of blurring process to move from discrete points $z_i \in \mathrm{Z}$, with Z defined in (2.42) to a continuous spatial domain $[0, \ell_{\mathrm{N}}]$. Thus, the next paragraphs motivate the so-called *inverse design approach* which basically starts already with the continuous formulation.

2.3 Inverse Design Approach

The previous two sections confirm that there is an equivalence between dynamic models of MASs with discrete nature and the models of distributed parameter systems described by PDEs under suitable assumptions. In general the modelling process follows abstraction steps from a network consisting of a number of agents to a mathematical representation in from of state variables. These stand for the features of the agents. The formal structure of the interaction between the agents and their dynamical behaviour is usually expressed by differential equations. The continuous model approach adds a further abstraction step since it results in a mathematical representation which is technically independent of the communication topology and the number of agents. However, it must be said that the structure of the network is limited when explicit discretisation methods apply. At the beginning of Section 2.2 the early and late lumping approaches were briefly discussed in terms of the consecutive modelling and control design steps. Apparently the investigated modelling procedure is the reverse of the early lumping approach in traditional control theory for DPSs. More precisely, in this work a MAS consisting of a discrete number of states is approximated by a continuous formulation, but the early lumping approach approximates continuum models by using discretisation methods. In contrast to that, the controller or observer synthesis for MASs which are modelled by continuous formulations can be represented as the process illustrated in Figure 2.11. Interestingly, this design concept

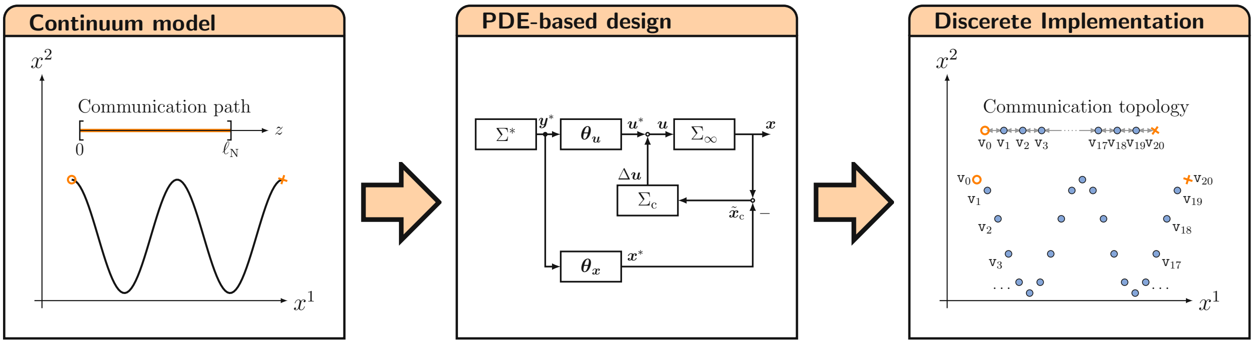


Figure 2.11: Concept of the *inverse design approach*: Starting with a continuous modelling approach, followed by controller and observer design and synthesis. As a last step the developed algorithms are discretised and applied to the MAS.

follows exactly the late lumping policy. With this insight it is understandable to step back from the sophisticated abstraction process. For networks without a-priori given protocols it is possible to impose a certain desired behaviour instead by assigned an appropriate PDE model. The subsequently proposed *inverse design approach* instantly introduces a continuous model for the MAS in form of a distributed parameter system with inherent boundary conditions and proper initial conditions. The determination induces both, the dynamic model of the agents with respect to time as well as the input protocol, cf. (2.19) and (2.22). Though the continuous set-up does not schedule any type of communication topology. For instance, considering a swarm of wheeled robots the commitment to model the swarm dynamics in the 2-dimensional (2D) plane by means of the parabolic partial differential equation

$$\frac{\partial \boldsymbol{x}(z,t)}{\partial t} = a\frac{\partial^2 \boldsymbol{x}(z,t)}{\partial z^2} + b\frac{\partial \boldsymbol{x}(z,t)}{\partial z} + c\boldsymbol{x}(z,t)\,, \qquad z \in (0, \ell_\mathrm{N})\,, \tag{2.100}$$

with dynamic inhomogeneous boundary conditions

$$\begin{aligned} \frac{\partial \boldsymbol{x}(0,t)}{\partial t} &= \boldsymbol{u}_0(t)\,, \\ \frac{\partial \boldsymbol{x}(\ell_\mathrm{N},t)}{\partial t} &= \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,, \end{aligned} \tag{2.101}$$

and $\boldsymbol{x}(z,t_0) = \boldsymbol{x}_0(z)$ induces a simple integrator model for the temporal dynamics of each single agent, including the leader and the anchor agent at the boundaries. According to the established input protocol the spatial dynamics of the swarm behaves equivalent to a physical DCRS. However, in general any type of partial differential equation may be considered for each coordinate $x^m(z,t)\,, m = 1\ldots n$ of the state vector

$$\boldsymbol{x}(z,t) = \Big[x^1(z,t), x^2(z,t), \ldots, x^n(z,t)\Big]^T \tag{2.102}$$

individually. Moreover they may be coupled systems or may be de-coupled from each other. Recalling the MAS example of wheeled robots in the 2D plane ($n = 2$) the left-hand side of (2.100) and (2.101) represents the velocity in each coordinate of the follower agents and the leader agents, respectively. The input protocol on the right-hand side is modelled by a decoupled system, meaning that the spatial dynamics of one coordinate $x^m(z,t)$ is completely independent from the other coordinates $x^p(z,t)$ with $m \neq p$.

In this thesis the discussion focuses exclusively on parabolic distributed parameter systems for both the uncoupled system as well as some particular cases with coupled representations. Consequently, the following chapter deals with the problem formulation of continuous models for MAS of both types. It starts with a short classification of the parabolic type of PDE and is followed by a very generic ansatz of the problem formulation. With this, some specific models are derived for both, coupled and uncoupled systems.

Chapter 3

Continuous Problem Formulation for Multi-Agent Dynamics

The previous chapter motivates to use continuous models in order to describe the dynamic behaviour of swarms or MASs. It gives an introduction to some preliminary graph theoretical aspects and shows step by step the existence of equivalent patterns between a discrete mathematical description in form of a set coupled ODEs and the approximation by using continuous models of DPSs. Therefore the following paragraphs introduce parabolic partial differential equations for the abstraction and the modelling of MA dynamics. Starting with a brief classification of the parabolic type of PDE the chapter works out some particular problems for both, coupled systems limited to specific properties as well as uncoupled systems.

3.1 Parabolic Partial Differential Equations

Note, the following discussion is based on an elaboration in [57]. Generally the classification of partial differential equation is important to characterise their dynamic behaviour in view of suitable control and observer design concepts. For studying the classification of distributed parameter systems let us consider the semi-linear PDE of second order, i.e.,

$$\begin{aligned} a_{11}(z,t)\frac{\partial^2 x(z,t)}{\partial z^2} + 2a_{12}(z,t)\frac{\partial^2 x(z,t)}{\partial z \partial t} + a_{22}(z,t)\frac{\partial^2 x(z,t)}{\partial t^2} + b_1\frac{\partial x(z,t)}{\partial z} + b_2\frac{\partial x(z,t)}{\partial t} \\ + f(z,t,x(z,t)) = 0 \end{aligned} \tag{3.1}$$

with the independent spatial coordinate z and temporal coordinate t. The task to find a solution $x(z,t)$ of (3.1) that satisfies certain conditions given on a curve on the (z,t)-surface is known in literature as the *Cauchy problem.* Assuming the solution $x(z,t)$ with regular values of $\partial x(z,t)/\partial t$ and $\partial x(z,t)/\partial z$ on the curve $\Gamma : (z,t) = (\alpha(s), \beta(s))$, $s \in \mathbb{R}$, starts with initial values according to the functions

$$x(z,t)|_\Gamma = h(s), \quad \left.\frac{\partial x(z,t)}{\partial z}\right|_\Gamma = \psi(s), \quad \left.\frac{\partial x(z,t)}{\partial t}\right|_\Gamma = \phi(s)\,, \tag{3.2}$$

then the total differentials reads

$$\mathrm{d}h(s) = \frac{\partial x(z,t)}{\partial z}\mathrm{d}z + \frac{\partial x(z,t)}{\partial t}\mathrm{d}t = \psi(s)\mathrm{d}z + \phi(s)\mathrm{d}t\,, \tag{3.3}$$

$$\begin{aligned} \mathrm{d}\psi(s) &= \frac{\partial^2 x(z,t)}{\partial z^2}\mathrm{d}z + \frac{\partial^2 x(z,t)}{\partial t \partial z}\mathrm{d}t\,, \\ \mathrm{d}\phi(s) &= \frac{\partial^2 x(z,t)}{\partial z \partial t}\mathrm{d}t + \frac{\partial^2 x(z,t)}{\partial t^2}\mathrm{d}z\,. \end{aligned} \tag{3.4}$$

Here (3.3) expresses the obvious condition that only two of three functions (3.2) can be assigned independently. Considering (3.4) a solution $x(z,t)$ of (3.1) on the curve (3.2) has to fulfil the set of equations

$$\underbrace{\begin{bmatrix} a_{11}(z,t) & 2a_{12}(z,t) & a_{22}(z,t) \\ \mathrm{d}z & \mathrm{d}t & 0 \\ 0 & \mathrm{d}z & \mathrm{d}t \end{bmatrix}}_{=:A} \begin{bmatrix} \frac{\partial^2 x(z,t)}{\partial z^2} \\ \frac{\partial^2 x(z,t)}{\partial t \partial z} \\ \frac{\partial^2 x(z,t)}{\partial t^2} \end{bmatrix} = \begin{bmatrix} -b_1\frac{\partial x(z,t)}{\partial z} - b_2\frac{\partial x(z,t)}{\partial t} - f(z,t)x(z,t) \\ \mathrm{d}\psi(s) \\ \mathrm{d}\phi(s) \end{bmatrix} \tag{3.5}$$

and the solution is unique if the determinant of A satisfies $\det(A) \neq 0$. More precisely $\det(A) = a_{11}(z,t)\mathrm{d}z^2 - 2a_{12}(z,t)\mathrm{d}z\mathrm{d}t + a_{22}(z,t)\mathrm{d}t^2 = 0$ defines the conditional equation of the so-called characteristic curve

$$\frac{\mathrm{d}t}{\mathrm{d}z} = \frac{a_{12}(z,t) \pm \sqrt{a_{12}^2(z,t) - a_{11}(z,t)\,.a_{22}(z,t)}}{a_{11}(z,t)} \tag{3.6}$$

With $a_{12}^2(z,t) - a_{11}(z,t)a_{22}(z,t) = 0$ and $b_2(z,t) \neq 0$ the partial differential equations is known as parabolic. Obviously (3.1) can always be classified as parabolic in case of $a_{22}(z,t) = a_{12}(z,t) = 0$. In the upcoming section the problem formulation starts with a generic parabolic arrangement and is subsequently specified to certain problems which are further investigated for formation control of MASs.

3.1.1 General Continuous Problem Formulation using Parabolic Partial Differential Equations

A generic formulation of a parabolic partial differential equation is given by

$$\frac{\partial \boldsymbol{x}(z,t)}{\partial t} = D(z)\frac{\partial^2 \boldsymbol{x}(z,t)}{\partial z^2} - \frac{\partial [V(z,t,\boldsymbol{x})\boldsymbol{x}(z,t)]}{\partial z} + \boldsymbol{r}(z,t,\boldsymbol{x}), \quad (z,t) \in \Omega(\ell_\mathrm{N}, t_0) \tag{3.7}$$

with $\Omega(\ell_\mathrm{N}, t_0) := (0, \ell_\mathrm{N}) \times \mathbb{R}_{t_0}^+$ and the state vector $\boldsymbol{x}(z,t) = [x^1(z,t), x^2(z,t), \ldots, x^n(z,t)]^T$, $n \geq 1$ as the number of coordinates, and the initial condition

$$\boldsymbol{x}(z,t_0) = \boldsymbol{x}_0(z)\,. \tag{3.8}$$

The inhomogeneous or dynamic boundary conditions

$$\begin{aligned} \boldsymbol{\phi}_1(t, \boldsymbol{x}(0,t), \partial_z \boldsymbol{x}(0,t), \partial_t \boldsymbol{x}(0,t)) &= \boldsymbol{u}_0(t)\,, \\ \boldsymbol{\phi}_2(t, \boldsymbol{x}(\ell_\mathrm{N},t), \partial_z \boldsymbol{x}(\ell_\mathrm{N},t), \partial_t \boldsymbol{x}(\ell_\mathrm{N},t)) &= \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,, \end{aligned} \qquad t > t_0 \tag{3.9}$$

respectively, define how boundary control inputs $\boldsymbol{u}_j(t) = [u_j^1(t), u_j^2(t), \dots, u_j^n(t)]^T$, $j = 0, \ell_\mathrm{N}$, for the n coordinates enter the system. The diagonal matrix $D(z)$ defined as

$$D(z) = \begin{bmatrix} d_1(z) & 0 & \cdots & 0 \\ 0 & d_2(z) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n(z) \end{bmatrix} \tag{3.10}$$

stands for the diffusion coefficients and the matrix $V(z, t, \boldsymbol{x})$ defines the convective contribution. The term $\boldsymbol{r}(z, t, \boldsymbol{x})$ can be interpreted as a distributed reaction process. Let us assume that $\boldsymbol{x}^*(z, t)$ with its proper initial state $\boldsymbol{x}^*(z, t_0)$ is a trajectory of (3.7) for the boundary inputs $\boldsymbol{u}_0(t) = \boldsymbol{u}_0^*(t)$ and $\boldsymbol{u}_{\ell_\mathrm{N}}(t) = \boldsymbol{u}_{\ell_\mathrm{N}}^*(t)$. Moreover $\partial_z \boldsymbol{x}^*(z, t)$ is the corresponding derivative with respect to z. Then the linearisation of the NL problem (3.7) around the trajectory $\{\boldsymbol{x}^*(z, t), \boldsymbol{u}_0^*(t), \boldsymbol{u}_{\ell_\mathrm{N}}^*(t)\}$ gives the LTV parabolic differential equation

$$\begin{aligned} \frac{\partial \Delta \boldsymbol{x}(z, t)}{\partial t} =& A(z) \frac{\partial^2 \Delta \boldsymbol{x}(z, t)}{\partial z^2} - V(z, t, \boldsymbol{x}^*) \frac{\partial \Delta \boldsymbol{x}(z, t)}{\partial z} \\ &+ \left[R(z, t, \boldsymbol{x}^*) - \tilde{V}(z, t, \boldsymbol{x}^*, \partial_z \boldsymbol{x}^*) \right] \Delta \boldsymbol{x}(z, t) , \end{aligned} \tag{3.11}$$

for the domain $(z, t) \in \Omega(\ell_\mathrm{N}, t_0)$ and $\Delta \boldsymbol{x}(z, t) = \boldsymbol{x}(z, t) - \boldsymbol{x}^*(z, t)$. Its initial state is

$$\Delta \boldsymbol{x}(z, t_0) = \boldsymbol{x}_0(z) - \boldsymbol{x}^*(z, t_0) \tag{3.12}$$

and the coefficient matrices are given by

$$\begin{aligned} A(z) &= D(z) , \\ R(z, t, \boldsymbol{x}^*) &= \left. \frac{\partial \boldsymbol{r}(z, t, \boldsymbol{x})}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x} = \boldsymbol{x}^*} , \text{ and} \\ \tilde{V}(z, t, \boldsymbol{x}^*, \partial_z \boldsymbol{x}^*) &= \left(\frac{\partial V(z, t, \boldsymbol{x})}{\partial z} + \frac{\partial}{\partial \boldsymbol{x}} \left(\frac{\partial V(z, t, \boldsymbol{x})}{\partial z} \right) \cdot \boldsymbol{x}^*(z, t) \right. \\ &\qquad \left. \left. + \frac{\partial}{\partial \boldsymbol{x}} \left(V(z, t, \boldsymbol{x}) \right) \cdot \frac{\partial \boldsymbol{x}^*(z, t)}{\partial z} \right) \right|_{\boldsymbol{x} = \boldsymbol{x}^*} . \end{aligned} \tag{3.13}$$

The linearised boundary conditions follow to

$$\begin{aligned} \Lambda^1(t) \Delta \boldsymbol{x}(0, t) + \Lambda_z^1(t) \partial_z \Delta \boldsymbol{x}(0, t) + \Lambda_t^1(t) \partial_t \Delta \boldsymbol{x}(0, t) &= \Delta \boldsymbol{u}_0(t) , \\ \Lambda^2(t) \Delta \boldsymbol{x}(\ell_\mathrm{N}, t) + \Lambda_z^2(t) \partial_z \Delta \boldsymbol{x}(\ell_\mathrm{N}, t) + \Lambda_t^2(t) \partial_t \Delta \boldsymbol{x}(\ell_\mathrm{N}, t) &= \Delta \boldsymbol{u}_{\ell_\mathrm{N}}(t) , \end{aligned} \tag{3.14}$$

with the coefficient matrices

$$\begin{aligned} \Lambda^j(t) &= \left. \frac{\partial \boldsymbol{\phi}_j}{\partial \boldsymbol{x}} (t, \boldsymbol{x}(z, t), \partial_z \boldsymbol{x}(z, t), \partial_t \boldsymbol{x}(z, t)) \right|_{\substack{\boldsymbol{x} = \boldsymbol{x}^* \\ \partial_z \boldsymbol{x} = \partial_z \boldsymbol{x}^* \\ \partial_t \boldsymbol{x} = \partial_t \boldsymbol{x}^*}} , \\ \Lambda_z^j(t) &= \left. \frac{\partial \boldsymbol{\phi}_j}{\partial (\partial_z \boldsymbol{x})} (t, \boldsymbol{x}(z, t), \partial_z \boldsymbol{x}(z, t), \partial_t \boldsymbol{x}(z, t)) \right|_{\substack{\boldsymbol{x} = \boldsymbol{x}^* \\ \partial_z \boldsymbol{x} = \partial_z \boldsymbol{x}^* \\ \partial_t \boldsymbol{x} = \partial_t \boldsymbol{x}^*}} , \quad z \in \{0, \ell_\mathrm{N}\} \\ \Lambda_t^j(t) &= \left. \frac{\partial \boldsymbol{\phi}_j}{\partial (\partial_t \boldsymbol{x})} (t, \boldsymbol{x}(z, t), \partial_z \boldsymbol{x}(z, t), \partial_t \boldsymbol{x}(z, t)) \right|_{\substack{\boldsymbol{x} = \boldsymbol{x}^* \\ \partial_z \boldsymbol{x} = \partial_z \boldsymbol{x}^* \\ \partial_t \boldsymbol{x} = \partial_t \boldsymbol{x}^*}} , \end{aligned} \tag{3.15}$$

and the inputs

$$\Delta \boldsymbol{u}_j(t) = \boldsymbol{u}_j(t) - \boldsymbol{u}_j^*(t) \tag{3.16}$$

for $j = \{0, \ell_{\mathrm{N}}\}$. Depending on the choice of V and $\boldsymbol{r}$ the generic problem formulation (3.7) can be reduced to the following families of parabolic PDEs, which are frequently examined in this work. It should be noted that the real time application to be considered later is a swarm of wheeled robots which operate in 2D domain described by x^1- and x^2- positions. Consequently, but without loss of generality the dimension of the state vector is $n = 2$ and the associated coefficients matrices are of 2×2.

Linear, Time-variant Diffusion-Convection-Reaction System

With the assumptions $V(z,t,\boldsymbol{x}) = B(z,t)$, $\boldsymbol{r}(z,t,\boldsymbol{x}) = [C(z,t) + \partial_z B(z,t)]\, \boldsymbol{x}(z,t)$, and a constant diffusion coefficient matrix $D(z) = A$ the PDE (3.7) can be reduced to the linear DCRS

$$\frac{\partial \boldsymbol{x}(z,t)}{\partial t} = A\frac{\partial^2 \boldsymbol{x}(z,t)}{\partial z^2} - B(z,t)\frac{\partial \boldsymbol{x}(z,t)}{\partial z} + C(z,t)\boldsymbol{x}(z,t), \quad (z,t) \in \Omega(\ell_{\mathrm{N}}, t_0) \tag{3.17}$$

defined on a 1-dimensional spatial domain z with initial state according to (3.8) and linear boundary conditions equivalently to (3.14).

Assuming $B(z,t) = \operatorname{diag}[(b^{11}(z,t), \cdots, b^{nn}(z,t)]$ to be diagonal (3.17) can be transformed into a new system to eliminate the convection term $B(z,t)\partial_z \boldsymbol{x}(z,t)$. A coordinate change by following the so-called Hopf-Cole state transformation

$$\begin{aligned} \boldsymbol{x}(z,t) \mapsto \boldsymbol{\chi}(z,t) := T^{-1}(z,t)\boldsymbol{x}(z,t)\,, \quad &\text{with} \quad T(z,t) = \exp\left(\Lambda(z,t)\right) \\ &\text{and} \quad \Lambda(z,t) = A^{-1}\int_0^z \frac{B(s,t)}{2}\,\mathrm{d}s\,, \end{aligned} \tag{3.18}$$

transforms the distributed parameters system (3.25) into the coupled linear, time-variant Diffusion-Reaction-System (DR)

$$\frac{\partial \boldsymbol{\chi}(z,t)}{\partial t} = A\frac{\partial^2 \boldsymbol{\chi}(z,t)}{\partial z^2} + \Gamma(z,t)\boldsymbol{\chi}(z,t) \tag{3.19}$$

with the transformed coefficient matrix of the reaction term

$$\Gamma(z,t) = T^{-1}(z,t)C(z,t)T(z,t) - A(\partial_z\Lambda(z,t))^2 + A\partial_z^2\Lambda(z,t) - \partial_t\Lambda(z,t). \tag{3.20}$$

Remark 3.8. *Note, that a constant matrix $\bar{B} = \operatorname{diag}[b^{11}, b^{22}]$ leads to a transformation matrix $T(z)$ which only evolves in space, i.e.,*

$$T(z) = \exp\left(\Lambda(z)\right) = \exp\left(\frac{1}{2}A^{-1}\bar{B}z\right). \tag{3.21}$$

Consequently, in this case a constant coefficient matrix $C(z,t) := \bar{C}$ is transformed into the spatially varying matrix $\Gamma(z) = T^{-1}(z)\bar{C}T(z) - \bar{B}A^{-1}\bar{B}/4$.

Modified, Viscous Burgers' Equation

With $V(z,t,\boldsymbol{x}) = B(z,t)X(z,t)/2$, where X is defined as the state matrix in diagonal form

$$X(z,t) := \operatorname{diag}\left[\boldsymbol{x}(z,t)\right] = \begin{bmatrix} x_1(z,t) & 0 & \cdots & 0 \\ 0 & x_2(z,t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_n(z,t) \end{bmatrix}, \tag{3.22}$$

$\boldsymbol{r}(z,t,\boldsymbol{x}) = \left[C(z,t) + \partial_z B(z,t)X(z,t)/2\right]\boldsymbol{x}(z,t)$, and the constant diagonal matrix $D(z) = A$, then (3.7) yields the MVBE

$$\frac{\partial \boldsymbol{x}(z,t)}{\partial t} = A\frac{\partial^2 \boldsymbol{x}(z,t)}{\partial z^2} - B(z,t)X(z,t)\frac{\partial \boldsymbol{x}(z,t)}{\partial z} + C(z,t)\boldsymbol{x}(z,t), \quad (z,t) \in \Omega(\ell_\mathrm{N}, t_0). \tag{3.23}$$

with the coefficient matrix $B(z,t)$ for the non-linear term $X(z,t)\partial_z \boldsymbol{x}(z,t)$. Note that $\partial_z X(z,t)\boldsymbol{x}(z,t) = X(z,t)\partial_z \boldsymbol{x}(z,t)$ holds true. Again, the initial and boundary conditions can be written in a very generic way as defined in (3.8) and (3.9). The linearisation according to (3.11) applied to the Burgers' equation (BE) results in

$$\begin{aligned}\frac{\partial \Delta\boldsymbol{x}(z,t)}{\partial t} = A\frac{\partial^2 \Delta\boldsymbol{x}(z,t)}{\partial z^2} - B(z,t)X^*(z,t)\frac{\partial \Delta\boldsymbol{x}(z,t)}{\partial z} \\ + \left(C(z,t) - B(z,t)\partial_z X^*(z,t)\right)\Delta\boldsymbol{x}(z,t)\,,\end{aligned} \tag{3.24}$$

with the state vector $\Delta\boldsymbol{x}(z,t) = \boldsymbol{x}(z,t) - \boldsymbol{x}^*(z,t)$, its proper initial state $\Delta\boldsymbol{x}(z,t_0)$ defined in (3.14), and the linear boundary conditions in accordance with the equations (3.14) –(3.16).

Remark 3.9. *Note, the linearisation of* (3.23) *around a desired trajectory leads to a coupled DCRS* (3.17) *with coefficient matrices for the convection and reaction term which evolve in space and time. This applies even for constant $B(z,t) := \bar{B}$ and $C(z,t) := \bar{C}$. In case of diagonal $B(z,t)$* (3.24) *can be simplified to the coupled linear, time-variant DRS* (3.19) *by making use of the Hopf-Cole transformation.*

In the following some properties on the system matrices A, $B(z,t)$, and $C(z,t)$ of the derived distributed parameters systems have to be considered for further proceedings such as stability analysis, steady states and control theoretical design concepts.

Assumption 3.1. *The parameter matrices A, B, and C in* (3.17), *and* (3.23) *shall satisfy the following conditions:*

(i). The entries a^{ii}, $i = 1\ldots n$ of the diagonal diffusion matrix A are positive constants. Moreover, they are bounded with $\bar{a}$ and $\underline{a}$, and shall be arranged in descending order according to $\infty > \bar{a} > a^{11} > a^{22} > \cdots > a^{nn} > \underline{a} > 0$.

(ii). The parameter coefficient matrix B of the convection term in (3.17) *and of the semi-linear term in* (3.23) *shall be in* $\mathbf{CG}^{1,\alpha_B}\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\right)$ *for* $\alpha_B \in [1,2]$*. Moreover, assume* $\boldsymbol{\nu} \in \mathbb{R}^n$ *with* $\|\boldsymbol{\nu}\|_2 = 1$ *then B shall either fulfil* $\boldsymbol{\nu}^T B(z,t)\boldsymbol{\nu} \geq 0$ *or* $\boldsymbol{\nu}^T B(z,t)\boldsymbol{\nu} \leq 0$ *over the entire domain* $(z,t) \in \Omega(\ell_{\mathrm{N}}, t_0)$*.*

(iii). The reaction parameter matrix C shall have elements of $\mathbf{CG}^{0,\alpha_C}\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\right)$ *for* $\alpha_C \in [1,2]$*.*

Note, for the appropriate formulation of the function classes with two variables see Definition 3.12 in Appendix C.2.

Remark 3.10. *Assuming A, B, and C in* (3.17)-(3.23) *as diagonal matrices leads to a simplified problem formulation, where the PDE of each state coordinate $x^i(z,t)$ is completely independent of all other coordinates $x^j(z,t)$, with $j \neq i$. In other words the MA dynamics is uncoupled in each state coordinate $x^i(z,t)$.*

Remark 3.11. *In this work the analysis of the MA dynamics is focused on the 2-dimensional plane, which represents, e.g., a swarm of wheeled robots, a fleet of unmanned ground vehicles, or other objects or distributed processes which can be characterized by a 2-dimensional (2D) state vector $\boldsymbol{x}(z,t)$. As a consequence, though without loss of generality, the number of coordinates is set to $n = 2$. Furthermore, if no other statement is present it is assumed that* $\boldsymbol{x}(z,t) \in \mathbf{L}^2\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, \bar{t}_0\right); \mathbb{R}^2\right)$*.*

The following paragraph deals with continuous problem formulation when the state coordinates x^i are independent of each other as hinted in Remark 3.10). In this case the MA dynamics may be represented by a scalar-valued distributed parameter system for each coordinate i.

3.1.2 Scalar Continuous Problem Formulation using Parabolic Partial Differential Equations

For the sake of completeness and for a comprehensive clarification of the nomenclature the problem formulations derived in Section 3.1.1 are presented for the scalar state x in the following paragraphs.

Scalar Linear, Time-variant Diffusion-Convection-Reaction System

From (3.17) the uncoupled, and hence, scalar-valued DCRS can be written as

$$\frac{\partial x(z,t)}{\partial t} = a\frac{\partial^2 x(z,t)}{\partial z^2} - b(z,t)\frac{\partial x(z,t)}{\partial z} + c(z,t)x(z,t)\,, \quad (z,t) \in \Omega(\ell_{\mathrm{N}}, t_0). \tag{3.25}$$

This structure always can be reduced to a simpler form by eliminating the convection term $b(z,t)\partial_z x(z,t)$. A coordinate change by following the Hopf-Cole state transformation

$$x(z,t) \mapsto \chi(z,t) := \exp\left(-\lambda(z,t)\right)x(z,t) \quad \text{and} \quad \lambda(z,t) = \int_0^z \frac{b(s,t)}{2a}\,\mathrm{d}s \tag{3.26}$$

transforms the distributed parameters system (3.25) into the linear, time-variant DRS

$$\frac{\partial \chi(z,t)}{\partial t} = a\frac{\partial^2 \chi(z,t)}{\partial z^2} + \gamma(z,t)\chi(z,t) \tag{3.27}$$

with the transformed coefficient of the reaction term

$$\gamma(z,t) = c(z,t) - a(\partial_z \lambda(z,t))^2 + a\partial_z^2 \lambda(z,t) - \partial_t \lambda(z,t). \tag{3.28}$$

Scalar Modified, Viscous Burgers' Equation

The scalar-valued formulation of the semi-linear viscous Burgers Equation can be deduced from (3.23) as

$$\frac{\partial x(z,t)}{\partial t} = a\frac{\partial^2 x(z,t)}{\partial z^2} - b(z,t)x(z,t)\frac{\partial x(z,t)}{\partial z} + c(z,t)x(z,t)\,, \quad (z,t) \in \Omega(\ell_{\mathrm{N}}, t_0) \tag{3.29}$$

Especially for the formulation of tracking control problems it is necessary to derive the linearisation of this type of non-linear distributed parameters system around a desired trajectory x^* and its derivative $\partial_t x^*$. With this and the introduction of the state $\Delta x(z,t) = x(z,t) - x^*(z,t)$ the linearisation of the modified viscous Burgers' equation leads to a DCRS of the form

$$\frac{\partial \Delta x(z,t)}{\partial t} = a\frac{\partial^2 \Delta x(z,t)}{\partial z^2} - \beta(z,t)\frac{\partial \Delta x(z,t)}{\partial z} + \gamma(z,t)\Delta x(z,t)\,, \tag{3.30}$$

which is equivalent to (3.25). The yielding parameter coefficients comply with

$$\begin{aligned} \beta(z,t) &= b(z,t)x^*(z,t)\,, \\ \gamma(z,t) &= c(z,t) - b(z,t)\partial_z x^*(z,t)\,. \end{aligned} \tag{3.31}$$

Obviously, recalling the discussion from above the linearised system (3.30) can be reduced to a DRS by utilizing the Hopf-Cole transformation (3.26).

3.1.3 Boundary Control

The generic BCs $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ in equation (3.9) define the control inputs $\boldsymbol{u}_j(t) = [u_j^1(t), u_j^2(t), \ldots, u_j^n(t)]^T$, $j = 0, \ell_{\mathrm{N}}$ of the MA dynamics for all $n \geq 1$ coordinates. In the following the control inputs u_0^i and $u_{\ell_{\mathrm{N}}}^i$ of each coordinate are assumed to depend only on the state x^i and its derivatives with respect to t or z of the same coordinate i. They shall be independent of all other states x^j or their derivatives $\partial_z x^j$ and $\partial_t x^j$, respectively, for coordinates $j \neq i$. As a consequence, it sufficient to proceed with the boundary condition of one single coordinate i. For simplicity the index i is neglected in the further statements. Moreover, for controller design of MASs especially the commonly known Dirichlet BC as well as dynamic BCs are well suited, and hence, discussed in the following paragraph.

Dirichlet Boundary Conditions

Introducing Dirichlet BCs, i.e,

$$x(0,t) = u_0(t), \quad x(\ell_\mathrm{N},t) = u_{\ell_\mathrm{N}}(t) \tag{3.32}$$

implies that the control inputs enforce a certain state of the agents, which are located at the boundaries of the virtual communication path $z \in \{0, \ell_\mathrm{N}\}$. In the sense of a mobile agent, e.g., a wheel-based robot, this means that the control inputs impose a certain position $x(0,t)$ and $x(\ell_\mathrm{N},t)$ of the boundary agents.

Obviously this defines a different dynamical behaviour compared to the dynamics of the other agents along $z \in (0, \ell_\mathrm{N})$. The latter are modelled as integrators, i.e., generally defined by the left-hand side of equation (3.7). The integrators are characterised by the right-hand side of the parabolic PDE, e.g., the uncoupled DCRS (3.25). The approach to standardise or, e.g., to mimic Laplacian control the dynamics of the boundary agents at $z = 0$ and $z = \ell_\mathrm{N}$ with the dynamical behaviour of all other agents along $z \in (0, \ell_\mathrm{N})$, leads to dynamic BCs.

Dynamic Boundary Conditions

As encouraged above, the choice of dynamic boundary conditions, defined as

$$\partial_t x(0,t) = u_0(t), \quad \partial_t x(\ell_\mathrm{N},t) = u_{\ell_\mathrm{N}}(t)\,, \tag{3.33}$$

harmonises the dynamics of all agents along the communication path $z \in [0, \ell_\mathrm{N}]$. In this case all agents are modelled as integrators. Particularly in (3.33) the control inputs are assigned to the integrator models of the boundary agents at $z = \{0, \ell_\mathrm{N}\}$. Pragmatically speaking, this means that the protocol and the control inputs impose a request which is set equal to time derivative $\partial_t x$ of the agent continuum. From a physical point of view the control input signal is equivalent to a requested velocity $v(z,t) = \partial_t x(z,t)$ of the agents.

3.2 Formation Profiles with Constant Model Parameters

A formation or formation profile of a MAS is defined as a certain arrangement or a specific state of equilibrium of the agents. In context of mobile wheeled robots a formation is a desired deployment of the robots in the 2D space. As a consequence, this demands the steady state solutions for[1] $t \geq \bar{t}$, with some $\bar{t} \in \mathbb{R}^+_{t_0}$, of the investigated continuous models. Subsequently, the evaluation of desired formation profiles is discussed for the coupled and uncoupled DCRS and MVBE, respectively. The stationary perspective demands $\partial_t x^i(z,t) \overset{!}{=} 0$ with constant model parameters for $t \geq \bar{t}$, which results in boundary value problems of ordinary differential

[1] This covers a stationary starting profile which requires $t \leq t_0$ as well. Just assume a theoretically reversed timeline, i.e. $t \to -t$, and $\bar{t} = t_0$.

equations. The choice of model parameters combined with the set-up of the boundary values define the steady state solutions of the underlying dynamic models. Hence, the coefficients of the PDE and its boundary values can be used to design desired shapes for formation control. The following sections deal with solutions of the stationary problems for the class of PDE discussed above with constant coefficients.

Steady States of the Coupled Diffusion-Convection-Reaction System

The stationary problem of the linear, time-variant DCRS defined in Section 3.1.1 but with constant coefficient matrices can be written as

$$A\partial_z^2\bar{\boldsymbol{x}}(z) - \bar{B}\partial_z\bar{\boldsymbol{x}}(z) + \bar{C}\bar{\boldsymbol{x}}(z) = \mathbf{0}_2\,, \quad \bar{\boldsymbol{x}}(z) = \left[\bar{x}^1(z)\ \bar{x}^2(z)\right]^T \tag{3.34a}$$

and the boundary conditions may be chosen by

$$\bar{\boldsymbol{x}}(0) = \bar{\boldsymbol{x}}_{\mathtt{a}}\,, \quad \bar{\boldsymbol{x}}(\ell_{\mathrm{N}}) = \bar{\boldsymbol{x}}_{\mathtt{1}}\,. \tag{3.34b}$$

At this stage it is pointed out that even for dynamic BCs the boundary values are imposed by the couple $\{\bar{\boldsymbol{x}}_{\mathtt{a}}, \bar{\boldsymbol{x}}_{\mathtt{1}}\}$ according to (3.34b) since they directly influence the steady solution and consequently the formation profile. Moreover, it is assumed that the coefficient matrix A for the diffusion term is positive definite and the coefficients of the matrices of the convection and reaction term are independent of the spatial domain z and set constant to $\bar{C} := C(z,t)$ and $\bar{B} := B(z,t)$ for $t \geq \bar{t}$. Therefore, they have the following structure

$$A = \begin{bmatrix} a^{11} & 0 \\ 0 & a^{22} \end{bmatrix}, \qquad \bar{B} = \begin{bmatrix} b^{11} & b^{12} \\ b^{21} & b^{22} \end{bmatrix}, \qquad \text{and} \qquad \bar{C} = \begin{bmatrix} c^{11} & c^{12} \\ c^{21} & c^{22} \end{bmatrix}. \tag{3.35}$$

Let us introduce the state vector

$$\bar{\boldsymbol{y}}(z) = \begin{bmatrix} \bar{\boldsymbol{y}}_1(z) \\ \bar{\boldsymbol{y}}_2(z) \end{bmatrix} = \begin{bmatrix} \bar{\boldsymbol{x}}(z) \\ \partial_z\bar{\boldsymbol{x}}(z) \end{bmatrix} \tag{3.36}$$

and substitute it into the stationary equation (3.34a). Then it can be written as the system of linear ODEs of first order

$$\underbrace{\begin{bmatrix} I_2 & 0 \\ 0 & A \end{bmatrix}}_{\Lambda} \begin{bmatrix} \partial_z\bar{\boldsymbol{y}}_1(z) \\ \partial_z\bar{\boldsymbol{y}}_2(z) \end{bmatrix} = \underbrace{\begin{bmatrix} 0_{2,2} & I_2 \\ -\bar{C} & \bar{B} \end{bmatrix}}_{M} \begin{bmatrix} \bar{\boldsymbol{y}}_1(z) \\ \bar{\boldsymbol{y}}_2(z) \end{bmatrix} \tag{3.37}$$

with $\Lambda, M \in \mathbb{R}^{4\times4}$. With this, let us formulate the algebraic eigenvalue problem for the matrix $\bar{M} = \Lambda^{-1}M$ as

$$(\nu\Lambda - M)\boldsymbol{v} = \begin{bmatrix} \nu & 0 & -1 & 0 \\ 0 & \nu & 0 & -1 \\ c^{11} & c^{12} & a^{11}\nu - b^{11} & -b^{12} \\ c^{21} & c^{22} & -b^{21} & a^{22}\nu - b^{22} \end{bmatrix}\boldsymbol{v} = \mathbf{0}_4 \tag{3.38}$$

with the eigenvalue ν and its corresponding eigenvector $\boldsymbol{v}$. Assuming $l \leq 4$ disjunct eigenvalues ν_j, with equal algebraic and geometric multiplicity μ_j, $\sum_{j=1}^{l} = \mu_j = 4$, then the eigenvectors $\boldsymbol{v}_{j_k} = \left[v_{j_k}^1, v_{j_k}^2, v_{j_k}^3, v_{j_k}^4\right]^T$, $k = 1 \ldots \mu_j$, $j = 1 \ldots l$ are linearly independent and provide an eigenbasis for $\mathbb{R}^4$. Then the general solution of the system of first order ODEs can be written as

$$\bar{\boldsymbol{y}}(z) = \sum_{j=1}^{l} \left(\boldsymbol{v}_{j_1} + \boldsymbol{v}_{j_2} z + \cdots + \boldsymbol{v}_{j_{\mu_j}} z^{\mu_j - 1}\right) \exp(\nu_j z)\,. \tag{3.39}$$

Considering (3.36) and relabelling the first two entries of $\boldsymbol{v}_{j_k}$ according to

$$\boldsymbol{\kappa}_{j_k} = \begin{bmatrix} \kappa_{j_k}^1 \\ \kappa_{j_k}^2 \end{bmatrix} = \begin{bmatrix} v_{j_k}^1 \\ v_{j_k}^2 \end{bmatrix}, \tag{3.40}$$

the general solution of the initial problem (3.34) has the same structure as (3.39), i.e.,

$$\bar{\boldsymbol{x}}(z) = \sum_{j=1}^{l} \left(\boldsymbol{\kappa}_{j_1} + \boldsymbol{\kappa}_{j_2} z + \cdots + \boldsymbol{\kappa}_{j_{\mu_j}} z^{\mu_j - 1}\right) \exp(\nu_j z)\,. \tag{3.41}$$

With this, the eigenvalues ν_j can be determined as the four roots of the characteristic polynomial

$$p_4 \nu^4 + p_3 \nu^3 + p_2 \nu^2 + p_1 \nu + p_0 = 0\,, \tag{3.42}$$

which follows from the determinate of the 2 × 2 block matrix [79]

$$\begin{aligned} \det(\nu\Lambda - M) &= \det \begin{bmatrix} \nu I_2 & -I_2 \\ \bar{C} & \nu A - \bar{B} \end{bmatrix} = \det(\nu^2 A - \nu \bar{B} + \bar{C}) \\ &= (a^{11}(\nu^2) - b^{11}\nu + c^{11})(a^{22}(\nu^2) - b^{22}\nu + c^{22}) - (b^{12}\nu - c^{12})(b^{21}\nu - c^{21}) \overset{!}{=} 0\,. \end{aligned} \tag{3.43}$$

In (3.42) the coefficients p_m, $m = 0, \ldots, 4$ are given by

$$\begin{aligned} p_4 &= a^{11}a^{22}\,, \\ p_3 &= -a^{22}b^{11} - a^{11}b22\,, \\ p_2 &= b^{11}b^{22} - b^{12}b^{21} + a^{11}c^{22} + a^{22}c^{11}\,, \\ p_1 &= -b^{11}c^{22} - b^{22}c^{11} + b^{12}c^{21} + b^{21}c^{12}\,, \\ p_0 &= c^{11}c^{22} - c^{12}c^{21}\,. \end{aligned} \tag{3.44}$$

Remark 3.12. *In the following the discussion is focused on parameter combinations for the matrices* (3.35) *which lead to $l = 4$ disjunct eigenvalues ν_j for the matrix $\bar{M}$, i.e, each eigenvalue shall have single algebraic multiplicity $\mu_j = 1$. For the sake of simplicity the subscript k is neglected for the eigenvectors $\boldsymbol{v}_j = \boldsymbol{v}_{j_1}$ and consequently $\boldsymbol{\kappa}_j = \boldsymbol{\kappa}_{j_1}$ with $j = 1 \ldots 4$.*

Taking Remark 3.12 into account the characteristic equation (3.42) has four disjunct solutions $j = 1, ..., 4$ which are either real-valued $\nu_j \in \mathbb{R}$ or appear as conjugated complex pairs

$\{\nu_j, \nu_j^*\} \in \mathbb{C}$. In this case each eigenvalue ν_j and its corresponding eigenvector $\boldsymbol{v}_j$ provide a fundamental solution $\bar{\boldsymbol{y}}_j(z) = \boldsymbol{v}_j \exp(\nu_j z)$ for (3.37). With this, the general solution (3.41) for the boundary problem (3.34) can be expressed as

$$\bar{\boldsymbol{x}}(z) = \sum_{j=1}^{4} \boldsymbol{\kappa}_j \exp(\nu_j z)\,. \tag{3.45}$$

The determination of $\boldsymbol{\kappa}_j$, $j = 1, \ldots, 4$, of dimension 2 require the consideration of the eigenvalue problem (3.38) and the boundary values (3.34b). Substituting the eigenvalues ν_j into (3.38) and considering (3.40) one can derive the dependencies

$$\kappa_j^2 = \frac{a^{11}(\nu_j)^2 - b^{11}\nu_j + c^{11}}{b^{12}\nu_j - c^{12}}\kappa_j^1 = f(\nu_j)\kappa_j^1\,, \quad \text{with} \quad b^{12}\nu_j - c^{12} \neq 0\,, \quad \text{or} \tag{3.46}$$

$$\kappa_j^1 = \frac{a^{22}(\nu_j)^2 - b^{22}\nu_j + c^{22}}{b^{21}\nu_j - c^{21}}\kappa_j^2 = g(\nu_j)\kappa_j^2\,, \quad \text{with} \quad b^{21}\nu_j - c^{21} \neq 0\,, \tag{3.47}$$

respectively, between the components of each vector $\boldsymbol{\kappa}_j$. As a consequence this motivates the ansatz

$$\boldsymbol{\kappa}_j = \kappa_j^1 \begin{bmatrix} 1 \\ f(\nu_j) \end{bmatrix} = \kappa_j^2 \begin{bmatrix} g(\nu_j) \\ 1 \end{bmatrix} \qquad j = 1 \ldots 4\,. \tag{3.48}$$

The remaining scaling of the vectors can be determined from the boundary conditions (3.34b) which provide the equations

$$\begin{aligned} \bar{\boldsymbol{x}}_\mathsf{a} &= \boldsymbol{\kappa}_1 + \boldsymbol{\kappa}_2 + \boldsymbol{\kappa}_3 + \boldsymbol{\kappa}_4 \\ \bar{\boldsymbol{x}}_1 &= \boldsymbol{\kappa}_1 \exp(\nu_1 \ell_\mathrm{N}) + \boldsymbol{\kappa}_2 \exp(\nu_2 \ell_\mathrm{N}) + \boldsymbol{\kappa}_2 \exp(\nu_3 \ell_\mathrm{N}) + \boldsymbol{\kappa}_4 \exp(\nu_4 \ell_\mathrm{N})\,. \end{aligned} \tag{3.49}$$

This leads to the inhomogeneous system of algebraic equations

$$\begin{bmatrix} \bar{x}_\mathsf{a}^1 \\ \bar{x}_\mathsf{a}^2 \\ \bar{x}_1^1 \\ \bar{x}_1^2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ f(\nu_1) & f(\nu_2) & f(\nu_3) & f(\nu_4) \\ \exp(\nu_1 \ell_\mathrm{N}) & \exp(\nu_2 \ell_\mathrm{N}) & \exp(\nu_3 \ell_\mathrm{N}) & \exp(\nu_4 \ell_\mathrm{N}) \\ f(\nu_1)\exp(\nu_1 \ell_\mathrm{N}) & f(\nu_2)\exp(\nu_2 \ell_\mathrm{N}) & f(\nu_3)\exp(\nu_3 \ell_\mathrm{N}) & f(\nu_4)\exp(\nu_4 \ell_\mathrm{N}) \end{bmatrix}}_{\mathcal{M}} \begin{bmatrix} \kappa_1^1 \\ \kappa_2^1 \\ \kappa_3^1 \\ \kappa_4^1 \end{bmatrix}. \tag{3.50}$$

Since $f(\nu_j)$ and $g(\nu_j)$ are non-linear for a positive definite coefficient matrix A and ν_j are assumed to be disjoint, see Remark 3.12, the coefficient matrix is regular[2]. Consequently, let the boundary values $\bar{\boldsymbol{x}}_\mathsf{a}$ and $\bar{\boldsymbol{x}}_1$ be chosen appropriately so that they satisfy

$$\operatorname{rank}(\mathcal{M}) = \operatorname{rank}\left(\mathcal{M} \,\middle|\, \begin{matrix} \bar{\boldsymbol{x}}_\mathsf{a} \\ \bar{\boldsymbol{x}}_1 \end{matrix}\right) = 4\,. \tag{3.51}$$

Then there exists a unique solution for $[\kappa_1^1\,\kappa_2^1\,\kappa_3^1\,\kappa_4^1]^T$. Further investigations and explicit derivations of steady state profiles are summarised in Appendix B.1.

[2] The first three lines of $\mathcal{M}$ are obviously linear independent. The fourth line is surly no linear combination of the first three lines. Similar thoughts apply to the columns.

Steady States of the Scalar Diffusion-Convection-Reaction System

The discussion of steady states for uncoupled, i.e., scalar parabolic systems can be found, e.g., in [30]. Therefore the following paragraphs shall give an comprehensive summary.

The model equation of an scalar DCRS as introduced in Section 3.1.2 can be evaluated for the stationary case equivalent to the discussion before. This leads to

$$a\partial_z^2 \bar{x}(z) - \bar{b}\partial_z \bar{x}(z) + \bar{c}\bar{x}(z) = 0\,, \tag{3.52}$$

with positive diffusion coefficient $a > 0$ as well as constant parameter $\bar{b} := b(z,t)$ and $\bar{c} := c(z,t)$ for some $t \geq \bar{t}$. Moreover, boundary conditions of the form

$$\bar{x}(0) = \bar{x}_{\mathtt{a}}\,, \quad \bar{x}(\ell_{\mathrm{N}}) = \bar{x}_{\mathtt{1}} \tag{3.53}$$

are necessary to compute a unique solution for (3.52). The ansatz with exponential eigenfunctions $\phi(z) = \kappa \exp(\nu z)$ yields the generic analytic solution

$$\bar{x}(z) = \kappa_1 \exp(\nu_1 z) + \kappa_2 \exp(\nu_2 z) \tag{3.54}$$

with generally complex (conjugate) parameters κ_j and ν_j. Considering the eigenfunction ϕ and utilising $\partial_z^k \phi(z) = \nu^k \phi(z), k = 1, 2$ the characteristic equation of (3.52) can be written as

$$a\nu^2 - \bar{b}\nu + \bar{c} = 0\,, \tag{3.55}$$

which is satisfied by the two solutions

$$\nu_{1,2} = \frac{\bar{b}}{2a} \pm \sqrt{\frac{\bar{b}^2}{4a^2} - \frac{\bar{c}}{a}} = \frac{\bar{b}}{2a} \pm \imath \frac{\sqrt{\bar{\gamma}}}{2a}\,. \tag{3.56}$$

Respecting $a > 0$ different cases have to be distinguished depending on the sign of the term $\bar{\gamma} = 4a\bar{c} - \bar{b}^2$. When the expression is positive, i.e., $\bar{\gamma} > 0$, the two solutions $\nu_{1,2}$ form a complex conjugated pair and the ansatz (3.54) can be written as

$$\bar{x}(z) = \kappa_1 \exp(\nu_1 z) + \kappa_1^* \exp(\nu_1^* z) = \exp\left(\sigma z\right)\left(k_{\mathrm{c}} \cos(\mu z) + k_{\mathrm{s}} \sin(\mu z)\right) \tag{3.57}$$

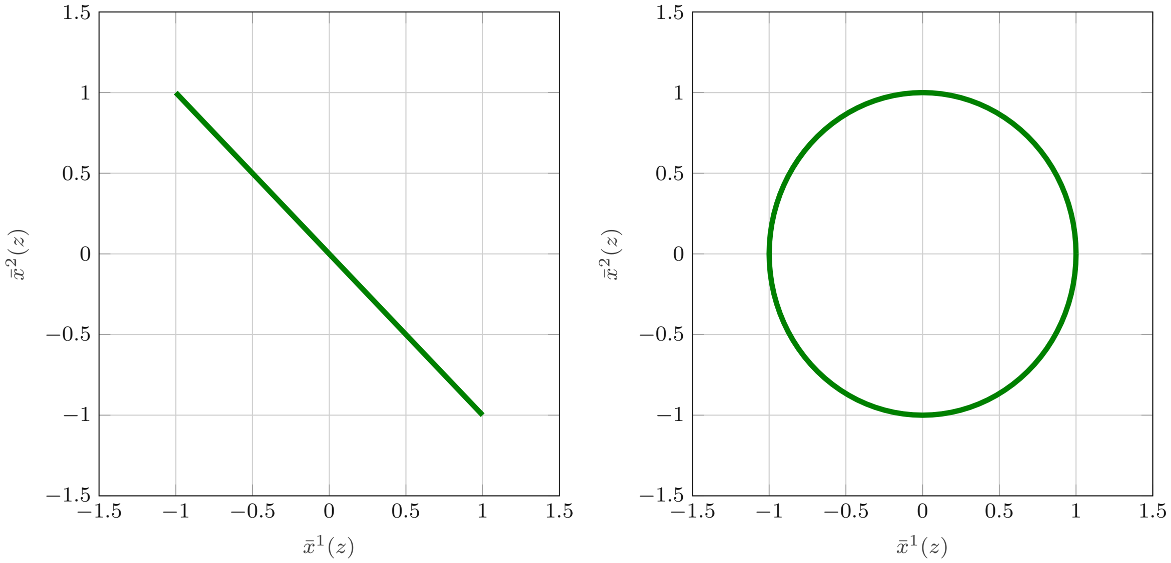
with the parameters $\kappa_1 = (k_{\mathrm{c}} - \imath k_{\mathrm{s}})/2$, $\kappa_2 = \kappa_1^*$, and $\nu_{1,2} = \sigma \pm \imath\mu$ with $\sigma = \bar{b}/(2a)$ and $\mu = \sqrt{\bar{\gamma}}/(2a)$. In case of $\bar{\gamma} < 0$ the solutions (3.56) of the characteristic equation are real valued and satisfy $\nu_{1,2} = \sigma \pm \mu$. Redefining the constants $\kappa_1 = (k_{\mathrm{c}} + k_{\mathrm{s}})/2$ and $\kappa_2 = (k_{\mathrm{c}} - k_{\mathrm{s}})/2$ yields

$$\bar{x}(z) = \frac{1}{2}(k_{\mathrm{c}} + k_{\mathrm{s}}) \exp(\nu_1 z) + \frac{1}{2}(k_{\mathrm{c}} - k_{\mathrm{s}}) \exp(\nu_2 z) = \exp(\sigma z)(k_{\mathrm{c}} \cosh(\mu z) + k_{\mathrm{s}} \sinh(\mu z))\,. \tag{3.58}$$

For the special case $\bar{\gamma} = 0$ and consequently $\nu_{1,2} = \sigma = \bar{b}/(2a)$ the ansatz

$$\bar{x}(z) = \exp(\sigma z)\left(k_0 + k_1 z\right) \tag{3.59}$$

is applicable. The choice of boundary conditions defined in (3.53) renders a system of two conditional equations for the coefficients k_{c} and k_{s}, or k_0 and k_1, respectively. Their further analysis is carried out in Appendix B.2.



(a) Line formation.

(b) Circle formation.

Figure 3.1: Examples of (normalised) formation profiles for the uncoupled DRS. The plots show a simple line formation in (a) and a circular profile in (b).

Remark 3.13. *Recalling the investigations on the equivalence between the scalar DCRS and the scalar DRS of the form* (3.25) *and* (3.27)*, respectively the same correspondence may be applied for the steady states. The determination of a steady state solution for* (3.52) *is equivalent to problem*

$$\begin{aligned} &a\partial_z^2\bar{\chi}(z) + \bar{\gamma}\bar{\chi}(z) = 0\,, \\ &\bar{\chi}(0) = \bar{x}_{\mathtt{a}}\,, \\ &\bar{\chi}(\ell_{\mathrm{N}}) = \exp\left(-\sigma\ell_{\mathrm{N}}\right)\bar{x}_{\mathtt{1}}\,. \end{aligned} \tag{3.60}$$

where $\bar{\gamma}$ *is exactly the transformed reaction term coefficient* (3.28) *when all parameters are chosen to be constant. Moreover, the term* $\exp(\sigma z)$ *represents the Hopf-Cole-transformation* (3.26) *with the correspondence* $\bar{x}(z) = \exp\left(\sigma z\right)\bar{\chi}(z)$*. Therefore the ansatz for the steady state solution of* (3.60) *is given by*

$$\bar{\chi}(z) = \exp\left(-\sigma z\right)\left(\kappa_1\exp(\nu_1 z) + \kappa_2\exp(\nu_2 z)\right) = \kappa_1\exp(\imath\mu z) + \kappa_2\exp(-\imath\mu z) \tag{3.61}$$

As an example a simple line formation and a circular profile are pictured in Figure 3.1. The line formation follows from the ansatz (3.59) for each of both coordinates $\bar{x}^1$ and $\bar{x}^2$ and their subsequent superposition in the 2D space. Accordingly, the circular shape can be established by the overlay of two steady state solutions derived in (3.57). The parameters for the underlying boundary value problem (3.52) are summarised in Table 3.1. With this it is easy to see that the setting of $\bar{x}_{\mathtt{a}}$ and $\bar{x}_{\mathtt{1}}$ to ±1 or 0, respectively, as well as $\ell_{\mathrm{N}} = 1$ normalise the deployment shapes inside the 2D plane.

Table 3.1: Parameters of the line and circular formation profiles with $\ell_{\mathrm{N}} = 1$.

Profile	Coord.	a	$\bar{b}$	$\bar{c}$	$\bar{x}_{\mathtt{a}}$	$\bar{x}_{\mathtt{1}}$
Line	(1) :	1	0	0	-1	1
	(2) :	1	0	0	1	-1
Circle	(1) :	1	0	$4\pi^2$	1	-1
	(2) :	1	0	$4\pi^2$	0	0

Table 3.2: Parameters of the *U-* and *Z-shaped* formation profiles with $\ell_{\mathrm{N}} = 1$.

Profile	Coord.	a	$\bar{b}$	$\bar{c}$	$\bar{x}_{\mathtt{a}}$	$\bar{x}_{\mathtt{1}}$
U-shape	(1) :	1	20	0	-1	-1
	(2) :	1	0	$4\pi^2$	1	-1
Z-shape	(1) :	1	20	0	0	0
	(2) :	1	0	$4\pi^2$	1	-1

Steady States of the Modified Viscous Burgers' Equation

The discussion about stationary formation profiles of the MVBE leads to the boundary value problem

$$A\partial_z^2 \bar{\boldsymbol{x}}(z) - \bar{B}\bar{\boldsymbol{x}}(z)\partial_z \bar{\boldsymbol{x}}(z) + \bar{C}\bar{\boldsymbol{x}}(z) = \mathbf{0}_2\,, \tag{3.62}$$

$$\bar{\boldsymbol{x}}(0) = \bar{\boldsymbol{x}}_{\mathtt{a}}\,, \quad \bar{\boldsymbol{x}}(\ell_{\mathrm{N}}) = \bar{\boldsymbol{x}}_{\mathtt{1}} \tag{3.63}$$

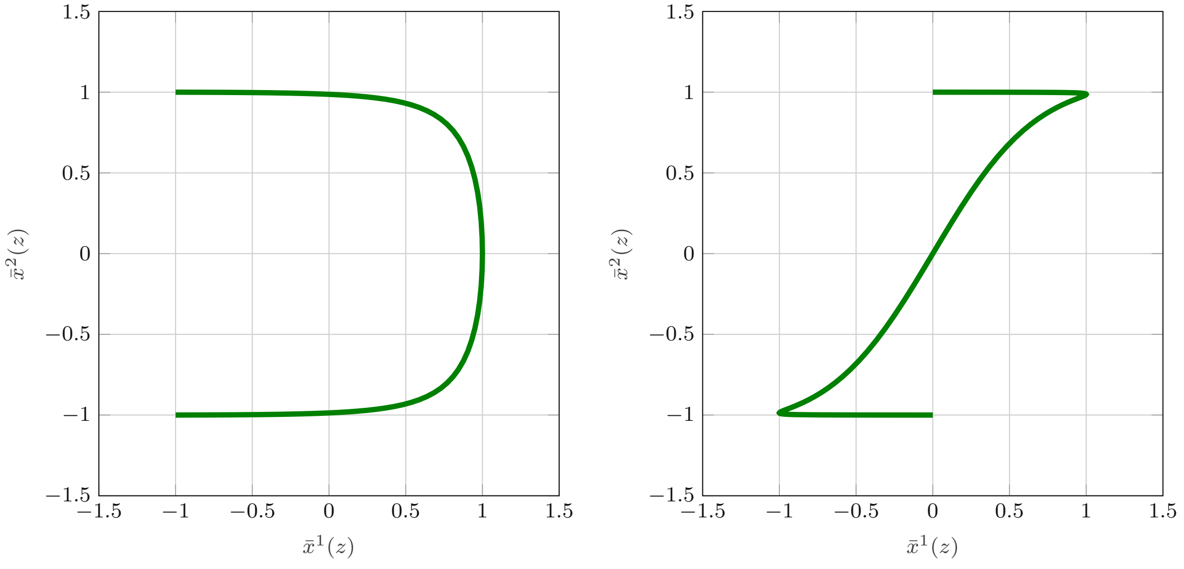
with positive definite diffusion coefficient matrix A, constant coefficient matrix $\bar{B} := B(z,t)$ for the semi-linear term as well as constant reaction parameter matrix $\bar{C} := C(z,t)$ for $t \geq \bar{t}$ with some $\bar{t} \in \mathbb{R}^+_{t_0}$. The corresponding scalar-valued problem can be written as

$$a\partial_z^2 \bar{x}(z) - \bar{b}\bar{x}(z)\partial_z \bar{x}(z) + \bar{c}\bar{x}(z) = 0\,, \tag{3.64}$$

$$\bar{x}(0) = \bar{x}_{\mathtt{a}}\,, \quad \bar{x}(\ell_{\mathrm{N}}) = \bar{x}_{\mathtt{1}} \tag{3.65}$$

with positive and constant diffusion coefficient a and the constant coefficients $\bar{b}$ and $\bar{c}$. In general no analytic solution can be derived for non-linear boundary value problems, i.e., neither for (3.62) nor for (3.64). Possible solutions may be calculated by means of suitable software tools for numerical computing. Figure 3.2 shows examples of possible formation profiles for uncoupled PDEs. In particular, a DRS is applied to coordinate $\bar{x}^1$ which is combined with a MVBE in coordinate $\bar{x}^2$. Analysing the parameter configuration in Table 3.2 clearly illustrates that the modification of the boundary values $\bar{x}_{\mathtt{a}}$ and $\bar{x}_{\mathtt{1}}$ can lead to completely different deployments. Again, the specific setting of $\bar{x}_{\mathtt{a}}$ and $\bar{x}_{\mathtt{1}}$ to ±1 or 0, and $\ell_{\mathrm{N}} = 1$ normalises the shape of the steady state formation profiles geometrically inside the $(x^1, x^2)-$plane.

This closes the discussion regarding continuous problem formulation for MAS dynamics and the explicit computation of steady state formation profiles in case of constant model parameters.



(a) *U-shaped* formation.

(b) *Z-shaped* formation.

Figure 3.2: Examples of (normalised) formation profiles for the modified viscous BE. The plots show a formation like the letter *U* in (a) and a *Z-shaped* profile in (b).

The following two chapters develop a so-called *2DOF* control strategy whereas the upcoming paragraph cope with the first part including feedforward control (FFC) and desired trajectory assignment.

Chapter 4

Motion Planning for the Agent Continuum

For the problem formulation (3.7) a 2DOF control concept is considered, which is composed of a flatness-based FF control approach and a backstepping based control for the tracking error dynamics. This chapter particularly concentrates on the design of the FF term, i.e. control inputs $\boldsymbol{u}^*{}_0$ and $\boldsymbol{u}^*_{\ell_\mathrm{N}}$ which, when applied to the system (3.7), can realise a desired state trajectory $\boldsymbol{x}^*$. In terms of MAS the FF control part shall guide the agents along $\boldsymbol{x}^*$ from a starting deployment $\bar{\boldsymbol{x}}_{t_0}$ to a different target formation profile $\bar{\boldsymbol{x}}_{t_0+\tau_{\boldsymbol{y}}}$ within the finite time interval $\tau_{\boldsymbol{y}}$. This requires to introduce a so-called flat output $\boldsymbol{y}$ which allows a formal state and input parametrisation of the system state $\boldsymbol{x}$ and the inputs $\boldsymbol{u}_0$ and $\boldsymbol{u}_{\ell_\mathrm{N}}$, respectively. This methodology originally comes from *differential flatness*, a concept introduced by Fliess and co-workers [22] for finite-dimensional linear and non-linear systems. It allows to compute the systems states and the control inputs in terms of the flat output and its time derivatives up to a certain order $\beta+1$. Therefore the following states the property for finite-dimensional dynamic systems in a formal way.

Definition 4.4. *A dynamic system* $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ *is called* differentially flat, *if there exists a flat output* $\boldsymbol{y} = \boldsymbol{\theta}(\boldsymbol{x}, \boldsymbol{u})$ *with* $\dim \boldsymbol{y} = \dim \boldsymbol{u}$ *so that* $\boldsymbol{x} = \boldsymbol{\theta}_{\boldsymbol{x}}\left(\boldsymbol{y}, \dot{\boldsymbol{y}}, \ldots, \boldsymbol{y}^{(\beta)}\right)$ *and* $\boldsymbol{u} = \theta_{\boldsymbol{u}}\left(\boldsymbol{y}, \dot{\boldsymbol{y}}, \ldots, \boldsymbol{y}^{(\beta+1)}\right)$.

However, the approach was eventually transferred to infinite-dimensional systems governed by PDE, e.g., see [75, 77, 54]. Based on this, the formal state and input parametrisation strategy reduces the entire FF control synthesis to the design problem of a desired trajectory $\boldsymbol{y}^*$.

4.1 Formal State and Input Parametrisation for Parabolic Partial Differential Equations

In the following a formal parametrisation of the state $\boldsymbol{x}$ and the inputs $\boldsymbol{u}_0$, $\boldsymbol{u}_{\ell_\mathrm{N}}$ in terms of a so-called flat output is determined. Building upon earlier works such as [1, 52], [54]−[61] as well as [76, 77], subsequently two inputs on opposite boundaries of the domain are considered and integrated into the formal integration approach. For this, the parabolic PDE in (3.7) is solved for $\partial_z^2 \boldsymbol{x}(z,t)$ and integrated twice with respect to z. Consequently the formal integration of (3.7) for $0 \leq z \leq \xi$ yields

$$\partial_z \boldsymbol{x}(z,t) = \partial_z \boldsymbol{x}(\xi,t) - \int_z^\xi \boldsymbol{f}(\sigma, t, \boldsymbol{x}, \partial_t \boldsymbol{x}, \partial_\sigma \boldsymbol{x})\, \mathrm{d}\sigma\,. \tag{4.1}$$

with $\partial_z \boldsymbol{x}(\xi,t) := \partial_z \boldsymbol{x}(z,t)|_{z=\xi}$ and

$$\boldsymbol{f}(z,t,\boldsymbol{x},\partial_t \boldsymbol{x},\partial_z \boldsymbol{x}) = D^{-1}(z)\left(\partial_t \boldsymbol{x}(z,t) - \partial_z\left[V(z,t,\boldsymbol{x})\boldsymbol{x}(z,t)\right] - \boldsymbol{r}(z,t,\boldsymbol{x})\right) \tag{4.2}$$

Integrating (4.1) again gives the implicit formula

$$\boldsymbol{x}(z,t) = \boldsymbol{x}(\xi,t) + \partial_z \boldsymbol{x}(\xi,t)(z-\xi) + \int_z^\xi \int_\eta^\xi \boldsymbol{f}(\sigma,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_\sigma\boldsymbol{x})\,\mathrm{d}\sigma\,\mathrm{d}\eta \tag{4.3}$$

for the calculation of the state vector $\boldsymbol{x}$. Furthermore, applying the same procedure for $\xi \leq z \leq \ell_{\mathrm{N}}$ leads to

$$\partial_z \boldsymbol{x}(z,t) = \partial_z \boldsymbol{x}(\xi,t) + \int_\xi^z \boldsymbol{f}(\sigma,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_\sigma\boldsymbol{x})\,\mathrm{d}\sigma \tag{4.4}$$

$$\boldsymbol{x}(z,t) = \boldsymbol{x}(\xi,t) + \partial_z \boldsymbol{x}(\xi,t)(z-\xi) + \int_\xi^z \int_\xi^\eta \boldsymbol{f}(\sigma,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_\sigma\boldsymbol{x})\,\mathrm{d}\sigma\,\mathrm{d}\eta\,, \tag{4.5}$$

which is the identical expression when the boundaries of the integrals in (4.1) and (4.3) are interchanged appropriately. Thus implicit formal parametrisations for $\boldsymbol{x}(z,t)$ and $\partial_z\boldsymbol{x}(z,t)$ are given by

$$\boldsymbol{x}(z,t) = \boldsymbol{y}_0(t) + \boldsymbol{y}_1(t)(z-\xi) + \int_\xi^z \int_\xi^\eta \boldsymbol{f}(\sigma,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_\sigma\boldsymbol{x})\,\mathrm{d}\sigma\,\mathrm{d}\eta \tag{4.6a}$$

$$\partial_z\boldsymbol{x}(z,t) = \boldsymbol{y}_1(t) + \int_\xi^z \boldsymbol{f}(\sigma,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_\sigma\boldsymbol{x})\,\mathrm{d}\sigma \tag{4.6b}$$

with the so-called flat output

$$\left[\boldsymbol{y}_0(t),\ \boldsymbol{y}_1(t)\right]^T = \left[\boldsymbol{x}(\xi,t)\,, \partial_z\boldsymbol{x}(\xi,t)\right]^T\,. \tag{4.7}$$

At this stage it has to be pointed out that for the following sequences of functions it is presumed that their limits exist and they are uniformly convergent. With this it is possible to render the implicit parametrisation (4.6) explicit and derive a recursive algorithm. For a rigorous proof the assumptions need to be verified subsequently for specific assignments of $\boldsymbol{f}$. Therefore, consider the sequences of functions $(\boldsymbol{x}_m(z,t))_{m\in\mathbb{N}}$, $(\partial_t\boldsymbol{x}_m(z,t))_{m\in\mathbb{N}}$, and $(\partial_z\boldsymbol{x}_m(z,t))_{m\in\mathbb{N}}$ with their limits

$$\begin{aligned}
&\lim_{m\to\infty} \boldsymbol{x}_m(z,t) = \boldsymbol{x}(z,t)\,,\\
&\lim_{m\to\infty} \partial_t\boldsymbol{x}_m(z,t) = \partial_t\boldsymbol{x}(z,t)\,,\\
&\lim_{m\to\infty} \partial_z\boldsymbol{x}_m(z,t) = \partial_z\boldsymbol{x}(z,t)\,.
\end{aligned} \tag{4.8}$$

With respect to the definition of $\boldsymbol{f}$ in (4.2), assume the terms $V(z,t,\boldsymbol{x})$ and $\boldsymbol{r}(z,t,\boldsymbol{x})$ are continuous in $\boldsymbol{x}$. Moreover, as mentioned above consider the sequences (4.8) as well as

$$
\begin{aligned}
&\lim_{m\to\infty} \boldsymbol{r}(z,t,\boldsymbol{x}_m) = \boldsymbol{r}(z,t,\boldsymbol{x})\,,\\
&\lim_{m\to\infty} V(z,t,\boldsymbol{x}_m) = V(z,t,\boldsymbol{x})\,,\\
&\lim_{m\to\infty} \partial_z V(z,t,\boldsymbol{x}_m) = \partial_z V(z,t,\boldsymbol{x})\,,
\end{aligned}
\tag{4.9}
$$

all to be uniformly convergent. Then the substitution of (4.2) into (4.6) gives

$$
\begin{aligned}
\lim_{m\to\infty} \partial_z \boldsymbol{x}_m(z,t) &= \boldsymbol{y}_1(t) + \int_\xi^z \boldsymbol{f}\left(\sigma, t, \lim_{m\to\infty}\boldsymbol{x}_m, \lim_{m\to\infty}\partial_t\boldsymbol{x}_m, \lim_{m\to\infty}\partial_\sigma\boldsymbol{x}_m\right) \mathrm{d}\sigma\\
&= \boldsymbol{y}_1(t) + \lim_{m\to\infty}\int_\xi^z \boldsymbol{f}\left(\sigma, t, \boldsymbol{x}_m, \partial_t\boldsymbol{x}_m, \partial_\sigma\boldsymbol{x}_m\right) \mathrm{d}\sigma\,,\\
\lim_{m\to\infty} \boldsymbol{x}_m(z,t) &= \boldsymbol{y}_0(t) + \boldsymbol{y}_1(t)(z-\xi)\\
&\quad + \int_\xi^z\int_\xi^\eta \boldsymbol{f}\left(\sigma, t, \lim_{m\to\infty}\boldsymbol{x}_m, \lim_{m\to\infty}\partial_t\boldsymbol{x}_m, \lim_{m\to\infty}\partial_\sigma\boldsymbol{x}_m\right) \mathrm{d}\sigma\,\mathrm{d}\eta\\
&= \boldsymbol{y}_0(t) + \boldsymbol{y}_1(t)(z-\xi) + \lim_{m\to\infty}\int_\xi^z\int_\xi^\eta \boldsymbol{f}\left(\sigma, t, \boldsymbol{x}_m, \partial_t\boldsymbol{x}_m, \partial_\sigma\boldsymbol{x}_m\right) \mathrm{d}\sigma\,\mathrm{d}\eta\,.
\end{aligned}
\tag{4.10}
$$

Furthermore, this and the assignment of the initial values $\boldsymbol{x}_1(z,t) = \boldsymbol{y}_0(t) + \boldsymbol{y}_1(t)(z-\xi)$, $\partial_z\boldsymbol{x}_1(z,t) = \boldsymbol{y}_1(t)$ allows to construct the recursive algorithm

$$
\begin{aligned}
\boldsymbol{x}_m(z,t) &= \int_\xi^z\int_\xi^\eta \boldsymbol{f}(\sigma, t, \boldsymbol{x}_{m-1}, \partial_t\boldsymbol{x}_{m-1}, \partial_\sigma\boldsymbol{x}_{m-1})\,\mathrm{d}\sigma\,\mathrm{d}\eta\,,\\
\partial_z\boldsymbol{x}_m(z,t) &= \int_\xi^z \boldsymbol{f}(\sigma, t, \boldsymbol{x}_{m-1}, \partial_t\boldsymbol{x}_{m-1}, \partial_\sigma\boldsymbol{x}_{m-1})\,\mathrm{d}\sigma
\end{aligned}
\tag{4.11}
$$

for $m \geq 2$. Respecting uniform convergence of the sequences (4.8) the time derivative can be computed by formal differentiation either of the the limiting value $\boldsymbol{x}(z,t)$ or directly by applying the operation to $\boldsymbol{x}_m(z,t)$. In this context the inputs for dynamic boundary control may be written as

$$
\boldsymbol{u}_0(t) = \partial_t\left(\lim_{m\to\infty}\boldsymbol{x}_m(0,t)\right) = \lim_{m\to\infty}\partial_t\boldsymbol{x}_m(0,t)\,, \tag{4.12a}
$$

$$
\boldsymbol{u}_{\ell_\mathrm{N}}(t) = \partial_t\left(\lim_{m\to\infty}\boldsymbol{x}_m(\ell_\mathrm{N},t)\right) = \lim_{m\to\infty}\partial_t\boldsymbol{x}_m(\ell_\mathrm{N},t)\,. \tag{4.12b}
$$

In contrast to the ansatz (4.8) where uniformly convergent sequences are applied, additional properties on $V(z,t,\boldsymbol{x})$ and $\boldsymbol{r}(z,t,\boldsymbol{x})$ allow an alternative approach using series to solve the implicit equations (4.6). As before, convergence properties are assumed for the ansatz to derive the recursive algorithm. Afterwards the hypothesis need to be proven for the particular assignments of $\boldsymbol{f}$. Thus, presume the infinite sums

$$\boldsymbol{x}(z,t) = \sum_{m=0}^{\infty} \boldsymbol{x}_m(z,t)\,, \tag{4.13a}$$

$$\partial_t \boldsymbol{x}(z,t) = \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m(z,t)\,, \tag{4.13b}$$

$$\partial_z \boldsymbol{x}(z,t) = \sum_{m=0}^{\infty} \partial_z \boldsymbol{x}_m(z,t)\,, \tag{4.13c}$$

to be uniformly convergent and look at the following premise.

Assumption 4.2. *Respecting* (4.2) *assume* $V(z,t,\boldsymbol{x})$ *and* $\boldsymbol{r}(z,t,\boldsymbol{x})$ *are defined such that* $\boldsymbol{f}$ *satisfies*

$$\begin{aligned}\boldsymbol{f}\left(z,t,\sum_{m=2}^{\infty}\boldsymbol{x}_{m-2},\sum_{m=2}^{\infty}\partial_t\boldsymbol{x}_{m-2},\sum_{m=2}^{\infty}\partial_z\boldsymbol{x}_{m-1}\right)&\\ =\sum_{m=2}^{\infty}\boldsymbol{g}_m\left(z,t,\boldsymbol{x}_0,\dots,\boldsymbol{x}_{m-2},\partial_t\boldsymbol{x}_{m-2},\partial_z\boldsymbol{x}_1,\dots,\partial_z\boldsymbol{x}_{m-1}\right)&\end{aligned} \tag{4.14}$$

and the series on the right hand side of (4.14) *is uniformly convergent. If* $\boldsymbol{f}$ *is linear in* $\boldsymbol{x}$, $\partial_t\boldsymbol{x}$ *and* $\partial_z\boldsymbol{x}$ *then obviously* $\boldsymbol{f}=\boldsymbol{g}_m=\boldsymbol{g}$ *and* $\boldsymbol{g}$ *may be written as* $\boldsymbol{g}\left(z,t,\boldsymbol{x}_{m-2},\partial_t\boldsymbol{x}_{m-2},\partial_z\boldsymbol{x}_{m-1}\right)$.

Then the substitution of (4.13) into (4.6) gives

$$\begin{aligned}\sum_{m=0}^{\infty}\boldsymbol{x}_m(z,t) &= \boldsymbol{y}_0(t)+\boldsymbol{y}_1(t)(z-\xi)\\ &\quad+\int_\xi^z\int_\xi^\eta \boldsymbol{f}\left(\sigma,t,\sum_{m=0}^{\infty}\boldsymbol{x}_m,\sum_{m=0}^{\infty}\partial_t\boldsymbol{x}_m,\sum_{m=0}^{\infty}\partial_\sigma\boldsymbol{x}_m\right)\mathrm{d}\sigma\,\mathrm{d}\eta\,,\\ \sum_{m=0}^{\infty}\partial_z\boldsymbol{x}_m(z,t) &= \boldsymbol{y}_1(t)+\int_\xi^z \boldsymbol{f}\left(\sigma,t,\sum_{m=0}^{\infty}\boldsymbol{x}_m,\sum_{m=0}^{\infty}\partial_t\boldsymbol{x}_m,\sum_{m=0}^{\infty}\partial_\sigma\boldsymbol{x}_m\right)\mathrm{d}\sigma\,.\end{aligned} \tag{4.15}$$

and allows the assignment of the initial series coefficients

$$\begin{aligned}\boldsymbol{x}_0(z,t) &= \boldsymbol{y}_0(t)\,, & \partial_z\boldsymbol{x}_0(z,t) &= 0\,,\\ \boldsymbol{x}_1(z,t) &= \boldsymbol{y}_1(t)(z-\xi)\,, & \partial_z\boldsymbol{x}_1(z,t) &= \boldsymbol{y}_1(t)\,.\end{aligned} \tag{4.16}$$

From this, and taking Assumption 4.2 into account the recursive part may be written as

$$
\begin{aligned}
\sum_{m=2}^{\infty} \boldsymbol{x}_m(z,t) &= \int_{\xi}^{z} \int_{\xi}^{\eta} \boldsymbol{f}\left(\sigma, t, \sum_{m=0}^{\infty} \boldsymbol{x}_m, \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m, \sum_{m=1}^{\infty} \partial_\sigma \boldsymbol{x}_m\right) \mathrm{d}\sigma\, \mathrm{d}\eta \\
&= \int_{\xi}^{z} \int_{\xi}^{\eta} \boldsymbol{f}\left(\sigma, t, \sum_{m=2}^{\infty} \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_t \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_\sigma \boldsymbol{x}_{m-1}\right) \mathrm{d}\sigma\, \mathrm{d}\eta \\
&= \sum_{m=2}^{\infty} \int_{\xi}^{z} \int_{\xi}^{\eta} \boldsymbol{g}_m(\sigma, t, \boldsymbol{x}_0, \ldots, \boldsymbol{x}_{m-2}, \partial_t \boldsymbol{x}_{m-2}, \partial_\sigma \boldsymbol{x}_1, \ldots, \partial_\sigma \boldsymbol{x}_{m-1})\, \mathrm{d}\sigma\, \mathrm{d}\eta\,, \\
\sum_{m=2}^{\infty} \partial_z \boldsymbol{x}_m(z,t) &= \int_{\xi}^{z} \boldsymbol{f}\left(\sigma, t, \sum_{m=0}^{\infty} \boldsymbol{x}_m, \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m, \sum_{m=1}^{\infty} \partial_\sigma \boldsymbol{x}_m\right) \mathrm{d}\sigma \\
&= \int_{\xi}^{z} \boldsymbol{f}\left(\sigma, t, \sum_{m=2}^{\infty} \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_t \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_\sigma \boldsymbol{x}_{m-1}\right) \mathrm{d}\sigma \\
&= \sum_{m=2}^{\infty} \int_{\xi}^{z} \boldsymbol{g}_m(\sigma, t, \boldsymbol{x}_0, \ldots, \boldsymbol{x}_{m-2}, \partial_t \boldsymbol{x}_{m-2}, \partial_\sigma \boldsymbol{x}_1, \ldots, \partial_\sigma \boldsymbol{x}_{m-1})\, \mathrm{d}\sigma\,.
\end{aligned}
\tag{4.17}
$$

This motivates the recursive computation of the series coefficients $\boldsymbol{x}_m$ summarised by the formalism

$$
\begin{aligned}
\boldsymbol{x}_m(z,t) &= \int_{\xi}^{z} \int_{\xi}^{\eta} \boldsymbol{g}_m(\sigma, t, \boldsymbol{x}_0, \ldots, \boldsymbol{x}_{m-2}, \partial_t \boldsymbol{x}_{m-2}, \partial_\sigma \boldsymbol{x}_1, \ldots, \partial_\sigma \boldsymbol{x}_{m-1})\, \mathrm{d}\sigma\, \mathrm{d}\eta\,, \\
\partial_z \boldsymbol{x}_m(z,t) &= \int_{\xi}^{z} \boldsymbol{g}_m(\sigma, t, \boldsymbol{x}_0, \ldots, \boldsymbol{x}_{m-2}, \partial_t \boldsymbol{x}_{m-2}, \partial_\sigma \boldsymbol{x}_1, \ldots, \partial_\sigma \boldsymbol{x}_{m-1})\, \mathrm{d}\sigma
\end{aligned}
\tag{4.18}
$$

for $m \geq 2$ together with the initial series coefficients according to (4.16). With this, the control inputs for Dirichlet boundary conditions are inherently given by

$$\boldsymbol{u}_0(t) = \sum_{m=0}^{\infty} \boldsymbol{x}_m(0,t)\,, \tag{4.19a}$$

$$\boldsymbol{u}_{\ell_\mathrm{N}}(t) = \sum_{m=0}^{\infty} \boldsymbol{x}_m(\ell_\mathrm{N},t)\,. \tag{4.19b}$$

Again, assuming uniform convergence of the series (4.13) the determination of $\partial_t \boldsymbol{x}$ can be achieved by formal differentiation of the infinite sum (4.13a) or by deriving it from each single series coefficient $\boldsymbol{x}_m$. As a consequence the inputs for dynamic boundary control can be written as

$$\boldsymbol{u}_0(t) = \partial_t \left(\sum_{m=0}^{\infty} \boldsymbol{x}_m(0,t)\right) = \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m(0,t)\,, \tag{4.20a}$$

$$\boldsymbol{u}_{\ell_\mathrm{N}}(t) = \partial_t \left(\sum_{m=0}^{\infty} \boldsymbol{x}_m(\ell_\mathrm{N},t)\right) = \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m(\ell_\mathrm{N},t)\,. \tag{4.20b}$$

Note, from (4.18) follows for the state vector $\boldsymbol{x}$ at the spatial position $z = \xi$

$$\begin{aligned} \boldsymbol{x}(\xi, t) &= \boldsymbol{x}_0(\xi, t) = \boldsymbol{y}_0(t) \\ \partial_z \boldsymbol{x}(\xi, t) &= \partial_z \boldsymbol{x}_1(\xi, t) = \boldsymbol{y}_1(t) \end{aligned} \tag{4.21}$$

and obviously

$$\begin{aligned} \boldsymbol{x}_m(\xi, t) &= 0 \quad \text{for } m \geq 1\,, \\ \partial_z \boldsymbol{x}_m(\xi, t) &= 0 \quad \text{for } m = 0 \text{ and } m \geq 2\,. \end{aligned} \tag{4.22}$$

The analysis (4.6)−(4.20) reveals that both, the state $\boldsymbol{x}$ and the inputs $\boldsymbol{u}_{\{0,\ell\}}$ are formally parametrised by the temporal output function (4.7), or so-called flat output $\boldsymbol{y}(t) = [\boldsymbol{y}_0(t),\, \boldsymbol{y}_1(t)]^T$. As mentioned before the FF control design is therefore reduced to a trajectory planning problem of the components $\boldsymbol{y}_0$ and $\boldsymbol{y}_1$. This leads to two further points for the discussion in the following paragraphs. First, it is necessary to proof uniform convergence for the determined parametrisations. Here this is conducted for the series ansatz (4.13) since $\boldsymbol{f}$ respects Assumption 4.2 when the terms $V(z, t, \boldsymbol{x})$ and $\boldsymbol{r}(z, t, \boldsymbol{x})$ are defined explicitly for the DCRS or the MVBE, respectively. The analysis imposes requirements for the assignment of desired trajectories $\boldsymbol{y}^*$ and the coefficient matrices of the PDEs given by (3.17) and (3.23). Second, the planning of admissible trajectories for the flat output is worked out subsequently. Therefore, the further discussion addresses the uniform and absolute convergence of (4.13) for the investigated systems.

4.2 Convergence Analysis

In the following convergence analysis is basically conducted in equivalent fashion as published in previous work, such as [55, 59]. However this approach focuses on a scalar spatial domain z but it extends the formalism for coupled parabolic PDEs. Moreover, in the following the flat output may be assigned at any arbitrary position ξ within the spatial domain $[0, \ell_{\mathrm{N}}]$. It is not necessarily fixed to the spatial boundary as in [62, 55]. Regardless of that, the achievements of the corresponding literature demand that the components $\boldsymbol{y}_j$, $j = 0, 1$ of the flat output $\boldsymbol{y}$ are presumed in so-called Gevrey classes $\mathbf{G}^{\delta_{\boldsymbol{y}_j}}_{\alpha_{\boldsymbol{y}_j}}\big(\mathbb{R}^+_{t_0}; \mathbb{R}^n\big)$ as well as the coefficient matrices B and C being in the space $\mathbf{CG}^{0,\alpha_{\{B,C\}}}\big(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\big)$, respectively. The mathematical introduction of Gevrey classes for scalar and matrix-valued functions can be found in Appendix C.2. Moreover consider Theorem 3.17 and Corollary 3.5 which state upper bound estimations for functions in in two independent arguments, i.e., the domain (z, t). Therefore the next sections discuss the analysis of (4.13) regarding uniform and absolute convergence in appropriate Gevrey classes for the coupled DCRS and the coupled MVBE, starting with the former in the following section.

4.2.1 Absolute and Uniform Convergence Analysis for the Coupled Diffusion-Convection-Reaction System

Observing the recursive formalism (4.18) together with Assumption 4.2 and the PDE (3.17) of the coupled linear, time-variant DCRS the function $\boldsymbol{g}$ is formally equal to $\boldsymbol{f}$. Consequently, it can be set to

$$\begin{aligned}\boldsymbol{g}(z,t,\boldsymbol{x}_{m-2},\partial_t\boldsymbol{x}_{m-2},\partial_z\boldsymbol{x}_{m-1}) = A^{-1}\left[\partial_t\boldsymbol{x}_{m-2}(z,t) + B(z,t)\partial_z\boldsymbol{x}_{m-1}(z,t)\right.\\ \left.-C(z,t)\boldsymbol{x}_{m-2}(z,t)\right]\,. \qquad (4.23)\end{aligned}$$

With this the computation of the series coefficients $\boldsymbol{x}_m(z,t)\,,\partial_z\boldsymbol{x}_m(z,t)\,, m\geq 2$ is given by

$$\begin{aligned}\boldsymbol{x}_m(z,t) &= A^{-1}\int_\xi^z\int_\xi^\eta \partial_t\boldsymbol{x}_{m-2}(\sigma,t) + B(\sigma,t)\partial_\sigma\boldsymbol{x}_{m-1}(\sigma,t) - C(\sigma,t)\boldsymbol{x}_{m-2}(\sigma,t)\,\mathrm{d}\sigma\,\mathrm{d}\eta\,,\\ \partial_z\boldsymbol{x}_m(z,t) &= A^{-1}\int_\xi^z \partial_t\boldsymbol{x}_{m-2}(\sigma,t) + B(\sigma,t)\partial_\sigma\boldsymbol{x}_{m-1}(\sigma,t) - C(\sigma,t)\boldsymbol{x}_{m-2}(\sigma,t)\,\mathrm{d}\sigma\,.\end{aligned} \tag{4.24}$$

Obviously both, $\boldsymbol{x}_m(z,t)$ and $\partial_z\boldsymbol{x}_m(z,t)$, depend on the time derivative $\partial_t\boldsymbol{x}_{m-2}(z,t)$. Therefore the convergence analysis of (4.13) requires a formulation of an upper bound for the growth of the time derivatives of each $\boldsymbol{x}_m(z,t)$ up to an arbitrary order k. From this, the absolute and uniform convergence of the series (4.13) can be shown by analysing the growth of the series coefficients. For this it is necessary to define norms for the states $\boldsymbol{x}(z,t)$ and the coefficient matrices, i.e., A, B, and C. Therefore one should examine the appropriate definitions and lemmas in Appendix C of the appendix. With this, note the following theorem.

Theorem 4.5. *Let* $B\in\mathbf{CG}^{0,\alpha_B}\big(\Omega\left(\bar\ell_\mathrm{N},t_0\right);\mathbb{R}^{n\times n}\big)$*,* $C\in\mathbf{CG}^{0,\alpha_C}\big(\Omega\left(\bar\ell_\mathrm{N},t_0\right);\mathbb{R}^{n\times n}\big)$ *and* $\boldsymbol{y}_j\in\mathbf{G}^{\delta_{\boldsymbol{y}_j}}_{\alpha_{\boldsymbol{y}_j}}\big(\mathbb{R}^+_{t_0};\mathbb{R}^n\big)\,, j=0,1$ *with* α_B*,* α_C*,* $\alpha_{\boldsymbol{y}_j}\geq 1$*, then for* $m\in\mathbb{N}$ *and* $\boldsymbol{g}$ *given by* (4.23) *the series coefficients* (4.18) *satisfy*

$$\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}}\partial_t^k\boldsymbol{x}_m\right\|_\infty(z)\leq\delta^{k+m}((k+m-1)!)^\alpha l_m\frac{|z-\xi|^m}{m!} \tag{4.25a}$$

$$\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}}\partial_t^k\partial_z\boldsymbol{x}_m\right\|_\infty(z)\leq\delta^{k+m}((k+m-1)!)^\alpha l_m\frac{|z-\xi|^{m-1}}{(m-1)!} \tag{4.25b}$$

with $\left\|\sup_{\mathbb{R}^+_{t_0}}(\,\cdot\,)\right\|_\infty(z)$ *defined in* (C.4)*,* $\delta=\max\left\{\delta_B,\delta_C,\delta_{\boldsymbol{y}_0},\delta_{\boldsymbol{y}_1}\right\}$*,* $\alpha=\max\left\{\alpha_B,\alpha_C,\alpha_{\boldsymbol{y}_0},\alpha_{\boldsymbol{y}_1}\right\}$ *and*

$$l_m=\left\|A^{-1}\right\|_\infty\left(\frac{\epsilon_{m-2}l_{m-2}}{\delta(m-1)^\alpha}+\frac{l_{m-1}}{(m-1)^\alpha}\right)\,,\qquad m\geq 3 \tag{4.26}$$

with $l_1=1$*,* $l_2=3\left\|A^{-1}\right\|_\infty$*, and* $\epsilon_m=1+m^{-\alpha}$*. Particularly, the series* (4.13) *converges absolutely and uniformly if* $1\leq\alpha\leq 2$ *for all* $|z-\xi|<\rho$*, where*

$$\rho=\begin{cases}\infty, & 1\leq\alpha<2\\ \left(\kappa h(\kappa)\right)^{-1}, & \alpha=2\,,\end{cases} \tag{4.27}$$

with $\kappa = \left\|A^{-1}\right\|_\infty \delta > 0$ *and*

$$h(\kappa) = \max\left\{\sqrt{\frac{\sqrt{2}}{\kappa} + \frac{3}{\sqrt{2}}}, \frac{1}{2\sqrt{3}} + \sqrt{\frac{1}{12} + \frac{1+\sqrt{6}}{\sqrt{6}\kappa}}\right\} . \tag{4.28}$$

Obviously, the convergence radius is infinite as long as $1 \leq \alpha < 2$, and is finite and dependent on the constant κ for the particular case $\alpha = 2$.

Proof. In the following Theorem 4.5 is proven by induction. First, it is shown that (4.25) holds for $m = 1, 2$. Second, the inequalities (4.25) are proven for $m + 1$ with the assumption that they hold for m. For this, please note some useful Theorems and Lemmas in Appendix C.3. Starting with the estimation for $\boldsymbol{x}_1$ and considering (4.18) as well as $\boldsymbol{y}_1 \in \mathbf{G}^{\delta}_{\alpha}\big(\mathbb{R}^+_{t_0}; \mathbb{R}^n\big) \supseteq \mathbf{G}^{\delta_{\boldsymbol{y}_1}}_{\alpha_{\boldsymbol{y}_1}}\big(\mathbb{R}^+_{t_0}; \mathbb{R}^n\big)$, then this trivially yields

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_1\right\|_\infty(z) &= \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{y}_1(t)(z-\xi)\right\|_\infty \\ &\leq \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{y}_1(t)\right\|_\infty \left|(z-\xi)\right| \leq \delta^{k+1}(k!)^\alpha l_1 \left|z-\xi\right| ,\end{aligned} \tag{4.29a}$$

$$\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \partial_z \boldsymbol{x}_1\right\|_\infty(z) = \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{y}_1(t)\right\|_\infty \leq \delta^{k+1}(k!)^\alpha l_1 . \tag{4.29b}$$

and consequently confirms (4.25) for $m = 1$. Proceeding with $m = 2$ the upper bound estimation for the coefficient $\boldsymbol{x}_2(z,t)$ in (4.18) can be written as

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_2\right\|_\infty(z) \leq \Bigg\|\sup_{\mathbb{R}^+_{t_0}} \Big(A^{-1} \int_\xi^z \int_\xi^\eta \partial_t^{k+1} \boldsymbol{x}_0(\sigma,t) + \partial_t^k \left[B(\sigma,t)\partial_\sigma \boldsymbol{x}_1(\sigma,t)\right] \\ + \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_0(\sigma,t)\right] \mathrm{d}\sigma \, \mathrm{d}\eta\Big)\Bigg\|_\infty \\ \leq \left\|A^{-1}\right\|_\infty \Bigg| \int_\xi^z \int_\xi^\eta \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{y}_0(t)\right\|_\infty + \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\boldsymbol{y}_1(t)\right]\right\|_\infty \\ + \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{y}_0(t)\right]\right\|_\infty \mathrm{d}\sigma \, \mathrm{d}\eta\Bigg|\end{aligned} \tag{4.30}$$

with $\left\|A^{-1}\right\|_\infty$ defined as the maximum absolute row sum of the matrix A^{-1}. Note, the maximum absolute row sum is sub-multiplicative. From this the integrands are investigated individually, i.e., applying the Leibniz rule (C.20) to the term $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{y}_0(t)\right]\right\|_\infty$, and considering $C \in \mathbf{CG}^{0,\alpha}\big(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\big)$ and $\boldsymbol{y}_0 \in \mathbf{G}^{\delta}_{\alpha}\big(\mathbb{R}^+_{t_0}; \mathbb{R}^n\big)$, respectively. Moreover, respecting the upper bounds (C.9), (C.19) as well as (C.23), then this yields

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{y}_0(t)\right]\right\|_\infty &\overset{\text{(C.9)}}{\leq} \sum_{j=0}^k \binom{k}{j} \left\|\sup_\Omega \partial_t^{k-j} C(\sigma,t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^j \boldsymbol{y}_0(t)\right\|_\infty \\ &\overset{\text{(C.19), (C.12)}}{\leq} \delta^{k+2} \sum_{j=0}^k \binom{k}{j} \left((k-j)!j!\right)^\alpha \overset{\text{(C.23)}}{\leq} \delta^{k+2} \left((k+1)!\right)^\alpha ,\end{aligned} \tag{4.31}$$

since δ_C, $\delta_{\boldsymbol{y}_0} \leq \delta$ and α_C, $\alpha_{\boldsymbol{y}_0} \leq \alpha$. The same estimation applies to $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\boldsymbol{y}_1(t)\right]\right\|_\infty$ with $\boldsymbol{y}_1(t) = \partial_\sigma \boldsymbol{x}_1 \sigma$ and to $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{x}_0(\sigma,t)\right\|_\infty = \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{y}_0(t)\right\|_\infty$ when δ_B, $\delta_{\boldsymbol{y}_{0,1}} \leq \delta$ and α_B, $\alpha_{\boldsymbol{y}_{0,1}} \leq \alpha$ are taken into account. Consequently, utilising (4.31) three times and reinserting the expressions in (4.30) gives

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_2\right\|_\infty(z) &\leq 3\delta^{k+2}\left((k+1)!\right)^\alpha \left\|A^{-1}\right\|_\infty \left|\frac{(z-\xi)^2}{2}\right| \\ &\leq \delta^{k+2}\left((k+1)!\right)^\alpha l_2 \frac{|z-\xi|^2}{2}\end{aligned} \tag{4.32}$$

which confirms (4.25a) for $m = 2$. The same procedure can be applied to the spatial derivative $\partial_z \boldsymbol{x}_2$. This yields

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \partial_z \boldsymbol{x}_2\right\|_\infty(z) \leq \left\|A^{-1}\right\|_\infty \bigg|\int_\xi^z &\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{y}_0(t)\right\|_\infty \\ &+ \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\boldsymbol{y}_1(t)\right]\right\|_\infty + \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{y}_0(t)\right]\right\|_\infty \mathrm{d}\sigma\bigg| \\ \leq \delta^{k+2}\left((k+1)!\right)^\alpha l_2 |z-\xi|&\end{aligned} \tag{4.33}$$

which is in accordance to (4.25b) for $m = 2$. Assume (4.25) holds for some $m \in \mathbb{N}$. Then for the induction step from m to $m+1$, $\forall m \geq 2$ the upper bound for the $k-$time derivative of the series coefficients $\boldsymbol{x}_{m+1}$ is given by

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_{m+1}\right\|_\infty(z) \leq \left\|A^{-1}\right\|_\infty \bigg|\int_\xi^z \int_\xi^\eta &\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \\ + \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\partial_\sigma \boldsymbol{x}_m(\sigma,t)\right]\right\|_\infty &+ \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty \mathrm{d}\sigma\, \mathrm{d}\eta\bigg|\end{aligned} \tag{4.34}$$

Again the differentiation of the addends is treated separately. Therefore, as conducted before the Leibniz rule for the term $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty$ together with the primary assumption (4.25a) for $\boldsymbol{x}_{m-1}$, $m \geq 2$, the consideration of $C \in \mathbf{CG}^{0,\alpha}\big(\Omega\left(\bar{\ell}_\mathrm{N}, t_0\right); \mathbb{R}^{n\times n}\big)$, and the upper bounds (C.19) and (C.23) provide the estimation

$$\begin{aligned}&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty \\ &\qquad\leq \sum_{j=0}^k \binom{k}{j} \left\|\sup_\Omega \partial_t^{k-j} C(\sigma,t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^j \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \\ &\qquad\overset{\text{(C.19), (4.25a)}}{\leq} \delta^{k+m} l_{m-1} \sum_{j=0}^k \binom{k}{j} ((k-j)!)^\alpha ((j+m-2)!)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \\ &\qquad\overset{\text{(C.23)}}{\leq} \delta^{k+m} l_{m-1} \left(\frac{(k+m-1)!}{m-1}\right)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!}\,.\end{aligned} \tag{4.35}$$

With this and the assumption that (4.25b) holds true for $m \geq 1$ the convective term $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\partial_\sigma \boldsymbol{x}_m(\sigma,t)\right]\right\|_\infty$ can be treated in equivalent fashion. As a consequence the second term in (4.34) yields

$$\begin{aligned}
&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)\partial_\sigma \boldsymbol{x}_m(\sigma,t)\right]\right\|_\infty \\
&\qquad \leq \sum_{j=0}^{k} \binom{k}{j} \left\|\sup_\Omega \partial_t^{k-j} B(\sigma,t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^j \partial_\sigma \boldsymbol{x}_m(\sigma,t)\right\|_\infty \\
&\qquad \overset{(C.16),\,(4.25a)}{\leq} \delta^{k+m+1} l_m \sum_{j=0}^{k} \binom{k}{j} ((k-j)!)^\alpha ((j+m-1)!)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \\
&\qquad \overset{(C.23)}{\leq} \delta^{k+m+1} l_m \left(\frac{(k+m)!}{m}\right)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \,.
\end{aligned} \tag{4.36}$$

Next, reinserting the upper bound of $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty$ which directly follows from (4.25a) for $m \geq 2$, together with the estimations (4.35) and (4.36) leads to

$$\begin{aligned}
\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_{m+1}\right\|_\infty(z) &\leq \delta^{k+m+1} \left\|A^{-1}\right\|_\infty \left[l_{m-1} \frac{((k+m-1)!)^\alpha}{\delta} \left(1 + \frac{1}{(m-1)^\alpha}\right) \right. \\
&\qquad\qquad \left. + l_m \frac{((k+m)!)^\alpha}{m^\alpha}\right] \left| \int_\xi^z \int_\xi^\eta \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \,\mathrm{d}\sigma\,\mathrm{d}\eta \right| \\
&\overset{(C.25)}{\leq} \delta^{k+m+1} ((k+m)!)^\alpha \left\|A^{-1}\right\|_\infty \left(\frac{l_{m-1}}{\delta (k+m)^\alpha}\left(1 + \frac{1}{(m-1)^\alpha}\right) + \frac{l_m}{m^\alpha}\right) \frac{|z-\xi|^{m+1}}{(m+1)!} \\
&\leq \delta^{k+m+1} ((k+m)!)^\alpha \left\|A^{-1}\right\|_\infty \left(\frac{\epsilon_{m-1} l_{m-1}}{\delta m^\alpha} + \frac{l_m}{m^\alpha}\right) \frac{|z-\xi|^{m+1}}{(m+1)!} \\
&\leq \delta^{k+m+1} ((k+m)!)^\alpha l_{m+1} \frac{|z-\xi|^{m+1}}{(m+1)!} \,.
\end{aligned} \tag{4.37}$$

Here it is necessary to utilise the auxiliary theorem (C.25) for the integration with $k \geq 0$ and the recursion $l_{m+1} = \|A^{-1}\|_\infty \left(\frac{\epsilon_{m-1} l_{m-1}}{\delta m^\alpha} + \frac{l_m}{m^\alpha}\right)$. Obviously the assumption (4.25a) for $m+1$ and (4.37) are equivalent. Taking (4.29) and (4.32) into account this means (4.25) holds $\forall m \in \mathbb{N}$ and consequently proves the inequalities (4.25a) and (4.25b) by induction.

With this the evaluation of the convergence radius follows. Making use of (4.25a) with $k = 0$

allows to bound (4.13) from above by a power series according to

$$\begin{aligned}\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty(z) &\le \sum_{m=0}^{\infty} \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \boldsymbol{x}_m\right\|_\infty(z) \\ &\le \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \boldsymbol{x}_0(z,t)\right\|_\infty + \sum_{m=1}^{\infty} \delta^m ((m-1)!)^\alpha l_m \frac{|z-\xi|^m}{m!} \\ &\le \delta(1+|z-\xi|) + 3\left\|A^{-1}\right\|_\infty \delta^2 \frac{|z-\xi|^2}{2} + \sum_{m=3}^{\infty} \underbrace{\delta^m \frac{((m-1)!)^\alpha}{m!} l_m}_{p_m} |z-\xi|^m .\end{aligned} \tag{4.38}$$

Note, for the latter expression $l_1 = 1$, $l_2 = 3\left\|A^{-1}\right\|_\infty$ is applied and starting index of the power series changes from $m = 1$ to $m = 3$. For estimation of the growth of the coefficients p_m, $m \ge 3$, consider the following lemma.

Lemma 4.6. *The recursion l_m defined in* (4.26) *satisfies*

$$l_m \le \left(\left\|A^{-1}\right\|_\infty\right)^{m-1} \frac{h(\kappa)^{m-1}}{((m-1)!)^{\frac{\alpha}{2}}}, \qquad m \ge 3 \tag{4.39}$$

with $\kappa = \left\|A^{-1}\right\|_\infty \delta > 0$, $h(\kappa)$ given by (4.28)*, and $1 \le \alpha \le 2$.*

The proof of Lemma 4.6 is achieved by mathematical induction again.

Proof. One may start with the base case which requires the computation of l_3. Consider $\alpha \ge 1$ and take the initial values $l_1 = 1$, $l_2 = 3\left\|A^{-1}\right\|_\infty$ as well as $\epsilon_m = 1 + m^{-\alpha}$ into account. Then l_3 satisfies the estimation (4.39) as follows

$$\begin{aligned}l_3 = \left\|A^{-1}\right\|_\infty \left(\frac{\epsilon_1 l_1}{\delta 2^\alpha} + \frac{l_2}{2^\alpha}\right) &= \left\|A^{-1}\right\|_\infty \left(\frac{(1+1^{-\alpha})}{\delta 2^\alpha} + \frac{3\left\|A^{-1}\right\|_\infty}{2^\alpha}\right) \\ &= \left(\left\|A^{-1}\right\|_\infty\right)^2 \left(\frac{2}{2^\alpha \kappa} + \frac{3}{2^\alpha}\right) \le \left(\left\|A^{-1}\right\|_\infty\right)^2 \left(\frac{2}{\kappa \sqrt{2}\, 2^{\frac{\alpha}{2}}} + \frac{3}{\sqrt{2}\, 2^{\frac{\alpha}{2}}}\right) \\ &\le \frac{\left(\left\|A^{-1}\right\|_\infty\right)^2}{(2)^{\frac{\alpha}{2}}} \left(\frac{\sqrt{2}}{\kappa} + \frac{3}{\sqrt{2}}\right) \le \left(\left\|A^{-1}\right\|_\infty\right)^2 \frac{h(\kappa)^2}{(2)^{\frac{\alpha}{2}}} \\ &\Rightarrow \quad h(\kappa) \ge \sqrt{\frac{\sqrt{2}}{\kappa} + \frac{3}{\sqrt{2}}} .\end{aligned} \tag{4.40}$$

The latter inequality holds true with respect to (4.28). With this and the consideration of

(4.26) for $m \geq 3$ again the induction step for the proof of (4.6) can be written as

$$\begin{aligned} l_{m+1} &= \left\|A^{-1}\right\|_\infty \left(\frac{\epsilon_{m-1} l_{m-1}}{\delta m^\alpha} + \frac{l_m}{m^\alpha} \right) \\ &\leq \left(\left\|A^{-1}\right\|_\infty\right)^m h(\kappa)^m \left(\frac{\epsilon_{m-1} h(\kappa)^{-2}}{\kappa\, m^\alpha ((m-2)!)^{\frac{\alpha}{2}}} + \frac{h(\kappa)^{-1}}{m^\alpha ((m-1)!)^{\frac{\alpha}{2}}} \right) \\ &\leq \left(\left\|A^{-1}\right\|_\infty\right)^m \frac{h(\kappa)^m}{(m!)^{\frac{\alpha}{2}}} \left(\left(1 + \frac{1}{(m-1)^\alpha}\right) \frac{(m-1)^{\frac{\alpha}{2}}}{\kappa\, m^{\frac{\alpha}{2}}} h(\kappa)^{-2} + \frac{1}{m^{\frac{\alpha}{2}}} h(\kappa)^{-1} \right) \\ &\leq \left(\left\|A^{-1}\right\|_\infty\right)^m \frac{h(\kappa)^m}{(m!)^{\frac{\alpha}{2}}} \times \\ &\qquad \left(\frac{1}{\kappa} \left(\frac{(m-1)^{\frac{\alpha}{2}}}{m^{\frac{\alpha}{2}}} + \frac{1}{m^{\frac{\alpha}{2}} (m-1)^{\frac{\alpha}{2}}} \right) h(\kappa)^{-2} + \frac{1}{m^{\frac{\alpha}{2}}} h(\kappa)^{-1} \right) . \end{aligned} \tag{4.41}$$

Note, since $(m-1)^{\frac{\alpha}{2}}/m^{\frac{\alpha}{2}} \leq 1$, $1/(m^{\frac{\alpha}{2}}(m-1)^{\frac{\alpha}{2}}) \leq 1/\sqrt{6}$, and $1/m^{\frac{\alpha}{2}} \leq 1/\sqrt{3}$ for $m \geq 3\,,\ \alpha \geq 1$ the recursion (4.26) can be further bounded from above by

$$l_{m+1} \leq \left(\left\|A^{-1}\right\|_\infty\right)^m \frac{h(\kappa)^m}{(m!)^{\frac{\alpha}{2}}} \underbrace{\left(\frac{1+\sqrt{6}}{\sqrt{6}\kappa} h(\kappa)^{-2} + \frac{1}{\sqrt{3}} h(\kappa)^{-1} \right)}_{\leq 1} \tag{4.42}$$

which defines the quadratic equation

$$h(\kappa)^2 - \frac{1}{\sqrt{3}} h(\kappa) - \frac{1+\sqrt{6}}{\sqrt{6}\kappa} \geq 0\,. \tag{4.43}$$

Its positive solution leads to the inequality

$$h(\kappa) \geq \frac{1}{2\sqrt{3}} + \sqrt{\frac{1}{12} + \frac{1+\sqrt{6}}{\sqrt{6}\kappa}} \tag{4.44}$$

which is satisfied when the definition of h in (4.28) is taken into account. □

Consequently, utilising Lemma 4.6 the coefficients p_m in (4.38) are bounded from above by

$$\begin{aligned} p_m = \delta^m \frac{(m-1)!)^\alpha}{m!} l_m &\leq \left(\left\|A^{-1}\right\|_\infty\right)^{m-1} \delta^m \frac{(m-1)!)^\alpha}{m!} \frac{h(\kappa)^{m-1}}{((m-1)!)^{\frac{\alpha}{2}}} \\ &\leq \delta (\kappa h(\kappa))^{m-1} \frac{((m-1)!)^{\frac{\alpha}{2}}}{m!} \end{aligned} \tag{4.45}$$

Pursuing the d'Alembert ratio test (C.28) as in [55] the convergence radius for the series (4.38) with powers of $|z-\xi|$ can be derived by

$$\begin{aligned} \rho_q = \lim_{m\to\infty} \left| \frac{p_m}{p_{m+1}} \right| &= (\kappa h(\kappa))^{-1} \lim_{m\to\infty} \frac{((m-1)!)^{\frac{\alpha}{2}}}{m!} \frac{(m+1)!}{(m!)^{\frac{\alpha}{2}}} = (\kappa h(\kappa))^{-1} \lim_{m\to\infty} \frac{m+1}{m^{\frac{\alpha}{2}}} \\ &= \begin{cases} \infty, & 1 \leq \alpha < 2 \\ (\kappa h(\kappa))^{-1}, & \alpha = 2\,, \end{cases} \end{aligned} \tag{4.46}$$

which confirms (4.27) and consequently closes the proof of Theorem 4.5. □

Remark 4.14. *The convergence analysis for the series* $\sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m(z,t)$ *allows the same conclusion as for* (4.13)*, i.e., absolute and uniform convergence with infinite radius for* $1 \leq \alpha < 2$*. With this it follows*

$$\partial_t \boldsymbol{x}(z,t) = \partial_t \left(\sum_{m=0}^{\infty} \boldsymbol{x}_m(z,t) \right) = \sum_{m=0}^{\infty} \partial_t \boldsymbol{x}_m(z,t)\,. \tag{4.47}$$

Note that this result allows to interchange the sum with derivative operator in (4.20)*.*

4.2.2 Absolute and Uniform Convergence Analysis for the Coupled Modified, Viscous Burgers' Equation

Let us recall the coupled MVBE which was introduced in (3.23) and consequently determines $\boldsymbol{f}$ as

$$\boldsymbol{f}(z,t,\boldsymbol{x},\partial_t\boldsymbol{x},\partial_z\boldsymbol{x}) = A^{-1}\left[\partial_t\boldsymbol{x}(z,t) + B(z,t)X(z,t)\partial_z\boldsymbol{x}(z,t) - C(z,t)\boldsymbol{x}(z,t)\right] \tag{4.48}$$

with the diagonal matrix $X = \operatorname{diag}[x^1, \cdots, x^n]$. Due to the term $X(z,t)\partial_z\boldsymbol{x}(z,t)$ the determination of $\boldsymbol{g}_m$ in (4.18) requires a more detailed analysis than that for the coupled DCRS in the paragraph before. Thus, inserting the series (4.13) in (4.48) and considering Theorem 3.24 for the non-linear term leads to

$$\begin{aligned}
&\boldsymbol{f}\left(\sigma, t, \sum_{m=2}^{\infty} \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_t \boldsymbol{x}_{m-2}, \sum_{m=2}^{\infty} \partial_z \boldsymbol{x}_{m-1}\right) \\
&\qquad = A^{-1}\Bigg[\sum_{m=2}^{\infty} \partial_t \boldsymbol{x}_{m-2}(z,t) + B(z,t)\sum_{m=2}^{\infty} X_{m-2}(z,t) \sum_{m=2}^{\infty} \partial_z \boldsymbol{x}_{m-1}(z,t) \\
&\qquad\qquad\qquad -C(z,t)\sum_{m=2}^{\infty} \boldsymbol{x}_{m-2}(z,t)\Bigg] \\
&\qquad = A^{-1}\Bigg[\sum_{m=2}^{\infty} \partial_t \boldsymbol{x}_{m-2}(z,t) + B(z,t)\sum_{m=2}^{\infty}\sum_{q=0}^{m-2} X_{m-2-q}(z,t)\partial_z \boldsymbol{x}_{q+1}(z,t) \\
&\qquad\qquad\qquad -C(z,t)\sum_{m=2}^{\infty} \boldsymbol{x}_{m-2}(z,t)\Bigg] \\
&\qquad = \sum_{m=2}^{\infty} A^{-1}\Bigg[\partial_t \boldsymbol{x}_{m-2}(z,t) + B(z,t)\sum_{q=0}^{m-2} X_{m-2-q}(z,t)\partial_z \boldsymbol{x}_{q+1}(z,t) \\
&\qquad\qquad\qquad - C(z,t)\boldsymbol{x}_{m-2}(z,t)\Bigg]
\end{aligned} \tag{4.49}$$

with $X_k = \operatorname{diag}\left[x_k^1, \ldots, x_k^n\right]$. As a result the function $\boldsymbol{g}_m$ for the coupled MVBE can be written as

$$\begin{aligned}\boldsymbol{g}_m\left(z, t, \boldsymbol{x}_0, \ldots, \boldsymbol{x}_{m-2}, \partial_t \boldsymbol{x}_{m-2}, \partial_z \boldsymbol{x}_1, \ldots, \partial_z \boldsymbol{x}_{m-1}\right) = A^{-1}\Bigg[&\partial_t \boldsymbol{x}_{m-2}(z,t) \\ &+ B(z,t)\sum_{q=0}^{m-2} X_{m-2-q}(z,t)\partial_z \boldsymbol{x}_{q+1}(z,t) - C(z,t)\boldsymbol{x}_{m-2}(z,t)\Bigg]. \qquad (4.50)\end{aligned}$$

Applying the results from above let us look at the following theorem.

Theorem 4.7. *Let* $B \in \mathbf{CG}^{0,\alpha_B}\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\right)$, $C \in \mathbf{CG}^{0,\alpha_C}\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right); \mathbb{R}^{n\times n}\right)$ *and* $\boldsymbol{y}_j \in \mathbf{G}^{\delta_{\boldsymbol{y}_j}}_{\alpha_{\boldsymbol{y}_j}}\left(\mathbb{R}^+_{t_0}; \mathbb{R}^n\right)$, $j = 0, 1$ *with* α_B, α_C, $\alpha_{\boldsymbol{y}_j} \geq 1$, *then for* $m \in \mathbb{N}$ *and* $\boldsymbol{g}_m$ *given by* (4.50) *the series coefficients* (4.18) *satisfy*

$$\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_m\right\|_\infty(z) \leq \delta^{k+m}\left((k+2m-2)!\right)^\alpha l_m \frac{|z-\xi|^m}{m!} \qquad (4.51a)$$

$$\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \partial_z \boldsymbol{x}_m\right\|_\infty(z) \leq \delta^{k+m}\left((k+2m-2)!\right)^\alpha l_m \frac{|z-\xi|^{m-1}}{(m-1)!} \qquad (4.51b)$$

with $\left\|\sup_{\mathbb{R}^+_{t_0}}(\,\cdot\,)\right\|_\infty(z)$ *defined in* (C.4), $\delta = \max\left\{\delta_B, \delta_C, \delta_{\boldsymbol{y}_0}, \delta_{\boldsymbol{y}_1}\right\}$, $\alpha = \max\left\{\alpha_B, \alpha_C, \alpha_{\boldsymbol{y}_0}, \alpha_{\boldsymbol{y}_1}\right\}$ *and*

$$\begin{aligned}l_m = \left\|A^{-1}\right\|_\infty \Bigg[&\frac{\epsilon_{m-2} l_{m-2}}{\delta(2m-4)^\alpha(2m-3)^\alpha(2m-2)^\alpha} + \frac{\delta l_{m-1}}{(2m-3)^\alpha(2m-2)^\alpha} + \\ &\sum_{q=0}^{m-3} \frac{l_{m-2-q} l_{q+1}}{(2m-5)^\alpha(2m-4)^\alpha(2m-3)^\alpha(2m-2)^\alpha}\binom{2m-6}{2q}^{-\alpha}\binom{m-2}{q}\Bigg], \qquad (4.52)\end{aligned}$$

for $m \geq 3$, *with* $l_1 = 1$, $l_2 = \left\|A^{-1}\right\|_\infty(2+\delta)/2^\alpha$, *and* $\epsilon_m = 1 + (2m-1)^{-\alpha}$. *Particularly, the series* (4.13) *converges absolutely and uniformly if* $1 \leq \alpha \leq 2$ *for all* $|z-\xi| < \rho$, *where*

$$\rho = \left(\left\|A^{-1}\right\|_\infty \delta\, h\left(\left\|A^{-1}\right\|_\infty, \delta\right)\right)^{-1}, \quad 1 \leq \alpha \leq 2\,, \qquad (4.53)$$

with

$$\begin{aligned}h\left(\left\|A^{-1}\right\|_\infty, \delta\right) = \max\Bigg\{&2+\delta\,, \sqrt{\frac{1}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2\left\|A^{-1}\right\|_\infty} + \delta + \frac{\delta^2}{2}}\,, \\ &\frac{\delta}{6} + \sqrt{\frac{\delta^2}{36} + \frac{4}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2}}\Bigg\}\,. \qquad (4.54)\end{aligned}$$

Proof. Theorem 4.7 is proven by induction. The induction start for $m = 1$ is omitted since it is equal to the analysis in (4.29). Thus, first it is shown that (4.51) holds for $m = 2$, i.e.,

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_2\right\|_\infty(z) \le{}& \Bigg\| \sup_{\mathbb{R}^+_{t_0}} \Big(A^{-1} \int_\xi^z \int_\xi^\eta \partial_t^{k+1} \boldsymbol{x}_0(\sigma,t) \\ &\qquad + \partial_t^k \left[B(\sigma,t)X_0(\sigma,t)\partial_\sigma \boldsymbol{x}_1(\sigma,t)\right] + \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_0(\sigma,t)\right] \mathrm{d}\sigma\, \mathrm{d}\eta\Big)\Bigg\|_\infty \\ \le{}& \left\|A^{-1}\right\|_\infty \Bigg| \int_\xi^z \int_\xi^\eta \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{y}_0(t)\right\|_\infty \\ &+ \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)Y_0(t)\boldsymbol{y}_1(t)\right]\right\|_\infty + \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{y}_0(t)\right]\right\|_\infty \mathrm{d}\sigma\, \mathrm{d}\eta \Bigg| .\end{aligned} \tag{4.55}$$

Considering (4.16) it directly follows $Y_0(t) = \operatorname{diag}\left[y_0^1(t), \dots, y_0^n(t)\right] = X_0(\sigma,t)$ and $\boldsymbol{y}_1(t) = \partial_\sigma \boldsymbol{x}_1(\sigma,t)$, respectively. The first and the last term inside the double integral were already analysed in the previous section, e.g. see (4.31). Thus, notice the upper bound estimation using Leibnitz's differentiation rule (C.20)

$$\begin{aligned}&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)X_0(\sigma,t)\partial_\sigma \boldsymbol{x}_1(\sigma,t)\right]\right\|_\infty \\ &\quad\le \sum_{j=0}^{k} \binom{k}{j} \left\|\sup_\Omega \partial_t^{k-j} B(\sigma,t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^j \left[Y_0(t)\boldsymbol{y}_1(t)\right]\right\|_\infty \\ &\quad\le \sum_{j=0}^{k} \binom{k}{j} \left\|\sup_\Omega \partial_t^{k-j} B(\sigma,t)\right\|_\infty \times \\ &\qquad\qquad \sum_{i=0}^{j} \binom{j}{i} \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{j-i} Y_0(t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^i \boldsymbol{y}_1(t)\right\|_\infty\end{aligned} \tag{4.56a}$$

and moreover applying (C.23) gives

$$\begin{aligned}&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)X_0(\sigma,t)\partial_\sigma \boldsymbol{x}_1(\sigma,t)\right]\right\|_\infty \\ &\quad\le \delta^{k+3} \sum_{j=0}^{k} \binom{k}{j} ((k-j)!)^\alpha \sum_{i=0}^{j} \binom{j}{i} ((j-i)!)^\alpha (i!)^\alpha \\ &\quad\le \delta^{k+3} \sum_{j=0}^{k} \binom{k}{j} ((k-j)!)^\alpha ((j+1)!)^\alpha \\ &\quad\le \delta^{k+3} \frac{((k+2)!)^\alpha}{2^\alpha} .\end{aligned} \tag{4.56b}$$

With this and utilising (4.31) twice the estimate for $\boldsymbol{x}_2$ leads to

$$\begin{aligned}\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_2\right\|_\infty(z) &\le \left\|A^{-1}\right\|_\infty \delta^{k+2} \left(2(k+1)!^\alpha + \delta \frac{(k+2)!^\alpha}{2^\alpha} \right) \left| \frac{(z-\xi)^2}{2} \right| \\ &\le \left\|A^{-1}\right\|_\infty \delta^{k+2} ((k+2)!)^\alpha \frac{2+\delta}{2^\alpha} \frac{|z-\xi|^2}{2} \le \delta^{k+2}(k+2)!^\alpha l_2 \frac{|z-\xi|^2}{2} ,\end{aligned} \tag{4.57}$$

which completes the base case of the proof by mathematical induction. The second step aims to show that the estimate (4.51a) holds for the next natural number $m+1$ provided that it is satisfied for m. Therefore the following terms under the double integral

$$\begin{aligned}
\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_{m+1}\right\|_\infty(z) \leq \left\|A^{-1}\right\|_\infty \bigg| \int_\xi^z \int_\xi^\eta & \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \\
& + \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k [B(\sigma,t) \sum_{q=0}^{m-1} X_{m-1-q}(\sigma,t) \partial_\sigma \boldsymbol{x}_{q+1}(\sigma,t)]\right\|_\infty \\
& + \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty \mathrm{d}\sigma\, \mathrm{d}\eta \bigg| \\
\leq \left\|A^{-1}\right\|_\infty \bigg| \int_\xi^z \int_\xi^\eta & \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^{k+1} \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \\
& + \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k [B(\sigma,t) \sum_{q=0}^{m-2} X_{m-1-q}(\sigma,t) \partial_\sigma \boldsymbol{x}_{q+1}(\sigma,t)]\right\|_\infty \\
& + \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k [B(\sigma,t) X_0(\sigma,t) \partial_\sigma \boldsymbol{x}_m(\sigma,t)]\right\|_\infty \\
& + \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty \mathrm{d}\sigma\, \mathrm{d}\eta \bigg|
\end{aligned} \tag{4.58}$$

require a more detailed analysis. In (4.58) the sum is separated into two parts since $X_0(\sigma,t) = Y_0(t)$ demands different treatment compared to $X_m(\sigma,t)$, $m \geq 1$. Starting with the last expression of the double integral it can be estimated by

$$\begin{aligned}
& \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \left[C(\sigma,t)\boldsymbol{x}_{m-1}(\sigma,t)\right]\right\|_\infty \\
& \quad \leq \sum_{j=0}^{k} \binom{k}{j} \left\|\sup\nolimits_{\Omega} \partial_t^{k-j} C(\sigma,t)\right\|_\infty \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^j \boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \\
& \quad \overset{\text{(C.19), (4.51a)}}{\leq} \delta^{k+m} l_{m-1} \sum_{j=0}^{k} \binom{k}{j} ((k-j)!)^\alpha (j+2m-4!)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \\
& \quad \overset{\text{(C.23)}}{\leq} \delta^{k+m} l_{m-1} \frac{((k+2m-3)!)^\alpha}{(2m-3)^\alpha} \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \,.
\end{aligned} \tag{4.59}$$

when making use of (C.19), (4.51a), and (C.23). Similarly the term involving $X_0(\sigma,t)$ can be bounded by

$$
\begin{aligned}
&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)X_0(\sigma,t)\partial_\sigma \boldsymbol{x}_m(\sigma,t)\right]\right\|_\infty \\
&\qquad \leq \sum_{j=0}^{k} \binom{k}{j} \left\|\sup_{\Omega} \partial_t^{k-j} B(\sigma,t)\right\|_\infty \times \\
&\qquad\qquad \sum_{i=0}^{j} \binom{j}{i} \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{j-i} Y_0(t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{i} \partial_\sigma \boldsymbol{x}_m(\sigma,t)\right\|_\infty \\
&\qquad \overset{(C.16),\ (4.51b)}{\leq} \delta^{k+m+2} l_m \sum_{j=0}^{k} \binom{k}{j} ((k-j)!)^\alpha \times \\
&\qquad\qquad \sum_{i=0}^{j} \binom{j}{i} ((j-i)!)^\alpha ((i+2m-2)!)^\alpha \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \\
&\qquad \overset{(C.23)}{\leq} \delta^{k+m+2} l_m \sum_{j=0}^{k} \binom{k}{j} \frac{((k-j)!)^\alpha ((j+2m-1)!)^\alpha}{(2m-1)^\alpha} \frac{|\sigma-\xi|^{m-1}}{(m-1)!} \\
&\qquad \overset{(C.23)}{\leq} \delta^{k+m+2} l_m \frac{((k+2m)!)^\alpha}{(2m-1)^\alpha (2m)^\alpha} \frac{|\sigma-\xi|^{m-1}}{(m-1)!}
\end{aligned}
\tag{4.60}
$$

when having a look at the estimation (4.51b) for the spatial derivative of $\boldsymbol{x}_{m-1}$. The term with the sum over q can be analysed by investigating the addends. Therefore applying the general Leibnitz rule (C.20) twice gives the cascaded sum

$$
\begin{aligned}
&\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \left[B(\sigma,t)X_{m-1-q}(\sigma,t)\partial_\sigma \boldsymbol{x}_{q+1}(\sigma,t)\right]\right\|_\infty \\
&\qquad \leq \sum_{j=0}^{k} \binom{k}{j} \left\|\sup_{\Omega} \partial_t^{k-j} B(\sigma,t)\right\|_\infty \times \\
&\qquad\qquad \sum_{i=0}^{j} \binom{j}{i} \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{j-i} X_{m-1-q}(\sigma,t)\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^{i} \partial_\sigma \boldsymbol{x}_{q+1}(\sigma,t)\right\|_\infty .
\end{aligned}
\tag{4.61}
$$

Again, reinserting the assumed estimations (4.51), applying the boundary (C.19) for B, and utilising the inequality (C.23) leads to

$$\begin{aligned}
&\sum_{j=0}^{k}\binom{k}{j}\left\|\sup\nolimits_{\Omega}\partial_t^{k-j}B\right\|_\infty \sum_{i=0}^{j}\binom{j}{i}\left\|\sup\nolimits_{\mathbb{R}_{t_0}^+}\partial_t^{j-i}X_{m-1-q}\right\|_\infty(\sigma)\left\|\sup\nolimits_{\mathbb{R}_{t_0}^+}\partial_t^{i}\partial_\sigma \boldsymbol{x}_{q+1}\right\|_\infty(\sigma)\\
&\quad\overset{(4.51),\,(C.19)}{\leq} \delta^{k+m+1} l_{m-1-q} l_{q+1}\sum_{j=0}^{k}\binom{k}{j}((k-j)!)^\alpha\times\\
&\qquad\qquad \sum_{i=0}^{j}\binom{j}{i}((j-i+2m-4-2q)!)^\alpha((i+2q)!)^\alpha\frac{|\sigma-\xi|^{m-1}}{(m-1-q)!q!}\\
&\quad\overset{(C.23)}{\leq} \delta^{k+m+1} l_{m-1-q} l_{q+1}\sum_{j=0}^{k}\binom{k}{j}((k-j)!)^\alpha\times\\
&\qquad\qquad \left(\frac{(2m-4-2q)!(2q)!(j+2m-3)!}{(2m-3)!}\right)^\alpha\binom{m-1}{q}\frac{|\sigma-\xi|^{m-1}}{(m-1)!}\\
&\quad\leq \delta^{k+m+1} l_{m-1-q} l_{q+1}\sum_{j=0}^{k}\binom{k}{j}\frac{((k-j)!)^\alpha((j+2m-3)!)^\alpha}{(2m-3)^\alpha}\times\\
&\qquad\qquad \binom{2m-4}{2q}^{-\alpha}\binom{m-1}{q}\frac{|\sigma-\xi|^{m-1}}{(m-1)!}\\
&\quad\overset{(C.23)}{\leq} \delta^{k+m+1} l_{m-1-q} l_{q+1}\frac{((k+2m-2)!)^\alpha}{(2m-3)^\alpha(2m-2)^\alpha}\times\\
&\qquad\qquad \binom{2m-4}{2q}^{-\alpha}\binom{m-1}{q}\frac{|\sigma-\xi|^{m-1}}{(m-1)!}\,.
\end{aligned} \tag{4.62}$$

Furthermore, the associated norm of the $k+1$–th time derivative of $\boldsymbol{x}_{m-1}$ obviously satisfies the inequality

$$\left\|\sup\nolimits_{\mathbb{R}_{t_0}^+}\partial_t^{k+1}\boldsymbol{x}_{m-1}(\sigma,t)\right\|_\infty \leq \delta^{k+m}((k+2m-3)!)^\alpha l_{m-1}\frac{|\sigma-\xi|^{m-1}}{(m-1)!} \tag{4.63}$$

just by respecting (4.51a). Now, by combining the bounds (4.59)–(4.63) and inserting them in (4.58) allows to further proceed with the induction step of the growth estimation (4.51a). Respecting the identity (C.25) and the abbreviation $\epsilon_m = 1+(2m-1)^{-\alpha}$ (4.58) can be further estimated as

$$
\begin{aligned}
&\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_{m+1}\right\|_\infty(z) \\
&\quad\le \delta^{k+m+1}\left\|A^{-1}\right\|_\infty \Bigg[l_{m-1}\frac{((k+2m-3)!)^\alpha}{\delta}\left(1+\frac{1}{(2m-3)^\alpha}\right) \\
&\qquad + l_m \frac{\delta((k+2m)!)^\alpha}{(2m-1)^\alpha(2m)^\alpha} + \sum_{q=0}^{m-2} l_{m-1-q}l_{q+1}\frac{((k+2m-2)!)^\alpha}{(2m-3)^\alpha(2m-2)^\alpha}\times \\
&\qquad\qquad \binom{2m-4}{2q}^{-\alpha}\binom{m-1}{q}\Bigg]\left|\int_\xi^z\int_\xi^\eta \frac{|\sigma-\xi|^{m-1}}{(m-1)!}\,\mathrm{d}\sigma\,\mathrm{d}\eta\right| \qquad (4.64a)\\
&\quad\le \delta^{k+m+1}((k+2m)!)^\alpha\left\|A^{-1}\right\|_\infty \Bigg[\frac{\epsilon_{m-1}l_{m-1}}{\delta(k+2m-2)^\alpha(k+2m-1)^\alpha(k+2m)^\alpha} \\
&\qquad + \frac{\delta l_m}{(2m-1)^\alpha(2m)^\alpha} + \sum_{q=0}^{m-2}\frac{l_{m-1-q}l_{q+1}}{(2m-3)^\alpha(2m-2)^\alpha(k+2m-1)^\alpha(k+2m)^\alpha}\times \\
&\qquad\qquad \binom{2m-4}{2q}^{-\alpha}\binom{m-1}{q}\Bigg]\frac{|z-\xi|^{m+1}}{(m+1)!}\,.
\end{aligned}
$$

Since $k \geq 0$ it can be set to $k=0$ in the denominators of the squared bracket. This allows to write the bound in the compact form

$$
\begin{aligned}
&\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \partial_t^k \boldsymbol{x}_{m+1}\right\|_\infty(z) \\
&\quad\le \delta^{k+m+1}((k+2m)!)^\alpha\left\|A^{-1}\right\|_\infty \Bigg[\frac{\epsilon_{m-1}l_{m-1}}{\delta(2m-2)^\alpha(2m-1)^\alpha(2m)^\alpha} \\
&\qquad\qquad + \frac{\delta l_m}{(2m-1)^\alpha(2m)^\alpha} + \sum_{q=0}^{m-2}\frac{l_{m-1-q}l_{q+1}}{(2m-3)^\alpha(2m-2)^\alpha(2m-1)^\alpha(2m)^\alpha}\times \qquad (4.64b)\\
&\qquad\qquad\qquad \binom{2m-4}{2q}^{-\alpha}\binom{m-1}{q}\Bigg]\frac{|z-\xi|^{m+1}}{(m+1)!} \\
&\quad\le \delta^{k+m+1}((k+2m)!)^\alpha l_{m+1}\frac{|z-\xi|^{m+1}}{(m+1)!}\,.
\end{aligned}
$$

which completes the proof for the state estimate (4.51a). The bound for the spatial derivative $\left\|\sup_{\mathbb{R}^+_{t_0}} \partial_t^k \partial_z \boldsymbol{x}_{m+1}\right\|_\infty(z)$ can be verified in equivalent fashion just by considering single integration instead of double integration in (4.57) as well as utilising Lemma (C.26) instead of (C.25) in (4.64a).

With this the radius of convergence can be determined by setting $k=0$ in (4.51a). This allows to bound the infinite series (4.13) with general coefficients $\boldsymbol{x}_m$ by a formal power series in

$|z-\xi|$. Hence, consider

$$\begin{aligned} \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty(z) &\leq \sum_{m=0}^{\infty} \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}_m\right\|_\infty(z) \leq \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}_0(z,t)\right\|_\infty + \\ &\sum_{m=1}^{\infty} \delta^m((2m-2)!)^\alpha l_m \frac{|z-\xi|^m}{m!} \leq \delta + \sum_{m=1}^{\infty} \underbrace{\delta^m \frac{((2m-2)!)^\alpha}{m!} l_m}_{p_m} |z-\xi|^m \,. \end{aligned} \tag{4.65}$$

As shown for the linear case above it is possible to find an upper bound for the coefficients p_m. Therefore, note the following inequality.

Lemma 4.8. *The coefficients l_m defined in* (4.26) *satisfy*

$$l_m \leq \left(\left\|A^{-1}\right\|_\infty\right)^{m-1} \frac{(m-1)!}{((2m-2)!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{m-1}, \qquad m \geq 1 \tag{4.66}$$

with $h\left(\left\|A^{-1}\right\|_\infty, \delta\right)$ given by (4.54) *and $1 \leq \alpha \leq 2$.*

Proof. Note, since the recursion (4.52) for l_m and $m \geq 3$ depends on all previous $l_q, q \in \{1, \dots m-1\}$ Lemma 4.8 is defined for $m \geq 1$ and consequently needs to be proven from $m = 1$ onward. However the case $m = 1$ is trivial and therefore omitted. Consequently the mathematical induction starts with the cases $m = 2$ and $m = 3$, respectively. The two cases lead to following, i.e.,

$$l_2 = \left\|A^{-1}\right\|_\infty \frac{2+\delta}{2^\alpha} \leq \left\|A^{-1}\right\|_\infty \frac{1!}{(2!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right) \tag{4.67}$$

$$\Rightarrow \quad h\left(\left\|A^{-1}\right\|_\infty, \delta\right) \geq 2 + \delta\,, \tag{4.68}$$

$$\begin{aligned} l_3 &= \left\|A^{-1}\right\|_\infty \left(\frac{\epsilon_1 l_1}{\delta 24^\alpha} + \frac{\delta l_2}{12^\alpha} + \frac{(l_1)^2}{24^\alpha}\right) \\ &= \left\|A^{-1}\right\|_\infty \left(\frac{1+1^{-\alpha}}{\delta 24^\alpha} + \frac{\left\|A^{-1}\right\|_\infty (2\delta+\delta^2)}{24^\alpha} + \frac{1}{24^\alpha}\right) \\ &\leq \left\|A^{-1}\right\|_\infty \frac{2\delta^{-1} + 1 + 2\left\|A^{-1}\right\|\delta + \left\|A^{-1}\right\|_\infty \delta^2}{(4!)^\alpha} \\ &\leq \left\|A^{-1}\right\|_\infty^2 \frac{2!}{(4!)^\alpha} \left(\frac{1}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2\left\|A^{-1}\right\|_\infty} + \delta + \frac{\delta^2}{2}\right) \\ &\leq \left\|A^{-1}\right\|_\infty^2 \frac{2!}{(4!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^2 \end{aligned} \tag{4.69}$$

$$\Rightarrow \quad h\left(\left\|A^{-1}\right\|_\infty, \delta\right) \geq \sqrt{\frac{1}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2\left\|A^{-1}\right\|_\infty} + \delta + \frac{\delta^2}{2}}\,. \tag{4.70}$$

With respect to (4.54) for the function h this satisfies the inequalities (4.68) and (4.70), respectively, and consequently confirms the starting cases l_2 and l_3. Conducting the step from m to

$m+1$ gives the estimation

$$\begin{aligned}
l_{m+1} &= \left\|A^{-1}\right\|_\infty \Bigg[\frac{\epsilon_{m-1} l_{m-1}}{\delta(2m-2)^\alpha(2m-1)^\alpha(2m)^\alpha} + \frac{\delta l_m}{(2m-1)^\alpha(2m)^\alpha} \\
&\quad + \sum_{q=0}^{m-2} \frac{l_{m-1-q} l_{q+1}}{(2m-3)^\alpha(2m-2)^\alpha(2m-1)^\alpha(2m)^\alpha} \binom{2m-4}{2q}^{-\alpha} \binom{m-1}{q}\Bigg] \\
&\leq \left\|A^{-1}\right\|_\infty^m \Bigg[\frac{\epsilon_{m-1}(2m-3)^\alpha(m-2)!}{\left\|A^{-1}\right\|_\infty \delta((2m)!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{m-2} \\
&\quad + \frac{\delta(m-1)!}{((2m)!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{m-1} \\
&\quad + \sum_{q=0}^{m-2} \frac{(m-2-q)!q!}{((2m)!)^\alpha} \binom{m-1}{q} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{m-2}\Bigg] \\
&\leq \left\|A^{-1}\right\|_\infty^m \frac{m!}{((2m)!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^m \Bigg[\frac{(2m-3)^\alpha+1}{\left\|A^{-1}\right\|_\infty \delta m(m-1)} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{-2} \\
&\quad + \frac{\delta}{m} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{-1} + \sum_{q=0}^{m-2} \frac{1}{m(m-1-q)} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{-2}\Bigg].
\end{aligned} \tag{4.71}$$

Note, since $((2m-3)^\alpha+1)/(m(m-1)) \leq 4$, $1/m \leq 1/3$ and $\sum_{q=0}^{m-2} 1/(m(m-1-q)) \leq 1/2$ for $m \geq 3$, $1 \leq \alpha \leq 2$ the recursion (4.26) can be further bounded from above by

$$\begin{aligned}
l_{m+1} &\leq \left\|A^{-1}\right\|_\infty^m \frac{m!}{((2m)!)^\alpha} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^m \times \\
&\quad \underbrace{\left[\frac{\delta}{3} h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{-1} + \left(\frac{4}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2}\right) h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^{-2}\right]}_{\leq 1}
\end{aligned} \tag{4.72}$$

which defines the quadratic equation

$$h\left(\left\|A^{-1}\right\|_\infty, \delta\right)^2 - \frac{\delta}{3} h\left(\left\|A^{-1}\right\|_\infty, \delta\right) - \left(\frac{4}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2}\right) \geq 0\,. \tag{4.73}$$

The positive solution can be written as

$$h\left(\left\|A^{-1}\right\|_\infty, \delta\right) \geq \frac{\delta}{6} + \sqrt{\frac{\delta^2}{36} + \frac{4}{\left\|A^{-1}\right\|_\infty \delta} + \frac{1}{2}}\,. \tag{4.74}$$

Again, the inequality for $h\left(\left\|A^{-1}\right\|_\infty, \delta\right)$ holds true when (4.54) is taking into account and eventually completes the proof for Lemma 4.8. □

With this and as suggested in [55] by taking the Cauchy-Hadamard Theorem 3.22 into account the convergence radius for the series (4.38) with powers of $|z-\xi|$ can be derived as

$$\begin{aligned}
\rho_q &= \frac{1}{\limsup\limits_{m\to\infty}\left(\sqrt[m]{|p_m|}\right)} = \delta^{-1} \limsup_{m\to\infty} \left(\frac{((2m-2)!)^\alpha}{m!} l_m\right)^{-\frac{1}{m}} \\
&= \delta^{-1} \lim_{m\to\infty} \sqrt[m]{m} \left(\left\|A^{-1}\right\|_\infty h\left(\left\|A^{-1}\right\|_\infty, \delta\right)\right)^{\frac{1-m}{m}} = \left(\left\|A^{-1}\right\|_\infty \delta h\left(\left\|A^{-1}\right\|_\infty, \delta\right)\right)^{-1}
\end{aligned} \tag{4.75}$$

for $\alpha \in [1,2]$ which is actually claimed in (4.53). □

This completes the convergence analysis for the investigated coupled PDEs. For uncoupled parabolic systems the formal state and input parametrisation is a well researched field. Therefore, the next section sketches a brief summary on this topic.

4.2.3 Absolute and Uniform Convergence Analysis for the Scalar Diffusion-Convection-Reaction System and the Scalar Modified, Viscous Burgers' Equation

This paragraph shall give some results from recent literature concerning the convergence analysis of flatness based state and input parametrisation for the scalar DCRS and the scalar MVBE. As already mentioned above the FFC concept of this chapter is based upon previous contributions such as [1, 52], [54]–[61] and [76, 77]. Therefore the proofs of the following theorems are omitted and may be studied in the corresponding referenced work. At this stage it has to be pointed out that in most contributions the flat output was assumed to be at the domain boundaries, i.e. $\xi = 0$ or $\xi = \ell_\mathrm{N}$. However in this work the flat output may be assigned to any fixed but arbitrary position $\xi \in [0, \ell_\mathrm{N}]$. Moreover the formal integration procedure and subsequent analysis, i.e. (4.1) to (4.22), can be borrowed for uncoupled systems in scalar-valued form.

Furthermore, in case of the scalar MVBE the entire discussion of Section 4.2.2, including Theorem 4.7 and Lemma 4.8, can be directly translated to the uncoupled case by setting the dimension $n = 1$ and by replacing vector- and matrix-valued definitions and expressions with scalar-valued terms. Though in case of the scalar DCRS the approach can be simplified when considering certain assumptions. Therefore, let us recall the PDE (3.25) and the fact that it can reduced to an equivalent DRS (3.27) by utilising (3.26). Then the infinite series (4.13) can be written in scalar-valued form as

$$x(z,t) = \exp\left(\lambda(z,t)\right)\chi(z,t) \ = \exp\left(\lambda(z,t)\right)\sum_{m=0}^{\infty}\chi_m(z,t)\,. \tag{4.76}$$

This shows that the computation process and subsequent convergence analysis may be performed in the transformed state $\chi(z,t)$. Since the recursive formula (4.18) of DRSs does not need to deal with spatial derivatives the first two coefficients can be combined and this finally results in the simplified algorithm

$$\begin{aligned}\chi_0(z,t) &= y_0(t) + y_1(t)(z-\xi)\,,\\ \chi_m(z,t) &= \frac{1}{a}\int_\xi^z\int_\xi^\eta \left(\partial_t\chi_{m-1}(\sigma,t) - \gamma(\sigma,t)\chi_{m-1}(\sigma,t)\right)\mathrm{d}\sigma\,\mathrm{d}\eta\,, \qquad m \geq 1\,.\end{aligned} \tag{4.77}$$

with the flat output $\boldsymbol{y}(t) = [y_0(t),\, y_1(t)]^T = \exp(-\lambda(\xi,t))\,[x(\xi,t),\, \partial_z x(\xi,t)]^T$. With this modifications it is possible to verify the following theorem for the scalar DRS (3.27).

Theorem 4.9 (Absolute and uniform convergence of formal state and input parametrisation for a DRS [55])**.** *Let* $\gamma \in \mathbf{CG}^{0,\alpha_\gamma}\big(\Omega\left(\overline{\ell}_\mathrm{N}, t_0\right);\mathbb{R}\big)$ *and* $y_j \in \mathbf{G}^{\delta_{y_j}}_{\alpha_{y_j}}\big(\mathbb{R}^+_{t_0};\mathbb{R}\big)$, $j = 0, 1$ *with* $\alpha \geq 1$.

Then the series coefficient χ_m *in* (4.77) *satisfies*

$$\sup_{\mathbb{R}^+_{t_0}} \left|\partial_t^k \chi_m\right|(z) \le \delta^{k+m+1} \left((k+m)!\right)^{\alpha} l_m \left(\frac{|z-\xi|^{2m}}{(2m)!} + \frac{|z-\xi|^{2m+1}}{(2m+1)!} \right) . \tag{4.78}$$

with $\delta = \max\{\delta_\gamma, \delta_{y_1}, \delta_{y_2}\}$, $\alpha = \max\{\alpha_\gamma, \alpha_{y_0}, \alpha_{y_1}\}$ *and*

$$l_m = \begin{cases} 1, & m = 0 \\ \frac{1}{a^m} \prod_{p=1}^{m} \left(1 + \frac{1}{p^\alpha}\right), & m \ge 1 \,. \end{cases} \tag{4.79}$$

Particularly, the series (4.13) *converges absolutely and uniformly if* $1 \le \alpha \le 2$ *for all* $|z-\xi| < \rho$*, where*

$$\rho = \begin{cases} \infty, & 1 \le \alpha < 2 \\ \frac{2}{a\sqrt{\delta}}, & \alpha = 2 \,. \end{cases} \tag{4.80}$$

Remark 4.15. *With reference to* (3.28) *and* $\lambda(z,t) = \int_0^z \frac{b(s,t)}{2a} \mathrm{d}s$ *it can be concluded that the convection term coefficient* b *of an appropriate DCRS has to be continuous in* z *and it has to be an element of an appropriate Gevrey class with respect to time* t*.*

Remark 4.16. *Bearing in mind the discussion in Section 3.1.1 it is possible to reduce a coupled DCRS but with diagonal* B *to an equivalent DRS comparable to the uncoupled case. Thus, assuming an adapted definition of norms and function spaces for vector-valued systems, e.g. see Appendix C.1, then the simplified recursive computation* (4.77) *and the general statement of Theorem 4.9 are valid for the class of coupled diffusion-reaction problems as well.*

In the following paragraph the discussion focuses on the remaining trajectory planning problem. As stated in the introduction of the chapter the FF control part shall guide the agents along $\boldsymbol{x}^*$ from a starting profile $\bar{\boldsymbol{x}}_{t_0}$ to a different target deployment $\bar{\boldsymbol{x}}_{t_0+\tau_{\boldsymbol{y}}}$ within a finite time interval $\tau_{\boldsymbol{y}}$. For this it is necessary to design and construct desirable trajectories $\boldsymbol{y}^*(t)$ for the flat output $\boldsymbol{y}$ and to plan corresponding temporal paths $B^*(z,t)$ and $C^*(z,t)$ for the matrix-valued parameter functions B and C.

4.3 Trajectory Assignment for the Flat Output and Time-Varying Parameters

Desired trajectories $\boldsymbol{y}_j^*(t)$ for the components of the flat output $\boldsymbol{y}_j$, $j = 0, 1$ are assigned as

$$\boldsymbol{y}_j^*(t) = \boldsymbol{y}_{j,t_0}^* + \left(\boldsymbol{y}_{j,t_0+\tau_{\boldsymbol{y}}}^* - \boldsymbol{y}_{j,t_0}^*\right) \Phi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}(t - t_0) \,. \tag{4.81}$$

Here, the Gevrey class function $\Phi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}} \in \mathbf{G}^{\delta_{\boldsymbol{y}}}_{\alpha_{\boldsymbol{y}}}(\mathbb{R}^+_{t_0};\mathbb{R})$, $\alpha_{\boldsymbol{y}} \in (1,2]$ is locally non-analytic, i.e., $\partial_t^k \Phi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}(t)|_{t\in\{0,\tau_{\boldsymbol{y}}\}} = 0$, $\forall k \ge 1$ while $\Phi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}(0) = 0$ and $\Phi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}(\tau_{\boldsymbol{y}}) = 1$. For a more detailed analysis of Gevrey classes and its properties the reader is referred to Definition 3.10 and in addition to the work [74]. The constants $\boldsymbol{y}^*_{j,t_0} = \boldsymbol{y}^*_j(t)$ for $t \leq t_0$ and $\boldsymbol{y}^*_{i,t_0+\tau_{\boldsymbol{y}}} = \boldsymbol{y}^*_i(t)$, for $t \geq t_0 + \tau_{\boldsymbol{y}}$ may be derived from the steady state analysis of the underlying continuous model as

$$\begin{aligned}
\boldsymbol{y}^*_{0,t_0} &= \bar{\boldsymbol{x}}_{t_0}\left(\xi; \bar{B}(\xi), \bar{C}(\xi), \bar{\boldsymbol{x}}_{\mathtt{a}}, \bar{\boldsymbol{x}}_{\mathtt{1}}\right) , \\
\boldsymbol{y}^*_{0,t_0+\tau_{\boldsymbol{y}}} &= \bar{\boldsymbol{x}}_{t_0+\tau_{\boldsymbol{y}}}\left(\xi; \bar{B}(\xi), \bar{C}(\xi), \bar{\boldsymbol{x}}_{\mathtt{a}}, \bar{\boldsymbol{x}}_{\mathtt{1}}\right) , \\
\boldsymbol{y}^*_{1,t_0} &= \partial_z \bar{\boldsymbol{x}}_{t_0}\left(\xi; \bar{B}(\xi), \bar{C}(\xi), \bar{\boldsymbol{x}}_{\mathtt{a}}, \bar{\boldsymbol{x}}_{\mathtt{1}}\right) , \\
\boldsymbol{y}^*_{1,t_0+\tau_{\boldsymbol{y}}} &= \partial_z \bar{\boldsymbol{x}}_{t_0+\tau_{\boldsymbol{y}}}\left(\xi; \bar{B}(\xi), \bar{C}(\xi), \bar{\boldsymbol{x}}_{\mathtt{a}}, \bar{\boldsymbol{x}}_{\mathtt{1}}\right) .
\end{aligned} \tag{4.82}$$

The parameters $\bar{\boldsymbol{x}}_{\mathtt{a}}$ and $\bar{\boldsymbol{x}}_{\mathtt{1}}$ define the steady states of the anchor and the leader agent for $t \leq t_0$ and $t \geq t_0 + \tau_{\boldsymbol{y}_i}$, respectively. Finally, a similar time-variant selection for the desired temporal evolution of each element of the matrix-valued parameter functions B and C, i.e.,

$$\begin{aligned}
b^{ij*}(z,t) &= \bar{b}^{ij}_{t_0}(z) + \left(\bar{b}^{ij}_{t_0+\tau_{b_{ij}}}(z) - \bar{b}^{ij}_{t_0}(z)\right) \varPhi_{\tau_{b^{ij}},\omega_{b^{ij}}}(t-t_0) \\
c^{ij*}(z,t) &= \bar{c}^{ij}_{t_0}(z) + \left(\bar{c}^{ij}_{t_0+\tau_{c_{ij}}}(z) - \bar{c}^{ij}_{t_0}(z)\right) \varPhi_{\tau_{c^{ij}},\omega_{c^{ij}}}(t-t_0)
\end{aligned} \tag{4.83}$$

enables the possibility to connect two different formation profiles within the time range $\tau_{\boldsymbol{y}}$. Obviously, for this the Gevrey functions $\varPhi_{\tau_{b_{ij}},\omega_{b_{ij}}} \in \mathbf{G}^{\delta_B}_{\alpha_B}\big(\mathbb{R}^+_{t_0}; \mathbb{R}\big)$ and $\varPhi_{\tau_{c_{ij}},\omega_{c_{ij}}} \in \mathbf{G}^{\delta_C}_{\alpha_C}\big(\mathbb{R}^+_{t_0}; \mathbb{R}\big)$ of the model coefficients need to fulfil the timing constraint $0 < \{\tau_{b_{ij}}\ \tau_{c_{ij}}\} \leq \tau_{\boldsymbol{y}}$, $i,j \in \{1,2\}$. With this, recall that a particular steady state parameter configuration for $t \geq t_0 + \tau_{\boldsymbol{y}}$ (or $t \leq t_0$) refers to a rich set of steady states of the underlying PDE. Example solutions of the corresponding stationary problem were discussed in Section 3.2 and Appendix B for constant parameter matrices $\bar{B}$ and $\bar{C}$ which do not evolve in space.

In summary the construction of desired trajectories for the flat output according to (4.81) as well as the temporal assignment for the model parameters in (4.83) closes the stated planning problem. Together with the formal state and input parametrisation developed above in Section 4.1 this establishes a powerful framework for guided transitions between desired deployments of MAs. More precisely, the procedure allows to design and realize desired state trajectories $\boldsymbol{x}^*(z,t)$ which allows the MAS to move between different stationary formation profiles, e.g., a starting deployment $\bar{\boldsymbol{x}}_{t_0}(z)$ to a target deployment $\bar{\boldsymbol{x}}_{t_0+\tau_{\boldsymbol{y}}}(z)$, within the finite time interval $\tau_{\boldsymbol{y}}$. The necessary FF terms of the control inputs are inherently given by $\boldsymbol{u}^*_0(t) = \partial_t \boldsymbol{x}^*(0,t)$ and $\boldsymbol{u}^*_{\ell_{\mathrm{N}}}(t) = \partial_t \boldsymbol{x}^*(\ell_{\mathrm{N}},t)$, and may be derived from (4.12).

The following copes with observer based feedback control which, e.g., is unconditionally necessary when agents approach an unstable steady state profile. For this, a concept for coupled PDEs is elaborated in detail based on the so-called backstepping approach.

Chapter 5

Observer based Tracking Control for Multi-Agent Systems

In Chapter 4 the focus of the discussion has been laid on the FF part of the 2DOF control design and so the upcoming paragraphs deal with the development of the tracking error control approach. For this, the main objective is to establish the so-called backstepping technique for controlling a MAS based on models featuring PDEs. Here the main idea of backstepping is to formulate a target system dynamics with known stability properties and then provide a dynamic correspondence to the actual given problem. In other words a Volterra integral transformation of second kind and its inverse establish a two-way linkage which has to ensure that the problem PDE with its associated boundary conditions approaches the exponentially stable target system. Since the backstepping technique naturally derives algorithms for boundary control this concept inherently supports the introduced leader-follower-topology of the MAS. In the following the main leader agent, or short leader, is meant to be at the boundary $z = \ell_{\mathrm{N}}$ and the so-called anchor agent, or short anchor, is located at $z = 0$. However, this convention is not mandatory and may be swapped if required. Since in the following sections some novelties concerning the backstepping for coupled PDEs are presented this chapter is a key part for this thesis. Thus, note the overviewing discourse on backstepping for PDEs in the following paragraphs.

In general the backstepping technique is a well established control design concept which is very closely linked to Prof. Miroslav Krstic at University of California, San Diego. In the early 90's the theory was initially developed as a non-linear control design technique for ODEs [44]. However in 2004 the approach has been extended to distributed parameters system [80] and since then it is well-researched topic a for wide range of classes of PDEs. From the early stages the articles [82, 81] are worth to mention where backstepping based boundary control was the main objective. Based on these papers different research groups extended the concept in various directions. For instance in [54, 60] and [61] backstepping is discussed in combination with flatness. A key potential of this approach is that it enables the design of 2DOF controllers which are pursued in this work. Non-linear PDEs were covered e.g. in [56], and higher-dimensional spatial domains are explicitly investigated in [61] or [86]. Backstepping based state estimation by means of a general Luenberger observer ansatz [94] and appropriate approximation algorithms in order to recursively solve the implicit solution of the arising transformation kernel were the main objectives in e.g., [38, 39].

Backstepping applied to MASs is surely one of the latter research topics. It comes naturally with the crossover approach of modelling swarm dynamics by means of distributed parameter systems instead of, e.g., using fundamental graph theory as in [53]. Moreover backstepping for PDEs is predestined for the design of boundary controller and thus the methodology inherently supports the leader-follower agent topology. It is fair to say that the investigations in [30] and [59] paved the way for other researchers, such as the contributions by the author of this thesis [23, 24, 25, 26] or from other research groups with the publications [71**?**] and very recently [91]. In this context the journal articles [8, 25] stand out since they show backstepping based controller design applied to a real time experiments. To the author's current knowledge [25] is not only a novelty for a MAS-application but for PDE-backstepping with dynamic BCs.

For future topics regarding backstepping control of distributed parameter systems the author of this thesis sees three main topics. They can be named as

- backstepping based tracking error control and state estimation of coupled PDEs,
- in-domain backstepping control, and
- bilateral backstepping control, where two collaborating actuators are configured at the opposite boundaries of the spatial domain.

The latter was discussed in [87] for the first time and extended recently in [14]. In-domain backstepping controller design was topic in [90] and [92], and in [26] it was a key novelty for the observer design of a Burgers-type PDE. Without proof the author of this work wants to venture with the claim that bilateral and in-domain backstepping are conceptionally related very closely. It seems they consolidate in similar way as duality does for controller and observer design, meaning bilateral controller design is equivalently to in-domain observer design and vice-versa. First hints to this proposition are provided in the work in [14], where the authors managed to put bilateral controller design and observer design with in-domain measurements under a common umbrella. Recently this concept is extended in [40] for coupled PDEs. Bilateral and in-domain backstepping are promising approaches for real applications. In the author's opinion bilateral backstepping has the main advantage that it allows to design fail-safe controllers and observers. Two collaborating actuators or sensors in combination with switching algorithms may compensate single hardware failures. Moreover, control signals may decrease in magnitude while they can still provide the same dynamic behaviour. Therefore, on the one hand hardware may be get smaller since less energy is required for actuation. However, on the other hand, applying an collaborating actuation or sensing system may increase costs and maintenance for real applications. When using in-domain backstepping one should benefit from increased stability properties or smaller kernels of the Volterra transformation and consequently smaller gains while having the same stability properties as with conventional backstepping. Probably the drawback is that in-domain actuators and sensors need to be of a special type or variant. More precisely speaking, they have to support actuating on or measuring, respectively, the state $\boldsymbol{x}$ and the spatial derivative $\partial_z \boldsymbol{x}$ at the same time. This may be hardly the case for most applications, tough it would be easily applicable for controlling swarms and MASs.

Backstepping based boundary control and state estimation of coupled system are two of the main topic of recent research within the PDE community. When dealing with coupled PDEs and backstepping one gets faced with sophisticated equations for the kernel of the Volterra transformation which cannot be solved trivially due to non-commutative matrix multiplications. Here, various contributions deal with DCRSs, e.g., in [88, 89, 17], and pursue the approach to keep the target system as simple as possible equipped with partly arbitrary parameters for design purposes. They shall at least theoretically allow a free of choice setting of the convergence rate. Unfortunately the approach leads to rather complex kernel equations which, among others, depend on the complexity of the underlying problem. Contemporary work tends to increase this complexity by adding higher requirements to the parameter matrices of the problem formulation, starting with constant and equal (scalar) diffusion and moving to constant but different diffusion parameters [3, 2, 70]. From this, complexity is increased by making parameter matrices evolving in space [88, 89], including time-varying properties in [41] and even considering partial integro-differential equations (PIDEs) in [17, 18, 40]. At this state it shall be pointed out that one of the first published approaches concerning coupled PDEs and backstepping differs from the described concept. More precisely in [4] the authors put constraints on the properties of the target system and try to formulate conditions for its parameters which ensure stability. However, the mentioned contribution limits the concept to problems with constant parameters and leaving out convection entirely. As a fact this work and the mentioned restriction was the major inspiration for the upcoming investigations where the approach of [4] is generalised for DCRSs with uncoupled but distinct diffusion parameters and fully coupled convection and reaction terms whereas the latter two parameter matrices may evolve in time and space. Moreover the following concept supports all major kinds of boundary conditions, i.e., Dirichlet, Neumann, Robin as well as first order integrators as introduced in this work. Another big encouragement to this idea has been [32] which discusses as similar approach for coupled semi-linear sub-diffusion systems. It is fair to say that the developed concept does not work for any arbitrary parameter setting, even though it covers a huge set of coupled DCRSs.

Next, the following section develop a backstepping based error tracking controller and a corresponding observer algorithm which estimates the state information.

5.1 Backstepping based Boundary Control for a Class of Coupled Parabolic Partial Differential Equations

For the following let us recall the distributed parameter systems introduced in Section 3.1. First, consider the PDE of a coupled DCRS written as

$$\partial_t \boldsymbol{x}(z,t) = A\partial_z^2 \boldsymbol{x}(z,t) - B(z,t)\partial_z \boldsymbol{x}(z,t) + C(z,t)\boldsymbol{x}(z,t) \tag{5.1a}$$

and second, the coupled MVBE formulated as

$$\partial_t \boldsymbol{x}(z,t) = A\partial_z^2 \boldsymbol{x}(z,t) - B(z,t)X(z,t)\partial_z \boldsymbol{x}(z,t) + C(z,t)\boldsymbol{x}(z,t)\,. \tag{5.1b}$$

Both are defined on the domain $(z,t) \in \Omega(\ell_{\mathrm{N}}, t_0)$ and have either Dirichlet BCs

$$\boldsymbol{x}(0,t) = \boldsymbol{u}_0(t)\,, \quad \boldsymbol{x}(\ell_{\mathrm{N}},t) = \boldsymbol{u}_{\ell_{\mathrm{N}}}(t) \tag{5.1c}$$

or dynamic BCs

$$\partial_t \boldsymbol{x}(0,t) = \boldsymbol{u}_0(t)\,, \quad \partial_t \boldsymbol{x}(\ell_{\mathrm{N}},t) = \boldsymbol{u}_{\ell_{\mathrm{N}}}(t)\,. \tag{5.1d}$$

A proper IC is defined by $\boldsymbol{x}(z,t_0) = \boldsymbol{x}_0(z)$. At this point it is referred to Assumption 3.1 for recalling the properties of the parameter matrix A and the matrix-valued functions B and C. Since this work deals with a 2DOF control concept it is necessary to design the feedback part for the so-called tracking error dynamics. In this context the tracking error is defined as the difference between the actual state and the desired state derived by formal state parametrisation derived in Chapter 4. Therefore, look at the next section for the determination of the tracking error system and the backstepping based design process.

5.1.1 Backstepping based Boundary Control

As mentioned above for the 2DOF control concept it is necessary to formulate the feedback control problem in terms of the error state

$$\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) = \boldsymbol{x}(z,t) - \boldsymbol{x}^*(z,t)\,. \tag{5.2}$$

With this and taking (5.1a) into account this leads to a linear, time-variant DCRS in the error state as well. Contrary (5.1b) evaluates to the semi-linear system

$$\begin{aligned}\partial_t \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) &= A\partial_z^2 \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - B(z,t)\left(X(z,t)\partial_z \boldsymbol{x}(z,t) - X^*(z,t)\partial_z \boldsymbol{x}^*(z,t)\right)\\ &\quad + C(z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t)\,.\end{aligned} \tag{5.3}$$

Assuming small tracking errors, i.e, $\|\tilde{\boldsymbol{x}}_{\mathrm{c}}\|_2 \approx 0$ and $\|\partial_z \tilde{\boldsymbol{x}}_{\mathrm{c}}\|_2 \approx 0$, the linearisation around the desired trajectory $\{\boldsymbol{x}^*, \partial_z \boldsymbol{x}^*\}$ as introduced in (3.24) formally leads to a DCRS as well and may be written as

$$\begin{aligned}\partial_t \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) &= A\partial_z^2 \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - B(z,t)X^*(z,t)\partial_z \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t)\\ &\quad + \left(C(z,t) - B(z,t)\partial_z X^*(z,t)\right)\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t)\,.\end{aligned} \tag{5.4}$$

Since the desired state $\boldsymbol{x}^*$ and its derivative $\partial_z \boldsymbol{x}^*$ can be pre-described the tracking error systems can always be rewritten as an equivalent DCRS. Thus, for the subsequent discussion let us concentrate on the representative distributed parameter system

$$\partial_t \tilde{\boldsymbol{x}}_\mathrm{c}(z,t) = A\partial_z^2 \tilde{\boldsymbol{x}}_\mathrm{c}(z,t) - B(z,t)\partial_z \tilde{\boldsymbol{x}}_\mathrm{c}(z,t) + C(z,t)\tilde{\boldsymbol{x}}_\mathrm{c}(z,t), \tag{5.5a}$$

for $(z,t) \in \Omega(\ell_\mathrm{N}, t_0)$ and the Dirichlet BCs

$$\tilde{\boldsymbol{x}}_\mathrm{c}(0,t) = \Delta \boldsymbol{u}_0(t)\,, \quad \tilde{\boldsymbol{x}}_\mathrm{c}(\ell_\mathrm{N},t) = \Delta \boldsymbol{u}_{\ell_\mathrm{N}}(t) \tag{5.5b}$$

or dynamic BCs

$$\partial_t \tilde{\boldsymbol{x}}_\mathrm{c}(0,t) = \Delta \boldsymbol{u}_0(t)\,, \quad \partial_t \tilde{\boldsymbol{x}}_\mathrm{c}(\ell_\mathrm{N},t) = \Delta \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,, \tag{5.5c}$$

respectively. The IC is inherently given by $\tilde{\boldsymbol{x}}_\mathrm{c}(z,t_0) = \tilde{\boldsymbol{x}}_{\mathrm{c},0}(z) = \boldsymbol{x}_0(z) - \boldsymbol{x}_0^*(z)$ and the control inputs follow from $\Delta\boldsymbol{u}_0(t) = \boldsymbol{u}_0(t) - \boldsymbol{u}_0^*(t)$ and $\Delta\boldsymbol{u}_{\ell_\mathrm{N}}(t) = \boldsymbol{u}_{\ell_\mathrm{N}}(t) - \boldsymbol{u}_{\ell_\mathrm{N}}^*(t)$. Now, the backstepping technique shall establish a dynamical linkage between the error state $\tilde{\boldsymbol{x}}_\mathrm{c}$ and an target state $\boldsymbol{w}$ by means of the Volterra integral transformation of the second kind

$$\boldsymbol{w}(z,t) = \tilde{\boldsymbol{x}}_\mathrm{c}(z,t) - \int_0^z K(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s \tag{5.6}$$

and its inverse transformation

$$\tilde{\boldsymbol{x}}_\mathrm{c}(z,t) = \boldsymbol{w}(z,t) + \int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\,. \tag{5.7}$$

The so-called backstepping kernels K and G shall be matrix-valued $\mathbf{C}^2-$functions in z and s and are defined on the triangular spatial domain $(z,s) \in \mathfrak{D}_\mathrm{K}(\ell_\mathrm{N}) := \{(z,s) \in \mathbb{R}^2 \,|\, z \in [0,\ell_\mathrm{N}]\,,\, s \in [0,z]\}$ which is illustrated in Figure 5.1a. At this stage it is pointed out that the introduced setup addresses fully coupled convection terms in context with the backstepping approach. As far as known to the author and for the time being this is only covered in [89]. Substituting (5.7) into (5.6) leads to

$$\begin{aligned}\mathbf{0}_n &= \int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s - \int_0^z K(z,s,t)\left[\boldsymbol{w}(s,t) + \int_0^s G(s,p,t)\boldsymbol{w}(p,t)\,\mathrm{d}p\right]\mathrm{d}s \\ &= \int_0^z [G(z,s,t) - K(z,s,t)]\,\boldsymbol{w}(s,t)\,\mathrm{d}s - \int_0^z\int_0^s K(z,s,t)G(s,p,t)\boldsymbol{w}(p,t)\,\mathrm{d}p\,\mathrm{d}s\,.\end{aligned} \tag{5.8}$$

Changing the order of the integration and subsequent swapping the variable names for s and p, i.e., $\int_0^z\int_0^s F(s,p)\,\mathrm{d}p\,\mathrm{d}s = \int_0^z\int_s^z F(p,s)\,\mathrm{d}p\,\mathrm{d}s$ gives

$$\mathbf{0}_n = \int_0^z \left[G(z,s,t) - K(z,s,t) - \int_s^z K(z,p,t)G(p,s,t)\,\mathrm{d}p\right]\boldsymbol{w}(s,t)\,\mathrm{d}s\,. \tag{5.9}$$

Since this has to hold for every $z \in [0,\ell_\mathrm{N}]$ the kernels K and G are connected by

$$G(z,s,t) - K(z,s,t) = \int_s^z K(z,p,t)G(p,s,t)\,\mathrm{d}p\,. \tag{5.10}$$

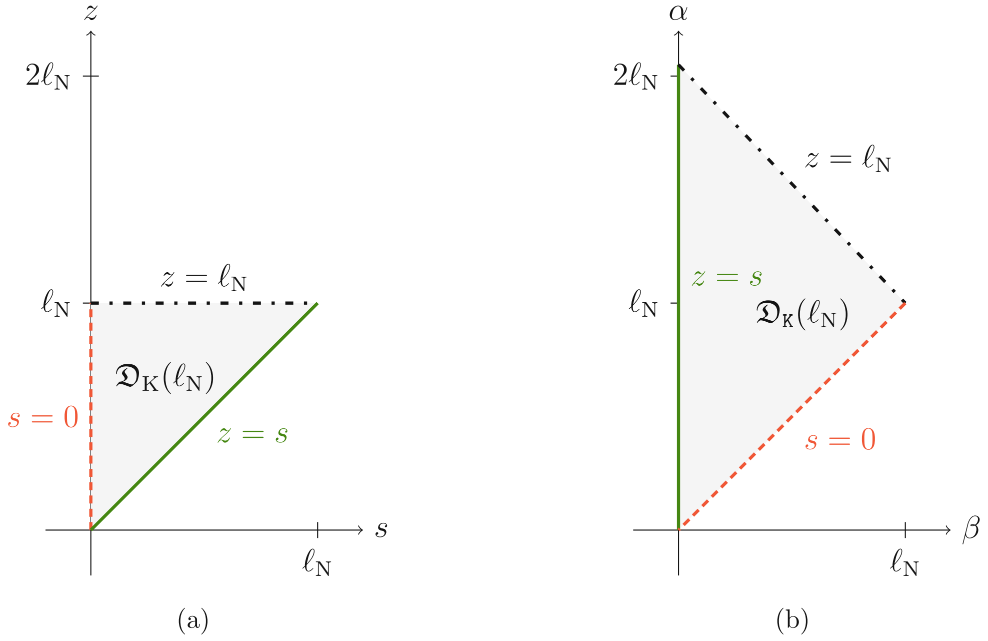


Figure 5.1: The triangular spatial domains of the backstepping kernel-PDE for tracking control: (a) in the $(z, s)-$plane for K and (b) in the scattering coordinates (α, β) for the transformed kernel $\mathtt{K}$. The size of the triangles varies with the length ℓ_{N} and therefore with the number of agents N.

Similarly, when (5.6) is substituted into (5.7) it gives the equivalent relationship

$$G(z, s, t) - K(z, s, t) = \int_s^z G(z, p, t)K(p, s, t)\,\mathrm{d}p\,. \tag{5.11}$$

For the target state $\boldsymbol{w}$ assume the following PIDE

$$\begin{aligned}\partial_t \boldsymbol{w}(z, t) &= A\partial_z^2 \boldsymbol{w}(z, t) - B(z, t)\partial_z \boldsymbol{w}(z, t) - (D(t) - M(z, t))\,\boldsymbol{w}(z, t)\\ &\quad + F(z, t)\boldsymbol{w}(0, t) + \int_0^z H(z, s, t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s, t)\,\mathrm{d}s,\end{aligned} \tag{5.12a}$$

for $(z, t) \in \Omega(\ell_{\mathrm{N}}, t_0)$ and with the Dirichlet BCs

$$\boldsymbol{w}(0, t) = \mathbf{0}_n\,, \quad \boldsymbol{w}(\ell_{\mathrm{N}}, t) = \mathbf{0}_n \tag{5.12b}$$

in case the tracking error system is formulated with (5.5b), or the dynamic BCs

$$\begin{aligned}\partial_t \boldsymbol{w}(0, t) &= -\,(D(t) - M(0, t))\,\boldsymbol{w}(0, t)\,,\\ \partial_t \boldsymbol{w}(\ell_{\mathrm{N}}, t) &= -\,(D(t) - M(\ell_{\mathrm{N}}, t))\,\boldsymbol{w}(\ell_{\mathrm{N}}, t)\end{aligned} \tag{5.12c}$$

when (5.5c) applies for the corresponding error system. Respecting the integral transformation (5.6) the initial state follows from $\boldsymbol{w}(z, t_0) = \boldsymbol{w}_0(z) = \tilde{\boldsymbol{x}}_{\mathrm{c},0}(z) - \int_0^z K_0(z, s)\tilde{\boldsymbol{x}}_{\mathrm{c},0}(s)\,\mathrm{d}s$ with the matrix $K_0(z, s)$ defined as the backstepping kernel K at $t = t_0$. In (5.12a) the matrices A and B are directly deduced from the error system (5.5a). For the reaction term the matrix D shall be diagonal and positive definite for every $t \in \mathbb{R}_{t_0}^+$. However, it will have to meet further requirements to ensure stability properties which are discussed in the following sections. Furthermore, in (5.12a) the matrix-valued functions M and F, and the integral kernel H are compensation terms which, again, are explicitly set in the upcoming paragraphs where the equations of the backstepping kernel K are derived in detail.

Before discussing the determination of the kernel equation two comments concerning notation are in order. First, assume the separation of the matrix M into the two parts

$$M(z,t) = M^{\mathrm{d}}(z,t) + M^{\mathrm{c}}(z,t) \tag{5.13}$$

where M^{d} notes the diagonal term $M^{\mathrm{d}}(z,t) = \operatorname{diag}\left[m_{11}(z,t), m_{22}(z,t)\ldots, m_{nn}(z,t)\right]$ and M^{c} has the form

$$M^{\mathrm{c}}(z,t) = \begin{bmatrix} 0 & m_{12}(z,t) & \cdots & m_{1n}(z,t) \\ m_{21}(z,t) & 0 & \cdots & m_{2n}(z,t) \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1}(z) & m_{n2}(z,t) & \cdots & 0 \end{bmatrix}. \tag{5.14}$$

Remark 5.17. *In the following for every arbitrary matrix $X \in \mathbb{R}^{n\times n}$ the notation X^{d} defines the diagonal part of X, X^{c} stands for the coupling part which has zeros as diagonal elements, X^{s} denotes the symmetric part, and X^{a} the anti-symmetric part. Note, assume $X, Y \in \mathbb{R}^{n\times n}$ then*

$$Z = X^{\mathrm{d}}Y - YX^{\mathrm{d}} = X^{\mathrm{d}}Y^{\mathrm{c}} - Y^{\mathrm{c}}X^{\mathrm{d}} = Z^{\mathrm{c}} \tag{5.15}$$

holds true, i.e., the result of the given difference has zero diagonal. Furthermore, the symmetric and anti-symmetric parts of Z can be written as

$$Z^{\mathrm{s}} = X^{\mathrm{d}}Y^{\mathrm{a}} - Y^{\mathrm{a}}X^{\mathrm{d}} \tag{5.16}$$

$$Z^{\mathrm{a}} = X^{\mathrm{d}}Y^{\mathrm{s}} - Y^{\mathrm{s}}X^{\mathrm{d}}\,. \tag{5.17}$$

Proof. The proof of (5.15) is trivial since diagonal matrices are commutative in multiplication. Consequently the difference cancels out the diagonal part, i.e.,

$$\begin{aligned} &X^{\mathrm{d}}\left(Y^{\mathrm{d}} + Y^{\mathrm{c}}\right) - \left(Y^{\mathrm{d}} + Y^{\mathrm{c}}\right)X^{\mathrm{d}} = X^{\mathrm{d}}Y^{\mathrm{d}} - Y^{\mathrm{d}}X^{\mathrm{d}} + X^{\mathrm{d}}Y^{\mathrm{c}} - Y^{\mathrm{c}}X^{\mathrm{d}} = \\ &X^{\mathrm{d}}Y^{\mathrm{c}} - Y^{\mathrm{c}}X^{\mathrm{d}} \end{aligned} \tag{5.18}$$

Furthermore, respect the properties of a symmetric matrix $\left(Z^{\mathrm{s}}\right)^T = Z^{\mathrm{s}}$ and of an anti-symmetric matrix $\left(Z^{\mathrm{a}}\right)^T = -Z^{\mathrm{a}}$. Then it can be shown that the difference (5.16) must be symmetric by

$$\left(X^{\mathrm{d}}Y^{\mathrm{a}} - Y^{\mathrm{a}}X^{\mathrm{d}}\right)^T = \left(Y^{\mathrm{a}}\right)^T\left(X^{\mathrm{d}}\right)^T - \left(X^{\mathrm{d}}\right)^T\left(Y^{\mathrm{a}}\right)^T = -Y^{\mathrm{a}}X^{\mathrm{d}} + X^{\mathrm{d}}Y^{\mathrm{a}} \tag{5.19}$$

In equivalent fashion it can be proven that the difference (5.17) has to be anti-symmetric. □

Second, the integral term in (5.12a) is written in terms of the error state $\tilde{\boldsymbol{x}}_{\mathrm{c}}$ to keep the expression short. However by making use of the inverse transformation (5.7) it always may be written in target coordinates $\boldsymbol{w}$ as

$$\begin{aligned}\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s &= \int_0^z H(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s \\ &\quad + \int_0^z H(z,s,t)\int_0^s G(s,p,t)\boldsymbol{w}(p,t)\,\mathrm{d}p\,\mathrm{d}s \\ &= \int_0^z \left(H(z,s,t) + \int_s^z H(z,p,t)G(p,s,t)\,\mathrm{d}p\right)\boldsymbol{w}(s,t)\,\mathrm{d}s\,.\end{aligned} \tag{5.20}$$

Subsequently, the next paragraph examines the derivation of the kernel equations and discusses two specializations which vastly simplify the composition of a solution of the kernel K.

5.1.2 Determination of the Backstepping Integral Kernel

As a next step to design the feedback controller it is necessary to derive the equations for the backstepping kernel. Thus, let us recall the Volterra integral transformation (5.6) and consider the following derivations in space and time

$$\partial_t\boldsymbol{w}(z,t) = \partial_t\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - \int_0^z \partial_t K(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^z K(z,s,t)\partial_t\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,, \tag{5.21}$$

$$\partial_z\boldsymbol{w}(z,t) = \partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - K(z,z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - \int_0^z \partial_z K(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,, \tag{5.22}$$

$$\begin{aligned}\partial_z^2\boldsymbol{w}(z,t) &= \partial_z^2\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - \frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - K(z,z,t)\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) \\ &\qquad - \partial_z K(z,z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - \int_0^z \partial_z^2 K(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\end{aligned} \tag{5.23}$$

with the total spatial derivative $\frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t) = [\partial_z K(z,s,t) + \partial_s K(z,s,t)]\,|_{z=s}$. Substituting the expressions (5.21)-(5.23) into the PIDE of the target system (5.12) and respecting the control error dynamics (5.5a) leads to

$$\begin{aligned}-\int_0^z \partial_t K(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^z K(z,s,t)\partial_t\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s &= -\Big[A\frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t) \\ &\quad + A\partial_z K(z,z,t) - B(z,t)K(z,z,t) + D(t) - M(z,t) + C(z,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) \\ &\quad - AK(z,z,t)\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) + F(z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) \\ &\quad + \int_0^z \Big[H(z,s,t) - A\partial_z^2 K(z,s,t) + B(z,t)\partial_z K(z,s,t) \\ &\quad + (D(t) - M(z,t))K(z,s,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\mathrm{d}s\,.\end{aligned} \tag{5.24}$$

Again, considering the error dynamics (5.5a) within the integral on the left hand side and subsequent sorting of the terms gives

$$
\begin{aligned}
\mathbf{0} = &\int_0^z K(z,s,t)\Big[A\partial_s^2\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t) - B(s,t)\partial_s\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t) + C(s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\Big]\,\mathrm{d}s \\
&- \Big[A\frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t) + A\partial_z K(z,z,t) - B(z,t)K(z,z,t) + D(t) - M(z,t) \\
&+ C(z,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - AK(z,z,t)\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) + F(z,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) \\
&+ \int_0^z \Big[H(z,s,t) + \partial_t K(z,s,t) - A\partial_z^2 K(z,s,t) + B(z,t)\partial_z K(z,s,t) \\
&+ (D(t) - M(z,t))K(z,s,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\mathrm{d}s\,.
\end{aligned}
\tag{5.25}
$$

From this, two times partial integration of the term

$$
\begin{aligned}
\int_0^z K(z,s,t)A\partial_s^2\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s = &\Big[K(z,s,t)A\partial_s\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t) - \partial_s K(z,s,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\Big]_{s=0}^{s=z} \\
&+ \int_0^z \partial_s^2 K(z,s,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,,
\end{aligned}
\tag{5.26}
$$

one time partial integration of the expression

$$
\begin{aligned}
\int_0^z K(z,s,t)B(s,t)\partial_s\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s = &\Big[K(z,s,t)B(s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\Big]_{s=0}^{s=z} \\
&- \int_0^z \Big[\partial_s K(z,s,t)B(s,t) + K(z,s,t)\partial_s B(s,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,,
\end{aligned}
\tag{5.27}
$$

and recognising the resulting terms in (5.25) allows to write the equation as

$$
\begin{aligned}
\mathbf{0} = &-\Big[A\frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t) + A\partial_z K(z,z,t) + \partial_s K(z,z,t)A \\
&- B(z,t)K(z,z,t) + K(z,z,t)B(z,t) + D(t) - M(z,t) + C(z,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) \\
&+ \Big[F(z,t) + \partial_s K(z,0,t)A + K(z,0,t)B(0,t)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) - K(z,0,t)A\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) \\
&+ \Big[K(z,z,t)A - AK(z,z,t)\Big]\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) + \int_0^z \Big[H(z,s,t) + \partial_t K(z,s,t) \\
&- A\partial_z^2 K(z,s,t) + \partial_s^2 K(z,s,t)A + B(z,t)\partial_z K(z,s,t) + \partial_s K(z,s,t)B(s,t) \\
&+ (D(t) - M(z,t))K(z,s,t) + K(z,s,t)\left(C(s,t) + \partial_s B(s,t)\right)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\mathrm{d}s\,.
\end{aligned}
\tag{5.28}
$$

Since (5.28) has to be satisfied for all $z \in [0, \ell_{\mathrm{N}}]$ and $t > t_0$ the kernel $K \in \mathbf{CG}^{2,2,\alpha}(\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \mathbb{R}_{t_0}^+; \mathbb{R}^{n\times n})$ with $1 \leq \alpha \leq 2$ has to fulfil the PDE

$$
\begin{aligned}
\partial_t K(z,s,t) = &A\partial_z^2 K(z,s,t) - \partial_s^2 K(z,s,t)A - B(z,t)\partial_z K(z,s,t) \\
&- \partial_s K(z,s,t)B(s,t) - (D(t) - M(z,t))K(z,s,t) \\
&- K(z,s,t)\left(C(s,t) + \partial_s B(s,t)\right) - H(z,s,t)
\end{aligned}
\tag{5.29a}
$$

defined on the open triangular domain $(z,s) \in [0,\ell_\mathrm{N}] \times [z,\ell_\mathrm{N}]$ and for all $t \in \mathbb{R}^+_{t_0}$. Setting $F(z,t) = 0_{n,n}$ in case of Dirichlet BCs (5.1c) and $F(z,t) = -\partial_s K(z,0,t)A$ for the dynamic BCs (5.1d) as well as respecting (5.13) then the BCs of the kernel at $z = s \in [0,\ell_\mathrm{N}]$ and $s = 0$, $z \in [0,\ell_\mathrm{N}]$ lead to

$$\begin{aligned} A\frac{\mathrm{d}}{\mathrm{d}z}K(z,z,t) + A\partial_z K(z,z,t) + \partial_s K(z,z,t)A = B(z,t)K(z,z,t) \\ - K(z,z,t)B(z,t) - D(t) + M(z,t) - C(z,t)\,, \end{aligned} \tag{5.29b}$$

$$K(z,z,t)A = AK(z,z,t)\,, \tag{5.29c}$$

$$K(z,0,t) = 0_{n,n}\,. \tag{5.29d}$$

In view of (5.6) the IC of the kernel at $t = t_0$ has to meet $\int_0^z K(z,s,t_0)\tilde{\boldsymbol{x}}_{\mathrm{c},0}(z)\mathrm{d}s = \tilde{\boldsymbol{x}}_{\mathrm{c},0}(z) - \boldsymbol{w}_0(z)$. Hence, the explicit assignment $K(z,s,t_0) = K_0(z,s)$ which shall be consistent with the BCs (5.29b)-(5.29d), implicitly defines the IC of the target state $\boldsymbol{w}_0(z)$.

Remark 5.18. *Assuming Neumann boundary conditions for the error system* (5.5) *the compensation term F demands the setting $F(z,t) = K(z,0,t)B(0,t)$ and the boundary condition* (5.29d) *of the kernel at $s = 0$ has to be modified to $\partial_s K(z,0,t) = 0_{n,n}$.*

Equations (5.29) reveal that commutative multiplication between the coefficient matrices A, B and the kernel K would largely simplify the construction of solutions for K. Moreover, since A is diagonal it is desirable to construct a diagonal backstepping kernel K at the boundary $z = s$, i.e. $K(z,z,t) = K^\mathrm{d}(z,z,t) = \mathrm{diag}\,[k_{11}(z,z,t),\dots,k_{nn}(z,z,t)]$. In other words imposing the diagonal kernel K^d at $z = s$ then the matrix-valued function M is enforced to the setting

$$M(z,t) = C^\mathrm{c}(z,t) + K^\mathrm{d}(z,z,t)B(z,t) - B(z,t)K^\mathrm{d}(z,z,t)\,, \tag{5.30}$$

and for diagonal $B(z,t) = B^\mathrm{d}(z,t)$ to

$$M(z,t) = C^\mathrm{c}(z,t)\,, \tag{5.31}$$

whereas C^c is the coupling part of C. Note that for diagonal kernel at $z = s$ the term $K^\mathrm{d}(z,z,t)B(z,t) - B(z,t)K^\mathrm{d}(z,z,t)$ has coupling format, see Remark 5.17 for details. In this case this means $M = M^\mathrm{c}$ and $M^\mathrm{d} = 0_{n,n}$. With this (5.29c) is satisfied trivially and (5.29b) simplifies to

$$\frac{\mathrm{d}}{\mathrm{d}z}K^\mathrm{d}(z,z,t) = -\frac{1}{2}A^{-1}\left(D(t) + C^\mathrm{d}(z,t)\right)\,. \tag{5.32}$$

Here C^d stands for the remaining diagonal part of C. Integrating (5.32) and subsequent substitution in (5.30) together with Remark 5.17 gives the explicit formulation

$$\begin{aligned} M(z,t) = C^\mathrm{c}(z,t) + \frac{z}{2}\left(A^{-1}D(t)B^\mathrm{c}(z,t) - B^\mathrm{c}(z,t)A^{-1}D(t)\right) \\ - \frac{1}{2}\int_0^z \left(A^{-1}C^\mathrm{d}(s,t)B^\mathrm{c}(z,t) - B^\mathrm{c}(z,t)A^{-1}C^\mathrm{d}(s,t)\right)\mathrm{d}s\,. \end{aligned} \tag{5.33}$$

Remark 5.19. *The kernel G for the inverse backstepping transformation* (5.7) *can be deduced in similar fashion as K above. Then the PDE for G can be written as*

$$\begin{aligned}\partial_t G(z,s,t) = A\partial_z^2 G(z,s,t) - \partial_s^2 G(z,s,t)A - B(z,t)\partial_z G(z,s,t)\\ - \partial_s G(z,s,t)B(s,t) + C(z,t)G(z,s,t)\\ + G(z,s,t)\left(D(t) - M(s,t) - \partial_s B(s,t)\right) - J(z,s,t)\end{aligned} \tag{5.34a}$$

with the boundary conditions

$$\begin{aligned}A\frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t) + A\partial_z G(z,z,t) + \partial_s G(z,z,t)A = B(z,t)G(z,z,t)\\ - G(z,z,t)B(z,t) - D(t) + M(z,t) - C(z,t)\,,\\ G(z,z,t)A = AG(z,z,t)\,,\\ G(z,0,t) = 0_{n,n},\,.\end{aligned} \tag{5.34b}$$

Here the matrix M is identical to the expression in (5.30) *and was initially introduced in the PIDE of the target system* (5.12a). *The term $J(z,s,t)$ has a similar task as $H(z,s,t)$ in* (5.29a). *Both act as compensation terms to simplify their corresponding PDE. However they are not independent from each other, which leads to the following definition.*

Definition 5.5. *Consider the matrix-valued compensation terms $H(z,s,t)$ and $J(z,s,t)$. The former is associated to the PDE* (5.29a) *for the determination of the kernel K, and the latter corresponds to the PDE* (5.34a) *which defines the kernel G of the related inverse backstepping transformation. The terms H and J are connected by the so-called transversal compensation condition (TCC) with respect to their backstepping kernels K and G, respectively. More precisely, they have to satisfy*

$$J(z,s,t) - H(z,s,t) = \int_s^z \left[J(z,p,t)K(p,s,t) + G(z,p,t)H(p,s,t)\right]\mathrm{d}p\,. \tag{5.35}$$

The entire discussion of the derivation of the inverse kernel and the TCC is conducted in Section 5.1.3. The interested reader is especially referred to D.2 where (5.35) is proven for the two different types of compensations which are presented in the following sections of this work.

As an intermediate summary, it can be stated that with (5.31) or (5.33), respectively, the compensation term M only depends on the parameter setting of the error system (5.5a) and the design coefficients of D. In this context the diagonal elements of D remain as degrees of freedom which allow to adjust appropriate stability properties. Furthermore the matrix-valued compensation M ensures that the kernel can be imposed to a diagonal form at the boundary $z = s$. However, this discussion still leaves behind the analysis of the kernel PDE (5.29a) with an appropriate setting of H. Therefore two particular cases are investigated in the following paragraphs. Both concepts have the common goal to simplify the kernel PDE in a way that it allows to use solutions and algorithms established for the computation of kernels arsing for DPSs governed by uncoupled PDEs. The first type of compensation aims for the goal to compensate the coupling terms. Then this allows to impose a diagonal kernel over the entire spatial domain. The design is based on the results in [3, 4, 70]. Though, here the concept largely extends and generalises the idea for non-constant parameter configurations and dynamic BCs. The second approach tries to modify the non-commutative terms which involve spatial derivatives into expressions with scalar coefficients.

Coupling Compensation

As mentioned above this approach imposes a backstepping kernel with diagonal structure. With this (5.29a) can be formulated as

$$\begin{aligned}\partial_t K^{\mathrm{d}}(z,s,t) &= A\left(\partial_z^2 K^{\mathrm{d}}(z,s,t) - \partial_s^2 K^{\mathrm{d}}(z,s,t)\right) - B^{\mathrm{d}}(z,t)\partial_z K^{\mathrm{d}}(z,s,t)\\ &\quad - B^{\mathrm{d}}(s,t)\partial_s K^{\mathrm{d}}(z,s,t) - \left(D(t) + C^{\mathrm{d}}(s,t) + \partial_s B^{\mathrm{d}}(s,t)\right) K^{\mathrm{d}}(z,s,t)\,,\end{aligned} \tag{5.36}$$

This demands a compensation term H defined as

$$\begin{aligned}H(z,s,t) &= -B^{\mathrm{c}}(z,t)\partial_z K^{\mathrm{d}}(z,s,t) - \partial_s K^{\mathrm{d}}(z,s,t)B^{\mathrm{c}}(s,t)\\ &\quad + M^{\mathrm{c}}(z,t)K^{\mathrm{d}}(z,s,t) - K^{\mathrm{d}}(z,s,t)\left(C^{\mathrm{c}}(s,t) + \partial_s B^{\mathrm{c}}(s,t)\right)\end{aligned} \tag{5.37}$$

and the matrix M is given by the setting (5.30). From here all matrices and matrix valued functions are diagonal and consequently the multiplication operation is commutative. Applying the Hopf-Cole-transformation as introduced in Section 3.1.1, with the transformation matrix $T(z,s,t) = \exp\left(\Lambda(z,t) - \Lambda(s,t)\right)$ and its argument $\Lambda(z,t) = (A^{-1}/2)\int_0^z B^{\mathrm{d}}(p,t)\mathrm{d}p$, the PDE of the transformed kernel $\breve{K}^{\mathrm{d}}(z,s,t) = T(z,s,t)K^{\mathrm{d}}(z,s,t)$ can be rewritten as

$$\partial_t \breve{K}^{\mathrm{d}}(z,s,t) = A\left(\partial_z^2 \breve{K}^{\mathrm{d}}(z,s,t) - \partial_s^2 \breve{K}^{\mathrm{d}}(z,s,t)\right) - \Gamma(z,s,t)\breve{K}^{\mathrm{d}}(z,s,t) \tag{5.38}$$

with $\Gamma(z,s,t) = D(t) + C^{\mathrm{d}}(s,t) + \partial_s B^{\mathrm{d}}(s,t) + A\left(\partial_z \Lambda(z,t)\right)^2 + A\left(\partial_s \Lambda(s,t)\right)^2 - A\partial_z^2 \Lambda(z,t) - A\partial_s^2 \Lambda(s,t) + \partial_t \Lambda(z,t) - \partial_t \Lambda(s,t)$. With this it is easy to see that Γ is diagonal as well, i.e., $\Gamma(z,s,t) = \operatorname{diag}\left[\gamma_{11}(z,s,t),\ldots,\gamma_{nn}(z,s,t)\right]$. As a consequence the kernel PDE may be solved independently for each element. Therefore, noting that $T(z,z,t) = I_n$ the uncoupled transformed kernel-PDEs can be formulated as

$$\begin{aligned}&\partial_t \breve{k}_{ii}(z,s,t) = a_{ii}\left(\partial_z^2 \breve{k}_{ii}(z,s,t) - \partial_s^2 \breve{k}_{ii}(z,s,t)\right) - \gamma_{ii}(z,s,t)\breve{k}_{ii}(z,s,t)\\ &\frac{\mathrm{d}}{\mathrm{d}z}\breve{k}_{ii}(z,z,t) = -\frac{1}{2a_{ii}}\left(d_{ii}(t) + c_{ii}(z,t)\right)\\ &\breve{k}_{ii}(z,0,t) = 0\end{aligned} \tag{5.39}$$

with $i = 1,\ldots,n$. Here the established methods developed for uncoupled backstepping controller design given distributed parameter systems can be applied to (5.39) in order to solve the kernel equations, e.g. see [55]. Thus, the scattering coordinates

$$\alpha = z + s \text{ and } \beta = z - s \tag{5.40}$$

are introduced, so that $\breve{k}(z,s,t) = \breve{k}\left(\frac{\alpha+\beta}{2}, \frac{\alpha-\beta}{2}, t\right) = \mathtt{k}\left(\alpha(z,s), \beta(z,s), t\right)$. With this the spatial domain adapts to $\mathfrak{D}_{\mathtt{K}}(\ell_{\mathrm{N}}) := \{(\alpha,\beta) \in \mathbb{R}^2 \,|\, \alpha \in [\beta, 2\ell_{\mathrm{N}} - \beta],\, \beta \in [0, \ell_{\mathrm{N}}]\}$ and is shown in Figure 5.1b. The differential operators modify to

$$
\begin{aligned}
\frac{\partial}{\partial z} &= \frac{\partial}{\partial \alpha} + \frac{\partial}{\partial \beta}\,, & \frac{\partial}{\partial s} &= \frac{\partial}{\partial \alpha} - \frac{\partial}{\partial \beta}\,,\\
\frac{\partial^2}{\partial z^2} &= \frac{\partial^2}{\partial \alpha^2} + 2\frac{\partial}{\partial \alpha}\frac{\partial}{\partial \beta} + \frac{\partial^2}{\partial \beta^2}\,, & \frac{\partial^2}{\partial s^2} &= \frac{\partial^2}{\partial \alpha^2} - 2\frac{\partial}{\partial \alpha}\frac{\partial}{\partial \beta} + \frac{\partial^2}{\partial \beta^2}\,.
\end{aligned}
\tag{5.41}
$$

Neglecting the double indices in (5.39) for simplicity and applying the coordinate change transforms the equations into the integrable form

$$
\begin{aligned}
&\partial_t \mathtt{k}(\alpha,\beta,t) = 4a\partial_\alpha\partial_\beta \mathtt{k}(\alpha,\beta,t) - \gamma\left(\frac{\alpha+\beta}{2}, \frac{\alpha-\beta}{2}, t\right)\mathtt{k}(\alpha,\beta,t)\,,\\
&\mathtt{k}(\alpha,0,t) = -\frac{1}{4a}\int_\beta^\alpha \left[d(t) + c\left(\frac{q}{2}, t\right)\right]\mathrm{d}q\,,\\
&\mathtt{k}(\alpha,\alpha,t) = 0\,.
\end{aligned}
\tag{5.42}
$$

Under the assumption $c \in \mathbf{CG}^{0,\delta}\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, t_0\right);\mathbb{R}\right)$ and $\gamma \in \mathbf{CG}^{0,0,\delta}\left(\mathfrak{D}_{\mathtt{K}}(\ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0};\mathbb{R}\right)$ with $1 \leq \delta \leq 2$ a strong solution can be determined by applying the formal integration method. Therefore, integrating spatially twice gives the implicit solution

$$
\begin{aligned}
\mathtt{k}(\alpha,\beta,t) = &-\frac{1}{4a}\int_\beta^\alpha \left[d(t) + c\left(\frac{q}{2}, t\right)\right]\mathrm{d}q\\
&+ \frac{1}{4a}\int_\beta^\alpha\int_0^\beta \left[\partial_t \mathtt{k}(q,p,t) + \gamma\left(\frac{q+p}{2}, \frac{q-p}{2}, t\right)\mathtt{k}(q,p,t)\right]\mathrm{d}p\mathrm{d}q\,.
\end{aligned}
\tag{5.43}
$$

which can be solved by recursive algorithms, e.g., successive approximation [38]. Assuming that the solution of the kernel can be formulated as the infinite functional series

$$
\mathtt{k}(\alpha,\beta,t) = \sum_{m=1}^{\infty} \mathtt{k}_m(\alpha,\beta,t)
\tag{5.44}
$$

with the terms computed according to

$$
\mathtt{k}_1(\alpha,\beta,t) = -\frac{1}{4a}\int_\beta^\alpha \left[d(t) + c\left(\frac{q}{2}, t\right)\right]\mathrm{d}q
\tag{5.45a}
$$

and

$$
\mathtt{k}_m(\alpha,\beta,t) = \frac{1}{4a}\int_\beta^\alpha\int_0^\beta \left[\partial_t \mathtt{k}_{m-1}(q,p,t) + \gamma\left(\frac{q+p}{2}, \frac{q-p}{2}, t\right)\mathtt{k}_{m-1}(q,p,t)\right]\mathrm{d}p\mathrm{d}q
\tag{5.45b}
$$

for $m > 1$, it can be shown that (5.44) is absolutely and uniformly convergent. Here, the proof is omitted and the interested reader is referred to the corresponding methods in [38, 39, 55] applied to control error systems with Dirichlet BCs. At this state it is pointed out that for dynamic BCs it is necessary to evaluate the first and second order of the derivative with respect to s

for the determination of controller gains, e.g., see [23]. Considering the differential operators (5.42) the first derivative can be computed analytically by

$$\partial_s \breve{k}(z,s,t) = \partial_\alpha \mathtt{k}(\alpha,\beta,t) - \partial_\beta \mathtt{k}(\alpha,\beta,t) \tag{5.46}$$

whereas $\partial_\alpha \mathtt{k}$ can be governed by single integration of (5.42) with respect to β and $\partial_\beta \mathtt{k}$ by formal differentiation of (5.43) with respect to β. Due to linearity both functions can be included in the recursive calculation of $\mathtt{k}$ and therefore $\partial_s \breve{k}$ already derived in parallel. The second derivative with respect to s as well as the differentiation to t needs to be computed by numerical methods. Finally, applying the inverse of the Hopf-Cole-Transformation, i.e., $K(z,s,t) = T^{-1}(z,s,t)\breve{K}(z,s,t)$ with $T^{-1}(z,s,t) = \exp(\Lambda(s,t) - \Lambda(z,t))$ the kernel K is obtained for further evaluation and for the remaining steps in terms of the controller design process. Derivatives of K follow from formal differentiation of the Hopf-Cole-Transformation accordingly.

Non-commutative Compensation

The second and alternative approach dissolves the non-commutativity of the kernel-PDE by means of substituting the matrix-valued parameter A and parameter function B with a scalar counterpart. By means of the abbreviations $\Delta A = A - aI_n$, $\Delta B(z,t) = B(z,t) - b(z,t)I_n$, and $M_{AB}(z,t) = M(z,t) - C^{\mathrm{c}}(z,t)$ the compensations terms are set to

$$\begin{aligned} H(z,s,t) &= \Delta A \partial_z^2 K(z,s,t) - \partial_s^2 K(z,s,t)\Delta A - \Delta B(z,t)\partial_z K(z,s,t) \\ &\quad - \partial_s K(z,s,t)\Delta B(s,t) - K(z,s,t)\partial_s \Delta B(s,t) + M_{AB}(z,t)K(z,s,t)\,, \end{aligned} \tag{5.47}$$

$$M(z,t) = 2\Delta A \frac{\mathrm{d}}{\mathrm{d}z} K^{\mathrm{d}}(z,z,t) + K^{\mathrm{d}}(z,z,t)B(z,t) - B(z,t)K^{\mathrm{d}}(z,z,t) + C^{\mathrm{c}}(z,t)\,. \tag{5.48}$$

Then the kernel dynamics can be expressed as

$$\begin{aligned} \partial_t K(z,s,t) &= a\left(\partial_z^2 K(z,s,t) - \partial_s^2 K(z,s,t)\right) - b(z,t)\partial_z K(z,s,t) \\ &\quad - b(s,t)\partial_s K(z,s,t) - \left(D(t) - C^{\mathrm{c}}(z,t)\right)K(z,s,t) \\ &\quad - K(z,s,t)\left(C(s,t) + \partial_s b(s,t) I_n\right)\,. \end{aligned} \tag{5.49}$$

Equivalently to the discussion above in Section 5.1.2, applying the Hopf-Cole-transformation with the scalar rule $\tau(z,s,t) = \exp(\lambda(z,t) - \lambda(s,t))$ and its argument $\lambda(z,t) = \frac{1}{2a}\int_0^z b(p,t)\mathrm{d}p$, allows the PDE to be formulated as

$$\begin{aligned} \partial_t \breve{K}(z,s,t) &= a\left(\partial_z^2 \breve{K}(z,s,t) - \partial_s^2 \breve{K}(z,s,t)\right) - \left(D(t) - C^{\mathrm{c}}(z,t)\right)\breve{K}(z,s,t) \\ &\quad - \breve{K}(z,s,t)\Gamma(z,s,t) \end{aligned} \tag{5.50}$$

with $\Gamma(z,s,t) = C(s,t) + \big(\partial_s b(s,t) + a\left(\partial_z \lambda(z,t)\right)^2 + a\left(\partial_s \lambda(s,t)\right)^2 - a\partial_z^2 \lambda(z,t) - a\partial_s^2 \lambda(s,t) + \partial_t \lambda(z,t) - \partial_t \lambda(s,t)\big) I_n$. Making use of $\tau(z,z,t) = 1$ the associated uncoupled boundary conditions derive to

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}z}\breve{K}(z,z,t) = -\frac{1}{2a}\left(D(t) + C^{\mathrm{d}}(z,t)\right) \\ &\breve{K}(z,0,t) = 0\,. \end{aligned} \tag{5.51}$$

In scattering coordinates (α, β) the implicit solution of the kernel $\breve{K}(z,s,t) = \breve{K}\left(\frac{\alpha+\beta}{2}, \frac{\alpha-\beta}{2}, t\right) = \mathtt{K}\left(\alpha(z,s), \beta(z,s), t\right)$ may be written as

$$\begin{aligned}\mathtt{K}(\alpha,\beta,t) = -\frac{1}{4a}\int_{\beta}^{\alpha}\left[D(t) + C^{\mathrm{d}}\left(\frac{q}{2}, t\right)\right]\mathrm{d}q + \frac{1}{4a}\int_{\beta}^{\alpha}\int_{0}^{\beta}\Bigg[\partial_t \mathtt{K}(q,p,t)\\ + \left(D(t) - C^{\mathrm{c}}\left(\frac{q+p}{2}\right)\right)\mathtt{K}(q,p,t) + \mathtt{K}(q,p,t)\Gamma\left(\frac{q+p}{2}, \frac{q-p}{2}, t\right)\Bigg]\mathrm{d}p\mathrm{d}q\,.\end{aligned} \tag{5.52}$$

Again, assuming recursive computation methods the solution of the kernel can be formulated as the infinite functional series $\mathtt{K}(\alpha,\beta,t) = \sum_{m=1}^{\infty}\mathtt{K}_m(\alpha,\beta,t)$ with the terms

$$\mathtt{K}_1(\alpha,\beta,t) = -\frac{1}{4a}\int_{\beta}^{\alpha}\left[D(t) + C^{\mathrm{d}}\left(\frac{q}{2}, t\right)\right]\mathrm{d}q \tag{5.53a}$$

$$\begin{aligned}\mathtt{K}_m(\alpha,\beta,t) = \frac{1}{4a}\int_{\beta}^{\alpha}\int_{0}^{\beta}\Bigg[\partial_t \mathtt{K}_{m-1}(q,p,t) + \left(D(t) - C^{\mathrm{c}}\left(\frac{q+p}{2}\right)\right)K_{m-1}(q,p,t)\\ + \mathtt{K}_{m-1}(q,p,t)\Gamma\left(\frac{q+p}{2}, \frac{q-p}{2}, t\right)\Bigg]\mathrm{d}p\mathrm{d}q\,, \qquad m > 1\,.\end{aligned} \tag{5.53b}$$

It can be shown that the series is absolutely and uniformly convergent. The proof may be conducted equivalently as for scalar systems. However the supremum of scalar functions may be exchanged by the induced operator max −norm as introduced in Definition 3.8. Therefore again, a detailed discussion on the convergence as well as further manipulation to obtain the kernel K and its derivatives with respect to s and t are refrained. The interested reader is referred to the remarks and literature mentioned above at the end of Section 5.1.2.

As a brief summary two compensation methods were introduced which explicitly define the matrix-valued function M and the integral kernel H, i.e., the compensation terms of the target system (5.12a). On the one hand, the coupling compensation concept enforces an diagonal kernel over the entire spatial domain and therefore demands the compensation of the coupling parts of the convection and reaction terms. On the other hand, the non-commutative compensation approach establishes scalar parameter values for the expressions with spatial derivatives. Depending on the parameter configuration of the error system (5.5) one approach may outperform the other. Apart from that, it has to be shown that the appropriate adjustment of the parameter matrix D allows the definition of an exponential stable target system (5.12) which includes the compensation terms M and H. Therefore in the next section the discussion is carried forward regarding stability properties of the target system and consequently ensuring closed loop stability of the error system.

5.1.3 Stability of the Tracking Error System

Basically the stability of the closed loop control error system is proven in two steps. First, the boundedness of the backstepping kernels K and G is mandatory to ensure stability. Second, the stability properties of the target system directly influence the stability of the tracking error system. In this context the boundedness for K is inherently given since in Section 5.1.2 it is shown that strong solutions exists for the two investigated cases. By means of (5.10) and (5.11), respectively, the boundedness of G may be directly deduced from the boundedness of K. More precisely, by applying the triangle inequality this leads to

$$\begin{aligned}&\varphi_{\mathrm{K}}(t)-\varphi_{\mathrm{G}}(t)\le\varphi_{\mathrm{K}}(t)\varphi_{\mathrm{G}}(t)\ell_{\mathrm{N}}\quad\text{and}\quad\varphi_{\mathrm{G}}(t)-\varphi_{\mathrm{K}}(t)\le\varphi_{\mathrm{G}}(t)\varphi_{\mathrm{K}}(t)\ell_{\mathrm{N}}\\&\to\varphi_{\mathrm{K}}(t)=\varphi_{\mathrm{G}}(t)\end{aligned}\tag{5.54}$$

with $\varphi_{\mathrm{K}}(t)=\max_{\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})}\big\|K(z,s,t)\big\|_2$ and $\varphi_{\mathrm{G}}(t)=\max_{\mathfrak{D}_{\mathrm{G}}(\ell_{\mathrm{N}})}\big\|G(z,s,t)\big\|_2$, see (C.10) for details. With this, the proof of the stability of the target system in appropriate spaces is the topic of the discussion in the next paragraphs.

Stability of the Target System for Dirichlet Boundary Conditions

Note, in this section $\boldsymbol{w}\in\mathbf{L}^2\left(\Omega\left(\bar{\ell}_{\mathrm{N}},\bar{t}_0\right);\mathbb{R}^n\right)$ with the $\mathbf{L}^2-$norm

$$\big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}(t)=\left(\int_0^{\ell_{\mathrm{N}}}\boldsymbol{w}^T(z,t)\boldsymbol{w}(z,t)\mathrm{d}z\right)^{\frac{1}{2}}.\tag{5.55}$$

is considered as a suitable state space for the target system. With this and respecting the following definition of functions stability properties are analysed subsequently. As already indicated above the time function $\varphi_{\mathrm{G}}(t)$ stands for the spectral norm of the backstepping kernel $G(z,s,t)$ of the inverse transformation (5.7) for every fixed time $t\in\mathbb{R}^+_{t_0}$ and $\varphi_{\mathrm{H}}(t)$ denotes the spectral norm of the integral kernel $H(z,s,t)$ of the target system (5.12) again for every fixed time $t\in\mathbb{R}^+_{t_0}$. Both define upper bounds for the spatial domain $\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})$. In general for these statements and all upcoming expressions regarding the mentioned spectral norms see Definition 3.9 in the Appendix. For details on the definition of $\max_{\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})}\big\|\cdot\big\|_2$ which applies to matrix-valued functions in two spatial variables see (C.10); for matrix-valued functions in a single spatial variable see (C.11). Moreover, note that M^{s} and B^{s} indicate the symmetric parts of the corresponding matrices M and B. With this the following theorem is formulated accordingly.

Theorem 5.10. *Let $D(t)$ be diagonal in* (5.12a) *and let $\boldsymbol{\nu}\in\mathbb{R}^n/\{\mathbf{0}_n\}$. If D fulfils the inequality*

$$\begin{aligned}\boldsymbol{\nu}^T D(t)\boldsymbol{\nu}\ge\boldsymbol{\nu}^T&\left(M^{\mathrm{s}}(z,t)+\frac{1}{2}\partial_z B^{\mathrm{s}}(z,t)-\frac{1}{4}B^{\mathrm{a}}(z,t)A^{-1}B^{\mathrm{a}}(z,t)\right)\boldsymbol{\nu}^T\\&+\left(\varphi_{\mathrm{H}}(t)\left(\frac{\ell_{\mathrm{N}}}{\sqrt{2}}+\varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}{}^2}{2}\right)+\lambda_{\min}(t)\right)\|\boldsymbol{\nu}\|_2^2\end{aligned}\tag{5.56}$$

for all $(z,t)\in\Omega(\ell_{\mathrm{N}},t_0)$ and for some $\lambda_{\min}(t)>0$ with $t>t_0$, then the zero equilibrium of the target system dynamics (5.12a) *with boundary conditions* (5.12b) *is exponentially stable in the norm $\big\|\cdot\big\|_{\mathbf{L}^2}$, i.e., there exists a constant $\rho>0$ so that the inequality*

$$\big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}(t)\le\rho\big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}(t_0)\exp\left(-\kappa_{\mathrm{c}}(t,t_0)\right)\tag{5.57}$$

holds true with $\kappa_{\mathrm{c}}(t,t_0)=\int_{t_0}^t\lambda_{\min}(\tau)\mathrm{d}\tau$.

Proof. For the following stability analysis consider the Lyapunov functional

$$V_1(t) = \frac{1}{2}\|\boldsymbol{w}\|^2_{\mathbf{L}^2}(t) = \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}(z,t)^T \boldsymbol{w}(z,t)\,\mathrm{d}z\,. \tag{5.58}$$

Then with (5.12a) the time derivative of V gives

$$\begin{aligned}\dot{V}_1(t) = \int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\partial_t \boldsymbol{w}(z,t)\,\mathrm{d}z &= -\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\left(D(t) - M(z,t)\right)\boldsymbol{w}(z,t)\,\mathrm{d}z\\ &+ \int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\left(A\partial_z^2 \boldsymbol{w}(z,t) - B(z,t)\partial_z \boldsymbol{w}(z,t)\right)\mathrm{d}z\\ &+ \int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,\mathrm{d}z\,.\end{aligned} \tag{5.59}$$

At this point the matrix B is decomposed into its symmetric and antisymmetric part, i.e., $B(z,t) = B^{\mathrm{s}}(z,t) + B^{\mathrm{a}}(z,t)$. Then partially integrating the expressions which include the parameter matrix A and the symmetric part B^{s} of B gives

$$\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)A\partial_z^2 \boldsymbol{w}(z,t)\,\mathrm{d}z = \left[\boldsymbol{w}^T(z,t)A\partial_z \boldsymbol{w}(z,t)\right]_0^{\ell_{\mathrm{N}}} - \int_0^{\ell_{\mathrm{N}}} \partial_z \boldsymbol{w}^T(z,t)A\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z\,, \tag{5.60}$$

$$\begin{aligned}\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)B^{\mathrm{s}}(z,t)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z &= \frac{1}{2}\left[\boldsymbol{w}^T(z,t)B^{\mathrm{s}}(z,t)\boldsymbol{w}(z,t)\right]_0^{\ell_{\mathrm{N}}}\\ &- \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\partial_z B^{\mathrm{s}}(z,t)\boldsymbol{w}(z,t)\,\mathrm{d}z\,.\end{aligned} \tag{5.61}$$

Moreover, the term involving the antisymmetric part B^{a} of B can be bounded by making use of the Young's inequality (C.35) as

$$\begin{aligned}\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)B^{\mathrm{a}}(z,t)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z \le \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} &\partial_z \boldsymbol{w}^T(z,t)P\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z\\ &- \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)B^{\mathrm{a}}(z,t)P^{-1}B^{\mathrm{a}}(z,t)\boldsymbol{w}(z,t)\,\mathrm{d}z\,.\end{aligned} \tag{5.62}$$

with $(B^{\mathrm{a}})^T = -B^{\mathrm{a}}$ and $P = \mathrm{diag}\left[r^{11}, r^{22}, \ldots, r^{nn}\right] > 0$. Further conditions on P are developed appropriately in the upcoming paragraphs. This and taking Dirichlet boundaries in (5.60) and (5.61) into account modifies the expression to

$$\begin{aligned}\dot{V}_1(t) \le -\int_0^{\ell_{\mathrm{N}}} \partial_z \boldsymbol{w}^T(z,t)\left(A - \frac{1}{2}P\right)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z &- \int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\bigg(D(t) - M(z,t)\\ &- \frac{1}{2}\partial_z B^{\mathrm{s}}(z,t) + \frac{1}{2}B^{\mathrm{a}}(z,t)P^{-1}B^{\mathrm{a}}(z,t)\bigg)\boldsymbol{w}(z,t)\mathrm{d}z\\ + \int_0^{\ell_{\mathrm{N}}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,\mathrm{d}z\,.&\end{aligned} \tag{5.63}$$

At last, the term which involves the integral kernel H can be estimated by making use of the Cauchy-Schwarz inequality (C.38) to obtain

$$\begin{aligned}\int_0^{\ell_\mathrm{N}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z &\le \left|\int_0^{\ell_\mathrm{N}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z\right| \\ &\le \left(\int_0^{\ell_\mathrm{N}}\!\!\int_0^z \tilde{\boldsymbol{x}}_\mathrm{c}(s,t)^T H^T(z,s,t)\,\mathrm{d}s\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z\right)^{\frac{1}{2}}\times \\ &\qquad\qquad \left(\int_0^{\ell_\mathrm{N}} \boldsymbol{w}^T(z,t)\boldsymbol{w}(z,t)\,\mathrm{d}z\right)^{\frac{1}{2}}.\end{aligned} \tag{5.64}$$

From this, and making use of the abbreviation $\boldsymbol{y}(z,s,t) = H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)$ the first factor in (5.64) can be further estimated by

$$\begin{aligned}\int_0^z \tilde{\boldsymbol{x}}_\mathrm{c}(s,t)^T H^T(z,s,t)\,\mathrm{d}s\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s &= \int_0^z \boldsymbol{y}^T(z,s,t)\,\mathrm{d}s\int_0^z \boldsymbol{y}(z,s,t)\,\mathrm{d}s \\ &= \begin{bmatrix}\int_0^z y^1(z,s,t)\,\mathrm{d}s & \cdots & \int_0^z y^n(z,s,t)\,\mathrm{d}s\end{bmatrix}\begin{bmatrix}\int_0^z y^1(z,s,t)\,\mathrm{d}s \\ \vdots \\ \int_0^z y^n(z,s,t)\,\mathrm{d}s\end{bmatrix} \\ &= \Sigma_{j=1}^n\left(\int_0^z y^j(z,s,t)\,\mathrm{d}s\right)^2 \le \Sigma_{j=1}^n\int_0^z |1|^2\,\mathrm{d}s\int_0^z \left|y^j(z,s,t)\right|^2\mathrm{d}s \\ &= z\int_0^z \Sigma_{j=1}^n\left|y^j(z,s,t)\right|^2\mathrm{d}s = z\int_0^z \boldsymbol{y}^T(z,s,t)\boldsymbol{y}(z,s,t)\,\mathrm{d}s \\ &= z\int_0^z \tilde{\boldsymbol{x}}_\mathrm{c}^T(s,t)H^T(z,s,t)H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s \le z\int_0^z \|H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\|_2^2\,\mathrm{d}s \\ &\le \underbrace{\max_{\mathfrak{D}_\mathrm{K}(\ell_\mathrm{N})}\left\|H(z,s,t)\right\|_2^2}_{(\varphi_\mathrm{H}(t))^2} z\int_0^z \|\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\|_2^2\,\mathrm{d}s \le (\varphi_\mathrm{H}(t))^2 z\left\|\tilde{\boldsymbol{x}}_\mathrm{c}\right\|_{\mathbf{L}^2}^2(t)\,.\end{aligned} \tag{5.65}$$

Consequently, considering (5.65) in (5.64) leads to

$$\begin{aligned}\int_0^{\ell_\mathrm{N}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z &\le \varphi_\mathrm{H}(t)\left\|\tilde{\boldsymbol{x}}_\mathrm{c}\right\|_{\mathbf{L}^2}(t)\left\|\boldsymbol{w}\right\|_{\mathbf{L}^2}(t)\left(\int_0^{\ell_\mathrm{N}} z\,\mathrm{d}z\right)^{\frac{1}{2}} \\ &\le \varphi_\mathrm{H}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\left\|\tilde{\boldsymbol{x}}_\mathrm{c}\right\|_{\mathbf{L}^2}(t)\left\|\boldsymbol{w}\right\|_{\mathbf{L}^2}(t)\,.\end{aligned} \tag{5.66}$$

Furthermore, assuming bounded backstepping kernels and applying the Minkowski inequality to (5.7), then $\left\|\tilde{\boldsymbol{x}}_\mathrm{c}\right\|_{\mathbf{L}^2}(t)$ can be estimated by

$$\left\|\tilde{\boldsymbol{x}}_\mathrm{c}(z,t)\right\|_{\mathbf{L}^2} \le \left\|\boldsymbol{w}(z,t)\right\|_{\mathbf{L}^2} + \left\|\int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\right\|_{\mathbf{L}^2}. \tag{5.67}$$

Here, the latter term can be further bounded by making use of the Cauchy-Schwarz inequality and applying the bound (5.65) again, i.e.,

$$\begin{aligned}
&\left\|\int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\right\|_{\mathbf{L}^2}\\
&\qquad= \left(\int_0^{\ell_\mathrm{N}}\!\!\int_0^z \boldsymbol{w}^T(s,t)G^T(z,s,t)\,\mathrm{d}s\int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\,\mathrm{d}z\right)^{\frac{1}{2}}\\
&\qquad\le \underbrace{\max\nolimits_{\mathfrak{D}_\mathrm{K}(\ell_\mathrm{N})}\left\|G(z,s,t)\right\|_2}_{\varphi_\mathrm{G}(t)}\left(\int_0^{\ell_\mathrm{N}}\!\!\int_0^z \boldsymbol{w}^T(s,t)\,\mathrm{d}s\int_0^z \boldsymbol{w}(s,t)\,\mathrm{d}s\,\mathrm{d}z\right)^{\frac{1}{2}} \qquad (5.68)\\
&\qquad\le \varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\left\|\boldsymbol{w}\right\|_{\mathbf{L}^2}(t)\,.
\end{aligned}$$

With this the inequality $\left\|\tilde{\boldsymbol{x}}_\mathrm{c}\right\|_{\mathbf{L}^2}(t) \le \left(1+\varphi_\mathrm{G}(t)\ell_\mathrm{N}/\sqrt{2}\right)\left\|\boldsymbol{w}\right\|_{\mathbf{L}^2}(t)$ holds true. Considering this, the upper bound (5.66) can be finally written as

$$\int_0^{\ell_\mathrm{N}} \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z \le \varphi_\mathrm{H}(t)\left(\frac{\ell_\mathrm{N}}{\sqrt{2}}+\varphi_\mathrm{G}(t)\frac{{\ell_\mathrm{N}}^2}{2}\right)\left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t)\,. \tag{5.69}$$

Inserting (5.69) in (5.63) $\dot{V}$ can be further estimated by

$$\begin{aligned}
\dot{V}_1(t) \le -\int_0^{\ell_\mathrm{N}}\partial_z\boldsymbol{w}^T(z,t)\left(A-\frac{1}{2}P\right)\partial_z\boldsymbol{w}(z,t)\,\mathrm{d}z - \int_0^{\ell_\mathrm{N}}\boldsymbol{w}^T(z,t)\Big(D(t)-M(z,t)&\\
-\frac{1}{2}\partial_z B^\mathrm{s}(z,t)+\frac{1}{2}B^\mathrm{a}(z,t)P^{-1}B^\mathrm{a}(z,t)\Big)\boldsymbol{w}(z,t)\mathrm{d}z& \qquad (5.70)\\
+\varphi_\mathrm{H}(t)\left(\frac{\ell_\mathrm{N}}{\sqrt{2}}+\varphi_\mathrm{G}(t)\frac{{\ell_\mathrm{N}}^2}{2}\right)\left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t)\,.&
\end{aligned}$$

With this, it is clear that requirements on the parameter matrix P and the matrix valued functions D have to be derived in order to ensure exponential stability of the target system (5.12a) with Dirichlet BCs. Obviously P has to fulfil

$$\boldsymbol{\nu}^T\left(A-\frac{1}{2}P\right)\boldsymbol{\nu}\ge 0 \qquad \to \qquad \boldsymbol{\nu}^T P\boldsymbol{\nu}\le 2\boldsymbol{\nu}^T A\boldsymbol{\nu} \tag{5.71}$$

$\boldsymbol{\nu}\in\mathbb{R}^n/\{\mathbf{0}_n\}$. Obviously (5.71) is fulfilled by $P=2A$ which is always applicable since the diffusion matrix A is preliminarily assumed to be positive definite. Since $\boldsymbol{\nu}^T M(z,t)\boldsymbol{\nu}=\boldsymbol{\nu}^T M^\mathrm{s}(z,t)\boldsymbol{\nu}$ the last term requires that

$$\begin{aligned}
\boldsymbol{\nu}^T\left(D(t)-M^\mathrm{s}(z,t)-\frac{1}{2}\partial_z B^\mathrm{s}(z,t)+\frac{1}{4}B^\mathrm{a}(z,t)A^{-1}B^\mathrm{a}(z,t)\right)\boldsymbol{\nu}\ge& \qquad (5.72)\\
\left(\lambda_\mathrm{min}(t)+\varphi_\mathrm{H}(t)\left(\frac{\ell_\mathrm{N}}{\sqrt{2}}+\varphi_\mathrm{G}(t)\frac{{\ell_\mathrm{N}}^2}{2}\right)\right)\left\|\boldsymbol{\nu}\right\|^2_2&
\end{aligned}$$

for all $\boldsymbol{\nu}\in\mathbb{R}^n/\{\mathbf{0}_n\}$ and with $\lambda_\mathrm{min}(t)>0$ for all $t\in\mathbb{R}^+_{t_0}$. However this can not be guaranteed in general and depends on the parameter configuration of the given error system (5.5). Therefore consider the assumption (5.56) of Theorem 5.10 which implicitly fulfils the inequality (5.72), then the rate of change of the Lyapunov functional (5.58) along a solution of (5.12a) to (5.12b) respects the upper bound

$$\dot{V}_1(t) \leq -\lambda_{\min}(t)\big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}(t) \leq -2\lambda_{\min}(t)V_1(t) \tag{5.73}$$

which is equivalent to

$$V_1(t) \leq V_1(t_0)\exp\left(-2\int_{t_0}^{t}\lambda_{\min}(\tau)\mathrm{d}\tau\right) . \tag{5.74}$$

This gives a proposition for the stability properties of the target dynamics (5.12a) with BCs (5.12b). In other words, taking (5.55) the Lyapunov functional (5.58) is obviously bounded by

$$\frac{\rho^-}{2}\left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t) \leq V_1(t) \leq \frac{\rho^+}{2}\left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t)\,, \tag{5.75}$$

e.g., trivially with $\rho^+ = \rho^- = 1$. This, together with (5.74) allows to deduce the sequence of inequalities

$$\left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t) \leq 2V_1(t) \leq 2V_1(t_0)\exp\left(-2\kappa_{\mathrm{c}}(t,t_0)\right) \leq \left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t_0)\exp\left(-2\kappa_{\mathrm{c}}(t,t_0)\right) \tag{5.76}$$

which implies (5.57) of Theorem 5.10 with the setting $\rho = 1$. □

Next, exponential stability is analysed in the Sobolev space $\mathbf{H}^1$. For this consider again Dirichlet BCs and assume $\boldsymbol{w} \in \mathbf{H}^1_0\left(\Omega\left(\bar{\ell}_{\mathrm{N}},\bar{t}_0\right);\mathbb{R}^n\right) = \left\{\boldsymbol{w} \in \mathbf{H}^1\left(\Omega\left(\bar{\ell}_{\mathrm{N}},\bar{t}_0\right);\mathbb{R}^n\right) \mid \boldsymbol{w}(0,t) = \boldsymbol{w}(\ell_{\mathrm{N}},t) = \mathbf{0}_n\right\}$. By taking Lemma 3.25 into account the norm for the given space can be defined as

$$\big\|\boldsymbol{w}\big\|_{\mathbf{H}^1_0}(t) = \left(\int_0^{\ell_{\mathrm{N}}}\partial_z\boldsymbol{w}^T(z,t)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z\right)^{\frac{1}{2}} . \tag{5.77}$$

Theorem 5.11. *Let $D(t)$ be diagonal in* (5.12a)*and let $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$. If D fulfils the inequality*

$$\begin{aligned}\boldsymbol{\nu}^T D(t)\boldsymbol{\nu} \geq \boldsymbol{\nu}^T\left(M^{\mathrm{s}}(z,t) + \frac{1}{2}B(z,t)A^{-1}B^T(z,t) + \frac{1}{2}\partial_z M(z,t)A^{-1}\partial_z M^T(z,t)\right)\boldsymbol{\nu}\\ + \boldsymbol{\nu}^T\left(2{\ell_{\mathrm{N}}}^2 A + \varphi^2_{\mathrm{H}}(t)\left(1 + \varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2{\ell_{\mathrm{N}}}^4 A^{-1}\right)\boldsymbol{\nu} + \lambda_{\min}(t)\left\|\boldsymbol{\nu}\right\|^2_2\end{aligned} \tag{5.78}$$

for all $(z,t) \in \Omega(\ell_{\mathrm{N}},t_0)$ and for some $\lambda_{\min}(t) > 0$ with $t > t_0$, then the zero equilibrium of the target system dynamics (5.12a) *with boundary conditions* (5.12b) *is exponentially stable in the norm $\big\|\cdot\big\|_{\mathbf{H}^1_0}$, i.e., there exists a constant $\rho > 0$ so that the inequality*

$$\big\|\boldsymbol{w}\big\|_{\mathbf{H}^1_0}(t) \leq \rho\big\|\boldsymbol{w}\big\|_{\mathbf{H}^1_0}(t_0)\exp\left(-\kappa_{\mathrm{c}}(t,t_0)\right) \tag{5.79}$$

holds true with $\kappa_{\mathrm{c}}(t,t_0) = \int_{t_0}^{t}\lambda_{\min}(\tau)\mathrm{d}\tau$.

Again, in (5.78) the scalar time function $\varphi_{(\cdot)}(t)$ denotes the spatial maximum of the induced operator norm, e.g., $\varphi_{\mathrm{H}}(t) = \max_{\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})}\big\|H(z,s,t)\big\|_2$.

Proof. Consider the Lyapunov functional

$$V_2(t) = \frac{1}{2}\|\boldsymbol{w}\|^2_{\mathbf{H}_0^1}(t) = \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}(z,t)^T \partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z\,. \tag{5.80}$$

Then the time derivative of V_2 gives

$$\begin{aligned}\dot{V}_2(t) &= \int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}^T(z,t)\partial_t\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z = \left[\partial_z \boldsymbol{w}^T(z,t)\partial_t \boldsymbol{w}(z,t)\right]_0^{\ell_\mathrm{N}} \\ &\qquad - \int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t)\partial_t \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &= -\int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t) A \partial_z^2 \boldsymbol{w}(z,t)\,\mathrm{d}z - \int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z \\ &\qquad + \int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t)\left(B(z,t)\partial_z \boldsymbol{w}(z,t) + \left(D(t) - M(z,t)\right)\boldsymbol{w}(z,t)\right)\mathrm{d}z\end{aligned} \tag{5.81}$$

by using partial integration and taking $\partial_t \boldsymbol{w}(0,t) = \partial_t \boldsymbol{w}(\ell_\mathrm{N},t) = \mathbf{0}_n$ into account since Dirichlet conditions are considered. The latter terms require further partial integration and upper bound estimations. Therefore integrating by parts gives

$$\begin{aligned}\int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t)\left(D(t) - M(z,t)\right)\boldsymbol{w}(z,t)\,\mathrm{d}z &= \left[\partial_z \boldsymbol{w}^T(z,t)\left(D(t) - M(z,t)\right)\boldsymbol{w}(z,t)\right]_0^{\ell_\mathrm{N}} \\ &\quad - \int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}^T(z,t)\left(D(t) - M^\mathrm{s}(z,t)\right)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &\quad + \int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}^T(z,t)\partial_z M(z,t)\boldsymbol{w}(z,t)\,\mathrm{d}z\,.\end{aligned} \tag{5.82}$$

Here the term involving $\partial_z M(z,t)$ is further analysed by applying Young's and the Poincaré inequalities, i.e., (C.35) and (C.40), respectively. This yields

$$\begin{aligned}\int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}^T(z,t)\partial_z M(z,t)\boldsymbol{w}(z,t)\,\mathrm{d}z &\leq \frac{1}{2}\int_0^{\ell_\mathrm{N}} \boldsymbol{w}(z,t)^T P \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &\quad + \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}(z,t)^T \partial_z M(z,t) P^{-1} \partial_z M^T(z,t)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &\leq \int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}(z,t)^T \left(\frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t) + 2{\ell_\mathrm{N}}^2 P\right)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &\quad + \ell_\mathrm{N}\boldsymbol{w}(\ell_\mathrm{N},t)^T P \boldsymbol{w}(\ell_\mathrm{N},t)\end{aligned} \tag{5.83}$$

with $P = \operatorname{diag}\left[r^{11}, r^{22}, \ldots, r^{nn}\right] > 0$. Similarly, the term in (5.81) involving the convection part can be bounded by

$$\begin{aligned}\int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t)B(z,t)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z &\leq \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}(z,t)^T Q \partial_z^2 \boldsymbol{w}(z,t)\,\mathrm{d}z \\ &\quad + \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z \boldsymbol{w}(z,t)^T B(z,t)Q^{-1}B^T(z,t)\partial_z \boldsymbol{w}(z,t)\,\mathrm{d}z\,,\end{aligned} \tag{5.84}$$

again with diagonal $Q > 0$. This leaves the estimation of the integral part of target PIDE. Equivalently to (5.64) and (5.69) the upper bound can be derived by using the Cauchy-Schwartz inequality (C.38). Here additionally Young's (C.35) and the Poincaré inequality (C.40) have to be taken into account, so that

$$\begin{aligned}
&\int_0^{\ell_\mathrm{N}} \partial_z^2 \boldsymbol{w}^T(z,t) \int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_\mathrm{c}(s,t)\,\mathrm{d}s\,\mathrm{d}z \\
&\qquad \le \varphi_\mathrm{H}(t)\left(\frac{\ell_\mathrm{N}}{\sqrt{2}} + \varphi_\mathrm{G}(t)\frac{{\ell_\mathrm{N}}^2}{2}\right)\|\boldsymbol{w}\|_{\mathbf{L}^2}(t)\|\partial_z^2\boldsymbol{w}\|_{\mathbf{L}^2}(t) \\
&\qquad \le \varphi_\mathrm{H}^2(t)\left(1 + \varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 \frac{{\ell_\mathrm{N}}^2}{4}\int_0^{\ell_\mathrm{N}} \boldsymbol{w}(z,t)^T R^{-1}\boldsymbol{w}(z,t)\,\mathrm{d}z \\
&\qquad\quad + \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z^2\boldsymbol{w}(z,t)^T R\,\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z \\
&\qquad \le \varphi_\mathrm{H}^2(t)\left(1 + \varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 \frac{{\ell_\mathrm{N}}^3}{2}\Bigg(2\ell_\mathrm{N}\int_0^{\ell_\mathrm{N}} \partial_z\boldsymbol{w}(z,t)^T R^{-1}\partial_z\boldsymbol{w}(z,t)\,\mathrm{d}z \\
&\qquad\qquad\qquad + \boldsymbol{w}(\ell_\mathrm{N},t)^T R^{-1}\boldsymbol{w}(\ell_\mathrm{N},t)\Bigg) + \frac{1}{2}\int_0^{\ell_\mathrm{N}} \partial_z^2\boldsymbol{w}(z,t)^T R\,\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z\,.
\end{aligned} \tag{5.85}$$

with diagonal $R > 0$. In the following and for convenience reasons let us assume $R = Q = P > 0$. Then by means of the analysis from (5.82) to (5.85) and considering that the states at the boundaries vanish due to Dirichlet conditions the time derivative of the Lyapunov functional (5.80) can be be bounded by

$$\begin{aligned}
\dot{V}_2(t) \le &-\int_0^{\ell_\mathrm{N}} \partial_z^2\boldsymbol{w}^T(z,t)\,(A-P)\,\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z - \int_0^{\ell_\mathrm{N}} \partial_z\boldsymbol{w}^T(z,t)\Big(D(t) - M^\mathrm{s}(z,t) \\
&- \frac{1}{2}B(z,t)P^{-1}B^T(z,t) - \frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t) - 2{\ell_\mathrm{N}}^2 P \\
&- \varphi_\mathrm{H}^2(t)\left(1 + \varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 {\ell_\mathrm{N}}^4 P^{-1}\Big)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z\,.
\end{aligned} \tag{5.86}$$

Obviously exponential stability requires the parameter setting

$$\boldsymbol{\nu}^T\,(A-P)\,\boldsymbol{\nu} \ge 0 \qquad \rightarrow \qquad \boldsymbol{\nu}^T P\boldsymbol{\nu} \le \boldsymbol{\nu}^T A\boldsymbol{\nu} > 0 \tag{5.87}$$

for all $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$, and with this the matrix D has to fulfil the condition

$$\begin{aligned}
\boldsymbol{\nu}^T D(t)\boldsymbol{\nu} > \boldsymbol{\nu}^T &\left(M^\mathrm{s}(z,t) + \frac{1}{2}B(z,t)A^{-1}B^T(z,t) + \frac{1}{2}\partial_z M(z,t)A^{-1}\partial_z M^T(z,t)\right)\boldsymbol{\nu} \\
&+ \boldsymbol{\nu}^T\left(2{\ell_\mathrm{N}}^2 A + \varphi_\mathrm{H}^2(t)\left(1 + \varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 {\ell_\mathrm{N}}^4 A^{-1}\right)\boldsymbol{\nu}\,.
\end{aligned} \tag{5.88}$$

However this is fulfilled when the conditional inequality (5.78) is taking it into account. Consequently, considering (5.80) and (5.78) the time derivative of the Lyapunov functional $V_2(t)$ respects the upper bound

$$\dot{V}_2(t) \le -\lambda_\mathrm{min}(t)\|\boldsymbol{w}\|_{\mathbf{H}_0^1}(t) \le -2\lambda_\mathrm{min}(t)V_2(t)\,. \tag{5.89}$$

Moreover, the Lyapunov functional $V_2(t)$ can obviously be bounded by making use of the norm (5.77), c.f. (5.58) and (5.75). Then this, together with (5.89) allows to deduce the inequality

$$\|\boldsymbol{w}\|^2_{\mathbf{H}^1_0}(t) \leq \|\boldsymbol{w}\|^2_{\mathbf{H}^1_0}(t_0) \exp\left(-2\int_{t_0}^{t} \lambda_{\min}(\tau)\mathrm{d}\tau\right) \tag{5.90}$$

which proves the exponential stability in the Sobolev space $\mathbf{H}^1$ for Dirichlet conditions when (5.78) applies. □

Corollary 5.2. *Assume inequality* (5.78) *is fulfilled, then the zero equilibrium of the target system dynamics* (5.12a) *is point-wise exponentially stable in the* max *−norm* $\|\cdot\|_\infty$ *, i.e., there exists a constant* $\sigma > 0$ *so that the inequality*

$$\max_{z\in(0,\ell_\mathrm{N})} |\boldsymbol{w}|(t) \leq \sigma \|\boldsymbol{w}\|_{\mathbf{H}^1_0}(t_0) \exp\left(-\kappa_\mathrm{c}(t,t_0)\right) \tag{5.91}$$

holds true with $\kappa_\mathrm{c}(t,t_0) = \int_{t_0}^{t} \lambda_{\min}(\tau)\mathrm{d}\tau$ *and* $\|\boldsymbol{f}\|^2_{\mathbf{H}^1_0} = \left\|\partial_z \boldsymbol{f}\right\|^2_{\mathbf{L}^2}$ *for* $\boldsymbol{f} \in \mathbf{H}^1_0\left(\Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right); \mathbb{R}^n\right)$.

Proof. Making use of Young's inequality (C.35), Agmon's inequality (C.43), and the Poincaré inequality (C.39) and taking the verifications of Theorem 5.11 into account the following estimation holds true

$$\begin{aligned}\max_{z\in(0,\ell_\mathrm{N})} |\boldsymbol{w}(z,t)|^2 &\leq \|\boldsymbol{w}(0)\|^2_2(t) + 2\left\|\boldsymbol{w}\right\|_{\mathbf{L}^2}(t)\left\|\partial_z\boldsymbol{w}\right\|_{\mathbf{L}^2}(t) \leq \left\|\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t) + \left\|\partial_z\boldsymbol{w}\right\|^2_{\mathbf{L}^2}(t)\\ &\leq \left(1+4{\ell_\mathrm{N}}^2\right)\|\boldsymbol{w}\|^2_{\mathbf{H}^1_0}(t) \leq \left(1+4{\ell_\mathrm{N}}^2\right)\|\boldsymbol{w}\|^2_{\mathbf{H}^1_0}(t_0)\exp\left(-2\kappa_\mathrm{c}(t,t_0)\right).\end{aligned} \tag{5.92}$$

With this, obviously inequality (5.91) is satisfied with $\sigma = \sqrt{1+4{\ell_\mathrm{N}}^2}$. □

Stability of the Target System for Dynamic Boundary Conditions

The stability analysis for dynamic BCs is closely related to the discussion conducted for Dirichlet BCs,i.e., $\boldsymbol{w} \in \mathbf{H}^1_0\left(\Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right); \mathbb{R}^n\right)$, covered by Theorem 5.11. The major differences are that for dynamic BCs the state vector does not vanish obligatorily at the boundary and there is a non-trivial inhomogeneity $F(z,t)$ in the PIDE of the target state. Thus, the stability analysis is performed in the space $\mathbf{X} = \mathbf{H}^1\left(\Omega\left(\bar{\ell}_\mathrm{N}, \bar{t}_0\right); \mathbb{R}^n\right)$ equipped with the inner product

$$\langle \boldsymbol{f}_1, \boldsymbol{f}_2\rangle_\mathbf{X} = \boldsymbol{f}_1(0)^T\boldsymbol{f}_2(0) + \boldsymbol{f}_1(\ell_\mathrm{N})^T\boldsymbol{f}_2(\ell_\mathrm{N}) + \langle \boldsymbol{f}_1, \boldsymbol{f}_2\rangle_{\mathbf{L}^2} + \langle \partial_z\boldsymbol{f}_1, \partial_z\boldsymbol{f}_2\rangle_{\mathbf{L}^2} \tag{5.93}$$

which induces the norm $\|\boldsymbol{f}\|_\mathbf{X} = \sqrt{\langle \boldsymbol{f}, \boldsymbol{f}\rangle} = \left(\|\boldsymbol{f}(0)\|^2_2 + \|\boldsymbol{f}(\ell_\mathrm{N})\|^2_2 + \left\|\boldsymbol{f}\right\|^2_{\mathbf{L}^2} + \left\|\partial_z\boldsymbol{f}\right\|^2_{\mathbf{L}^2}\right)^{\frac{1}{2}}$.

Theorem 5.12. *Let* $D(t)$ *be diagonal in* (5.12a) *and let* $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$*. If* D *fulfils the inequality*

$$\begin{aligned}\boldsymbol{\nu}^T D(t)\boldsymbol{\nu} \geq{}& \boldsymbol{\nu}^T\left(M^\mathrm{s}(z,t) + \frac{3}{4}\left[\partial_z M(z,t)A^{-1}\partial_z M^T(z,t) + B(z,t)A^{-1}B^T(z,t)\right]\right)\boldsymbol{\nu}\\ &+ \boldsymbol{\nu}^T\left(\varphi^2_\mathrm{H}(t)\left(1+\varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2\frac{3{\ell_\mathrm{N}}^4}{2}A^{-1} + \frac{4}{3}{\ell_\mathrm{N}}^2 A\right]\boldsymbol{\nu} + \lambda_{\min}(t)\|\boldsymbol{\nu}\|^2_2\,,\end{aligned} \tag{5.94a}$$

for $z \in (0, \ell_{\mathrm{N}})$ *and at the boundaries* $z \in \{0, \ell_{\mathrm{N}}\}$ *the condition*

$$\boldsymbol{\nu}^T D(t)\boldsymbol{\nu} > \boldsymbol{\nu}^T M^{\mathrm{s}}(z,t)\boldsymbol{\nu} + \lambda_{\min}(t)\,\|\boldsymbol{\nu}\|_2^2 \tag{5.94b}$$

holds true for some $\lambda_{\min}(t) > 0$ *and* $t > t_0$*, then the zero equilibrium of the target system dynamics* (5.12a) *with boundary conditions* (5.12c) *is exponentially stable in the norm* $\|\cdot\|_{\mathbf{X}}$ *, i.e., there exists a constant* $\sigma > 0$ *so that the inequality*

$$\|\boldsymbol{w}\|_{\mathbf{X}}(t) \le \sigma\,\|\boldsymbol{w}\|_{\mathbf{X}}(t_0)\exp\left(-\kappa_{\mathrm{c}}(t,t_0)\right) \tag{5.95}$$

holds true with $\kappa_{\mathrm{c}}(t,t_0) = \int_{t_0}^{t} \lambda_{\min}(\tau)\mathrm{d}\tau$*.*

Subsequently, note the verification of Theorem 5.12.

Proof. For the stability analysis consider the Lyapunov functional

$$V_3(t) = \frac{r}{2}\,\|\boldsymbol{w}(0)\|_2^2\,(t) + \frac{r}{2}\,\|\boldsymbol{w}(\ell_{\mathrm{N}})\|_2^2\,(t) + \frac{1}{2}\big\|\partial_z\boldsymbol{w}\big\|_{\mathbf{L}^2}^2(t) \tag{5.96}$$

with the parameter $r > 0$. With this, the time derivative of V_3 gives

$$\begin{aligned}\dot{V}_3(t) =\ & r\boldsymbol{w}^T(0,t)\partial_t\boldsymbol{w}(0,t) + r\boldsymbol{w}^T(\ell_{\mathrm{N}},t)\partial_t\boldsymbol{w}(\ell_{\mathrm{N}},t) + \int_0^{\ell_{\mathrm{N}}} \partial_z\boldsymbol{w}^T(z,t)\partial_t\partial_z\boldsymbol{w}(z,t)\,\mathrm{d}z\\ =& - r\boldsymbol{w}^T(0,t)\left(D(t) - M(0,t)\right)\boldsymbol{w}(0,t) - r\boldsymbol{w}^T(\ell_{\mathrm{N}},t)\left(D(t) - M(\ell_{\mathrm{N}},t)\right)\boldsymbol{w}(\ell_{\mathrm{N}},t)\\ &- \left[\partial_z\boldsymbol{w}^T(z,t)\partial_t\boldsymbol{w}(z,t)\right]_0^{\ell_{\mathrm{N}}} - \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\partial_t\boldsymbol{w}(z,t)\,\mathrm{d}z\\ =& - r\boldsymbol{w}^T(0,t)\left(D(t) - M(0,t)\right)\boldsymbol{w}(0,t) - r\boldsymbol{w}^T(\ell_{\mathrm{N}},t)\left(D(t) - M(\ell_{\mathrm{N}},t)\right)\boldsymbol{w}(\ell_{\mathrm{N}},t)\\ &- \left[\partial_z\boldsymbol{w}^T(z,t)\partial_t\boldsymbol{w}(z,t)\right]_0^{\ell_{\mathrm{N}}} - \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)A\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z\\ &+ \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\left(B(z,t)\partial_z\boldsymbol{w}(z,t) + \left(D(t) - M(z,t)\right)\boldsymbol{w}(z,t)\right)\mathrm{d}z\\ &+ \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z - \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\int_0^{z} H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,\mathrm{d}z\,.\end{aligned} \tag{5.97}$$

At this point the verification is taking advantage of the proof of Theorem 5.11, i.e., substituting the estimations (5.82) and (5.83) into (5.97) then the latter modifies to

$$\begin{aligned}\dot{V}_3(t) \le& -r\boldsymbol{w}^T(0,t)D_0^s(t)\boldsymbol{w}(0,t) - \boldsymbol{w}^T(\ell_{\mathrm{N}},t)\left(rD_{\ell_{\mathrm{N}}}^s(t) - \ell_{\mathrm{N}}P\right)\boldsymbol{w}(\ell_{\mathrm{N}},t)\\ &- \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)A\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z + \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)B(z,t)\partial_z\boldsymbol{w}(z,t)\,\mathrm{d}z\\ &- \int_0^{\ell_{\mathrm{N}}} \partial_z\boldsymbol{w}^T(z,t)\bigg(D(t) - M^{\mathrm{s}}(z,t) - \frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t)\\ &\qquad\qquad - 2{\ell_{\mathrm{N}}}^2 P\bigg)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z + \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z\\ &- \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\int_0^{z} H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,\mathrm{d}z\end{aligned} \tag{5.98}$$

with the abbreviations $D_0^s(t) = D(t) - M^{\mathrm{s}}(0,t)$, $D_{\ell_{\mathrm{N}}}^s(t) = D(t) - M^{\mathrm{s}}(\ell_{\mathrm{N}},t)$, and diagonal matrix $P > 0$. Equivalently, considering (5.84) as well as (5.85) leads to

$$
\begin{aligned}
\dot{V}_3(t) \le{}& -r\boldsymbol{w}^T(0,t)D_0^s(t)\boldsymbol{w}(0,t) - \boldsymbol{w}^T(\ell_{\mathrm{N}},t)\left(rD_{\ell_{\mathrm{N}}}^s(t) - \ell_{\mathrm{N}}P\right)\boldsymbol{w}(\ell_{\mathrm{N}},t)\\
&- \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\left(A - \frac{1}{2}P\right)\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z - \int_0^{\ell_{\mathrm{N}}} \partial_z\boldsymbol{w}^T(z,t)\Big(D(t) - M^{\mathrm{s}}(z,t)\\
&- 2{\ell_{\mathrm{N}}}^2P - \frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t) - \frac{1}{2}B(z,t)P^{-1}B^T(z,t)\Big)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z\\
&+ \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z - \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\,\mathrm{d}z\\
\le{}& -r\boldsymbol{w}^T(0,t)D_0^s(t)\boldsymbol{w}(0,t)\\
&- \boldsymbol{w}^T(\ell_{\mathrm{N}},t)\left(rD_{\ell_{\mathrm{N}}}^s(t) - \ell_{\mathrm{N}}P - \varphi_{\mathrm{H}}^2(t)\left(1 + \varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2\frac{{\ell_{\mathrm{N}}}^3}{2}P^{-1}\right)\boldsymbol{w}(\ell_{\mathrm{N}},t)\\
&- \int_0^{\ell_{\mathrm{N}}} \partial_z\boldsymbol{w}^T(z,t)\Big(D(t) - M^{\mathrm{s}}(z,t) - \frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t) - 2{\ell_{\mathrm{N}}}^2P\\
&\quad - \frac{1}{2}B(z,t)P^{-1}B^T(z,t) - \varphi_{\mathrm{H}}^2(t)\left(1 + \varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2{\ell_{\mathrm{N}}}^4P^{-1}\Big)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z\\
&- \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)\left(A - P\right)\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z + \int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z\,.
\end{aligned}
\tag{5.99}
$$

From this, again applying the Cauchy-Schwarz inequality (C.37) to the compensation term at the boundary $z = 0$ leads to the estimation

$$
\begin{aligned}
\int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z \le{}& \int_0^{\ell_{\mathrm{N}}} \left\|\partial_z^2\boldsymbol{w}(z,t)\right\|_2\left\|F(z,t)\boldsymbol{w}(0,t)\right\|_2\mathrm{d}z\\
\le{}& \int_0^{\ell_{\mathrm{N}}} \left(\partial_z^2\boldsymbol{w}(z,t)^T\partial_z^2\boldsymbol{w}(z,t)\right)^{\frac{1}{2}}\left(\boldsymbol{w}^T(0,t)F^T(z,t)F(z,t)\boldsymbol{w}(0,t)\right)^{\frac{1}{2}}\mathrm{d}z\,.
\end{aligned}
\tag{5.100}
$$

Since the matrix $F^T(z,t)F(z,t)$ is symmetric and positive semi-definite for every fixed $(z,t) \in \Omega\left(\bar{\ell}_{\mathrm{N}}, \bar{t}_0\right)$ making use of the Young's inequality (C.35) further leads to

$$
\begin{aligned}
\int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)F(z,t)\boldsymbol{w}(0,t)\,\mathrm{d}z \le{}& \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} \partial_z^2\boldsymbol{w}^T(z,t)P\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z\\
&+ \boldsymbol{w}^T(0,t)\Theta_P(t)\boldsymbol{w}(0,t)
\end{aligned}
\tag{5.101}
$$

with $\Theta_P(t) = \frac{1}{2}\int_0^{\ell_{\mathrm{N}}} F^T(z,t)P^{-1}F(z,t)\,\mathrm{d}z \ge 0$ and the identical weighting matrix $P > 0$ as used above. Respecting (5.101) in (5.98) $\dot V_3$ can be further estimated by

$$\begin{aligned}\dot V_3(t) \leq{}& -\boldsymbol{w}^T(0,t)\left(rD_0^s(t)-\Theta_P(t)\right)\boldsymbol{w}(0,t) - \boldsymbol{w}^T(\ell_\mathrm{N},t)\Bigg(rD^s_{\ell_\mathrm{N}}(t) - \ell_\mathrm{N} P \\ &- \varphi_\mathrm{H}^2(t)\left(1+\varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 \frac{{\ell_\mathrm{N}}^3}{2}P^{-1}\Bigg)\boldsymbol{w}(\ell_\mathrm{N},t) \\ &- \int_0^{\ell_\mathrm{N}} \partial_z\boldsymbol{w}^T(z,t)\Bigg(D(t) - M^\mathrm{s}(z,t) - 2{\ell_\mathrm{N}}^2P - \frac{1}{2}\partial_z M(z,t)P^{-1}\partial_z M^T(z,t) \\ &- \frac{1}{2}B(z,t)P^{-1}B^T(z,t) - \varphi_\mathrm{H}^2(t)\left(1+\varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2 {\ell_\mathrm{N}}^4P^{-1}\Bigg)\partial_z\boldsymbol{w}(z,t)\mathrm{d}z \\ &- \int_0^{\ell_\mathrm{N}} \partial_z^2\boldsymbol{w}^T(z,t)\left(A-\frac{3}{2}P\right)\partial_z^2\boldsymbol{w}(z,t)\,\mathrm{d}z\end{aligned} \tag{5.102}$$

With this, obviously requirements for the weighting matrix P, the parameter r, and the matrix-valued function D have to be derived in order to ensure exponential stability of the Lyapunov functional V_3. As a consequence P has to fulfil

$$\boldsymbol{\nu}^T\left(A-\frac{3}{2}P\right)\boldsymbol{\nu}\geq 0 \qquad\Longrightarrow\qquad 0<P\leq\frac{2}{3}A\,. \tag{5.103}$$

Note, (5.103) can be fulfilled certainly since the diffusion matrix A is positive definite. Moreover, setting the requirements for the parameter r to

$$\begin{aligned}&\left.\begin{aligned}&\boldsymbol{\nu}^T\left(rD_0^s(t)-\Theta_P(t)\right)\boldsymbol{\nu}\geq r\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2>0 \quad\text{and}\\ &\boldsymbol{\nu}^T\left(rD^s_{\ell_\mathrm{N}}(t)-\Psi_P(t)\right)\boldsymbol{\nu}\geq r\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2>0\end{aligned}\right.\quad\Longrightarrow \\ &r\geq r_\mathrm{min}=\max\left\{\frac{\boldsymbol{\nu}^T\Theta_P(t)\boldsymbol{\nu}}{\boldsymbol{\nu}^TD_0^s(t)\boldsymbol{\nu}-\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2},\ \frac{\boldsymbol{\nu}^T\Psi_P(t)\boldsymbol{\nu}}{\boldsymbol{\nu}^TD^s_{\ell_\mathrm{N}}(t)\boldsymbol{\nu}-\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2},\varepsilon\right\}\end{aligned} \tag{5.104}$$

for all $\boldsymbol{\nu}\in\mathbb{R}^n/\left\{\mathbf{0}_n\right\}$, $\varepsilon>0$ and with $\Psi_P(t)=\varphi_\mathrm{H}^2(t)\left(1+\varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2\frac{{\ell_\mathrm{N}}^3}{2}P^{-1}$. Clearly (5.104) demands $\boldsymbol{\nu}^TD_0^s(t)\boldsymbol{\nu}>\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2$ and $\boldsymbol{\nu}^TD^s_{\ell_\mathrm{N}}(t)\boldsymbol{\nu}>\lambda_\mathrm{min}(t)\left\|\boldsymbol{\nu}\right\|_2^2$ which is covered by the assumption (5.94b). Finally the remaining integral term in (5.102) requires that

$$\begin{aligned}\boldsymbol{\nu}^TD(t)\boldsymbol{\nu}>\boldsymbol{\nu}^T\Bigg(&M^\mathrm{s}(z,t)+2{\ell_\mathrm{N}}^2P+\frac{1}{2}\partial_zM(z,t)P^{-1}\partial_zM^T(z,t) \\ &+\frac{1}{2}B(z,t)P^{-1}B^T(z,t)+\varphi_\mathrm{H}^2(t)\left(1+\varphi_\mathrm{G}(t)\frac{\ell_\mathrm{N}}{\sqrt{2}}\right)^2{\ell_\mathrm{N}}^4P^{-1}\Bigg)\boldsymbol{\nu}\,.\end{aligned} \tag{5.105}$$

This again depends on the parameter configuration of the given error system (5.5). Therefore consider the assumption (5.94) of Theorem 5.12 which implicitly fulfils the inequality (5.103) by making use of the setting $P=2A/3$. With this the rate of change of the Lyapunov functional (5.96) along a solution of (5.12) fulfils

$$\begin{aligned}\dot V_3(t)\leq{}& -r\lambda_\mathrm{min}(t)\boldsymbol{w}^T(0,t)\boldsymbol{w}(0,t)-r\lambda_\mathrm{min}(t)\boldsymbol{w}^T(\ell_\mathrm{N},t)\boldsymbol{w}(\ell_\mathrm{N},t) \\ &-\lambda_\mathrm{min}(t)\int_0^{\ell_\mathrm{N}}\partial_z\boldsymbol{w}^T(z,t)\partial_z\boldsymbol{w}(z,t)\,\mathrm{d}z\leq-2\lambda_\mathrm{min}(t)V_3(t)\end{aligned} \tag{5.106}$$

or

$$V(t) \leq V(t_0) \exp\left(-2\int_{t_0}^{t} \lambda_{\min}(\tau)\mathrm{d}\tau\right). \tag{5.107}$$

Next, acknowledging the definition of the norm $\|\cdot\|_{\mathbf{X}}$ allows to bound the Lyapunov functional V_3 by

$$\frac{\sigma^-}{2}\|\boldsymbol{w}\|_{\mathbf{X}}^2(t) \leq V_3(t) \leq \frac{\sigma^+}{2}\|\boldsymbol{w}\|_{\mathbf{X}}^2(t) \tag{5.108}$$

with $\sigma^+ > \sigma^- > 0$. Obviously σ^+ follows directly from the definition (5.96) and can be set to, e.g. $\sigma^+ = \max\{1, r\}$. The determination of σ^- requires the splitting $\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2} = (1-q)\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2} + q\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2}$ with $1 > q > 0$ and the consideration of the Poincaré inequality (C.40) written in the form

$$\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2}^2(t) \geq \frac{1}{4{\ell_{\mathrm{N}}}^2}\|\boldsymbol{w}\|_{\mathbf{L}^2}^2(t) - \frac{1}{2\ell_{\mathrm{N}}}\|\boldsymbol{w}(0)\|_2^2(t). \tag{5.109}$$

Applying these to (5.96) gives

$$\begin{aligned}
V_3(t) &= \frac{r}{2}\|\boldsymbol{w}(0)\|_2^2(t) + \frac{r}{2}\|\boldsymbol{w}(\ell_{\mathrm{N}})\|_2^2(t) + \frac{1-q}{2}\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2}^2(t) + \frac{q}{2}\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2}^2(t)\\
&\geq \frac{1}{2}\left(r - \frac{q}{2\ell_{\mathrm{N}}}\right)\|\boldsymbol{w}(0)\|_2^2(t) + \frac{r}{2}\|\boldsymbol{w}(\ell_{\mathrm{N}})\|_2^2(t) + \frac{1-q}{2}\|\partial_z \boldsymbol{w}\|_{\mathbf{L}^2}^2(t)\\
&\quad + \frac{q}{8{\ell_{\mathrm{N}}}^2}\|\boldsymbol{w}\|_{\mathbf{L}^2}^2(t)\\
&\geq \frac{\sigma^-}{2}\|\boldsymbol{w}\|_{\mathbf{X}}(t)
\end{aligned} \tag{5.110}$$

and consequently leads to $\sigma^- = \min\left\{r - q/(2\ell_{\mathrm{N}}), r, 1-q, q/(4{\ell_{\mathrm{N}}}^2)\right\}$. With this and still respecting (5.107) combined with the analysis of the weighting matrix P and the parameter r as conducted above in (5.103) and (5.104) the sequence

$$\|\boldsymbol{w}\|_{\mathbf{X}}^2(t) \leq \frac{2}{\sigma^-}V_3(t) \leq \frac{2}{\sigma^-}V_3(t_0)\exp\left(-2\kappa_{\mathrm{c}}(t,t_0)\right) \leq \frac{\sigma^+}{\sigma^-}\|\boldsymbol{w}\|_{\mathbf{X}}^2(t_0)\exp\left(-2\kappa_{\mathrm{c}}(t,t_0)\right) \tag{5.111}$$

verifies the claim raised in Theorem 5.12 with $\sigma = \sqrt{\sigma^+/\sigma^-}$. □

The following corollary implies point-wise exponential stability of the target state $\boldsymbol{w}$ with dynamic boundary conditions.

Corollary 5.3. *Assume inequality* (5.94) *is fulfilled, then the zero equilibrium of the target system dynamics* (5.12a) *is point-wise exponentially stable in the* max *−norm* $\|\cdot\|_\infty$ *, i.e., there exists a constant* $\sigma > 0$ *so that the inequality*

$$\max_{z\in(0,\ell_{\mathrm{N}})}|\boldsymbol{w}|(t) \leq \sigma\|\boldsymbol{w}\|_{\mathbf{X}}(t_0)\exp\left(-\kappa_{\mathrm{c}}(t,t_0)\right) \tag{5.112}$$

holds true with $\kappa_{\mathrm{c}}(t,t_0) = \int_{t_0}^{t}\lambda_{\min}(\tau)\mathrm{d}\tau$ *and* $\|\boldsymbol{f}\|_{\mathbf{X}}^2 = \|\boldsymbol{f}(0)\|_2^2 + \|\boldsymbol{f}(\ell_{\mathrm{N}})\|_2^2 + \|\boldsymbol{f}\|_{\mathbf{L}^2}^2 + \|\partial_z\boldsymbol{f}\|_{\mathbf{L}^2}^2$ *for* $\boldsymbol{f} \in \mathbf{H}^1\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, \bar{t}_0\right); \mathbb{R}^n\right)$.

Proof. Making use of Young's (C.35) and Agmon's inequality (C.43), and taking the verifications of Theorem 5.12 into account the following estimation holds

$$\max_{z\in(0,\ell_\mathrm{N})} |\boldsymbol{w}(z,t)|^2 \leq \|\boldsymbol{w}(0)\|_2^2(t) + 2\big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}(t)\big\|\partial_z\boldsymbol{w}\big\|_{\mathbf{L}^2}(t) \leq \|\boldsymbol{w}(0)\|_2^2(t) \\ + \big\|\boldsymbol{w}\big\|_{\mathbf{L}^2}^2(t) + \big\|\partial_z\boldsymbol{w}\big\|_{\mathbf{L}^2}^2(t) \leq \|\boldsymbol{w}\|_{\mathbf{X}}^2(t) \leq \sigma\,\|\boldsymbol{w}\|_{\mathbf{X}}^2(t_0)\exp\left(-2\kappa_\mathrm{c}(t,t_0)\right). \quad (5.113)$$

With this, it is clear that inequality (5.112) is satisfied with σ as analysed above in the proof of Theorem 5.12. □

Remark 5.20. *Obviously Theorem 5.12 and Corollary 5.3 apply for the one dimensional case, i.e $\boldsymbol{w} = w$ and $D = d$ as well. It is well documented, e.g. in [23, 30], that for $n = 1$, the DCRS with dynamic boundary conditions is exponentially stable in the norm $\|\cdot\|_{\mathbf{X}}$ without structural limitations. More precisely the compensations terms $M(z,t)$ and $H(z,s,t)$ can be set identical to zero matrix.*

This finalises the stability analysis of the chosen target dynamics (5.12). In summary the $n-$dimensional target state $\boldsymbol{w}$ is exponentially stable in the space $\mathbf{L}^2$ and $\mathbf{H}_0^1$ for Dirichlet BCs and for dynamic BCs in space the $\mathbf{X} = \mathbf{H}^1$ with its inner product (5.93), if the associated inequalities for the tuning parameter matrix $D(t)$ with corresponding setting of the compensation terms $H(z,s,t)$ and $M(z,t)$ are satisfied. Moreover, point-wise exponentially stability can be shown for both types of BCs. From literature it is known that for the scalar case the target state w satisfies the stability properties in the $\|w\|_{\mathbf{X}}$ without any compensation terms. The convergence rate may be adjusted by the diagonal matrix-valued parameter function D or the scalar valued parameter function d respectively. From this, recalling that the backstepping kernel respect the norm(s) $\varphi_\mathrm{K}(t) = \max_{\mathfrak{D}_\mathrm{K}(\ell_\mathrm{N})}\big\|K\big\|_2(t)$ and $\varphi_\mathrm{G}(t) = \max_{\mathfrak{D}_\mathrm{K}(\ell_\mathrm{N})}\big\|G\big\|_2(t)$ let us introduce the constants $l_\mathrm{G} \leq 1+\sup_{t\in\bar{\mathbb{R}}_{t_0}^+}\varphi_\mathrm{G}(t)\ell_\mathrm{N}/\sqrt{2}$ and $l_\mathrm{K} \leq 1+\sup_{t\in\bar{\mathbb{R}}_{t_0}^+}\varphi_\mathrm{K}(t)\ell_\mathrm{N}/\sqrt{2}$, for their determination cf. (5.68). Then it is possible to deduce the following sequence of estimations

$$\|\tilde{\boldsymbol{x}}_\mathrm{c}\|_{\mathbf{L}^2}(t) \leq l_\mathrm{G}\,\|\boldsymbol{w}\|_{\mathbf{L}^2}(t) \leq l_\mathrm{G}\sigma\,\|\boldsymbol{w}\|_{\mathbf{L}^2}(t_0)\exp\left(-2\kappa_\mathrm{c}(t,t_0)\right) \\ \leq \frac{l_\mathrm{G}}{l_\mathrm{K}}\sigma\,\|\tilde{\boldsymbol{x}}_\mathrm{c}\|_{\mathbf{L}^2}(t_0)\exp\left(-2\kappa_\mathrm{c}(t,t_0)\right) \quad (5.114)$$

exemplarily for the norm $\|\cdot\|_{\mathbf{L}^2}$. Obviously the procedure can be repeated for $\mathbf{H}_0^1$ and $\mathbf{X}$ with appropriate constants l_G and l_K, respectively. Then this allows to conclude that the closed loop of the error system inherits the stability properties of the target system. As a consequence this means that the statements from Theorems 5.10 to 5.12 and Corollaries 5.2 and 5.3 can be directly applied to the control error system.

With this, the following paragraph deals with the computation of the control inputs $\Delta\boldsymbol{u}_0$ and $\Delta\boldsymbol{u}_{\ell_\mathrm{N}}$ for feedback control of the tracking error.

5.1.4 Determination of the Boundary Control Inputs

Next, the control law for $\Delta\boldsymbol{u}_0(t)$ is deduced by inserting (5.6) and, depending on the BCs, (5.12b) or (5.12c) into the first BC of either (5.5b) or (5.5c), respectively. For Dirichlet conditions the control law for $z = 0$ is trivially set to

$$\Delta \boldsymbol{u}_0(t) = \mathbf{0}_n \,. \tag{5.115}$$

and for dynamic BCs the simple algorithm

$$\Delta \boldsymbol{u}_0(t) = \partial_t \tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) = -D_0(t)\boldsymbol{w}(0,t) = -D_0(t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) \,, \tag{5.116}$$

can be derived with $D_0(t) = D(t) - M(0,t)$. This may be interpreted as impressing a first order (coupled) error dynamics at $z = 0$. The associated agent is called *the anchor* because practically the control algorithm only depends on its own control error $\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t)$ and therefore is completely independent of state of the agents. In comparison the feedback control $\Delta \boldsymbol{u}_{\ell_{\mathrm{N}}}(t)$ is determined as

$$\Delta \boldsymbol{u}_{\ell_{\mathrm{N}}}(t) = \tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) = \int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \tag{5.117}$$

by utilising (5.6) and $\boldsymbol{w}(\ell_{\mathrm{N}},t) = \mathbf{0}_n$ in case of Dirichlet conditions. For dynamic BCs the determination of $\Delta \boldsymbol{u}_{\ell_{\mathrm{N}}}(t)$ is more sophisticated since the time derivative of (5.6) has to be evaluated at $z = \ell_{\mathrm{N}}$ and the boundary does not vanish, i.e.

$$\begin{aligned}
\Delta \boldsymbol{u}_{\ell_{\mathrm{N}}}(t) &= \partial_t \tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) = \partial_t \boldsymbol{w}(\ell_{\mathrm{N}},t) + \frac{\mathrm{d}}{\mathrm{d}t}\int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\
&= -D_{\ell_{\mathrm{N}}}(t)\boldsymbol{w}(\ell_{\mathrm{N}},t) + \int_0^{\ell_{\mathrm{N}}} \partial_t K(\ell_{\mathrm{N}},s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s + \int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)\partial_t \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\
&= \int_0^{\ell_{\mathrm{N}}} \left(\partial_t K(\ell_{\mathrm{N}},s,t) + D_{\ell_{\mathrm{N}}}(t)K(\ell_{\mathrm{N}},s,t) + K(\ell_{\mathrm{N}},s,t)C(s,t)\right)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\
&\quad -D_{\ell_{\mathrm{N}}}(t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) + \int_0^{\ell_{\mathrm{N}}} \left[K(\ell_{\mathrm{N}},s,t)A\partial_s^2 \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t) - K(\ell_{\mathrm{N}},s,t)B(s,t)\partial_s \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\right]\mathrm{d}s \,.
\end{aligned} \tag{5.118}$$

Here, the partial integration of the last two terms gives

$$\begin{aligned}
\int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)A\partial_s^2 \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s &= \left[K(\ell_{\mathrm{N}},s,t)A\partial_s \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\right]_0^{\ell_{\mathrm{N}}} \\
&\quad - \left[\partial_s K(\ell_{\mathrm{N}},s,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\right]_0^{\ell_{\mathrm{N}}} + \int_0^{\ell_{\mathrm{N}}} \partial_s^2 K(\ell_{\mathrm{N}},s,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\
&= K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)A\partial_z \tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) - \partial_s K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) \\
&\quad + \partial_s K(\ell_{\mathrm{N}},0,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t) + \int_0^{\ell_{\mathrm{N}}} \partial_s^2 K(\ell_{\mathrm{N}},s,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\
\int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)B(s,t)\partial_s \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s &= K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)B(\ell_{\mathrm{N}},t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) \\
&\quad - \int_0^{\ell_{\mathrm{N}}} \partial_s K(\ell_{\mathrm{N}},s,t)B(s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^{\ell_{\mathrm{N}}} K(\ell_{\mathrm{N}},s,t)\partial_s B(s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s
\end{aligned} \tag{5.119}$$

and allows us to write the feedback law as

$$\begin{aligned}\Delta\boldsymbol{u}_{\ell_{\mathrm{N}}}(t) = &- \left[D_{\ell_{\mathrm{N}}}(t) + \partial_s K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)A + K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)B(\ell_{\mathrm{N}},t)\right]\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t)\\ &+ K(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)A\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell_{\mathrm{N}},t) + \partial_s K(\ell_{\mathrm{N}},0,t)A\tilde{\boldsymbol{x}}_{\mathrm{c}}(0,t)\\ &+ \int_0^{\ell_{\mathrm{N}}} \Big[\partial_t K(\ell_{\mathrm{N}},s,t) + \partial_s^2 K(\ell_{\mathrm{N}},s,t)A + \partial_s K(\ell_{\mathrm{N}},s,t)B(s,t)\\ &+ D_{\ell_{\mathrm{N}}}(t)K(\ell_{\mathrm{N}},s,t) + K(\ell_{\mathrm{N}},s,t)\left(C(s,t) + \partial_s B(s,t)\right)\Big]\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\mathrm{d}s\,.\end{aligned} \tag{5.120}$$

Clearly the dependency of $\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)$ in the integral over the entire spatial domain in (5.117) or (5.120), respectively, motivates the labelling *leader agent* at $z = \ell_{\mathrm{N}}$. In summary the feedback control inputs $\Delta\boldsymbol{u}_0(t)$ and $\Delta\boldsymbol{u}_{\ell_{\mathrm{N}}}(t)$ evidently depend on the dynamic behaviour of the error state $\tilde{\boldsymbol{x}}_{\mathrm{c}}$ and the integral kernel K evaluated at $z = 0$ and $z = \ell_{\mathrm{N}}$. Apparently the leader has to have the knowledge of the control error state of all agents of a MAS. However, in order to ensure independence throughout the follower agents it is not intended that the leader agent gets measurements from all other agents. This motivates the development of estimation algorithm and thus, this is topic of the next paragraphs.

5.2 Backstepping based Luenberger-type Observer for a Class of Coupled Parabolic Partial Differential Equations

In general a Luenberger-type observer consists of two parts. First, a copy of the plant, and second, one or more output injections. Therefore note the following ansatz for an observer of a coupled DCRS

$$\begin{aligned}\partial_t\hat{\boldsymbol{x}}(z,t) = &A\partial_z^2\hat{\boldsymbol{x}}(z,t) - B(z,t)\hat{\boldsymbol{x}}(z,t) + C(z,t)\hat{\boldsymbol{x}}(z,t)\\ &+ \Phi(z,t)\left(\boldsymbol{x}(\ell_{\mathrm{N}},t) - \hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t)\right) + \Psi(z,t)\left(\partial_z\boldsymbol{x}(\ell_{\mathrm{N}},t) - \partial_z\hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t)\right)\,,\end{aligned} \tag{5.121a}$$

and that of a coupled MVBE

$$\begin{aligned}\partial_t\hat{\boldsymbol{x}}(z,t) = &A\partial_z^2\hat{\boldsymbol{x}}(z,t) - B(z,t)\hat{X}(z,t)\hat{\boldsymbol{x}}(z,t) + C(z,t)\hat{\boldsymbol{x}}(z,t)\\ &+ \Phi(z,t)\left(\boldsymbol{x}(\ell_{\mathrm{N}},t) - \hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t)\right) + \Psi(z,t)\left(\partial_z\boldsymbol{x}(\ell_{\mathrm{N}},t) - \partial_z\hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t)\right)\end{aligned} \tag{5.121b}$$

both for the domain $(z,t) \in \Omega(\ell_{\mathrm{N}},t_0)$. The correction part consists of two output injections with the matrix-valued observer gains Φ and Ψ. Moreover, consider either

$$\begin{aligned}\hat{\boldsymbol{x}}(0,t) &= \boldsymbol{u}_0(t)\,,\\ \hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t) &= \boldsymbol{u}_{\ell_{\mathrm{N}}}(t)\,.\end{aligned} \tag{5.121c}$$

in case of Dirichlet BCs or in particular the two ODEs

$$\begin{aligned}\partial_t\hat{\boldsymbol{x}}(0,t) &= \boldsymbol{u}_0(t) + \Phi_0(t)\left(\boldsymbol{x}(0,t) - \hat{\boldsymbol{x}}(0,t)\right)\,,\\ \partial_t\hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t) &= \boldsymbol{u}_{\ell_{\mathrm{N}}}(t) + \Phi_{\ell_{\mathrm{N}}}(t)\left(\boldsymbol{x}(\ell_{\mathrm{N}},t) - \hat{\boldsymbol{x}}(\ell_{\mathrm{N}},t)\right)\,.\end{aligned} \tag{5.121d}$$

for the dynamic boundary conditions at $z = 0$ and $z = \ell_{\mathrm{N}}$, respectively. A proper initial condition may be given by $\hat{\boldsymbol{x}}(z, t_0) = \hat{\boldsymbol{x}}_0(z)$. The boundary control concept introduced in Section 5.1 requires state information along the entire spatial domain. Therefore it is necessary to establish an estimation algorithm which allows to reproduce the state information from boundary measurements only. Thus, the next paragraphs deal with the determination of an appropriate error system and the observer design process based on the backstepping technique.

5.2.1 Backstepping based Observer Design with Boundary Output Injection

Generally the backstepping based design process for tracking control and for a Luenberger-type observer are very similar. Therefore let us introduce observer error state

$$\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) = \boldsymbol{x}(z,t) - \hat{\boldsymbol{x}}(z,t)\,. \tag{5.122}$$

With this, obviously the error system of the DCRS can be written as

$$\begin{aligned}\partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) &= A\partial_z^2 \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) - B(z,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) + C(z,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t)\\ &\quad - \Phi(z,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t) - \Psi(z,t)\partial_z \tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t)\end{aligned} \tag{5.123a}$$

which is defined on the domain $(z,t) \in \Omega(\ell_{\mathrm{N}}, t_0)$. The boundaries are either formulated as Dirichlet BCs

$$\begin{aligned}\tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) &= \mathbf{0}_n\,,\\ \tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t) &= \mathbf{0}_n\,,\end{aligned} \tag{5.123b}$$

or as dynamic BCs

$$\begin{aligned}\partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) &= -\Phi_0(t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t)\,,\\ \partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t) &= -\Phi_{\ell_{\mathrm{N}}}(t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t)\,,\end{aligned} \tag{5.123c}$$

and the initial condition is set to $\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t_0) = \tilde{\boldsymbol{x}}_{\mathrm{o},0}(z)$. From this the focus shall be on the linearisation of the error system of the MVBE which leads to

$$\begin{aligned}\partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) &= A\partial_z^2 \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) - B(z,t)\hat{X}(z,t)\partial_z \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t)\\ &\quad + \Big(C(z,t) - B(z,t)\partial_z \hat{X}(z,t)\Big)\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) - \Phi(z,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t) - \Psi(z,t)\partial_z \tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t)\,.\end{aligned} \tag{5.124}$$

Obviously the convection and reaction term in (5.124) depend on the currently estimated state $\hat{\boldsymbol{x}}$ and its spatial derivative $\partial_z \hat{\boldsymbol{x}}$. Since observer state values cannot be pre-determined the estimation algorithm requires cyclic update methods to compute the time-variant observer gains. This is one of the major distinction compared with the tracking control design approach. For the latter the linearised terms depend on the desired state $\boldsymbol{x}^*$ and its spatial derivative $\partial_z \boldsymbol{x}^*$, which can be clearly computed in advance (cf. the linearised control error system (5.4)).

Remark 5.21. *Instead of estimating the state value $\boldsymbol{x}$ it is possible to design an observer for the tracking control error $\tilde{\boldsymbol{x}}_{\mathrm{c}} = \boldsymbol{x} - \boldsymbol{x}^*$. With this obviously the state value can be observed by $\hat{\boldsymbol{x}} = \boldsymbol{x}^* + \hat{\mathbf{e}}$, where $\hat{\mathbf{e}}$ stands for the estimation of the tracking control error. Even though this approach is not further pursued in this work it can be easily demonstrated that this concept would avoid additional computational costs for an observer implementation of the problem PDE* (5.1b)*.*

Apart from this and keeping this computational burden in mind, again, it is sufficient to consider the linear, time-variant parabolic system (5.123a), either with Dirichlet BCs (5.123b) or dynamic BCs (5.123c), respectively. From a structural point of view, i.e., the determination of the matrix-valued observer gains Φ and Ψ, this covers both, the observer error dynamics of a DCRS as well as of the linearised error system of the MVBE. Here, the backstepping transformation, again, a Volterra integral transformation of the second kind,

$$\tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) = \boldsymbol{v}(z,t) - \int_z^{\ell_{\mathrm{N}}} L(z,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s \tag{5.125}$$

and its inverse transformation

$$\boldsymbol{v}(z,t) = \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) + \int_z^{\ell_{\mathrm{N}}} S(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\,\mathrm{d}s\,, \tag{5.126}$$

shall dynamically connect the error state $\tilde{\boldsymbol{x}}_{\mathrm{o}}$ and the target state $\boldsymbol{v}$. The backstepping kernels L and S are matrix-valued $\mathbf{C}^2-$ functions in z and s. They are defined on the triangular spatial domain $(z,s) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}}) := \{(z,s) \in \mathbb{R}^2 \,|\, z \in [0,\ell_{\mathrm{N}}]\,,\, s \in [z,\ell_{\mathrm{N}}]\}$ which is illustrated in Figure 5.2a.

Remark 5.22. *There is a linkage between the backstepping kernels L and S. More precisely and equivalently to* (5.10) *and* (5.11) *the kernels are connected via*

$$S(z,s,t) - L(z,s,t) = \int_s^z S(z,p,t)L(p,s,t)\,\mathrm{d}p\,, \tag{5.127}$$

or

$$L(z,s,t) - S(z,s,t) = \int_s^z L(z,p,t)S(p,s,t)\,\mathrm{d}p\,. \tag{5.128}$$

Let us assume that the dynamics of the target state $\boldsymbol{v}$ can be written as the PIDE

$$\begin{aligned}\partial_t\boldsymbol{v}(z,t) = A\partial_z^2\boldsymbol{v}(z,t) - B(z,t)\partial_z\boldsymbol{v}(z,t) - \left(E(t) - N(z,t)\right)\boldsymbol{v}(z,t)\\ - \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\,\mathrm{d}s,\end{aligned} \tag{5.129a}$$

with either Dirichlet conditions

$$\begin{aligned}&\boldsymbol{v}(0,t) = \mathbf{0}_n\,, \quad \text{and}\\ &\boldsymbol{v}(\ell_{\mathrm{N}},t) = \mathbf{0}_n\,,\end{aligned} \tag{5.129b}$$

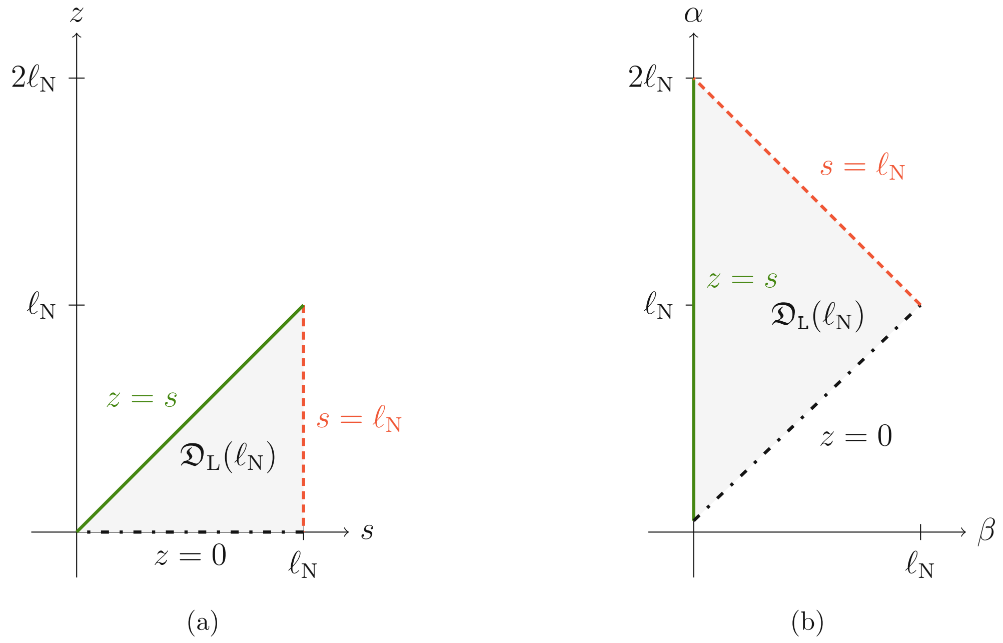


Figure 5.2: The triangular spatial domains of the backstepping kernel-PDE for the Luenberger-type observer: (a) in the $(z, s)-$plane for L and (b) in the scattering coordinates (α, β) for the transformed kernel $\mathtt{L}$. The size of the triangles varies with the length parameter $\ell_{\rm N}$ and therefore with the number of agents N.

or dynamic BCs

$$\begin{aligned} \partial_t \boldsymbol{v}(0,t) &= -(E(t) - N(0,t))(t)\boldsymbol{v}(0,t)\,, \quad \text{and} \\ \partial_t \boldsymbol{v}(\ell_{\rm N},t) &= -(E(t) - N(\ell_{\rm N},t))\boldsymbol{v}(\ell_{\rm N},t)\,, \end{aligned} \tag{5.129c}$$

respectively, and the initial condition $\boldsymbol{v}(z, t_0) = \boldsymbol{v}_0(z)$. In (5.129a) clearly the matrices A and B can be directly read from the error system (5.123a). The matrix E shall be diagonal, be positive definite for $t > t_0$, and will have to meet specific requirements to ensure stability of the target system. Furthermore, the matrix-valued function N and the integral kernel Q are introduced for compensation purposes. A thorough specification and characterisation of the design and compensation parameters is topic of the upcoming paragraphs starting with the determination of the equations for the backstepping kernel L.

5.2.2 Determination of the Backstepping Integral Kernel and the Observer Gains

As mentioned above the construction of a backstepping based observer demands the explicit knowledge of the integral kernel L. Similar to Section 5.1.2 two specific cases are analysed which support straightforward solutions of the arising kernel-PDE. For this, let us substitute the derivations in space and time of (5.125), written as

$$\partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) = \partial_t \boldsymbol{v}(z,t) - \int_z^{\ell_{\mathrm{N}}} \partial_t L(z,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s - \int_z^{\ell_{\mathrm{N}}} L(z,s,t)\partial_t \boldsymbol{v}(s,t)\,\mathrm{d}s\,, \tag{5.130}$$

$$\partial_z \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) = \partial_z \boldsymbol{v}(z,t) + L(z,z,t)\boldsymbol{v}(z,t) - \int_z^{\ell_{\mathrm{N}}} \partial_z L(z,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s\,, \tag{5.131}$$

$$\begin{aligned}\partial_z^2 \tilde{\boldsymbol{x}}_{\mathrm{o}}(z,t) &= \partial_z^2 \boldsymbol{v}(z,t) + \frac{\mathrm{d}}{\mathrm{d}z}L(z,z,t)\boldsymbol{v} + L(z,z,t)\partial_z \boldsymbol{v}(z,t) \\ &\quad + \partial_z L(z,z,t)\boldsymbol{v}(z,t) - \int_z^{\ell_{\mathrm{N}}} \partial_z^2 L(z,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s\,,\end{aligned} \tag{5.132}$$

into the observer error system (5.123a). After combining some terms this gives

$$\begin{aligned}&-\int_z^{\ell_{\mathrm{N}}} \partial_t L(z,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s - \int_z^{\ell_{\mathrm{N}}} L(z,s,t)\partial_t \boldsymbol{v}(s,t)\,\mathrm{d}s = AL(z,z,t)\partial_z \boldsymbol{v}(z,t) \\ &\quad + \left[A\frac{\mathrm{d}}{\mathrm{d}z}L(z,z,t) + A\partial_z L(z,z,t) - B(z,t)L(z,z,t) + E(t) - N(z,t)\right. \\ &\quad \left. + C(z,t)\right]\boldsymbol{v}(z,t) - \Big[\Phi(z,t) + \Psi(z,t)L(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)\Big]\boldsymbol{v}(\ell_{\mathrm{N}},t) - \Psi(z,t)\partial_z \boldsymbol{v}(\ell_{\mathrm{N}},t) \\ &\quad + \int_z^{\ell_{\mathrm{N}}} \Big[-A\partial_z^2 L(z,s,t) + B(z,t)\partial_z L(z,s,t) - C(z,t)L(z,s,t)\Big]\boldsymbol{v}(s,t)\mathrm{d}s \\ &\quad + \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s\,.\end{aligned} \tag{5.133}$$

Inserting the PIDE (5.129a) on the left hand side of equation (5.133) and subsequently summarising all terms on the right hand side results in

$$\begin{aligned}\boldsymbol{0} &= \int_z^{\ell_{\mathrm{N}}} L(z,s,t)\left[A\partial_s^2 \boldsymbol{v}(s,t) - B(z,t)\partial_s \boldsymbol{v}(s,t)\right]\mathrm{d}s + AL(z,z,t)\partial_z \boldsymbol{v}(z,t) \\ &\quad + \left[A\frac{\mathrm{d}}{\mathrm{d}z}L(z,z,t) + A\partial_z L(z,z,t) - B(z,t)L(z,z,t) + E(t) - N(z,t)\right. \\ &\quad \left. + C(z,t)\right]\boldsymbol{v}(z,t) - \Big[\Phi(z,t) + \Psi(z,t)L(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t)\Big]\boldsymbol{v}(\ell_{\mathrm{N}},t) - \Psi(z,t)\partial_z \boldsymbol{v}(\ell_{\mathrm{N}},t) \\ &\quad + \int_z^{\ell_{\mathrm{N}}} \Big[\partial_t L(z,s,t) - A\partial_z^2 L(z,s,t) + B(z,t)\partial_z L(z,s,t) - C(z,t)L(z,s,t) \\ &\quad - L(z,s,t)\left(E(t) - N(s,t)\right)\Big]\boldsymbol{v}(s,t)\mathrm{d}s + \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s \\ &\quad - \int_z^{\ell_{\mathrm{N}}}\int_s^{\ell_{\mathrm{N}}} L(z,s,t)Q(s,p,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(p,t)\mathrm{d}p\,\mathrm{d}s\,.\end{aligned} \tag{5.134}$$

Furthermore, integration by parts of the diffusion and convection terms leads to

$$\begin{aligned}\int_z^{\ell_{\mathrm{N}}} L(z,s,t)A\partial_s^2 \boldsymbol{v}(s,t)\,\mathrm{d}s &= \left[L(z,s,t)A\partial_s \boldsymbol{v}(s,t)\right]_{s=z}^{s=\ell_{\mathrm{N}}} \\ &\quad - \left[\partial_s L(z,s,t)A\boldsymbol{v}(s,t)\right]_{s=z}^{s=\ell_{\mathrm{N}}} + \int_z^{\ell_{\mathrm{N}}} \partial_s^2 L(z,s,t)A\boldsymbol{v}(s,t)\,\mathrm{d}s\,,\end{aligned} \tag{5.135}$$

as well as

$$\begin{aligned}\int_z^{\ell_{\mathrm{N}}} L(z,s,t)B(s,t)\partial_s \boldsymbol{v}(s,t)\,\mathrm{d}s &= \left[L(z,s,t)B(s,t)\boldsymbol{v}(s,t)\right]_{s=z}^{s=\ell_{\mathrm{N}}} \\ &\quad - \int_z^{\ell_{\mathrm{N}}} \left[\partial_s L(z,s,t)B(s,t) + L(z,s,t)\partial_s B(s,t)\right]\boldsymbol{v}(s,t)\,\mathrm{d}s\,.\end{aligned} \tag{5.136}$$

Moreover, changing the order of the double integration of the last term in (5.134) according to $\int_z^{\ell_{\mathrm{N}}}\int_s^{\ell_{\mathrm{N}}} F(s,p)\,\mathrm{d}p\,\mathrm{d}s = \int_z^{\ell_{\mathrm{N}}}\int_z^{s} F(p,s)\,\mathrm{d}p\,\mathrm{d}s = -\int_z^{\ell_{\mathrm{N}}}\int_s^{z} F(p,s)\,\mathrm{d}p\,\mathrm{d}s$ and inserting the two intermediate expressions (5.135) and (5.136) allows to rewrite the conditional equation as

$$\begin{aligned}\mathbf{0}_n = \Big[&A\frac{\mathrm{d}}{\mathrm{d}z}L(z,z,t) + A\partial_z L(z,z,t) + \partial_s L(z,z,t)A - B(z,t)L(z,z,t) \\ &+ L(z,z,t)B(z,t) + E(t) - N(z,t) + C(z,t)\Big]\boldsymbol{v}(z,t) \\ &+ \Big[L(z,\ell_{\mathrm{N}},t)A - \Psi(z,t)\Big]\partial_z\boldsymbol{v}(\ell_{\mathrm{N}},t) + \Big[AL(z,z,t) - L(z,z,t)A\Big]\partial_z\boldsymbol{v}(z,t) \\ &- \Big[\Phi(z,t) + \Psi(z,t)L(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t) + \partial_s L(z,\ell_{\mathrm{N}},t)A + L(z,\ell_{\mathrm{N}},t)B(\ell_{\mathrm{N}},t)\Big]\boldsymbol{v}(\ell_{\mathrm{N}},t) \\ &+ \int_z^{\ell_{\mathrm{N}}} \Big[\partial_t L(z,s,t) - A\partial_z^2 L(z,s,t) + \partial_s^2 L(z,s,t)A + B(z,t)\partial_z L(z,s,t) \\ &+ \partial_s L(z,s,t)B(s,t) - L(z,s,t)\left(E(t) - N(s,t) - \partial_s B(s,t)\right) \\ &- C(z,t)L(z,s,t)\Big]\boldsymbol{v}(s,t)\mathrm{d}s \\ &+ \int_z^{\ell_{\mathrm{N}}}\Big[Q(z,s,t) + \int_s^z L(z,p,t)Q(p,s,t)\mathrm{d}p\Big]\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s\,.\end{aligned} \tag{5.137}$$

At this stage the situation is similar to the derivation of the PDE for the inverse kernel G of the backstepping based tracking control design in Section D.1. In particular compare the current discussion with (D.19) and (D.20). Here, extending the integral over s with $\pm P(z,s,t)\boldsymbol{v}(s,t)$ in (5.137) one can formulate the condition

$$\int_z^{\ell_{\mathrm{N}}}\Big[Q(z,s,t) + \int_s^z L(z,p,t)Q(p,s,t)\mathrm{d}p\Big]\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s = \int_z^{\ell_{\mathrm{N}}} P(z,s,t)\boldsymbol{v}(s,t)\mathrm{d}s \tag{5.138}$$

Making use of the inverse backstepping transformation (5.126) allows to modify the condition to

$$\begin{aligned}&\int_z^{\ell_{\mathrm{N}}}\Big[Q(z,s,t) - P(z,s,t) + \int_s^z L(z,p,t)Q(p,s,t)\mathrm{d}p\Big]\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s \\ &\quad - \int_z^{\ell_{\mathrm{N}}}\int_s^{\ell_{\mathrm{N}}} P(z,s,t)S(s,p,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(p,t)\mathrm{d}p\,\mathrm{d}s = \mathbf{0}_n\,,\end{aligned} \tag{5.139}$$

and the subsequent change of the order of the double integral leads to

$$\begin{aligned}&\int_z^{\ell_{\mathrm{N}}}\Big[Q(z,s,t) - P(z,s,t) + \int_s^z L(z,p,t)Q(p,s,t)\mathrm{d}p\Big]\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s \\ &+ \int_s^z P(z,p,t)S(p,s,t)\mathrm{d}p\Big]\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\mathrm{d}s\,.\end{aligned} \tag{5.140}$$

Since (5.140) has to hold for every $z \in (0, \ell_{\mathrm{N}})$ one can derive the transversal compensation condition of the backstepping kernels for the observer equations as

$$P(z,s,t) - Q(z,s,t) = \int_s^z \left[P(z,p,t)S(p,s,t) + L(z,p,t)Q(p,s,t)\right] \mathrm{d}p\,. \tag{5.141}$$

Next, in (5.137) evaluating the terms with $\boldsymbol{v}(\ell_{\mathrm{N}},t)$ and $\partial_z \boldsymbol{v}(\ell_{\mathrm{N}},t)$ and considering that the target system has Dirichlet BCs, this leads to the explicit formulations

$$\begin{aligned} \Phi(z,t) &= 0_{n,n}\,, \\ \Psi(z,t) &= L(z,\ell_{\mathrm{N}},t)A\,, \end{aligned} \tag{5.142}$$

for the observer gains. In case of dynamic BCs these gains can be determined as

$$\Phi(z,t) = -L(z,\ell_{\mathrm{N}},t)AL(\ell_{\mathrm{N}},\ell_{\mathrm{N}},t) - \partial_s L(z,\ell_{\mathrm{N}},t)A - L(z,\ell_{\mathrm{N}},t)B(\ell_{\mathrm{N}},t)\,, \tag{5.143a}$$

$$\Psi(z,t) = L(z,\ell_{\mathrm{N}},t)A\,. \tag{5.143b}$$

Moreover, the combination of the dynamic boundary conditions (5.129c) with the evaluation of the backstepping transformation (5.125) and its derivatives (5.130) at the boundaries $z = 0$ and $z = \ell_{\mathrm{N}}$ leads to

$$\begin{aligned} \partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) + \int_0^{\ell_{\mathrm{N}}} \partial_t L(0,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s + \int_0^{\ell_{\mathrm{N}}} L(0,s,t)\partial_t \boldsymbol{v}(s,t)\,\mathrm{d}s = \\ -\left(E(t) - N(0,t)\right)\left(\tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) + \int_0^{\ell_{\mathrm{N}}} L(0,s,t)\boldsymbol{v}(s,t)\,\mathrm{d}s\right)\,, \end{aligned} \tag{5.144}$$

$$\partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t) = -\left(E(t) - N(\ell_{\mathrm{N}},t)\right)\tilde{\boldsymbol{x}}_{\mathrm{o}}(\ell_{\mathrm{N}},t)\,. \tag{5.145}$$

The first equation can be written as

$$\begin{aligned} \partial_t \tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) = -\left(E(t) - N(0,t)\right)\tilde{\boldsymbol{x}}_{\mathrm{o}}(0,t) - \int_0^{\ell_{\mathrm{N}}} L(0,s,t)\partial_t \boldsymbol{v}(s,t)\,\mathrm{d}s \\ - \int_0^{\ell_{\mathrm{N}}} \left[\partial_t L(0,s,t) + \left(E(t) - N(0,t)\right)L(0,s,t)\right]\boldsymbol{v}(s,t)\,\mathrm{d}s\,. \end{aligned} \tag{5.146}$$

Comparing (5.145) and (5.146) with the boundary conditions (5.123c) of the observer error system leads to the remaining observer gains

$$\begin{aligned} \Phi_0(t) &= E(t) - N(0,t) \\ \Phi_{\ell_{\mathrm{N}}}(t) &= E(t) - N(\ell_{\mathrm{N}},t)\,, \end{aligned}$$

and the boundary conditions for the kernel at $z = 0$

$$L(0,s,t) = \partial_t L(0,s,t) = 0_{n,n}\,. \tag{5.147}$$

With this and taking (5.137) into account one obtains the set of kernel equations

$$
\begin{aligned}
\partial_t L(z,s,t) = A\partial_z^2 L(z,s,t) - \partial_s^2 L(z,s,t)A - B(z,t)\partial_z L(z,s,t)& \\
- \partial_s L(z,s,t)B(s,t) + C(z,t)L(z,s,t)& \\
+ L(z,s,t)\left(E(t) - N(s,t) - \partial_s B(s,t)\right) - P(z,s,t)\,,&
\end{aligned}
\tag{5.148}
$$

$$
\begin{aligned}
A\frac{\mathrm{d}}{\mathrm{d}z}L(z,z,t) + A\partial_z L(z,z,t) + \partial_s L(z,z,t)A = B(z,t)L(z,z,t)& \\
- L(z,z,t)B(z,t) - E(t) + N(z,t) - C(z,t)\,,& \\
L(z,z,t)A = AL(z,z,t)\,,& \\
L(0,s,t) = 0_{n,n}\,.&
\end{aligned}
\tag{5.149}
$$

Remark 5.23. *The equations of the backstepping kernel S for the inverse transformation* (5.126) *can be derived in similar fashion as the equations of the backstepping kernel* (5.29) *for the transformation of the tracking controller in Section 5.1.2. The PDE with corresponding BCs read*

$$
\begin{aligned}
\partial_t S(z,s,t) = A\partial_z^2 S(z,s,t) - \partial_s^2 S(z,s,t)A - B(z,t)\partial_z S(z,s,t)& \\
- \partial_s S(z,s,t)B(s,t) - \left(E(t) - N(z,t)\right)S(z,s,t)& \\
- S(z,s,t)\left(C(s,t) + \partial_s B(s,t)\right) - Q(z,s,t)\,,&
\end{aligned}
\tag{5.150}
$$

$$
\begin{aligned}
A\frac{\mathrm{d}}{\mathrm{d}z}S(z,z,t) + A\partial_z S(z,z,t) + \partial_s S(z,z,t)A = B(z,t)S(z,z,t)& \\
- S(z,z,t)B(z,t) - E(t) + N(z,t) - C(z,t)\,,& \\
S(z,z,t)A = AS(z,z,t)\,,& \\
S(0,s,t) = 0_{n,n}\,.&
\end{aligned}
\tag{5.151}
$$

In the following the discussion is directed to two specific types of compensations. They allow to solve the kernel equations with methods which are comprehensively investigated in recent literature. At this stage the interested reader is referred to Section 5.1.2.

Coupling Compensation

Imposing diagonal backstepping kernel $L(z,s,t) = L^{\mathrm{d}}(z,s,t)$ over the entire spatial domain $(z,s) \in \mathfrak{D}_{\mathrm{L}}(\ell_{\mathrm{N}})$ leads to the compensation terms

$$
N(z,t) = N^{\mathrm{c}}(z,t) = L^{\mathrm{d}}(z,z,t)B(z,t) - B(z,t)L^{\mathrm{d}}(z,z,t) + C^{\mathrm{c}}(z,t)\,,
\tag{5.152}
$$

$$
\begin{aligned}
P(z,s,t) = -B^{\mathrm{c}}(z,t)\partial_z L^{\mathrm{d}}(z,s,t) - \partial_s L^{\mathrm{d}}(z,s,t)B^{\mathrm{c}}(s,t)& \\
+ C^{\mathrm{c}}(z,t)L^{\mathrm{d}}(z,s,t) - L^{\mathrm{d}}(z,s,t)\left(N^{\mathrm{c}}(s,t) + \partial_s B^{\mathrm{c}}(s,t)\right)\,.&
\end{aligned}
\tag{5.153}
$$

Similar to the coupling compensation discussion in Section 5.1.2 for the control design the uncoupled equations can be written element-wise as

$$\begin{aligned}
\partial_t l_{ii}(z,s,t) &= a_{ii}\left(\partial_z^2 l_{ii}(z,s,t) - \partial_s^2 l_{ii}(z,s,t)\right) - b_{ii}(z,t)\partial_z l_{ii}(z,s,t) \\
&\qquad - b_{ii}(s,t)\partial_s l_{ii}(z,s,t) + \left(e_{ii}(t) + c_{ii}(s,t) - \partial_s b_{ii}(s,t)\right)(z,s,t) l_{ii}(z,s,t)\,, \\
\frac{\mathrm{d}}{\mathrm{d}z} l_{ii}(z,z,t) &= -\frac{1}{2a_{ii}}\left(e_{ii}(t) + c_{ii}(z,t)\right)\,, \\
l_{ii}(z,0,t) &= 0\,.
\end{aligned} \tag{5.154}$$

Applying the Hopf-Cole-transformation and subsequent transfer into scattering coordinates allows to rewrite the PDE in a integrable form. This allows to solve each kernel element by means of recursive algorithms. A more detailed description has been dispensed since this pattern is already a standard procedure in literature, e.g. [55], and is equivalent to the remarks of the associated analysis in Section 5.1.2.

Non-commutative Compensation

Alternatively the goal of the following approach is to compensate the non-commutativity of the kernel-PDE (5.148). More precisely speaking, the setting of the compensation terms

$$N(z,t) = 2\Delta A \frac{\mathrm{d}}{\mathrm{d}z} L^{\mathrm{d}}(z,z,t) + L^{\mathrm{d}}(z,z,t)B(z,t) - B(z,t)L^{\mathrm{d}}(z,z,t) + C^{\mathrm{c}}(z,t)\,, \tag{5.155}$$

$$\begin{aligned}
P(z,s,t) &= \Delta A \partial_z^2 L(z,s,t) - \partial_s^2 L(z,s,t)\Delta A - \Delta B(z,t)\partial_z L(z,s,t) \\
&\qquad - \partial_s L(z,s,t)\Delta B(s,t) - L(z,s,t)\left(N_{AB}(s,t) + \partial_s \Delta B(s,t)\right)\,.
\end{aligned} \tag{5.156}$$

with the the abbreviations $\Delta A = A - aI_n$, $\Delta B(z,t) = B(z,t) - b(z,t)I_n$, and $N_{AB}(z,t) = N(z,t) - C^{\mathrm{c}}(z,t)$ allows to write the kernel-PDE as

$$\begin{aligned}
\partial_t L(z,s,t) &= a\left(\partial_z^2 L(z,s,t) - \partial_s^2 L(z,s,t)\right) - b(z,t)\partial_z L(z,s,t) \\
&\qquad - b(s,t)\partial_s L(z,s,t) + \left(E(t) - C^{\mathrm{c}}(z,t)\right)L(z,s,t) \\
&\qquad + L(z,s,t)\left(C(s,t) - \partial_s b(s,t)I_n\right)\,,
\end{aligned} \tag{5.157}$$

where the parameter matrix A and the matrix-valued function B are substituted by a scalar parameter a and a scalar function b. Moreover, the boundary conditions are uncoupled as stated above in (5.154). Again, a further discussion is omitted since standard procedures allow to solve the kernel equations as stated.

5.2.3 Stability of the Observer Error System

As discussed for control design in Section 5.1.3 the closed loop stability of the observer error system demands the boundedness of the backstepping kernels L and S and the stability of the target system (5.129). The former is inherently given since by referring to the the discussion above strong solutions exist for L and S. The latter requirement is topic of the next two sections.

Stability of the Target System for Dirichlet Boundary Conditions

Note the following two Theorems.

Theorem 5.13. *Let $E(t)$ be diagonal in* (5.129a) *and let $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$. If E fulfils the inequality*

$$\boldsymbol{\nu}^T E(t)\boldsymbol{\nu} \geq \boldsymbol{\nu}^T \left(N^{\mathrm{s}}(z,t) + \frac{1}{2}\partial_z B^{\mathrm{s}}(z,t) - \frac{1}{4}B^{\mathrm{a}}(z,t)A^{-1}B^{\mathrm{a}}(z,t) \right) \boldsymbol{\nu}^T \\ + \left(\varphi_{\mathrm{P}}(t) \left(\frac{\ell_{\mathrm{N}}}{\sqrt{2}} + \varphi_{\mathrm{S}}(t)\frac{{\ell_{\mathrm{N}}}^2}{2} \right) + \mu_{\min}(t) \right) \|\boldsymbol{\nu}\|_2^2 \tag{5.158}$$

for all $(z,t) \in \Omega(\ell_{\mathrm{N}}, t_0)$ and for some $\mu_{\min}(t) > 0$ with $t > t_0$, then the zero equilibrium of the target system dynamics (5.129a) *with boundary conditions* (5.129b) *is exponentially stable in the norm $\big\|\cdot\big\|_{\mathbf{L}^2}$, i.e., there exists a constant $\rho > 0$ so that the inequality*

$$\big\|\boldsymbol{v}\big\|_{\mathbf{L}^2}(t) \leq \rho\big\|\boldsymbol{v}\big\|_{\mathbf{L}^2}(t_0)\exp\left(-\kappa_{\mathrm{o}}(t,t_0)\right) \tag{5.159}$$

holds true with $\kappa_{\mathrm{o}}(t,t_0) = \int_{t_0}^{t} \mu_{\min}(\tau)\mathrm{d}\tau$.

Theorem 5.14. *Let $E(t)$ be diagonal in* (5.129a) *and let $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$. If E fulfils the inequality*

$$\boldsymbol{\nu}^T E(t)\boldsymbol{\nu} \geq \boldsymbol{\nu}^T \left(N^{\mathrm{s}}(z,t) + \frac{1}{2}B(z,t)A^{-1}B^T(z,t) + \frac{1}{2}\partial_z N(z,t)A^{-1}\partial_z N^T(z,t) \right) \boldsymbol{\nu} \\ + \boldsymbol{\nu}^T \left(2{\ell_{\mathrm{N}}}^2 A + \varphi_{\mathrm{P}}^2(t)\left(1 + \varphi_{\mathrm{S}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2 {\ell_{\mathrm{N}}}^4 A^{-1} \right) \boldsymbol{\nu} + \mu_{\min}(t)\,\|\boldsymbol{\nu}\|_2^2 \tag{5.160}$$

for all $(z,t) \in \Omega(\ell_{\mathrm{N}}, t_0)$ and for some $\mu_{\min}(t) > 0$ with $t > t_0$, then the zero equilibrium of the target system dynamics (5.129a) *with boundary conditions* (5.129b) *is exponentially stable in the norm $\big\|\cdot\big\|_{\mathbf{H}_0^1}$ as defined in* (5.77) *, i.e., there exists a constant $\rho > 0$ so that the inequality*

$$\big\|\boldsymbol{v}\big\|_{\mathbf{H}_0^1}(t) \leq \rho\big\|\boldsymbol{v}\big\|_{\mathbf{H}_0^1}(t_0)\exp\left(-\kappa_{\mathrm{o}}(t,t_0)\right) \tag{5.161}$$

holds true with $\kappa_{\mathrm{o}}(t,t_0) = \int_{t_0}^{t} \mu_{\min}(\tau)\mathrm{d}\tau$.

Obviously there is a correlation between the labelling of matrices and functions introduced for the design of the tracking controller and the Luenberger-type observer. Consider the correspondence list

Lyapunov functionals: V_i	$\longleftrightarrow$	W_i ,	Target state: $\boldsymbol{w}$	$\longleftrightarrow$	$\boldsymbol{v}$,
Parameter matrix: D	$\longleftrightarrow$	E ,	Compensation matrix: M	$\longleftrightarrow$	N ,
Compensation kernel: H	$\longleftrightarrow$	P ,	(Inverse) Compensation kernel: J	$\longleftrightarrow$	Q ,
Backstepping kernel: K	$\longleftrightarrow$	L ,	(Inverse) Backstepping kernel: G	$\longleftrightarrow$	S ,

and note the following paragraph.

Proof. Generally the verifications of Theorems 5.13 and 5.14 is shortened since the proofs are equivalent to the analysis done for Theorems 5.10 and 5.11 concerning the stability of the tracking error controller. The only difference is the upper bound estimation of the term including $\varphi_{\mathrm{P}}(t)$ and $\varphi_{\mathrm{S}}(t)$ which is performed briefly. For this, utilising the backstepping transformation (5.125) and transversal compensation condition of the observer (5.141) the term equivalent to (5.64) can be formulated as

$$\begin{aligned}
&\int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\,\mathrm{d}s\,\mathrm{d}z \\
&\quad= \int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\left[\boldsymbol{v}(s,t) - \int_s^{\ell_{\mathrm{N}}} L(z,p,t)\boldsymbol{v}(p,t)\,\mathrm{d}p\right]\mathrm{d}s\,\mathrm{d}z \\
&\quad= \int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}}\left[Q(z,s,t) + \int_s^{z} Q(z,p,t)L(p,s,t)\,\mathrm{d}p\right]\boldsymbol{v}(s,t)\,\mathrm{d}s\,\mathrm{d}z \\
&\quad= \int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}}\left[P(z,s,t) - \int_s^{z} P(z,p,t)S(p,s,t)\,\mathrm{d}p\right]\boldsymbol{v}(s,t)\,\mathrm{d}s\,\mathrm{d}z\,.
\end{aligned} \tag{5.162}$$

From this, equivalently to (5.69) the upper bound estimation holds true

$$\begin{aligned}
&\int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}} Q(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{o}}(s,t)\,\mathrm{d}s\,\mathrm{d}z \\
&\quad\leq \varphi_{\mathrm{P}}(t)\left(\frac{\ell_{\mathrm{N}}}{\sqrt{2}} + \varphi_{\mathrm{S}}(t)\frac{{\ell_{\mathrm{N}}}^2}{2}\right)) \int_0^{\ell_{\mathrm{N}}} \boldsymbol{v}^T(z,t) \int_z^{\ell_{\mathrm{N}}} \boldsymbol{v}(s,t)\,\mathrm{d}s\,\mathrm{d}z\,.
\end{aligned} \tag{5.163}$$

□

Stability of the Target System for Dynamic Boundary Conditions

Consider the following Theorem and Corollary for a stability statement with dynamic BCs in the space $\mathbf{X} = \mathbf{H}^1\left(\Omega\left(\bar{\ell}_{\mathrm{N}}, \bar{t}_0\right); \mathbb{R}^n\right)$ with the norm induced by the inner product (5.93).

Theorem 5.15. *Let* $E(t)$ *be diagonal in* (5.129a) *and let* $\boldsymbol{\nu} \in \mathbb{R}^n/\{\mathbf{0}_n\}$. *If* E *fulfils the inequality*

$$\begin{aligned}
\boldsymbol{\nu}^T E(t)\boldsymbol{\nu} \geq \boldsymbol{\nu}^T &\left(N^{\mathrm{s}}(z,t) + \frac{3}{4}\left[\partial_z N(z,t)A^{-1}\partial_z N^T(z,t) + B(z,t)A^{-1}B^T(z,t)\right]\right)\boldsymbol{\nu} \\
&+ \boldsymbol{\nu}^T\left(\varphi_{\mathrm{P}}^2(t)\left(1+\varphi_{\mathrm{S}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2\frac{3{\ell_{\mathrm{N}}}^4}{2}A^{-1} + \frac{4}{3}{\ell_{\mathrm{N}}}^2 A\right)\boldsymbol{\nu} + \mu_{\min}(t)\left\|\boldsymbol{\nu}\right\|_2^2\,,
\end{aligned} \tag{5.164a}$$

for $z \in (0, \ell_{\mathrm{N}})$ *and at the boundaries* $z \in \{0, \ell_{\mathrm{N}}\}$ *the condition*

$$\boldsymbol{\nu}^T E(t)\boldsymbol{\nu} > \boldsymbol{\nu}^T N^{\mathrm{s}}(z,t)\boldsymbol{\nu} + \mu_{\min}(t)\left\|\boldsymbol{\nu}\right\|_2^2\,, \tag{5.164b}$$

holds true for some $\lambda_{\min}(t) > 0$ *and* $t > t_0$, *then the zero equilibrium of the target system dynamics* (5.129a) *with boundary conditions* (5.129c) *is exponentially stable in the norm* $\left\|\cdot\right\|_{\mathbf{X}}$, *i.e., there exists a constant* $\sigma > 0$ *so that the inequality*

$$\left\|\boldsymbol{v}\right\|_{\mathbf{X}}(t) \leq \sigma\left\|\boldsymbol{v}\right\|_{\mathbf{X}}(t_0)\exp\left(-\kappa_{\mathrm{o}}(t,t_0)\right) \tag{5.165}$$

holds true with $\kappa_{\mathrm{o}}(t,t_0)=\int_{t_0}^{t}\mu_{\min}(\tau)\mathrm{d}\tau$ *and* $\|\boldsymbol{v}\|_{\mathbf{X}}^2(t)=\|\boldsymbol{v}(0)\|_2^2(t)+\|\boldsymbol{v}(\ell_{\mathrm{N}})\|_2^2(t)+\left\|\boldsymbol{v}\right\|_{\mathbf{L}^2}^2(t)+\left\|\partial_z\boldsymbol{v}\right\|_{\mathbf{L}^2}^2(t)$.

Corollary 5.4. *Assume inequality* (5.164) *is fulfilled, then the zero equilibrium of the target system dynamics* (5.129a) *is point-wise exponentially stable in the* max *−norm* $\|\cdot\|_\infty$ *, i.e., there exists a constant* $\sigma>0$ *so that the inequality*

$$\max_{z\in(0,\ell_{\mathrm{N}})}|\boldsymbol{v}(z,t)|\leq\sigma\left\|\boldsymbol{v}\right\|_{\mathbf{X}}(t_0)\exp\left(-\kappa_{\mathrm{o}}(t,t_0)\right)\tag{5.166}$$

holds true with $\kappa_{\mathrm{o}}(t,t_0)=\int_{t_0}^{t}\mu_{\min}(\tau)\mathrm{d}\tau$ *and for* $\boldsymbol{v}\in\mathbf{H}^1\left(\Omega\left(\bar{\ell}_{\mathrm{N}},\bar{t}_0\right);\mathbb{R}^n\right)$.

Proof. Here, the formal evidence of Theorem 5.15 and (5.4) is omitted since it is completely equivalent to the proof of Theorem 5.12 in view of the Lyapunov functional

$$W_3(t)=\frac{r}{2}\left\|\boldsymbol{v}(0)\right\|_2^2(t)+\frac{r}{2}\left\|\boldsymbol{v}(\ell_{\mathrm{N}})\right\|_2^2(t)+\frac{1}{2}\left\|\partial_z\boldsymbol{v}\right\|_{\mathbf{L}^2}^2(t)\tag{5.167}$$

with positive parameter $r>0$. The same applies to the analysis required for Corollary 5.4 which is identical to discussion already for conducted Corollary 5.3. □

Remark 5.24. *It is clear that Theorem 5.15 and Corollary 5.4 also apply for the case* $n=1$ *and consequently* $\boldsymbol{v}=v$*. However, e.g. in [23, 30], it is shown that the one dimensional DCRS with dynamic boundary conditions is exponentially stable in the norm* $\|\cdot\|_{\mathbf{X}}$ *without structural limitations. For the compensations terms this means* $N(z,t)=P(z,s,t)=0_{n,n}$.

In the following an extensive simulations study enhances the presented backstepping approach for PDEs with dynamic boundary conditions. Both, coupled and uncoupled scenarios shall demonstrate the 2DOF control concept which combines the FF developed in Chapter 4 and the observer-based tracking control part of the discussion above.

Chapter 6

Simulation Studies and Real Time Experiment

In this chapter the discussed concept for MASs and the developed control methods shall be verified by means of extensive simulations studies. On top of that the continuous modelling concept as well as the backstepping control approach in general is validated experimentally for an uncoupled formation scenario processed with a small swarm of caterpillar robots in a real time test-bed. While simulation studies of MAS based on PDE formulations are published occasionally, e.g., in [56, 30, 23, 26, 24, 91], a specific real time validation has been only performed in [25]. In the following the discussion recalls the discussion of Section 2.2 and Section 2.3 which leads to a discretisation and mapping process of the continuous model formulation and its synthesis in terms of the 2DOF control and Luenberger-type observer development. The four following simulation scenarios cover continuous model formulations of an uncoupled DRS, an uncoupled MVBE, and a coupled DRS which finally is extended to a coupled DCRS model. All four scenarios utilise dynamic boundary conditions. Last but not least the uncoupled Diffusion-Reaction case is validated in a real time experiment. With this let us start with the preparation of the simulation scenarios by semi-discretisation of the PDE formulations and the associated algorithms.

6.1 Semi-Discretisation of the Continuous Synthesis by FDM

The subsequent simulations scenarios evaluate on the one hand the PDE-based (inverse) design approach for dynamic models of MAS and on the other hand the 2DOF control concept based on a FF part utilising flatness properties and a FB part where backstepping for distributed parameter systems is the main motivator. For this, let us assume each agent is assigned with a 2−dimensional state $\boldsymbol{x} = [x^1\, x^2]^T$. In general the interpretation or physical meaning of the states is arbitrary. However let us assume that x^1 and x^2 are coordinates in the 2D plane since this is the actual meaning for the real time validation experiment which shall match the first simulation scenarios.

Next, let us recall the concept worked out in Section 2.2 and especially in Section 2.2.2. There the discussion leads to a $2-$step process in order to transform the continuous model of MAS to a corresponding discrete form which is based on Graph theoretical concepts. In summary, first, the PDE model is spatially discretised, e.g., by FDM and the discretisation step $\Delta z = \frac{\ell_{\mathrm{N}}}{\mathrm{N}-1}$. Second, the bijective function ϕ, defined in (2.52), formally maps the discretisation points z_i, $i = 1, 2 \ldots, \mathrm{N}$ to the vertices $\mathtt{v}_i$ of the underlying graph representation of the MAS. For instance the state vector $\boldsymbol{x}$ undergoes the transformation

$$\boldsymbol{x}(z,t) \quad \xRightarrow{\text{FDM}} \quad \boldsymbol{x}(z_i,t) \quad \xRightarrow{z_i=\phi^{-1}(\mathtt{v}_i)} \quad \boldsymbol{x}(\phi^{-1}(\mathtt{v}_i),t) := \boldsymbol{x}(\mathtt{v}_i,t) \tag{6.1}$$

with the abbreviation $z_i = (i-1)\Delta z$ and the function $\phi : \mathrm{Z} \rightarrow \mathtt{S}(\mathcal{G}_{\mathrm{s}})$. Here, the domain Z is the set of discretisation points, see (2.42), and the range $\mathtt{S}(\mathcal{G}_{\mathrm{s}})$ is the set of vertices of the associated underlying subgraph. As a consequence all analytical functions involved in the controller or observer design process have to be spatially discretised and mapped accordingly. The following list just represents an short excerpt, e.g.,

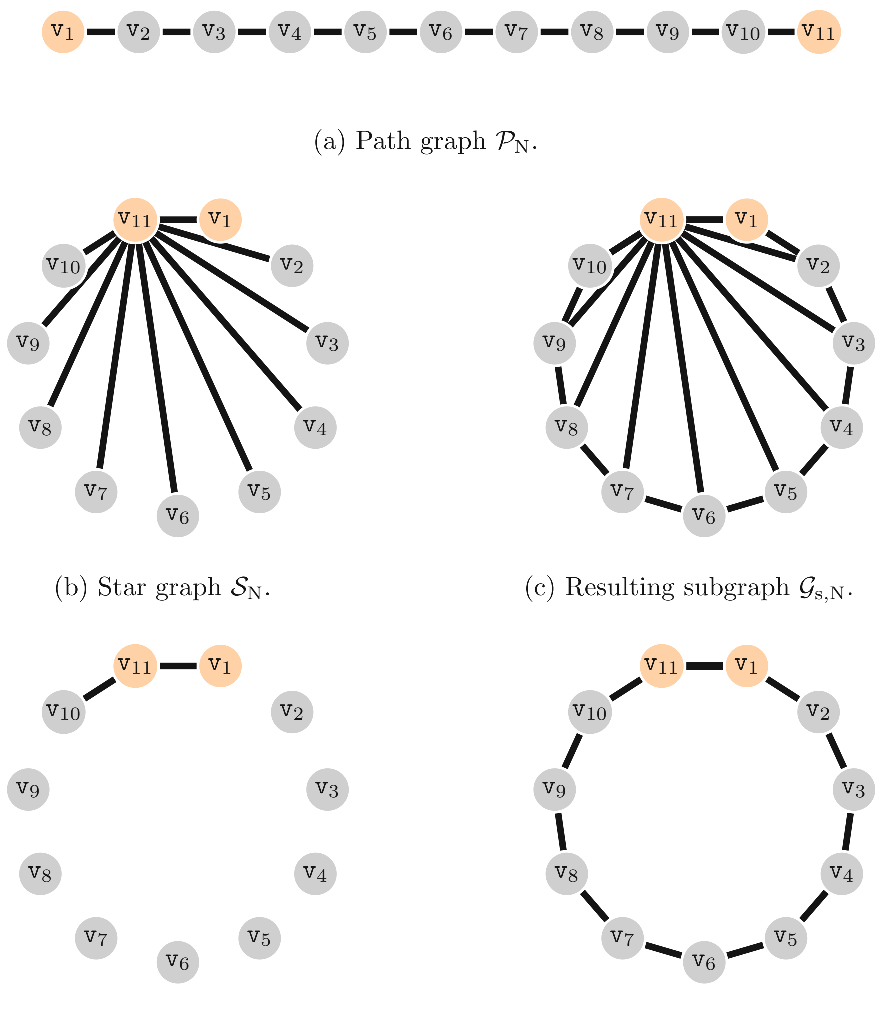
$$\begin{aligned} &B(z,t) \mapsto B(\mathtt{v}_i,t) && C(z,t) \mapsto C(\mathtt{v}_i,t) \\ &K(z,s,t) \mapsto K(\mathtt{v}_i,\mathtt{v}_j,t) && L(z,s,t) \mapsto L(\mathtt{v}_i,\mathtt{v}_j,t) \\ &\partial_s K(z,s,t) \mapsto \partial_s K(\mathtt{v}_i,\mathtt{v}_j,t) && \partial_s L(z,s,t) \mapsto \partial_s L(\mathtt{v}_i,\mathtt{v}_j,t) \end{aligned}$$

At this stage it is referred to (5.46), where $\partial_s K(z,s,t)$ is computed analytically. However, for instance the second order spatial derivative of the backstepping kernel $\partial_s^2 K(z,s,t)$ as well as the spatial derivatives of the continuous model formulations (3.17) and (3.23) are required for observer applications but are not available in an analytical way. Therefore numerical differentiation is applied, which, e.g., leads to

$$\partial_s^2 K(z,s,t) = \partial_s\left(\partial_s K(z,s,t)\right) \mapsto {}_{k}^{l}\mathcal{D}^{(1)}(\partial_s K)(\mathtt{v}_i,\mathtt{v}_j,t)$$

with ${}_{k}^{l}\mathcal{D}^{(\alpha)}$ defined by (2.76) as the discrete differential operator for central, forward and backward FDM. At this point it has to be emphasised that the determination of the actual type and order of the FDM defines and induces the mandatory network topology for the distribution of the states and consequently the necessary edge set of the underlying subgraph representation. For this, let us assume a *next neighbour* or $1-$*nearest neighbour* topology for state distribution which sets the operator ${}_{-1}^{\ 1}\mathcal{D}^{(1)}$ and ${}_{-1}^{\ 1}\mathcal{D}^{(2)}$ for all follower agents, ${}_{0}^{1}\mathcal{D}^{(1)}$ for the anchor agent, and ${}_{-1}^{\ 0}\mathcal{D}^{(1)}$ for the leader agent. In this case the process induces a path graph $\mathcal{P}_{\mathrm{N}}$ representation.

Remark 6.25. *The terminology regarding network topology explicitly refers to state distribution and is chosen on purpose. Parameter values may be time-dependent and therefore they need to be synchronised with all participating agents. This may lead to a more complex overall topology and underlying subgraph representation. Time synchronisation can be accomplished, e.g., by an external instance which sends trigger signals, or by means of another internal MAS which is*



(a) Path graph $\mathcal{P}_\mathrm{N}$.

(b) Star graph $\mathcal{S}_\mathrm{N}$.

(c) Resulting subgraph $\mathcal{G}_{\mathrm{s,N}}$.

(d) Subgraph $\mathcal{S}_{\mathrm{s,N}}$.

(e) Resulting N-cycle graph $\mathcal{C}_\mathrm{N}$.

Figure 6.1: Graph representations for state distribution of a swarm of $\mathrm{N} = 11$ agents. The path graph in (a) is induced by discretisation and mapping process. The star graph shown in (b) results from the backstepping-based controller design *without* state estimation. The overlap of these graphs leads to the graph depicted in (c). The illustrations in (d) and (e) show simplified representations when state estimation is applicable for FBC.

utilised for parameter distribution. In literature it is denoted as a so-called exogenous system*. For instance, in [26] the reaction term coefficient c^m is set up as an independent DRS which allows rather complex parameter evolution in time and space, i.e. throughout all agents. Since parameters are purely information and no physical dynamics is involved no disruption is expected apart from possible communication delays or message losses. As a consequence, when modelling the parameter distribution as a stable DRS it is sufficient to control the exogenous system in open-loop only. In this case the network topology which is induced for state distribution is preserved and consequently defines the overall topology.*

A graphical illustration for a path graph with $\mathrm{N} = 11$ is sketched in Figure 6.1a. Utilising appropriate coefficients for central FDM from Table A.1 the transformation of the PDE of a DCRS leads to

$$\begin{aligned}\partial_t \boldsymbol{x}(\mathtt{v}_i,t) &= A\left(\boldsymbol{x}(\mathtt{v}_{i+1},t) - 2\boldsymbol{x}(\mathtt{v}_i,t) + \boldsymbol{x}(\mathtt{v}_{i-1},t)\right) - \frac{B(\mathtt{v}_i,t)}{2}\left(\boldsymbol{x}(\mathtt{v}_{i+1},t) - \boldsymbol{x}(\mathtt{v}_{i-1},t)\right)\\ &\quad + C(\mathtt{v}_i,t)\boldsymbol{x}(\mathtt{v}_i,t)\,,\end{aligned} \tag{6.2}$$

and the discrete formulation of the MVBE is written as

$$\begin{aligned}\partial_t \boldsymbol{x}(\mathtt{v}_i,t) &= A\left(\boldsymbol{x}(\mathtt{v}_{i+1},t) - 2\boldsymbol{x}(\mathtt{v}_i,t) + \boldsymbol{x}(\mathtt{v}_{i-1},t)\right)\\ &\quad - \frac{B(\mathtt{v}_i,t)}{2} X(\mathtt{v}_i,t)\left(\boldsymbol{x}(\mathtt{v}_{i+1},t) - \boldsymbol{x}(\mathtt{v}_{i-1},t)\right) + C(\mathtt{v}_i,t)\boldsymbol{x}(\mathtt{v}_i,t)\,,\end{aligned} \tag{6.3}$$

with $i = \{2,\dots,\mathrm{N}-1\}$. Moreover, the IC may be obtained by $x(\mathtt{v}_i,t_0) = x_0(\mathtt{v}_i) \mapsfrom x_0(z)$. Considering dynamic BCs the transferred formulation of the anchor and leader agent,

$$\begin{aligned}\partial_t \boldsymbol{x}(\mathtt{v}_1,t) &= \boldsymbol{u}_0(t)\,,\\ \partial_t \boldsymbol{x}(\mathtt{v}_\mathrm{N},t) &= \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,,\end{aligned} \tag{6.4}$$

is induced by the associated discretisation and mapping of the analytical formulation of their control inputs. Therefore, by applying the 2−step transformation process on (5.116) and (5.120) the processed formulation of the control inputs can be written as

$$\boldsymbol{u}_0(t) = \boldsymbol{u}_0^*(t) - \left(D(t) - M(\mathtt{v}_1,t)\right)\tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_1,t)\,, \tag{6.5a}$$

$$\begin{aligned}\boldsymbol{u}_{\ell_\mathrm{N}}(t) &= \boldsymbol{u}_{\ell_\mathrm{N}}^*(t) - K_\mathrm{N}(t)\tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_\mathrm{N},t) + K_{\mathrm{N,N-1}}(t)\left(\tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_\mathrm{N},t) - \tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_{\mathrm{N}-1},t)\right)\\ &\quad + K_1(t)\tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_1,t) + \sum_{i=1}^{\mathrm{N}} K_\mathrm{S}(\mathtt{v}_i,t)\tilde{\boldsymbol{x}}_\mathrm{c}(\mathtt{v}_i,t)\end{aligned} \tag{6.5b}$$

whereas the gain abbreviations for the input of the leader agent are set to

$$\begin{aligned}K_\mathrm{N}(t) &= D(t) - M(\mathtt{v}_\mathrm{N},t) + \partial_s K(\mathtt{v}_\mathrm{N},\mathtt{v}_\mathrm{N},t)A + K(\mathtt{v}_\mathrm{N},\mathtt{v}_\mathrm{N},t)B(\mathtt{v}_\mathrm{N},t)\,,\\ K_{\mathrm{N,N-1}}(t) &= K(\mathtt{v}_\mathrm{N},\mathtt{v}_\mathrm{N},t)A\,,\\ K_1(t) &= \partial_s K(\mathtt{v}_\mathrm{N},\mathtt{v}_1,t)A\,,\end{aligned} \tag{6.6a}$$

and

$$\begin{aligned}K_\mathrm{S}(\mathtt{v}_i,t) &= \partial_t K(\mathtt{v}_\mathrm{N},\mathtt{v}_i,t) + {}_{-1}^{\,1}\mathcal{D}^{(1)}(\partial_s K)(\mathtt{v}_\mathrm{N},\mathtt{v}_i,t)A + \partial_s K(\mathtt{v}_\mathrm{N},\mathtt{v}_i,t)B(\mathtt{v}_i,t)\\ &\quad + \left(D(t) - M(\mathtt{v}_\mathrm{N},t)\right)K(\mathtt{v}_\mathrm{N},\mathtt{v}_i,t) + K(\mathtt{v}_\mathrm{N},\mathtt{v}_i,t)\left(C(\mathtt{v}_i,t) + \partial_s B(\mathtt{v}_i,t)\right)\,,\end{aligned} \tag{6.6b}$$

respectively. From the leader protocol in (6.5) it is obvious that the formulation demands an additional subgraph representation due to the leader-follower configuration. More precisely speaking the backstepping based control law induces a star graph $\mathcal{S}_\mathrm{N}$ which is show exemplarily in Figure 6.1b. Together with the path graph representation discussed above this forms a rather complex subgraph $\mathcal{G}_{\mathrm{s,N}}$ for $|\mathtt{V}| = \mathrm{N}$. The structure is briefly sketched in Figure 6.1c. This demands a sophisticated network topology which in general is the main motivator for the design of state estimators. With this, let us transform the observer equations into the spatially discretised form

$$
\begin{aligned}
\partial_t \hat{\boldsymbol{x}}(\mathtt{v}_1,t) &= \boldsymbol{u}_0(t) + (E(t) - N(\mathtt{v}_1,t))\left(\boldsymbol{y}_{\mathtt{v}_1}(t) - \hat{\boldsymbol{x}}(\mathtt{v}_1,t)\right)\,, \\
\partial_t \hat{\boldsymbol{x}}(\mathtt{v}_i,t) &= A\left(\hat{\boldsymbol{x}}(\mathtt{v}_{i+1},t) - 2\hat{\boldsymbol{x}}(\mathtt{v}_i,t) + \hat{\boldsymbol{x}}(\mathtt{v}_{i-1},t)\right) - \frac{B(\mathtt{v}_i,t)}{2}\left(\hat{\boldsymbol{x}}(\mathtt{v}_{i+1},t) - \hat{\boldsymbol{x}}(\mathtt{v}_{i-1},t)\right) \\
&\quad + C(\mathtt{v}_i,t)\hat{\boldsymbol{x}}(\mathtt{v}_i,t) + \Phi(\mathtt{v}_i,t)\left(\boldsymbol{y}_{\mathtt{v}_\mathrm{N}}(t) - \hat{\boldsymbol{x}}(\mathtt{v}_\mathrm{N},t)\right) \\
&\quad + \Psi(\mathtt{v}_i,t)\left(\boldsymbol{y}_{\mathtt{v}_\mathrm{N},\mathtt{v}_{\mathrm{N}-1}}(t) - {}_{-1}^{\;\,0}\mathcal{D}^{(1)}(\hat{\boldsymbol{x}})(\mathtt{v}_\mathrm{N},t)\right)\,, \\
\partial_t \hat{\boldsymbol{x}}(\mathtt{v}_\mathrm{N},t) &= \boldsymbol{u}_{\ell_\mathrm{N}}(t) + (E(t) - N(\mathtt{v}_\mathrm{N},t))\left(\boldsymbol{y}_{\mathtt{v}_\mathrm{N}}(t) - \hat{\boldsymbol{x}}(\mathtt{v}_\mathrm{N},t)\right)
\end{aligned}
\tag{6.7}
$$

with the three measurements $\boldsymbol{y}_{\mathtt{v}_1}(t) = \boldsymbol{x}(\mathtt{v}_1,t)$, $\boldsymbol{y}_{\mathtt{v}_\mathrm{N}}(t) = \boldsymbol{x}(\mathtt{v}_\mathrm{N},t)$, and $\boldsymbol{y}_{\mathtt{v}_\mathrm{N},\mathtt{v}_{\mathrm{N}-1}}(t) = \partial_z \boldsymbol{x}(\mathtt{v}_\mathrm{N},t)$. In case $\partial_z \boldsymbol{x}(\mathtt{v}_\mathrm{N},t)$ cannot be measured directly the latter measurement may be approximated by $\boldsymbol{y}_{\mathtt{v}_\mathrm{N},\mathtt{v}_{\mathrm{N}-1}}(t) = {}_{-1}^{\;\,0}\mathcal{D}^{(1)}(\boldsymbol{x})(\mathtt{v}_\mathrm{N},t) = \boldsymbol{x}(\mathtt{v}_\mathrm{N},t) - \boldsymbol{x}(\mathtt{v}_{\mathrm{N}-1},t)$. For completeness the transformed observer PDE for a MVBE is written as

$$
\begin{aligned}
\partial_t \hat{\boldsymbol{x}}(\mathtt{v}_i,t) &= A\left(\hat{\boldsymbol{x}}(\mathtt{v}_{i+1},t) - 2\hat{\boldsymbol{x}}(\mathtt{v}_i,t) + \hat{\boldsymbol{x}}(\mathtt{v}_{i-1},t)\right) \\
&\quad - \frac{B(\mathtt{v}_i,t)}{2} X(\mathtt{v}_i,t)\left(\hat{\boldsymbol{x}}(\mathtt{v}_{i+1},t) - \hat{\boldsymbol{x}}(\mathtt{v}_{i-1},t)\right) + C(\mathtt{v}_i,t)\hat{\boldsymbol{x}}(\mathtt{v}_i,t) \\
&\quad + \Phi(\mathtt{v}_i,t)\left(\boldsymbol{y}_{\mathtt{v}_\mathrm{N}}(t) - \hat{\boldsymbol{x}}(\mathtt{v}_\mathrm{N},t)\right) + \Psi(\mathtt{v}_i,t)\left(\boldsymbol{y}_{\mathtt{v}_\mathrm{N},\mathtt{v}_{\mathrm{N}-1}}(t) - {}_{-1}^{\;\,0}\mathcal{D}^{(1)}(\hat{\boldsymbol{x}})(\mathtt{v}_\mathrm{N},t)\right)\,.
\end{aligned}
\tag{6.8}
$$

With this, replacing the state vector with the estimated state vector in the leader protocol (6.5), i.e., $\boldsymbol{x} \to \hat{\boldsymbol{x}}$, simplifies the required topology of the communication network enormously. Basically it is defined by the measurement setup of the associated estimation algorithm. From the measurements applied to (6.7) or (6.8) one can easily derive the simplified graph representation plotted in Figure 6.1d. Combined with the path graph in Figure 6.1a this leads to a N-cycle graph $\mathcal{C}_\mathrm{N}$ which is exemplarily depicted in (6.1e). Obviously the representation is more readable and the presence of an state estimation algorithm leads to a slimmer network topology.

Remark 6.26. *Applying Dirichlet BCs instead of dynamic conditions would not change the graph representation when the system lacks state estimation. Again, the combined graph would result in the rather complex subgraph shown in* (6.1c). *However including a backstepping based observer algorithm into the feedback control loop would not add any mandatory communication paths to the graph representation induced due to numerical differentiation by FDM. The resulting graph would remain as the path graph shown in* (6.1a).

Basically this section finishes the theoretical preparation in order to conduct the following comprehensive simulation studies and real time experiments. Therefore, in the following let us focus on four challenging scenarios with various formations transitions of a simulated swarm of agents.

6.2 Simulation Scenarios

In the following an extensive simulations study shall verify the general concept of formation control of MASs by means of a continuous model description. Apart from that the studies shall gain evidence for the design of a 2DOF controller and an observer algorithm based on methods which are actually developed for distributed parameter systems. For the latter the scenarios shall serve as first feasibility study of the flatness-based FFC discussed in Chapter 4 and the backstepping-based FBC including the Luenberger-type observer design shown in Chapter 5. Both concepts are developed for coupled parabolic PDEs. From this, for a qualifying discussion of the simulations results it is useful to define the discretised $\mathbf{L}^2-$norms of important error variables, e.g.,

$$\left\|\tilde{\boldsymbol{x}}_{\mathrm{c}}\right\|_{\mathbf{L}^2}(t) := \left(\sum_{i=1}^{\mathrm{N}} \left(\boldsymbol{x}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\right)^T \left(\boldsymbol{x}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\right)\right)^{\frac{1}{2}} , \tag{6.9a}$$

$$\left\|\tilde{\boldsymbol{x}}_{\mathrm{o}}\right\|_{\mathbf{L}^2}(t) := \left(\sum_{i=1}^{\mathrm{N}} \left(\boldsymbol{x}(\mathrm{v}_i,t) - \hat{\boldsymbol{x}}(\mathrm{v}_i,t)\right)^T \left(\boldsymbol{x}(\mathrm{v}_i,t) - \hat{\boldsymbol{x}}(\mathrm{v}_i,t)\right)\right)^{\frac{1}{2}} , \tag{6.9b}$$

$$\left\|\tilde{\boldsymbol{x}}_{\mathrm{c}} - \tilde{\boldsymbol{x}}_{\mathrm{o}}\right\|_{\mathbf{L}^2}(t) := \left(\sum_{i=1}^{\mathrm{N}} \left(\hat{\boldsymbol{x}}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\right)^T \left(\hat{\boldsymbol{x}}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\right)\right)^{\frac{1}{2}} , \tag{6.9c}$$

where the latter stands for the discrete $\mathbf{L}^2-$norm of the estimated control error which is applied to the leader protocol (6.5) in case of an observer-based FBC approach.

Next, it follows a discussion about the simulation results of the first scenario where a simulated MAS performs two successive formation transitions. There the swarm dynamics is modelled as an uncoupled DRS.

6.2.1 Simulation Results for an Uncoupled DRS Model

The results of the first scenario are shown as a 3D plot in Figure 6.2. It illustrates the simulated state vectors $\boldsymbol{x}(\mathrm{v}_i,t)$ of a swarm of 11 agents whereas the MAS dynamic is modelled by a continuous formulation governed from an uncoupled DRS. At this point it is worth to mention that all simulation scenarios start in a line configuration and successively transit to various other steady state formation profiles. For instance, for this scenario the deployment moves to a half circle and finally to a circular shaped profile. The scenario benefits from an observer-based tracking controller in order to stabilise the control error between the simulated states and the desired target states. The simulation time for each transition was set to $\tau = 0.8{\ell_{\mathrm{N}}}^2\,\mathrm{s} = 0.8(\mathrm{N}-1)^2\,\mathrm{s} = 80\,\mathrm{s}$ whereas the entire simulation time is configured to $T_{\mathrm{sim}} = 200\,\mathrm{s}$ which leaves some additional time to underline the theoretically verified stability property. The cycle time of the applied observer algorithm was set to 10 ms and is denoted as observer cycle time in the following. In equivalent manner the controller cycle time defines the frequency of the execution of the controller algorithm. It is slowed down for the simulation scenario to 30 ms in order

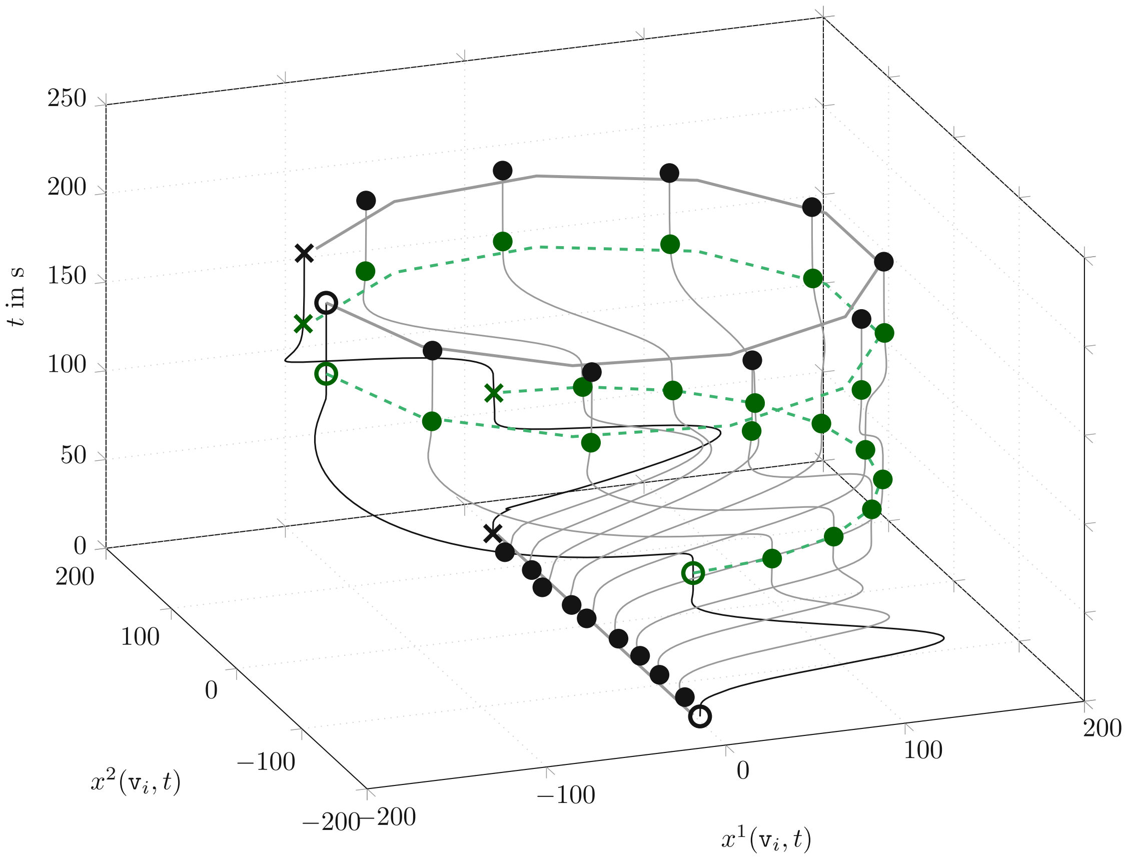
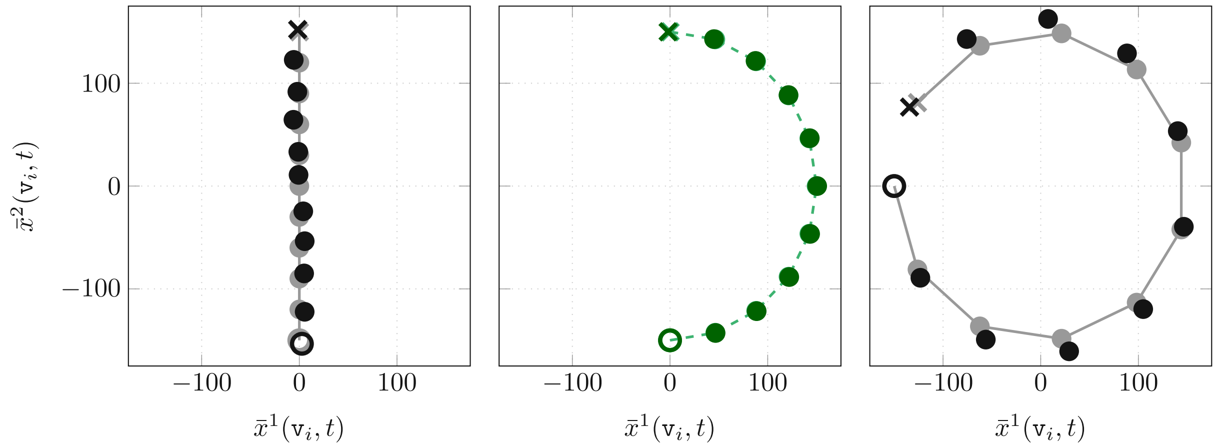


Figure 6.2: Simulation result of formation transitions for an MAS modelled as an uncoupled DRS. Simulation time is $T_{\text{sim}} = 200\,\text{s}$, observer cycle time is set to $\tau_{\text{o}} = 10\,\text{ms}$, control cycle time is set to $\tau_{\text{c}} = 3\tau_{\text{o}}$, and the (x^1, x^2) coordinates are illustrated in cm. The anchor agent is indicated as ○ and the leader agent as ×. Followers are denoted as ●.

Table 6.1: Parameter values for an uncoupled DRS scenario with $\tau = 80\,\text{s}$. Geometric values are given in cm.

		$t = 0\,\text{s}$			$t = \tau$			$t = 2\tau$		
Coord.	a	$\bar{c}^*$	$\bar{x}^*_{\mathtt{a}}$	$\bar{x}^*_{\mathtt{1}}$	$\bar{c}^*$	$\bar{x}^*_{\mathtt{a}}$	$\bar{x}^*_{\mathtt{1}}$	$\bar{c}^*$	$\bar{x}^*_{\mathtt{a}}$	$\bar{x}^*_{\mathtt{1}}$
x^1:	0.5	0	0	0	$a\frac{\pi^2}{{\ell_{\text{N}}}^2}$	0	0	$a\frac{4\pi^2}{(\ell_{\text{N}}+1)^2}$	−150	$-150\cos\left(\frac{2\pi\ell_{\text{N}}}{\ell_{\text{N}}+1}\right)$
x^2:	0.5	0	−150	150	$a\frac{\pi^2}{{\ell_{\text{N}}}^2}$	−150	150	$a\frac{4\pi^2}{(\ell_{\text{N}}+1)^2}$	0	$-150\sin\left(\frac{2\pi\ell_{\text{N}}}{\ell_{\text{N}}+1}\right)$

to demonstrate the robustness of the backstepping controller. Additionally an initial observer estimation error $\tilde{\boldsymbol{x}}_{\text{o}}$ at $t = t_0 = 0\,\text{s}$ is introduced in order to analyse the transient behaviour. The system parameters of the continuous DRS model and the steady state values of the anchor and leader agent are listed in Table 6.1. Regarding the final circular steady state formation profile it is worth to mention that the $(\ell_{\text{N}} + 1)-$dependency of the parameter c and the leader position $\bar{x}_1$ at $t = 2\tau$ allows to deploy the leader and anchor agent without collision at the



(a) Starting line profile. (b) Intermediate half circle. (c) Final circular profile.

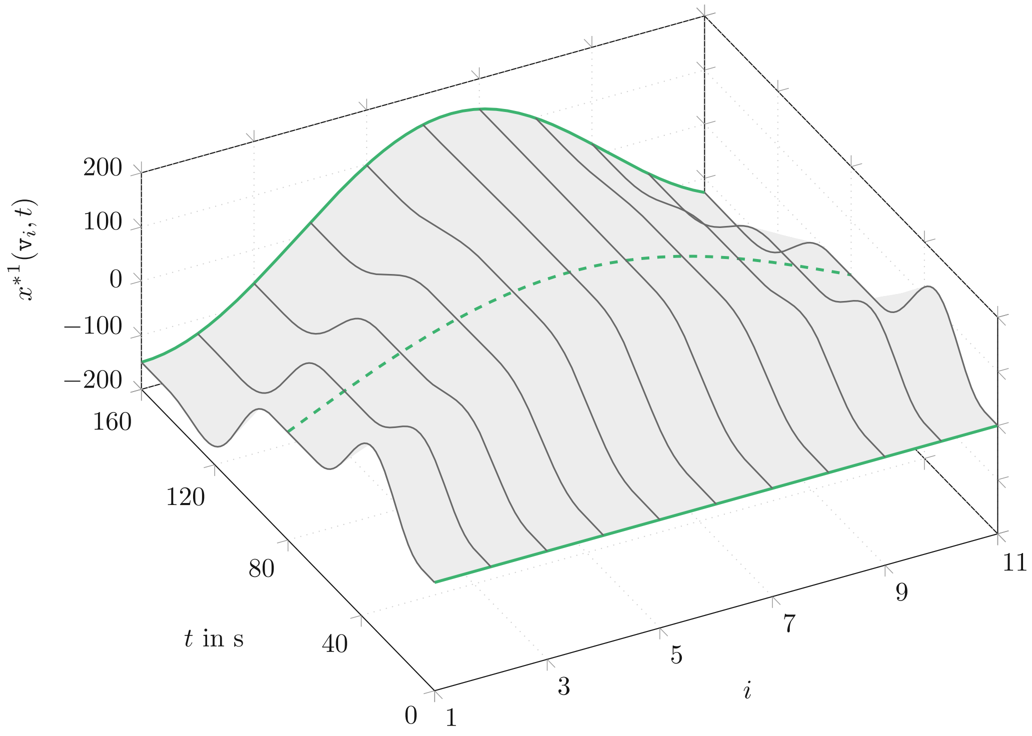
Figure 6.3: Steady state profiles in the (x^1, x^2)-plane: (a) A line as starting deployment, (b) a half circle as intermediate profile, and (c) a circular shape as final formation. Bright colours (●,●) indicate desired values and dark colours (●,●) simulated values.

predefined steady state situation. In other words, the circle is not closed and consequently, leader and anchor do not overlap at $t = 2\tau$. At this stage the reader who is interested in steady state evaluation of uncoupled DRSs is referred to Section 3.2 as well as Appendix B.2.

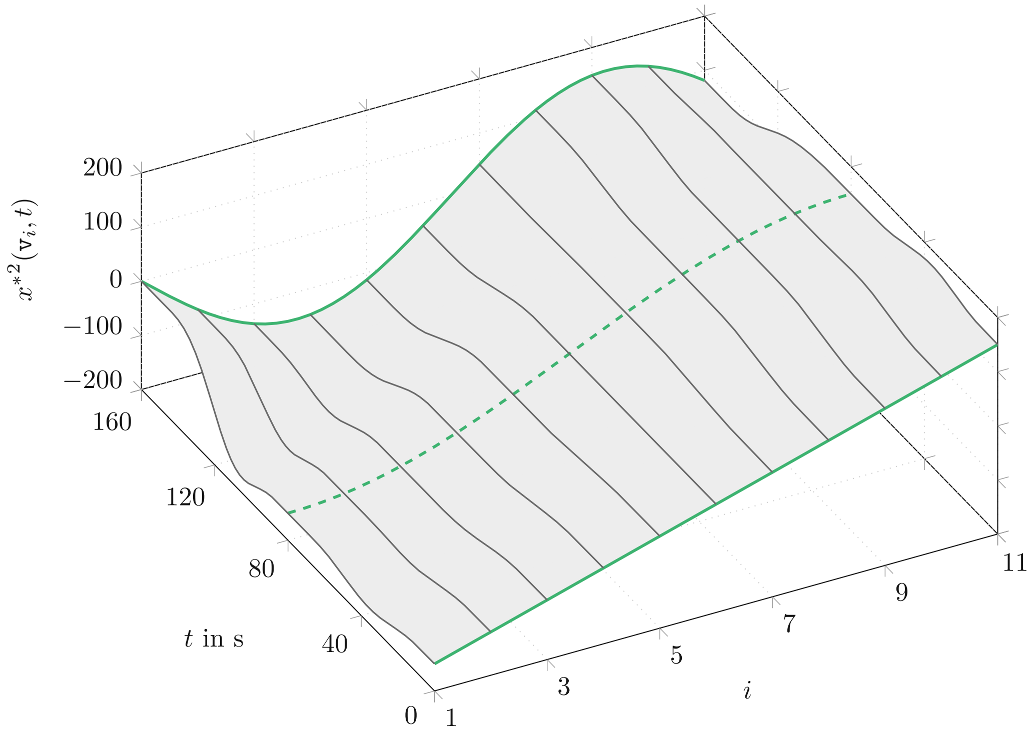
With this, the plots in Figure 6.3 show both simulated and desired state values for the starting line deployment, the intermediate half circle, and the final configuration at $t = T_{\mathrm{sim}}$. Assuming nominal observer states $\hat{\boldsymbol{x}}(\mathtt{v}_i, 0) = \boldsymbol{x}^*(\mathtt{v}_i, 0)$ and simulated state values placed off from the nominal line at $t = 0\,\mathrm{s}$, i.e., $\boldsymbol{x}(\mathtt{v}_i, 0) = \boldsymbol{x}^*(\mathtt{v}_i, 0) + \tilde{\boldsymbol{x}}_{\mathrm{c}}(\mathtt{v}_i, 0)$, Figure 6.3a clearly depicts the initial state error. This procedure matches situations which are faced in real time experiments. The almost exact matching of the simulated state $\boldsymbol{x}$ with the desired state $\boldsymbol{x}^*$ in Figure 6.3b proofs that the control strategy copes with the irritation at the beginning of the simulation and Figure 6.3c shows that the observer-based tracking controller stabilises the unstable circular profile. Moreover, the remaining offset between the simulated state and the desired state reveals that the backstepping stabilisation cannot compensate the control error ideally and reminds of a conventional state or P−controller, respectively.

Motion planning and Feedforward Control

The next two figures show results particularly for the flatness-based motion planning and FFC. Figure 6.4a depicts desired trajectories $\boldsymbol{x}^*$ of the agent states for coordinate 1 and Figure 6.4b those of coordinate 2. The grey shadowed surface illustrates the synthesis in continuous description in terms of the spatial domain $z \in [0, \ell_{\mathrm{N}}]$. Considering both plots in Figure 6.4 one can easily imagine the temporal transformation from the line configuration at $t = 0\,\mathrm{s}$ to the subsequent circular shapes which are formed by means of an overlap of sin− and cos−functions. For this, the parameter value for the associated spatial period is listed in Table 6.1. The de-

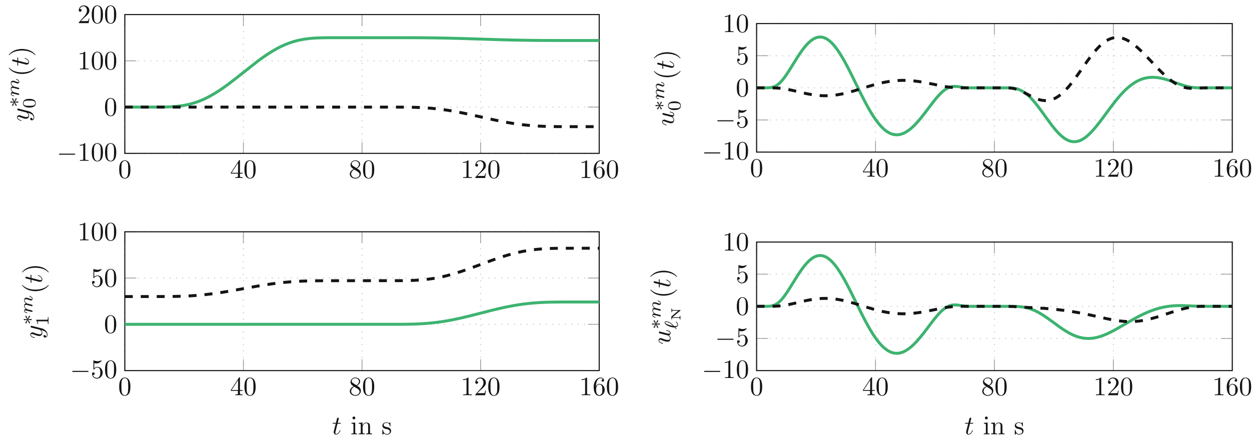


(a) Desired trajectories for coordinate x^1.



(b) Desired trajectories for coordinate x^2.

Figure 6.4: Desired trajectories $\boldsymbol{x}^*$ for the planning period $t \in [0, 160]\,\mathrm{s}$ computed by means of formal state parametrisation of a scalar DRS, see Section 4.1. The grey lines (―) show the nominal trajectories for 11 agents, the green lines (― , – –) shall symbolise the steady states situations at $t = \bar{t} \in \{0\,\mathrm{s}, 80\,\mathrm{s}, 160\,\mathrm{s}\}$ according to the applied motion planning procedure.



(a) Trajectories of the flat outputs.

(b) Feed forward control inputs.

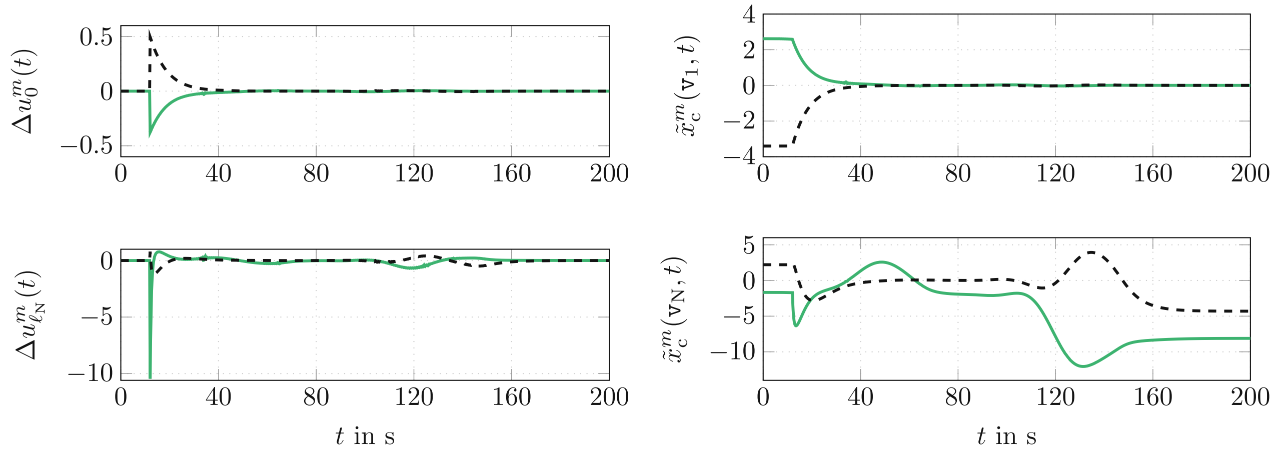
Figure 6.5: Desired and planned trajectories for the motion planning period $t \in [0, 160]$ s: (a) Desired trajectories $\{y_0^{*m}, y_1^{*m}\}$ of the components $\boldsymbol{y}^m$ of the flat output; (b) planned FF control inputs $\{u_0^{*m}, u_{\ell_N}^{*m}\}$ for dynamic BCs. Green solid lines (—) show trajectories for coordinate $m = 1$ and black dashed lines (- -) indicate time functions for coordinate $m = 2$.

sired state vector $\boldsymbol{x}^*(z,t)$ of the representing DRS is calculated according to the developed motion planning concept in Chapter 4. In particular the recursive algorithm (4.23) of the series ansatz is used together with the trajectory assignments (4.81) and (4.83), respectively. Here, the bell-shaped function (C.13) is utilised to govern the Gevrey function $\varPhi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}$ and $\varPhi_{\tau_c,\omega_c}$. For this simulation scenario the Gevrey order is chosen as $\alpha_{\boldsymbol{y}} = \alpha_C = 1.05$ with a planning horizon of $\tau_{\boldsymbol{y}} = \tau_c = 0.8(\mathrm{N}-1)^2\,\mathrm{s} = 80\,\mathrm{s}$ per formation transition. With this, the flatness-based motion planning approach leads to the desired trajectories y_0^{*m} and y_1^{*m} for the flat output $\boldsymbol{y}^m(t) = [y_0^m(t), y_1^m(t)]^T = [x^m(\xi,t), \partial_z x^m(\xi,t)]^T$ presented in Figure 6.5a with $\xi = \ell_\mathrm{N}/2 = (\mathrm{N}-1)/2$. Moreover, when making use of (4.20) the results of the FF control inputs are displayed in (6.5b). Here it is worth to mention that in the plots coordinate 1 is illustrated by a solid green line and graphs for coordinate 2 are plotted as dashed black lines.

Observer-based Tracking Control

While the last paragraph dealt with results concerning motion planning and FFC the following discussion focuses on FB and error tracking control. As already mentioned in this scenario the control loop is equipped with a Luenberger-type state observer. The design concept applied is worked out in Chapter 5. However for this simulation it is scaled down to individual uncoupled problems. This configuration leads to the a graph representation shown in Figure 6.1c.

As a first, Figure 6.6 presents the control inputs as well as the control error $\tilde{\boldsymbol{x}}_c$ for both coordinates $m \in \{1, 2\}$. All plots clearly show the initial settling time of the injected observer error. During the first 15 s the backstepping-based error tracking controller is disabled which leads to almost constant control errors for both, the leader agent and the anchor agent. As soon



(a) Tracking control inputs.

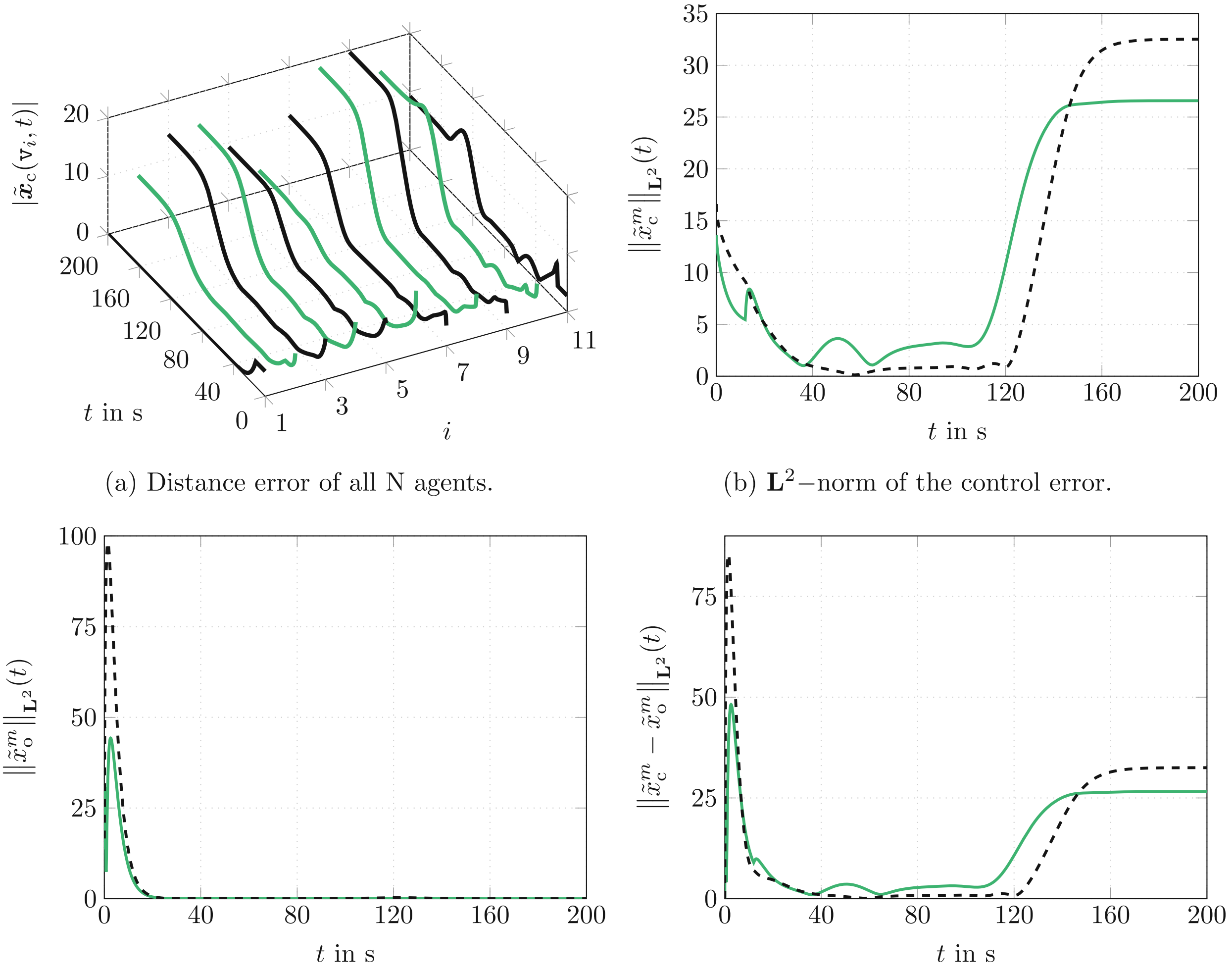
(b) Control error of the cognitive agents.

Figure 6.6: Feedback control inputs and errors for the entire simulation period $t \in [0, 200]$ s: (a) Tracking or feedback control inputs $\Delta u_0^m(t)$ of the anchor and $\Delta u_{\ell_N}^m(t)$ of the leader; (b) coordinate-wise position errors $\tilde{x}_c^m(\mathrm{v}_1, t)$ of the anchor and $\tilde{x}_c^m(\mathrm{v}_N, t)$ of the leader. Again, green solid lines (━━) illustrates values for coordinate $m = 1$ and black dashed lines (╍╍) for coordinate $m = 2$.

as the control loop is closed on the one hand relatively high control inputs ensure stability of the entire swarm formation and on the other hand general tracking of the desired states. The upper plots in Figure 6.6a and Figure 6.6b refer to results from the anchor agent. Recalling the discussion regarding (5.116) PT$_1$−behaviour is claimed by design for the tracking error $\tilde{x}_c^m(\mathrm{v}_i, t)$ of the anchor. It is fair to say that the related plots substantiate this analysis. While the control inputs of the anchor quickly converge to zero the lower plot in Figure 6.6a shows that the leader agent works quite intensively to keep the entire swarm on track. Moreover Figure 6.6b reveals that it seems the leader cannot hold its position during the formation transition but it stabilises at the desired steady state situations at $t = 80$ s and $t = 160$ s. The additional simulation time up to 200 s clearly indicates an remaining tracking error which is subsequently discussed in detail.

The following plots in Figure 6.7 outline the involved error values. Figure 6.7a shows the temporal evolution of euclidean distance between the desired and simulated position for each agent. The figure gives a lot of additional information, which shall be briefly listed in the following.

- Since the starting line profile is stable the initial error gets compensated for the follower agents without tracking control in place.
- The anchor just compensates its initial error and from then it is stable in all coordinates throughout the entire simulation.
- The error distance gets much larger for all other agents during the transition from the half circle to the circular shape. This happens since the coefficient of the reaction term



(a) Distance error of all N agents.

(b) $\mathbf{L}^2-$norm of the control error.

(c) $\mathbf{L}^2-$norm of the observer error.

(d) $\mathbf{L}^2-$norm of the estimated control error.

Figure 6.7: Absolute position errors and component-wise error norms in cm: (a) Temporal evolution of the agent's position error $|\tilde{\boldsymbol{x}}_\mathrm{c}(\mathrm{v}_i,t)| = \|\boldsymbol{x}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\|_2$ as euclidean distance in the $(x^1,x^2)-$plane; (b) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}^m_\mathrm{c}(\mathrm{v}_i,t)$ between simulated state and desired state; (c) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}^m_\mathrm{o}(\mathrm{v}_i,t)$ between simulated state and estimated state gained from the observer; (d) discrete $\mathbf{L}^2-$norm of the error between the estimated observer state and the desired state. The $\mathbf{L}^2-$norms are computed according to (6.9) for each coordinate $m = \{1,2\}$ individually. Green solid lines (—) show the norms for $m = 1$ and black dashed lines (– –) for $m = 2$.

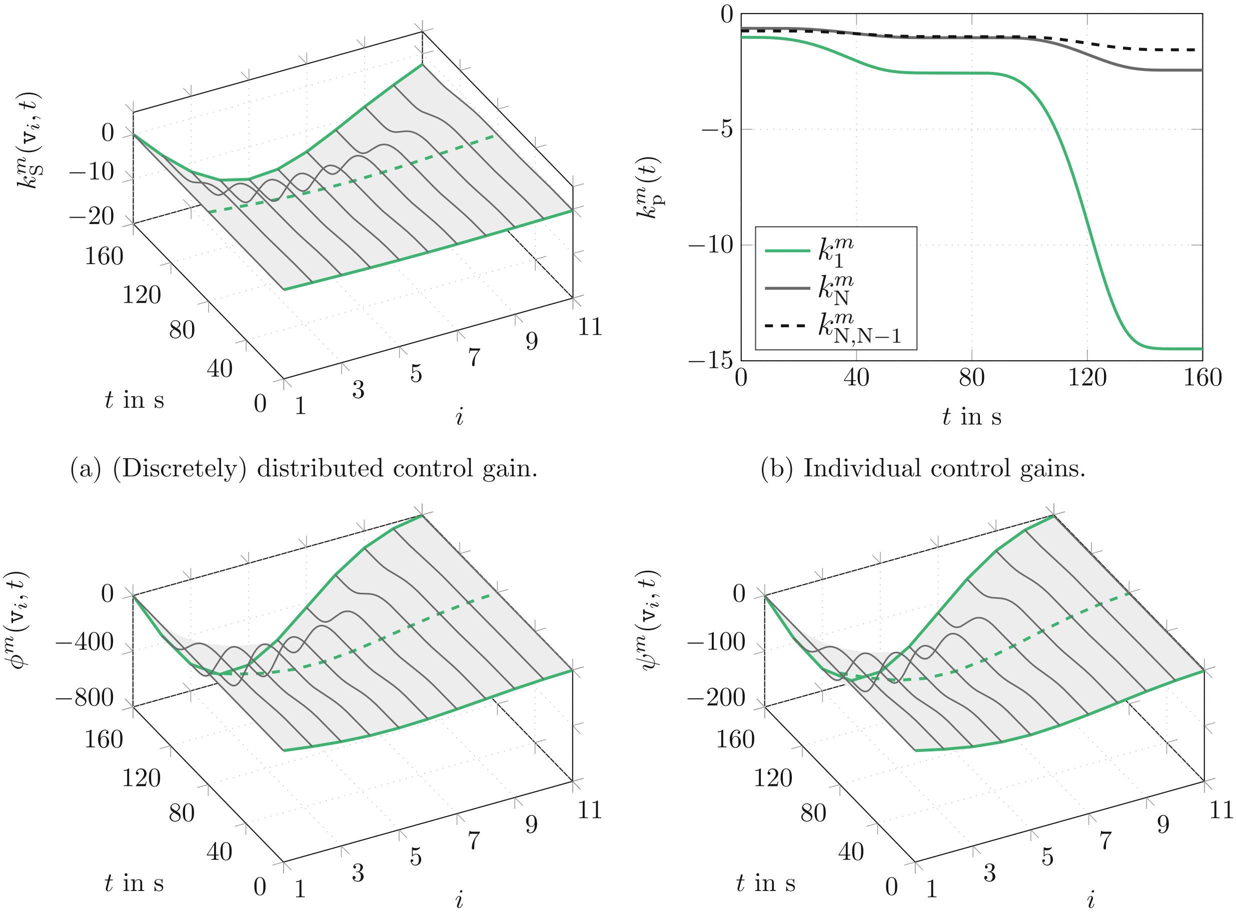
increases with time and so does the instability of the steady state formation profile, cf. the parameter values for c in Table 6.1.

- The final error distance reminds of a function which consists of a polynomial part and an additional $|\sin|-$part.

For the latter point one can conclude that the remaining error distance has two origins. First, the continuous design approach induces an error since the PDE formulation is an approximation of the simulated MAS represented by a N$-$dimensional set of ODEs. Consequently, the computed steady states $\boldsymbol{x}^*$ of the continuous system are as well an approximation and do not necessarily match the steady states of the discrete system. This effect decreases when the number of agents N increases. However, as already discussed before the tracking controller can be compared with a traditional P$-$control algorithm. Therefore the desired error dynamics defined by the target system (5.12) is reached entirely for all agents. This may introduce an additional remaining offset to the desired steady state values. Figures 6.7b to 6.7d depict different discretised $\mathbf{L}^2-$ norms for each coordinate individually. Here it can be said, that the illustration of the norm of the control error $\tilde{x}_\mathrm{c}$ in Figure 6.7b supports the statements from above. Moreover, in Figure 6.7c it is important to see that the $\mathbf{L}^2-$ norm of the observation error $\tilde{x}_\mathrm{c}$ rapidly recovers from the initial error and converges to zero. An interesting plot is shown in 6.7d where the norm of the estimated control error is sketched for each coordinate. At the beginning it is clearly dominated by the observation error and later by the actual control error. Therefore it does make sense to disable the tracking controller until state estimation is settled to a certain level. For this note the small peaks after 15 s in Figure 6.7b and Figure 6.7d.

The last plots of the simulation results give an impression of the magnitudes for the tracking controller and state observer. Here, it has to be pointed out that for this simulation scenario the numerical results of controller and observer design are completely identical for coordinate 1 and 2. Note, the system parameters a and c in Table 6.1 are equal and the tuning parameters are set to constant values, i.e., $d^m(t) = d^m = 0.15$ and $e^m(t) = e^m = 0.30$ for both coordinates. Thus, Figure 6.8a sketches the controller gains which consider the control errors distributed over all agents of the MAS. Contrarily, Figure 6.8b presents the time functions $k_1^m(t)$ and $k_\mathrm{N}^m(t)$ which respect the errors for the anchor and leader agent individually, as well as $k_{\mathrm{N,N-1}}^m(t)$ which corrects the estimated error between the leader agent and its neighbouring follower agent represented by index $\mathrm{N}-1$. Here the controller gains are computed according to the analysis above in (6.6). Furthermore, Figure 6.8c and Figure 6.8d present the distributed gain values for the Luenberger-type state observer. Here ϕ^m, the scalar equivalent of the matrix Φ in (6.7) and (6.8), considers the error of the leader's state estimation $\tilde{x}_\mathrm{o}^m(\mathrm{v_N}, t) = x^m(\mathrm{v_N}, t) - \hat{x}^m(\mathrm{v_N}, t)$. Furthermore, ψ^m, again the corresponding scalar observer gain of Ψ, amplifies the error increase between the leader agent and its next neighbour, i.e., $\Delta\tilde{x}_\mathrm{o}^m(\mathrm{v_N}, t) = \tilde{x}_\mathrm{o}^m(\mathrm{v_N}, t) - \tilde{x}_\mathrm{o}^m(\mathrm{v_{N-1}}, t)$. In general it can be stated that the magnitudes of all gains increase with time. This again is related to the increasing reaction term parameter c. Having a look at the steady state situations which are indicated by the green lines in Figure 6.8a, (c) and (d) the trained eye can identify curves which are very typical for the backstepping approach with parabolic PDEs. They can be constructed by modified Bessel functions of order one. For details the interested reader is referred to, e.g, [80].

The next simulation scenario copes with an underlying uncoupled MVBE model in one coordinate. Apart from dealing with non-linearities it leads to interesting results for comparison since system and simulation parameters are changed enormously and state observation is left out completely.



(a) (Discretely) distributed control gain.

(b) Individual control gains.

(c) Observer gain for $\tilde{x}_{\mathrm{o}}^{m}$.

(d) Observer gain for $\Delta\tilde{x}_{\mathrm{o}}^{m}$.

Figure 6.8: Control and observer gains: (a) (Discretely) distributed control gain according to (6.6a) which amplifies the control error $\tilde{x}_{\mathrm{c}}^{m}$ of every agent; amplified errors are summed up to form a control input signal. (b) individual control gains according to (6.6b) which respect the control error of the anchor, the control error of the leader and the control error of the *state difference* between the leader and its next follower; (c) observer gain according to (5.143a) which injects the estimation error $\tilde{x}_{\mathrm{o}}^{m}$ into the estimation algorithm; (d) observer gain according to (5.143b) which aims to correct the estimation error $\Delta\tilde{x}_{\mathrm{o}}^{m}$. In the 3D-plots green lines (—— , – –) indicate steady state situations in the sense of the applied motion planning procedure.

6.2.2 Simulation Results for an Uncoupled MVBE Model

Numerical results of the second scenario are presented in Figure 6.9. It shows in 3D the simulation of the state vector $\boldsymbol{x}$ associated to 21 agents whereas the MAS dynamics is modelled by a MVBE for coordinate 1 and by a DRS for coordinate 2. As already mentioned above the deployment always starts in a line configuration, then it successively moves to an $U-$shaped steady state formation profile and later to a profile which is reminiscent of an $\infty-$symbol. The scenario benefits from tracking control without state observation in order to stabilise the control error between the simulated states and the desired target states. Each transition lasts

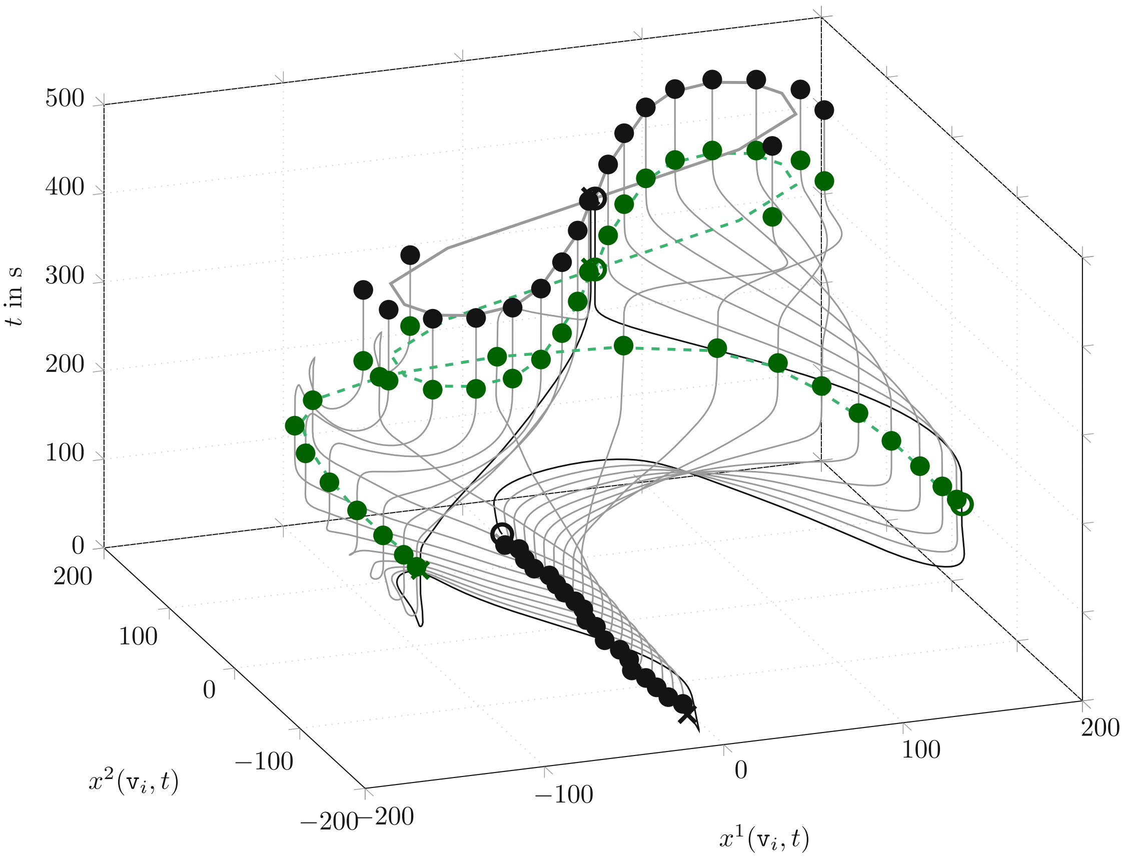
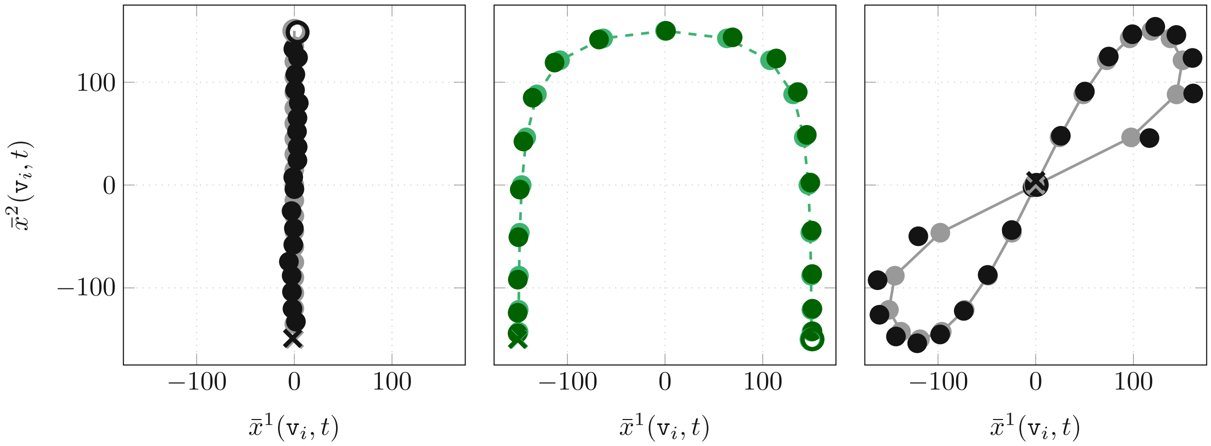


Figure 6.9: Simulation result of formation transitions for an MAS modelled as an uncoupled MVBE. Simulation time is $T_{\text{sim}} = 480\,\text{s}$, control cycle time is set to $\tau_{\text{c}} = 100\,\text{ms}$ and the (x^1, x^2) coordinates are illustrated in cm. The anchor agent is indicated as ○ and the leader agent as ×. Followers are denoted as ●.

Table 6.2: Parameter values for an uncoupled MVBE scenario with $\tau = 200\,\text{s}$. Geometric values are given in cm.

			$t = 0\,\text{s}$			$t = \tau$			$t = 2\tau$		
Coord.	a	b	c	$\bar{x}_\mathtt{a}$	$\bar{x}_\mathtt{1}$	c	$\bar{x}_\mathtt{a}$	$\bar{x}_\mathtt{1}$	c	$\bar{x}_\mathtt{a}$	$\bar{x}_\mathtt{1}$
x^1:	1.0	$a\frac{18}{\bar{x}_A \ell_\text{N}}$	0	150	−150	0	150	−150	$a\frac{60}{{\ell_\text{N}}^2}$	0	0
x^2:	1.0	0	0	0	0	$a\frac{4\pi^2}{{\ell_\text{N}}^2}$	−150	−150	$a\frac{4\pi^2}{{\ell_\text{N}}^2}$	0	0

$\tau = 0.5(\text{N} - 1)^2\,\text{s} = 200\,\text{s}$ whereas the entire simulation time is set to $T_{\text{sim}} = 480\,\text{s}$ in order to conclude on stability properties. The cycle time of the simulation itself is set to 20 ms and the cycle period of the controller is configured to 100 ms which allows to test the robustness of the backstepping approach. Additionally an initial position error $\tilde{\boldsymbol{x}}_\text{c}$ at $t = 0\,\text{s}$ shall emulate a behaviour as it would be the case in a real time experiment. Furthermore, Table 6.2 summarises the system parameters of the continuous models and values of the anchor and leader agent for steady state computation.



(a) Starting line profile.

(b) Intermediate $U-$shaped profile.

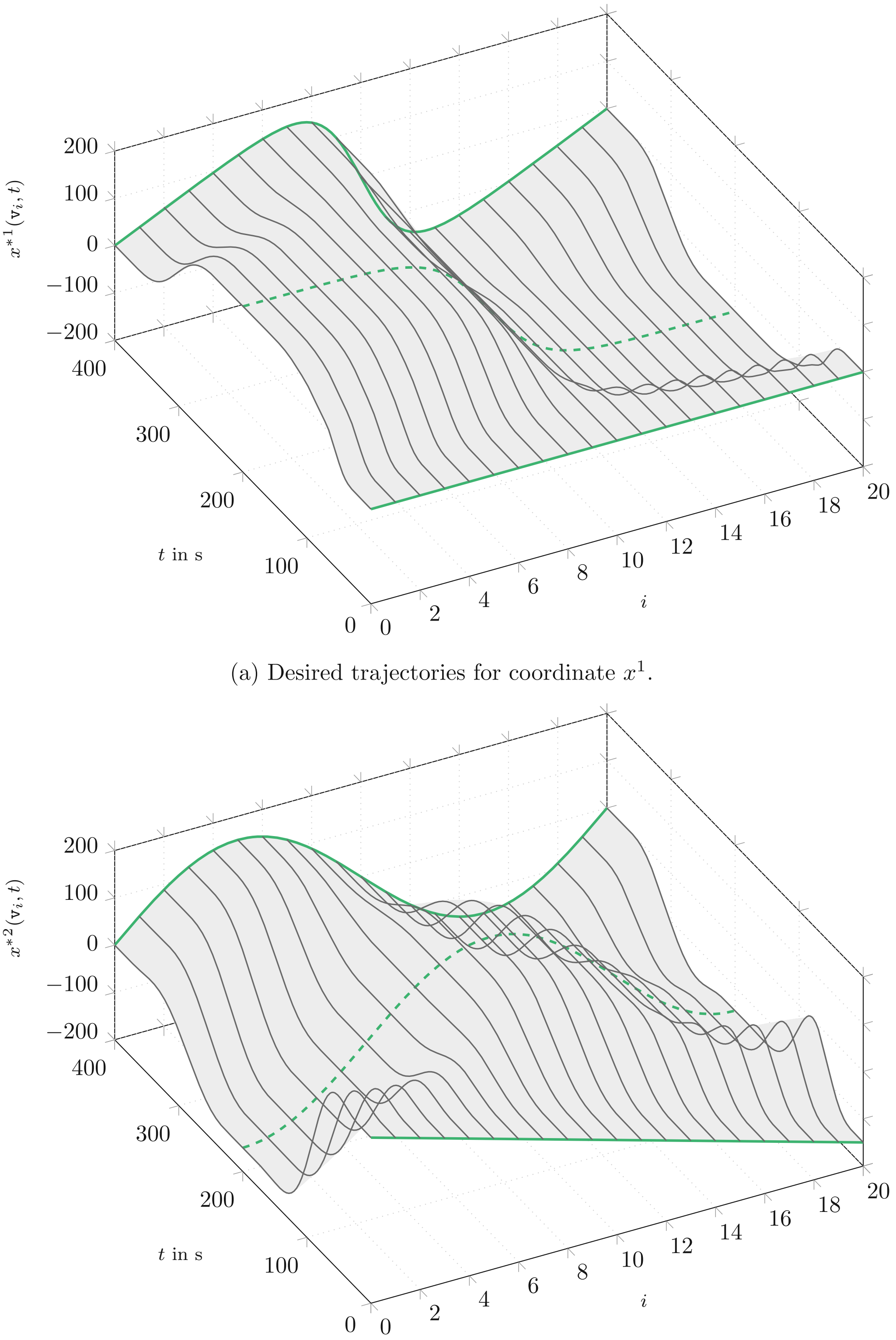
(c) Final $\infty-$symbol profile.

Figure 6.10: Steady state profiles in the (x^1, x^2)-plane: (a) A line as starting deployment, (b) an intermediate formation where the agents are deployed in an $U-$shape, and (c) an $\infty-$symbol as final formation. Bright colours (●,●) stand for desired values and dark colours (●,●) for simulated values.

Next, Figure 6.10 shows three plots, each with 21 simulated and desired state values at steady state situations in terms of their formation profile. Figure 6.10a shows the starting deployment at $t = 0\,\mathrm{s}$ where the simulated state values are positioned off from the nominal line, i.e. $\boldsymbol{x}(\mathrm{v}_i, 0) = \boldsymbol{x}^*(\mathrm{v}_i, 0) + \tilde{\boldsymbol{x}}_\mathrm{c}(\mathrm{v}_i, 0)$. This is followed by the intermediate $U-$shape at $t = \tau$ and sketched in Figure 6.10b. The subplot clearly shows that the control strategy can compensate the initial irritation. Finally, reaching the $\infty-$symbol at $t = 2\tau$ challenges the tracking controller harder than the intermediate profile. However, as pictured in Figure 6.10c it can maintain a stable behaviour at least until $t = T_\mathrm{sim}$.

Motion Planning and Feedforward Control

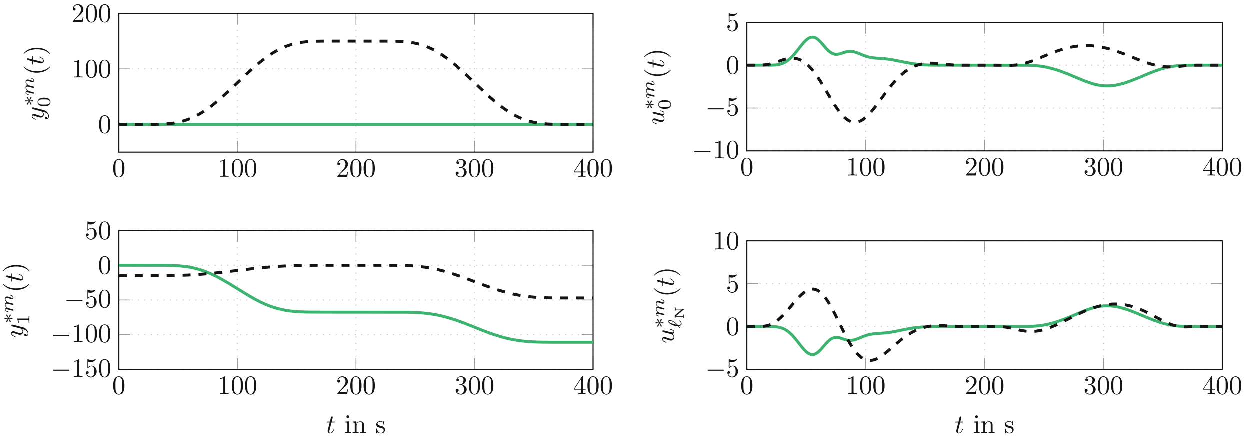
As in the simulation scenario before, the following two figures present results of computations related to motion planning and FFC. Thus, Figure 6.11a shows the nominal states $\boldsymbol{x}^*$ for coordinate 1 and Figure 6.11b those of coordinate 2. Again, the grey surface gives an impression of the result in continuous description of the underlying MVBE and DCRS, respectively. Figure 6.11a illustrates a steep slope in the middle of the spatial domain towards the temporal end. This is known as a shock-like profile which is typical for the Burgers' equation. Figure 6.11b shows the state trajectories of the DRS from a linear to a sin- function. Geometric parameters are listed in Table 6.2. As discussed above, the flatness-based concept in Chapter 4 is exploited for the computation of the desired state vector $\boldsymbol{x}^*(z,t)$. In particular the Gevrey functions $\varPhi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}$ and $\varPhi_{\tau_c,\omega_c}$ implemented by (C.13) use the parameter setting $\alpha_{\boldsymbol{y}} = 1.20$, $\alpha_C = 1.15$ for coordinate 1 and $\alpha_{\boldsymbol{y}} = 1.05$, $\alpha_C = 1.10$ for coordinate 2. The timing is equally set to $\tau_{\boldsymbol{y}} = \tau_c = 0.5(\mathrm{N}-1)^2 = 200\,\mathrm{s}$ per formation transition. Here, desired trajectories $\{y_0^{*m}, y_1^{*m}\}$ for the components $\boldsymbol{y}^m(t) = [y_0^m(t) y_1^m(t)]^T$ of the flat output are illustrated in Figure 6.12a and



(a) Desired trajectories for coordinate x^1.

(b) Desired trajectories for coordinate x^2.

Figure 6.11: Desired trajectories x^* for the planning period $t \in [0, 400]\,\mathrm{s}$. They are computed by means of formal state parametrisation of a scalar MVBE, see Section 4.1. The grey lines (—) show the nominal trajectories for 21 agents, the green lines (— , – –) shall symbolise the steady states situations at $t = \bar{t} \in \{0\,\mathrm{s}, 200\,\mathrm{s}, 400\,\mathrm{s}\}$ according to the applied motion planning procedure.



(a) Trajectories of the flat outputs.

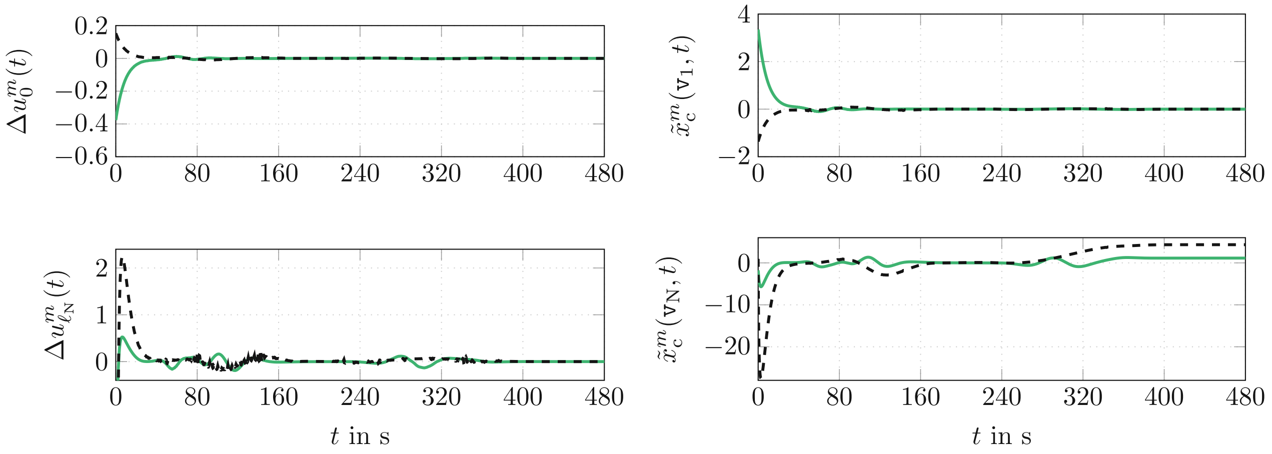
(b) Feed forward control inputs.

Figure 6.12: Desired and planned trajectories for the motion planning period $t \in [0, 400]$ s: (a) Desired trajectories $\{y_0^{*m}, y_1^{*m}\}$ of the components $\boldsymbol{y}^m$ of the flat output; (b) planned FF control inputs $\{u_0^{*m}, u_{\ell_N}^{*m}\}$ for dynamic BCs. Green solid lines (—) show trajectories for coordinate $m = 1$ and black dashed lines (- -) indicate time functions for coordinate $m = 2$.

results of the FF control inputs are plotted in (6.12b) for both coordinates.

Measurement-based Tracking Control

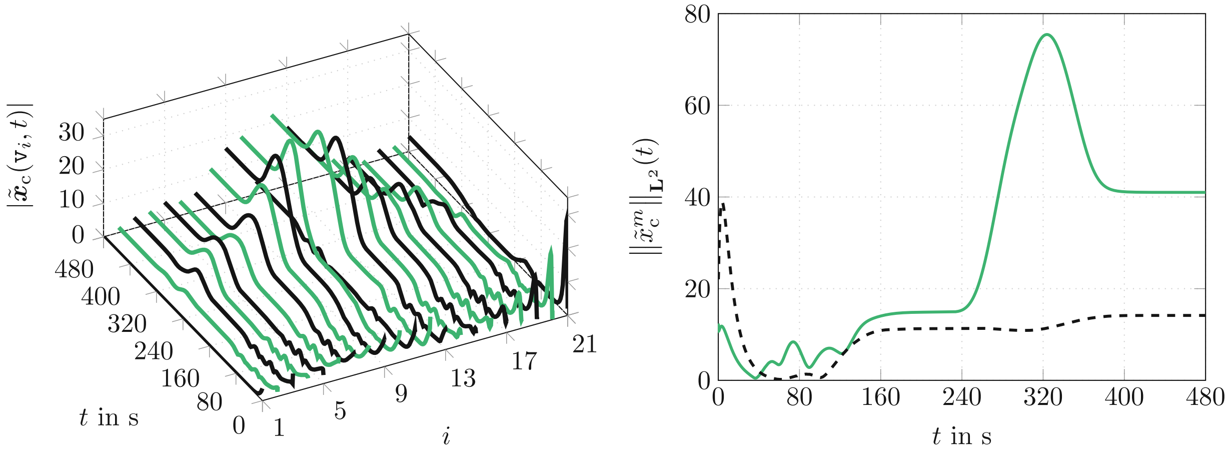
For this scenario the following paragraph discusses FB and error tracking control without the presence of a state estimator. The resulting graph representation for this simulation study is relatively complex and is sketched in Figure 6.1c. Equivalent to the analysis above Figure 6.13a and Figure 6.13b present the graphs of the leader and anchor agent related values for both coordinates $m \in \{1, 2\}$. Since no state estimation is in place the error tracking algorithm is enabled right from the beginning of the simulation. Basically the plots in Figure 6.13 indicate three things. First, the anchor shows its typical PT_1-behaviour due to the induced initial tracking error. Second, the tracking controller is mostly challenged by coordinate 1 which is based on the non-linear MVBE. And third, the leader control signal of coordinate 2 includes high frequencies. The latter observation gets clear when the setting of the tuning parameter is taken into account. For scenario 2 it set to the constant value $d^m = 45/(\mathrm{N}-1)^2$ for both coordinates. In relative numbers this is a three times higher value compared to the setting of the DRS in scenario 1. Due to the strong compensation the leader errors remain in a moderate range. Coming back to the second point it has to be emphasised that for the controller design the MVBE has to be linearised around the pair $\{x^{*1}, \partial_z x^{*1}\}$ which raises the requirement $\tilde{x}_c^1(z,t)\partial_z\tilde{x}_c^1(z,t) \approx 0$, see (5.3)ff. for details. Recalling the shock-like behaviour of the desired state x^{*1} shown in Figure 6.11a it is clear that the precondition is tough to meet. Consequently this tests the robustness of the closed system already by design. From the author's experience the closed loop is not robust enough and may get unstable easily when state estimation is included. However as shown in [26] the integration of an additional in-domain measurement



(a) Tracking control inputs.

(b) Control error of the cognitive agents.

Figure 6.13: Feedback control inputs and errors for the entire simulation period $t \in [0, 480]$ s: (a) Tracking or feedback control inputs $\Delta u_0^m(t)$ of the anchor and $\Delta u_{\ell_N}^m(t)$ of the leader; (b) coordinate-wise position errors $\tilde{x}_c^m(v_1, t)$ of the anchor and $\tilde{x}_c^m(v_N, t)$ of the leader. Again, green solid lines (—) illustrates values for coordinate $m = 1$ and black dashed lines (- -) for coordinate $m = 2$.



(a) Distance error of all N agents.

(b) $\mathbf{L}^2-$ norm of the control error.

Figure 6.14: Absolute position errors and component-wise error norms in cm: (a) Temporal evolution of the agent's position error $|\tilde{\boldsymbol{x}}_c(v_i, t)| = \|\boldsymbol{x}(v_i, t) - \boldsymbol{x}^*(v_i, t)\|_2$ as euclidean distance in the $(x^1, x^2)-$plane; (b) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}_c^m(v_i, t)$ between simulated state and desired state. The $\mathbf{L}^2-$ norm is computed according to (6.9) for each coordinate $m = \{1, 2\}$ individually. Green solid lines (—) show the norms for $m = 1$ and black dashed lines (- -) for $m = 2$.

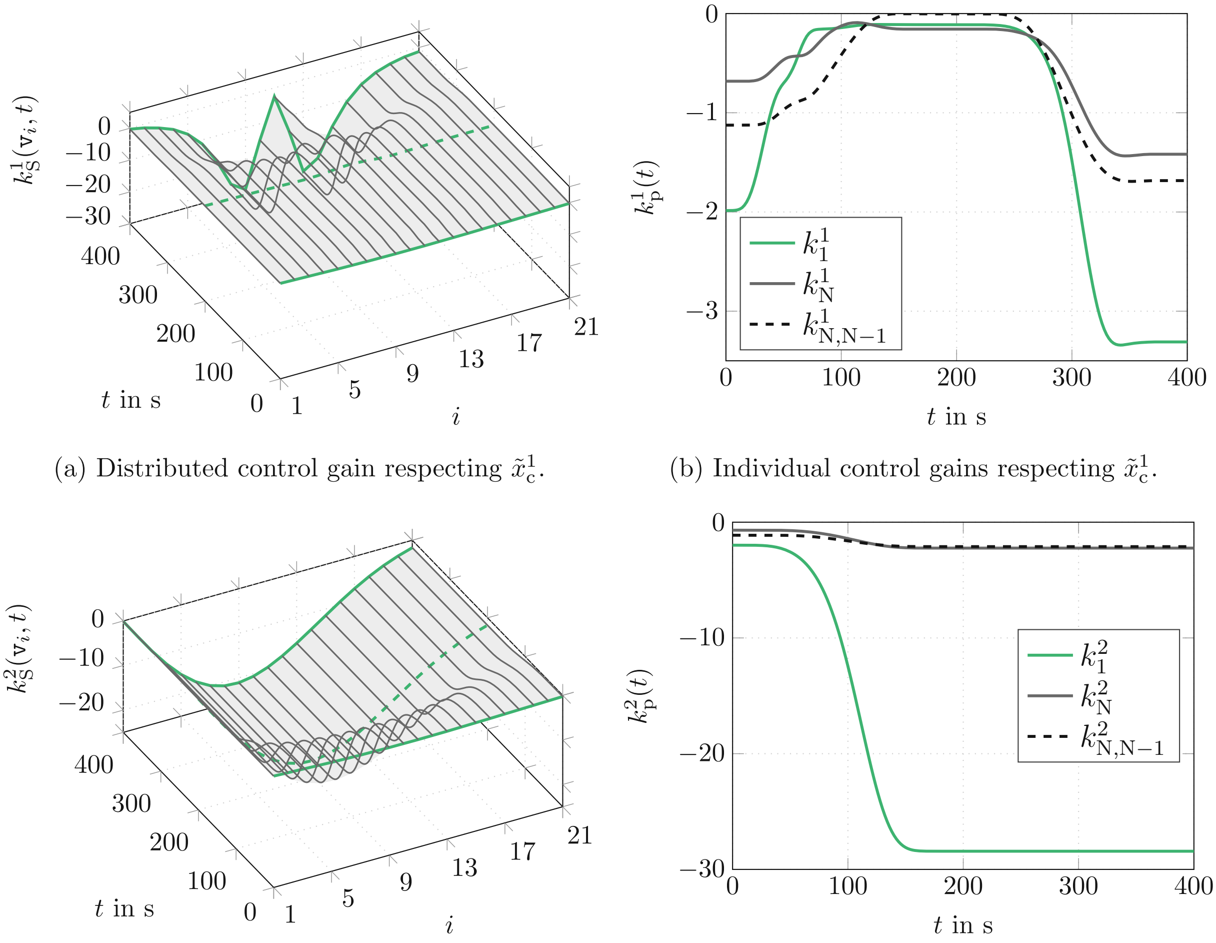
point allows to stabilise the configuration in a robust way. In recent literature, such as [14], the technique is labelled as *domain folding*.

The next two plots in Figure 6.14 deal with the analysis of the control errors $\tilde{\boldsymbol{x}}_c$. Equivalently to above Figure 6.14a shows the temporal evolution of euclidean distance between the simulated and desired state values for each agent. The interpretation is conducted similar as before, i.e.

- the initial error gets compensated for all agents,
- the anchor just needs to compensate its initial error,
- the error distance gets larger as the coefficient of the reaction term increases, cf. the parameter c in Table 6.2.
- the final error distance is dominated by the agents with an index close to $(\mathrm{N}-1)/2$ due to the shock-like target profile which induces high demands at $z = \ell_{\mathrm{N}}/2$, cf Figure 6.11a.

Obviously the discrete $\mathbf{L}^2-$ norms of the control error, shown Figure 6.14b for each coordinate individually, supports this analysis by all means. Apart from that the plot allows an interesting comparison with Figure 6.7b discussed for the simulation scenario beforehand. Note, that x^2 of the current scenario moves from a linear steady state profile to an intermediate cos−deployment and ends up in a sin−function in space. These two latter steady states matches the complete final steady state situation of coordinate 1 and 2 of the first scenario. Now, the comparison of the $\mathbf{L}^2-$ norm of coordinate 2, plotted by a dashed black line in Figure 6.14b, with the graphs in Figure 6.7b allows the confirmation of an important fact. The relatively higher tuning parameter d^2 in the current scenario and the larger number of agents (here 21 agents vs. 11 beforehand) cause much lower $\mathbf{L}^2-$ norm values for the control error, i.e. smaller overall position errors. At this stage it is pointed out again that the continuous problem formulation, especially the determination of the desired steady state formation profiles, can be interpreted as an approximation for the swarm dynamics of a MAS which consists of a discrete number of agents.

The closing plots of the simulation study show the numerical results of the controller gains for coordinate 1 and 2. Thus, Figure 6.15a and Figure 6.15c depict the gains covering the control error of all agents of the MAS, and Figure 6.15b and Figure 6.15d sketch gains which consider particular control errors, especially for the anchor, the leader agent, and the difference of the control error of the leader agent and its next neighbour within the chained network topology. For details to the individual computation of k_1^m, k_{N}^m, and $k_{\mathrm{N},\mathrm{N}-1}^m$ see (6.6). Again, in Figure 6.15a and Figure 6.15c green lines indicate at the steady state situations. The plot in Figure 6.15a is dominated by the parameters induced due to the linearisation process around $\{x^{*1}, \partial_z x^{*1}\}$. This leads to parameter values which depend on the target state and its spatial derivative. They consequently evolve in space and time, cf. (5.4)ff. As already hinted above, for the latter of the two 3D plots one might identify the typical curves of the backstepping approach with parabolic PDEs, see [80]. Here the comparison to Figure 6.8a and Figure 6.8b of the first simulation study draws an interesting picture and extends the analysis from above. Although in the current study the tuning parameter $d^2 = 45/(\mathrm{N}-1)^2$ is three times higher in relative figures and the diffusion coefficient a is twice as big compared to value in the previous study the magnitudes of the gains remain in the same numerical decade. This can be explained because the current simulation scenario deals with 21 agents compared to 11 agents in the previous. As a consequence it can be approximately assumed that the swarm dynamics of the



(a) Distributed control gain respecting $\tilde{x}_{\mathrm{c}}^{1}$.

(b) Individual control gains respecting $\tilde{x}_{\mathrm{c}}^{1}$.

(c) Distributed control gain respecting $\tilde{x}_{\mathrm{c}}^{2}$.

(d) Individual control gains respecting $\tilde{x}_{\mathrm{c}}^{1}$.

Figure 6.15: Controller gains: (a) and (c) (Discretely) distributed control gain for each coordinate according to (6.6a) which amplifies the control error $\tilde{x}_{\mathrm{c}}^{m}$ of every agent; amplified errors are summed up to form a control input signal. (b) and (d) individual control gains for each coordinate according to (6.6b) which respect the control error of the anchor, the control error of the leader and the control error of the 'state difference' between the leader and its next follower. In the 3D-plots green lines (— , – –) indicate steady state situations in the sense of the applied motion planning procedure.

current situation is four times slower compared to the first simulation study (just by comparing the factor $(\mathrm{N}-1)^2$). With this, absolute figures of the controller parameters can be four times smaller in the current scenario compared to the previous one in order to make a equivalent comparison. This means, even though the gains are similar in absolute figures the swarm as a system experiences a much higher amplification of the associated error values. A more detailed and systematic approach on this topic such as the difference between swarm and agent dynamics is conduced in Section 2.2.

The upcoming two simulation studies deal with a coupled DRS and a coupled DCRS as the underlying model of the MAS dynamics. Just by design this leads to a rather complex setup and sophisticated analysis process in the next paragraphs.

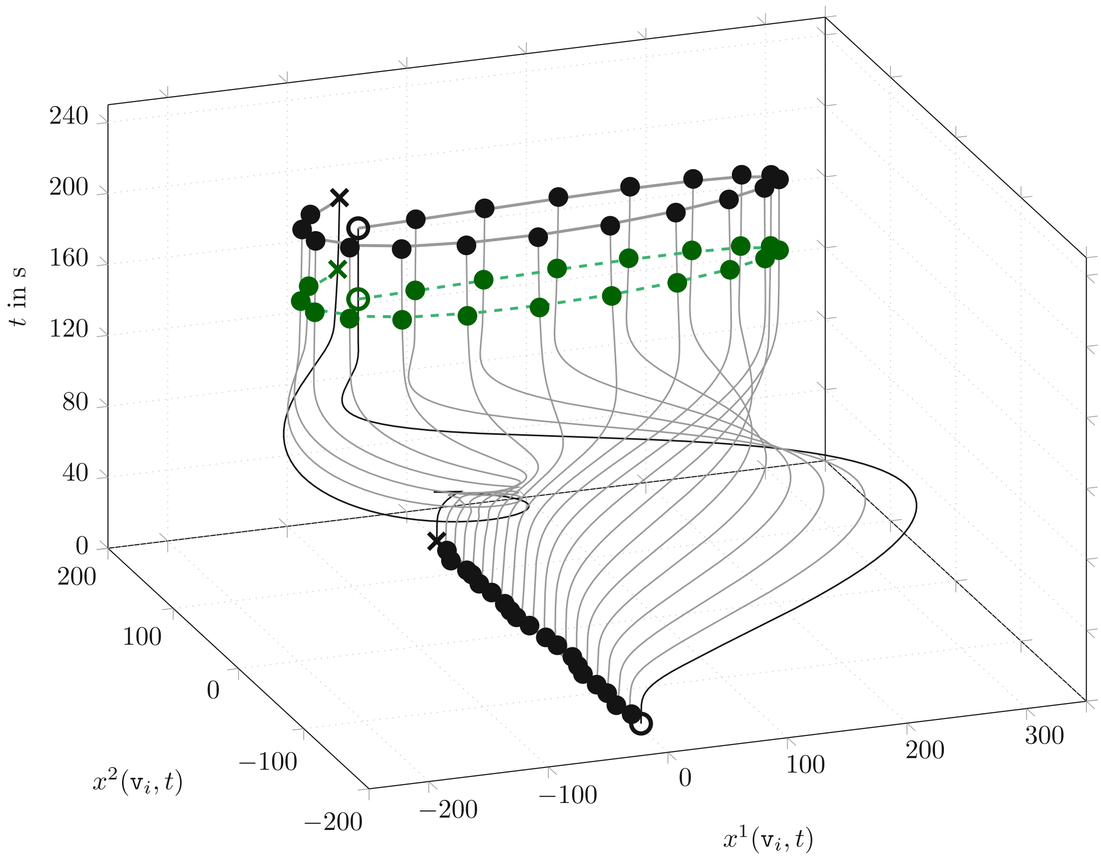


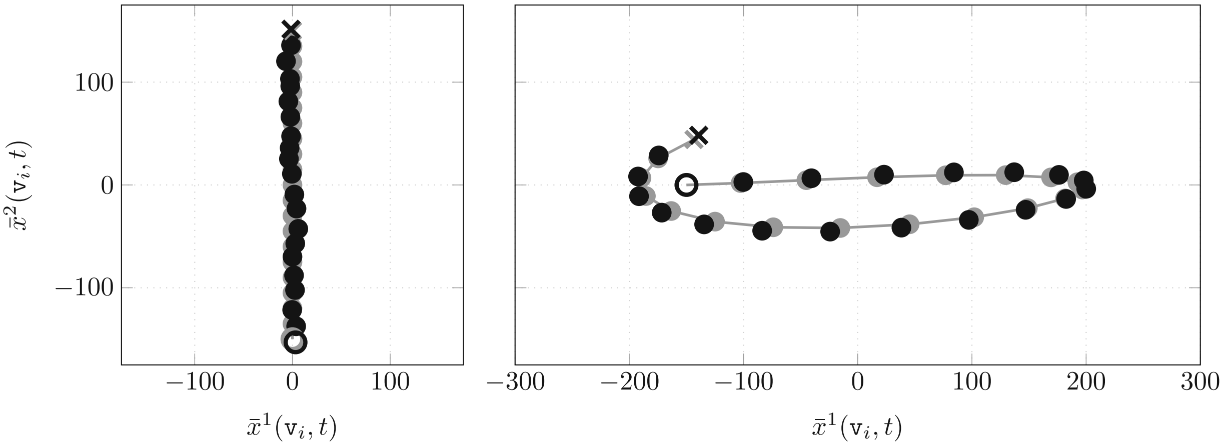
Figure 6.16: Simulation result of formation transitions for an MAS modelled as a coupled DRS. Simulation time is $T_{\mathrm{sim}} = 120\,\mathrm{s}$, control cycle time is set to $\tau_{\mathrm{c}} = 30\,\mathrm{ms}$, and the (x^1, x^2) coordinates are illustrated in cm. The anchor agent is indicated as ○ and the leader agent as ×. Followers are denoted as ●.

6.2.3 Simulation Results of a Coupled DRS Model

In the following a scenario is presented which uses a model dynamics described by a PDE of a coupled DRS. Figure 6.16 shows the simulated state vector $\boldsymbol{x}$ associated to $\mathrm{N} = 21$ agents. As with the other scenarios the starting configuration of the agents is a straight line, then they move to a rather complex steady state which has the shape of a kind of an *open ellipse*. The scenario has tracking control enabled which benefits from an observer-based estimation of all state variables. The transition lasts $\tau = 0.5(\mathrm{N} - 1)^2\,\mathrm{s} = 200\,\mathrm{s}$ however the entire simulation time is set to $T_{\mathrm{sim}} = 240\,\mathrm{s}$ in order to allow a statement about stability. The cycle time of the observer is set to 20 ms, but the cycle period of the controller is configured five times slower to challenge the robustness of the utilised backstepping approach for coupled PDEs. Again, an initial position error imitates real time behaviour. Moreover, Table 6.3 states the system parameters of the coupled DRS and the steady state values of the anchor and leader agent for $t = 0\,\mathrm{s}$ and $t = \tau$. Here it is worth to mention that the parameters are very similar to those of the first scenario where the agent move to a circular shape. The only difference is the number of agents, the modification of the diffusion parameter for coordinate 2 to $a^{22} = 1.2a^{11}$, and the

Table 6.3: Parameter values for the coupled DRS scenario with $\tau = 200\,\mathrm{s}$ and $h = 2\pi\ell_\mathrm{N}/(\ell_\mathrm{N}+1)$. Diffusion parameters are chosen as $a^{11} = 0.5$ and $a^{22} = 0.6$. Geometric values are given in cm.

	$t = 0\,\mathrm{s}$			$t = \tau$		
Coord.	$\bar{C}^*$	$\bar{\boldsymbol{x}}^*_\mathtt{a}$	$\bar{\boldsymbol{x}}^*_\mathtt{1}$	$\bar{C}^*$	$\bar{\boldsymbol{x}}^*_\mathtt{a}$	$\bar{\boldsymbol{x}}^*_\mathtt{1}$
$\begin{bmatrix} x^1 \\ x^2 \end{bmatrix}$:	$0_{2,2}$	$\begin{bmatrix} 0 \\ -150 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 150 \end{bmatrix}$	$\begin{bmatrix} a^{11} & \frac{1}{9}a^{11} \\ \frac{1}{9}a^{22} & a^{22} \end{bmatrix} 4(\pi/\ell_\mathrm{N})^2$	$\begin{bmatrix} -150 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -150\cos(h) \\ -150\sin(h) \end{bmatrix}$



(a) Starting line profile.

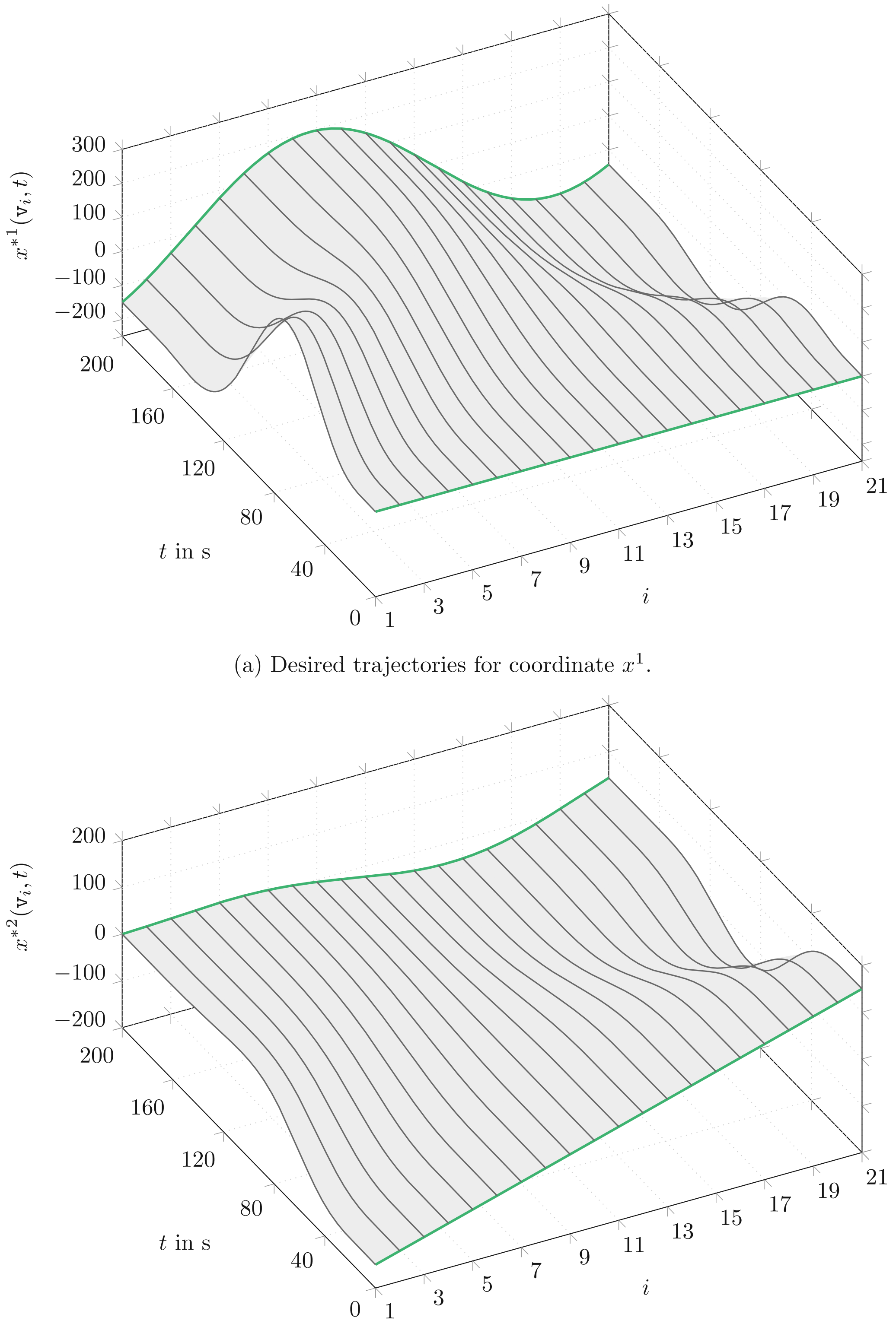
(b) Final shape.

Figure 6.17: Steady state profiles in the (x^1, x^2)-plane: (a) A line as starting deployment and (b) an *open ellipse* as final formation. Colour grey (●) stand for desired values and black symbols (●) for simulated values.

coupling of the reaction term which is expressed by the non-diagonal elements of $\bar{C}^*$ at $t = \tau$, i.e. $c^{12} = c^{11}/9$, $c^{21} = c^{22}/9$. These adaptions result in a completely different steady state formation profile compared to the uncoupled scenario. It is pictured in Figure 6.17 or more precisely speaking, the starting line with the initial error is shown in 6.17a and the final steady state is depicted in 6.17b.

Motion Planning and Feedforward Control

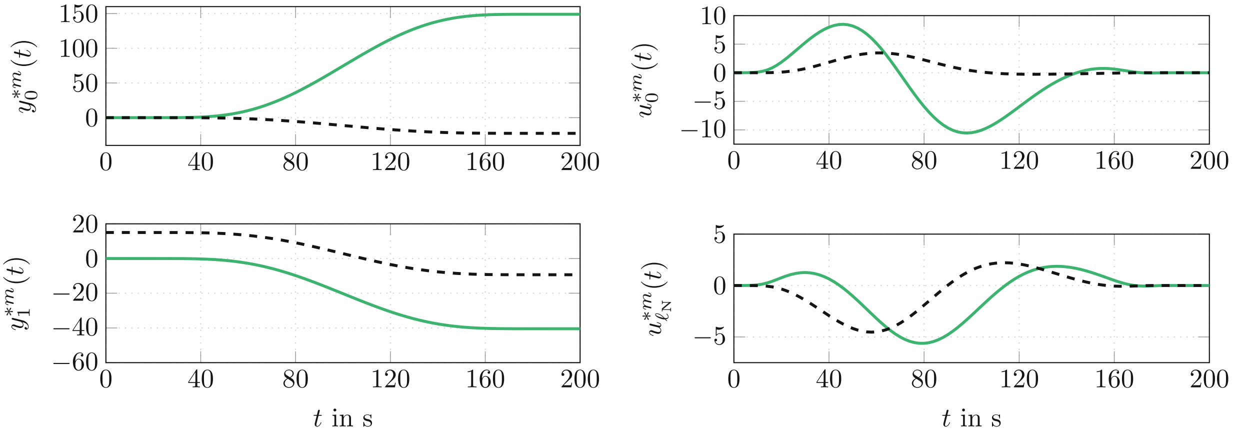
As in the equivalent sections above the next two figures give impressions to the applied motion planning and FFC. For this, Figure 6.18 shows the desired temporal evolution of the states $\boldsymbol{x}^*$ for both coordinates. As usual in this part of the analysis, the grey surface imitates the result in continuous description. Moreover, the plots in Figure 6.18a and Figure 6.18b show state trajectories which guide the MAS from a line configuration to a steady state deployment build upon sin− and cos−functions. In this context the applied flatness-based motion planning algorithm utilises the Gevrey functions Φ_{τ_y,ω_y} and Φ_{τ_C,ω_C} implemented by (C.13) with the parameter setting $\alpha_y = \alpha_C = 1.05$ in both coordinates. The timing parameters are



(a) Desired trajectories for coordinate x^1.

(b) Desired trajectories for coordinate x^2.

Figure 6.18: Desired trajectories $\boldsymbol{x}^*$ for the planning period $t \in [0, 200]\,\mathrm{s}$ computed by means of formal state parametrisation of a coupled DRS, see Section 4.1. The grey lines (—) show the nominal trajectories for 21 agents, the green lines (—) shall symbolise the steady states situations at $t = \bar{t} \in \{0\,\mathrm{s}, 200\,\mathrm{s}\}$ according to the continuous motion planning procedure.



(a) Trajectories of the flat outputs.

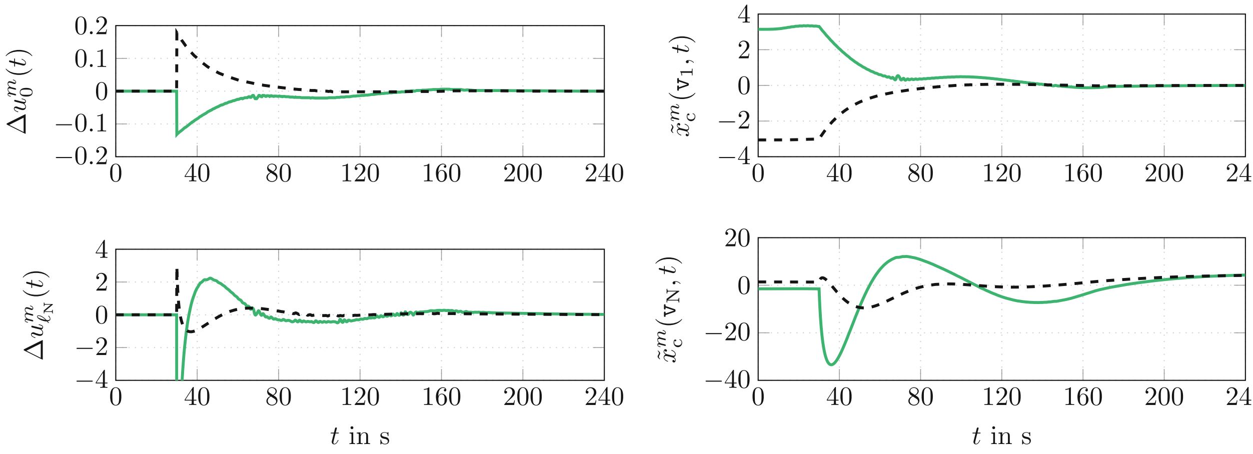
(b) Feed forward control inputs.

Figure 6.19: Desired and planned trajectories for the motion planning period $t \in [0, 200]\,$s: (a) Desired trajectories $\{y_0^{*m}, y_1^{*m}\}$ of the components $\boldsymbol{y}^m$ of the flat output; (b) planned FF control inputs $\{u_0^{*m}, u_{\ell_N}^{*m}\}$ for dynamic BCs. Green solid lines (—) stand for coordinate $m = 1$ and black dashed lines (– –) for coordinate $m = 2$.

equally set to $\tau_{\boldsymbol{y}} = \tau_C = 200\,$s. The interested reader is referred to Chapter 4 for details. For completeness the plots in the subsequent graphics show the computed desired trajectories $\boldsymbol{y}^{*m}(t) = [y_0^{*m}(t),\, y_1^{*m}(t)]^T$ of the flat output $\boldsymbol{y}$ in Figure 6.19a and results of the FF control inputs in Figure 6.19b.

Observer-based Tracking Control utilising Coupling Compensation

As stated above observer-based FBC is applied to the current scenario. For this the *coupling compensation* strategy allows to formulate scalar equations for the computation of the involved backstepping kernels. Figure 6.20 shows the resulting FB control inputs and the corresponding tracking errors of the leader and anchor agent, respectively. Obviously the magnitudes of the inputs and errors are similar compared with the previous scenarios. In other words, the entire concept preserves the limits of the actuator dynamics although the number of agents is changed and the coupled swarm dynamics is controlled by means of uncoupled backstepping kernel equations. Moreover, it has to be said that the feedback controller is firstly enabled after 30 s when the initial observer error is settled to a moderate level. For a deeper analysis different values considering tracking and observation errors of the complete swarm are depicted in Figure 6.21. Both, the individual distance error in Figure 6.21a and the discrete $\mathbf{L}^2-$ norm of the control error in Figure 6.21b give a picture which is comparable to the experienced results above. The peaks after $t \geq 30\,$s clearly originate from the self-induced initial position error, which challenges both, the controller and the observer. This can be seen in the remaining subplots as well. First, the discrete $\mathbf{L}^2-$ norm of the observer error is shown in Figure 6.21c, and second, Figure 6.21d illustrates the discrete $\mathbf{L}^2-$ norm of the estimated control error. It is important to keep in mind that the individual estimated control errors are the values which



(a) Tracking control inputs.

(b) Control error of the cognitive agents.

Figure 6.20: Feedback control inputs and errors for the entire simulation period $t \in [0, 240]$ s: (a) Tracking or feedback control inputs $\Delta u_0^m(t)$ of the anchor and $\Delta u_{\ell_{\mathrm{N}}}^m(t)$ of the leader; (b) coordinate-wise position errors $\tilde{x}_{\mathrm{c}}^m(\mathrm{v}_1, t)$ of the anchor and $\tilde{x}_{\mathrm{c}}^m(\mathrm{v}_{\mathrm{N}}, t)$ of the leader. Again, green solid lines (—) refer to coordinate $m = 1$ and black dashed lines (--) to coordinate $m = 2$.

are applied to the actual control algorithm.

Now, for the design and configuration of the backstepping-based FBC with *coupling compensation* one can rely on the sufficient inequality (5.94) which ensures exponential stability in the norm induced by (5.93) and in case of dynamic boundary conditions. For this particular scenario the inequality can be written as

$$\boldsymbol{\nu}^T D(t)\boldsymbol{\nu} \geq \boldsymbol{\nu}^T\left(M^{\mathrm{s}}(z,t)+\varphi_{\mathrm{H}}^2(t)\left(1+\varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2 {\ell_{\mathrm{N}}}^4 A^{-1}\right)\boldsymbol{\nu}+\lambda_{\min}\left\|\boldsymbol{\nu}\right\|_2^2\,, \tag{6.10}$$

since $M^{\mathrm{s}}(z,t) := M^{\mathrm{s}}(t) = \frac{1}{2}\left(C^{\mathrm{c}}(t) + (C^{\mathrm{c}})^T(t)\right)$ is independent of z and $B(z,t) = 0_{n,n}$. Moreover, note that $H(z,s,t) = C^{\mathrm{c}}(t)K^{\mathrm{d}}(z,s,t) - K^{\mathrm{d}}(z,s,t)C^{\mathrm{c}}(t)$ and consequently the matrix

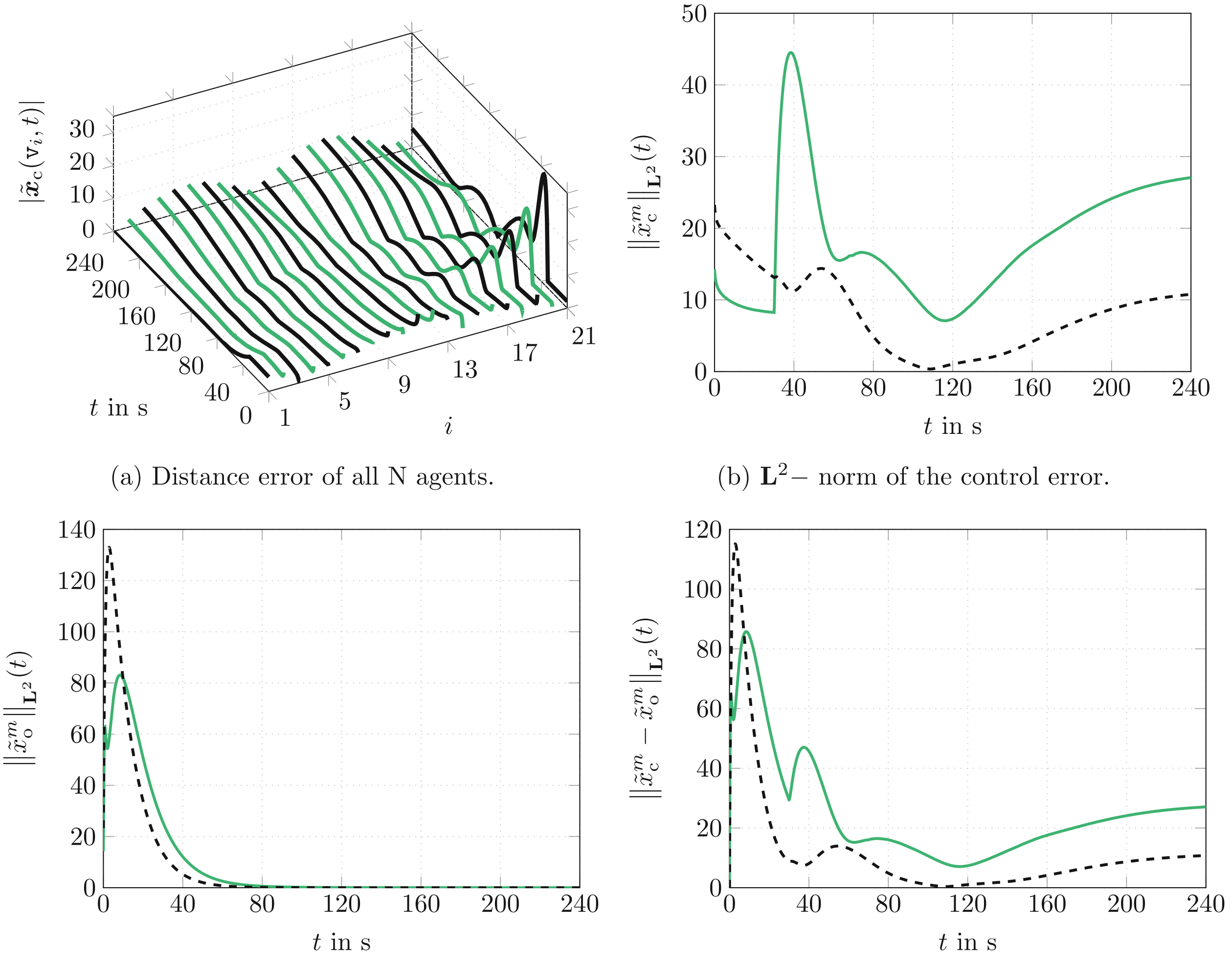
$$H^T(z,s,t)H(z,s,t) = \begin{bmatrix} \left(c^{12}(t)\right)^2 & 0 \\ 0 & \left(c^{21}(t)\right)^2 \end{bmatrix} \left(k^{11}(z,s,t) - k^{22}(z,s,t)\right)^2 \tag{6.11}$$

is diagonal and positive semi-definite for all $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \overline{\mathbb{R}}_{t_0}^+$. With this and (C.10) the upper bound $\varphi_{\mathrm{H}}(t)$ can be identified as

$$\begin{aligned}\varphi_{\mathrm{H}}(t) = \max\nolimits_{\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})}\left\|H\right\|_2(t) &= \sqrt{\max_{(z,s)\in\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})} \lambda\left(H^T(z,s,t)H(z,s,t)\right)} \\ &= \max\{\left|c_{12}(t)\right|, \left|c_{21}(t)\right|\} \times \max_{(z,s)\in\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})} \left|k^{11}(z,s,t) - k^{22}(z,s,t)\right|\,.\end{aligned} \tag{6.12}$$

Now, considering (5.45a) obviously the setting

$$\frac{1}{a^{11}}\left(d^{11}(t) + c^{11}(t)\right) = \frac{1}{a^{22}}\left(d^{22}(t) + c^{22}(t)\right) = \text{const.} \tag{6.13}$$



(a) Distance error of all N agents.

(b) $\mathbf{L}^2-$ norm of the control error.

(c) $\mathbf{L}^2-$ norm of the observer error.

(d) $\mathbf{L}^2-$ norm of the estimated control error.

Figure 6.21: Absolute position errors and component-wise error norms in cm according to (6.9): (a) Temporal evolution of the agent's position error $|\tilde{\boldsymbol{x}}_{\mathrm{c}}(\mathrm{v}_i,t)| = \|\boldsymbol{x}(\mathrm{v}_i,t) - \boldsymbol{x}^*(\mathrm{v}_i,t)\|_2$ as euclidean distance in the $(x^1,x^2)-$plane; (b) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}_{\mathrm{c}}^m(\mathrm{v}_i,t)$ between simulated state and desired state; (c) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}_{\mathrm{o}}^m(\mathrm{v}_i,t)$ between simulated state and estimated state gained from the Luenberger-type observer; (d) discrete $\mathbf{L}^2-$norm of the error between the estimated observer state and the desired state. The $\mathbf{L}^2-$ norms are computed for each coordinate $m = \{1,2\}$ individually. For these, green solid lines (—) refer to the norms related to coordinate 1 and black dashed lines (– –) to coordinate 2.

leads to equal and time-invariant kernel elements $k^{11}(z,s) = k^{22}(z,s)$ and consequently to $\varphi_{\mathrm{H}}(t) = 0$. However, considering the parameter setting in Table 6.3 the coupling part of the coefficient matrix C obviously fulfils $\boldsymbol{\nu}^T C^{\mathrm{c}}(t)\boldsymbol{\nu} \leq \boldsymbol{\nu}^T C^{\mathrm{c}*}(\tau)\boldsymbol{\nu}$ and therefore $t = \tau$ is considered as the *worst case setting*. This motivates a design with time-invariant matrix D, i.e. $(d^{11} + c^{11}(\tau))/a^{11} = (d^{22} + c^{22}(\tau))/a^{22}$ which results in $\varphi_{\mathrm{H}}(t \geq \tau) = 0$. Neglecting dynamic effects, i.e. $k^{mm} \gg \frac{\mathrm{d}}{\mathrm{d}t}k^{mm}$ for $t \in (0,\tau)$, the inequality (6.10) simplifies for design purposes to

$$d_{\min} \geq \lambda_{\max}(C^{\mathrm{c}*}(\tau)) + \lambda_{\min}(t) \tag{6.14}$$

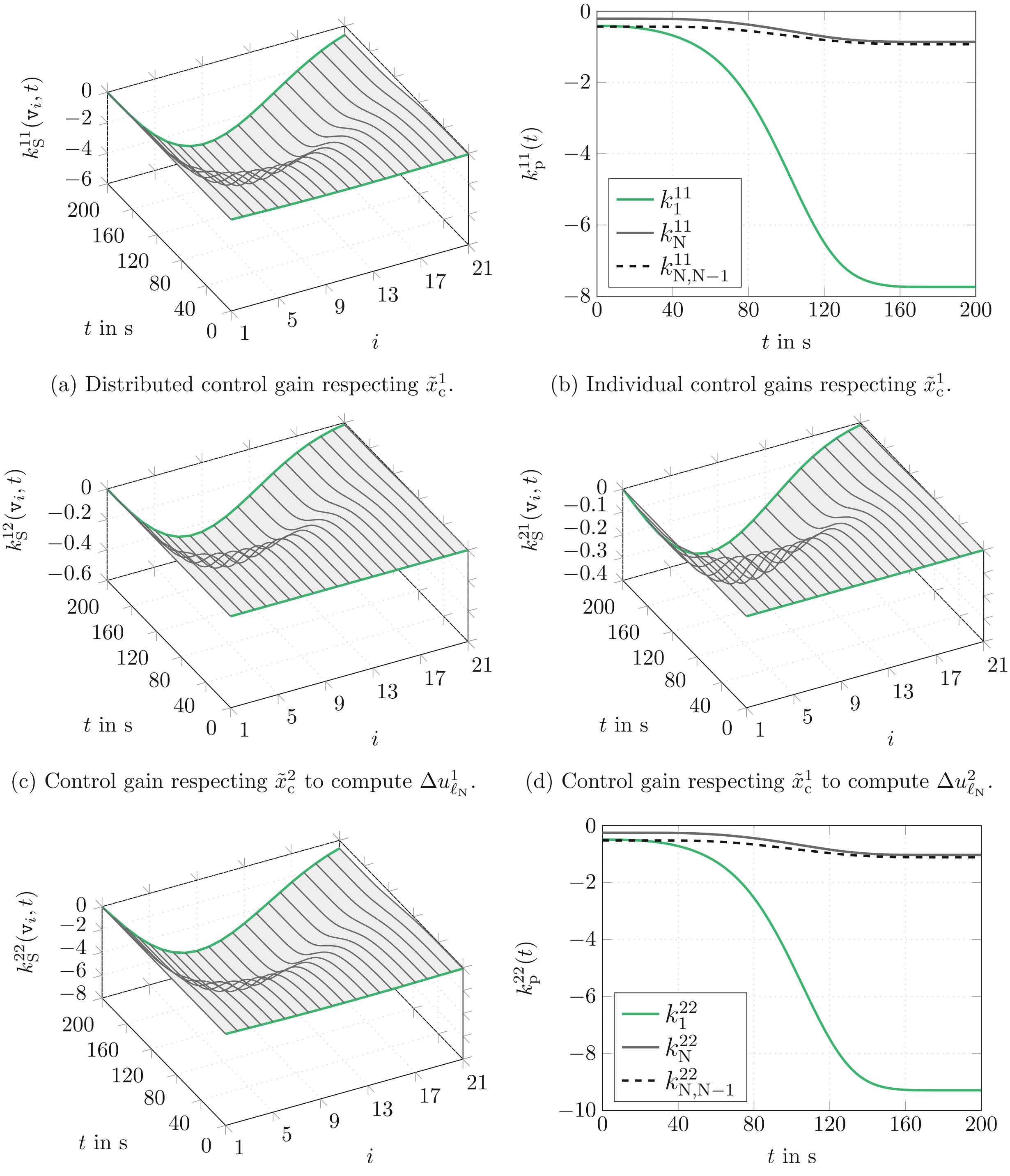
and ensures exponential stability for $t \geq \tau$. Stability check during the transition is done numerically in simulation since estimations like $|k^{mm}| \leq \rho \exp(2\rho z)$, for some $\rho > 0$, as stated in [80] are not appropriate for this case. These inequalities prove the boundedness of the kernel but they do not make any real statement of the maximum absolute value of the kernel. Therefore the numerical setting $D = \operatorname{diag}[0.0436, 0.0524]$ ensures $\lambda_{\min} \geq 15/(\mathrm{N}-1)^2 = 0.0375$ for $t \geq \tau$. The computed controller gains for this scenario are shown in Figure 6.22. Due to the increased number of agents all magnitudes are at a moderate level. Comparing the plots in Figure 6.22a and Figure 6.22b which only amplifies errors of coordinate 1 with Figure 6.22e and Figure 6.22f, where only errors of coordinate 2 are taken into account, the values of the latter two subplots are slightly bigger. Since the stationary values of the kernels k^{11} and k^{22} are equal by design these deviations origin from the different system parameter setting which is taken into account for the computation of the controller gains (6.5). Furthermore, this is the reason why the control gains plotted in Figure 6.22c and Figure 6.22d are not identical to zero. These introduce a cross-coupling between the tracking errors of one coordinate to the control input signal of the *other* coordinate. More precisely speaking, the gain values in Figure 6.22c amplify the tracking control error $\tilde{x}_{\mathrm{c}}^2(\mathrm{v}_i, t)$ and adds it to the computation of the input signal $\Delta u_{\ell_{\mathrm{N}}}^1(t)$, and the same applies to the gain values shown in Figure 6.22d.

With this, the same design procedure as above applies for the configuration of the observer setting. Referring to (5.164) and $\varphi_{\mathrm{P}}(t \geq \tau) = 0$ with $(e^{11} + c^{11}(\tau))/a^{11} = (e^{22} + c^{22}(\tau))/a^{22}$ the simplified inequality can be written as

$$e_{\min} \geq \mu_{\max}(C^{\mathrm{c}*}(\tau)) + \mu_{\min}(t) \tag{6.15}$$

and guarantees exponential stability for the observer when the final steady state is reached at $t = \tau$. Especially the setting $E = \operatorname{diag}[0.0811, 0.0974]$ results in $\mu_{\min} \geq 0.0750$ for $t \geq \tau$. Results of the observer gains are presented in Figure 6.23. Again, due to the different parameter system setting of the diffusion term one can observe increased magnitudes in the gains of coordinate 2 which are printed in the subplots Figure 6.23c and Figure 6.23d compared to those of coordinate 2 illustrated in Figure 6.23a and Figure 6.23b. However, here no *cross-coupling* distributed gains are applicable since (5.143) does not involve the system parameter matrix C which is responsible for the coupling phenomenon in this scenario. Overall the values remain at a similar level as already shown in the first scenario with state estimation.

This closes the third simulation scenario. The next simulation study is based on the current one but adds more system complexity. Apart from that the *non-commutative compensation* strategy replaces the coupling compensation procedure utilised in this section.



(a) Distributed control gain respecting $\tilde{x}_\mathrm{c}^1$.

(b) Individual control gains respecting $\tilde{x}_\mathrm{c}^1$.

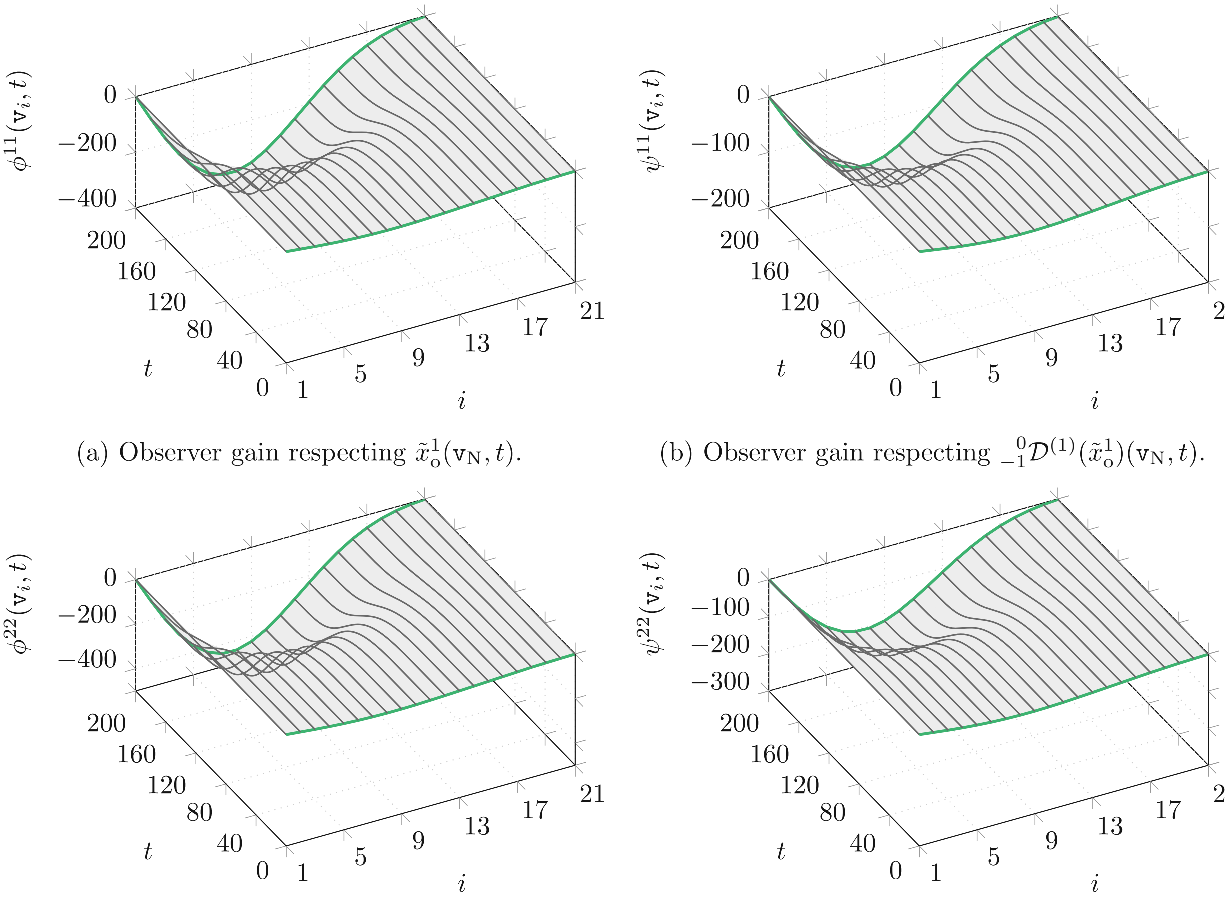
(c) Control gain respecting $\tilde{x}_\mathrm{c}^2$ to compute $\Delta u_{\ell_\mathrm{N}}^1$.

(d) Control gain respecting $\tilde{x}_\mathrm{c}^1$ to compute $\Delta u_{\ell_\mathrm{N}}^2$.

(e) Distributed control gain respecting $\tilde{x}_\mathrm{c}^2$.

(f) Individual control gains respecting $\tilde{x}_\mathrm{c}^2$.

Figure 6.22: Controller gains: (a) (Discretely) distributed control gain and (b) individual control gains, they respect control errors of coordinate 1 in order to compute the control signal $\Delta u_{\{0,\ell_\mathrm{N}\}}^1$; (c) and (d) are distributed control gains which consider control errors from the *other* coordinate; (e) distributed control gain and (f) individual control gains, both respecting $\tilde{x}_\mathrm{c}^2$ to provide $\Delta u_{\{0,\ell_\mathrm{N}\}}^2$.



(a) Observer gain respecting $\tilde{x}^1_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)$.

(b) Observer gain respecting ${}^{\;0}_{-1}\mathcal{D}^{(1)}(\tilde{x}^1_{\mathrm{o}})(\mathtt{v}_{\mathrm{N}},t)$.

(c) Observer gain respecting $\tilde{x}^2_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)$.

(d) Observer gain respecting ${}^{\;0}_{-1}\mathcal{D}^{(1)}(\tilde{x}^2_{\mathrm{o}})(\mathtt{v}_{\mathrm{N}},t)$.

Figure 6.23: Observer gains: (a) (Discretely) distributed observer gain which amplifies the observer error of the leader agent $\tilde{x}^1_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)$ and (b) respects the observer error difference $(\tilde{x}^1_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)-\tilde{x}^1_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}-1},t))$ between the leader agent and its next neighbour; (c) (discretely) distributed observer gain which considers $\tilde{x}^2_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)$, and finally (d) takes $(\tilde{x}^2_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}},t)-\tilde{x}^2_{\mathrm{o}}(\mathtt{v}_{\mathrm{N}+1},t))$ into account.

6.2.4 Simulation Results of a Coupled DCRS Model

The last simulation scenario is similar to the previous one, though it adds additional coupling in form of a time-variant convection term and the number of agents is reduced to $\mathrm{N}=11$. Therefore the matrix B is configured at steady states as

$$\begin{aligned} B^*(0) &= 0_{2,2}\,, \\ B^*(\tau) &= \begin{bmatrix} 0 & -\frac{1}{3\ell_{\mathrm{N}}} \\ -\frac{1}{3\ell_{\mathrm{N}}} & 0 \end{bmatrix}\,. \end{aligned} \tag{6.16}$$

Equivalently to the design for C the matrix-valued parameters are temporally connected by means of a the Gevrey function (C.13), i.e. $B(t)=B^*(0)+\Phi_{\tau_B,\omega_B}\left(B^*(\tau)-B^*(0)\right)$. The other system parameters remain the same as the set-up before and are outlined in Table 6.3. Moreover feedback control follows the *non-commutative compensation* strategy instead of *coupling compensation* and it forgoes state observation but uses measurements of the states. With this,

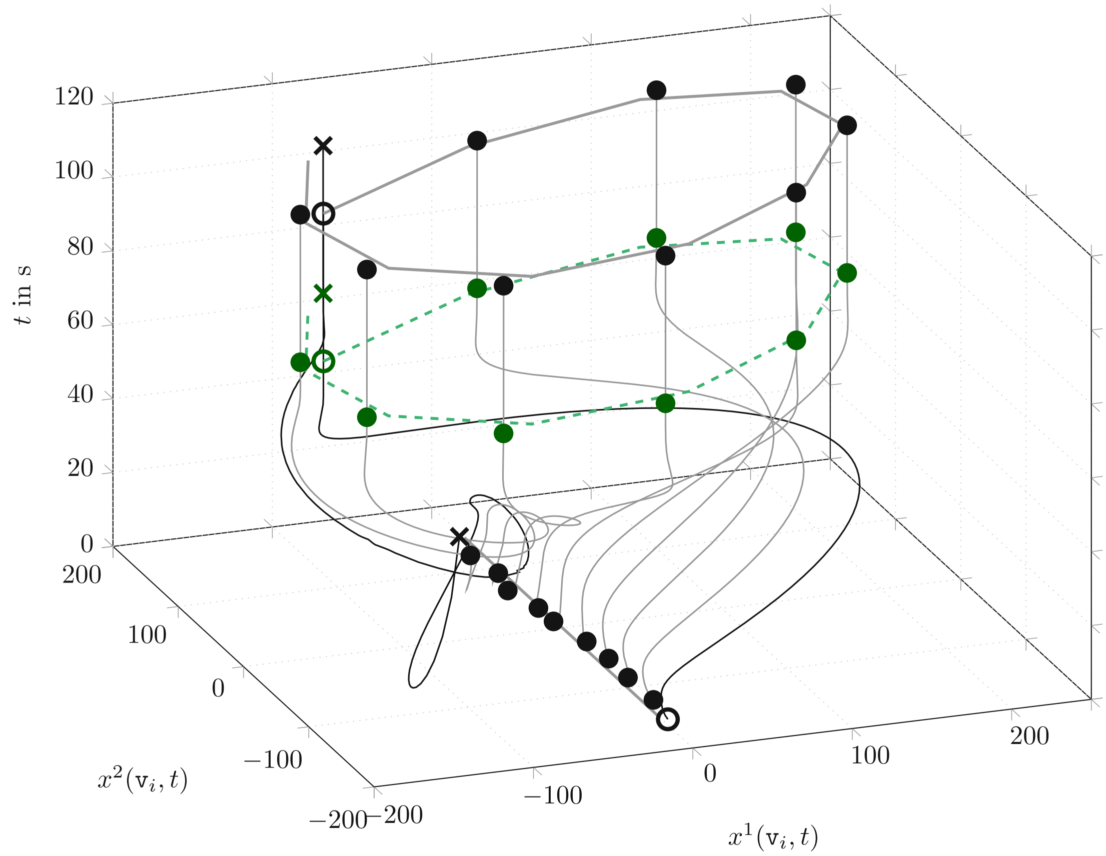
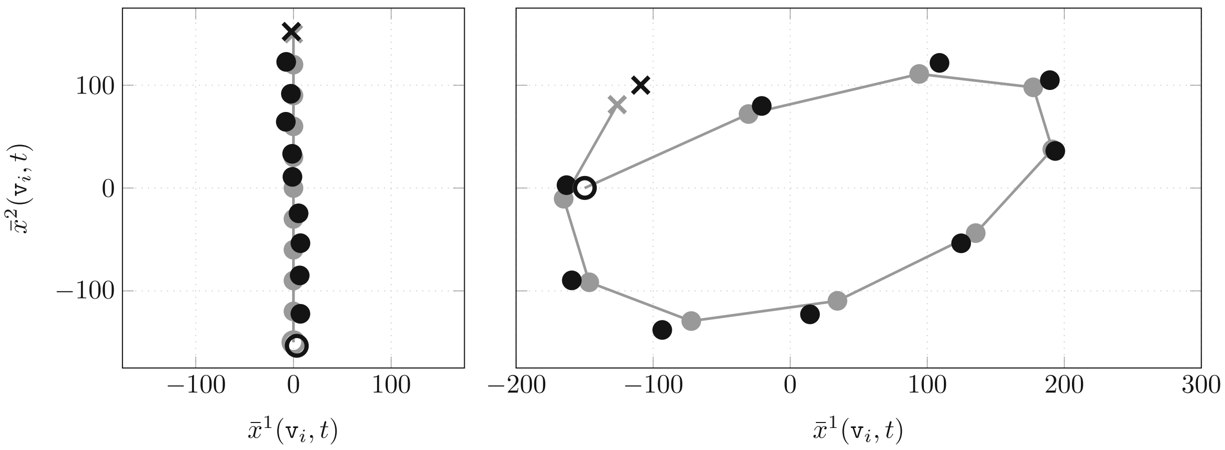


Figure 6.24: Simulation result of formation transitions for an MAS modelled as a coupled DCRS. Simulation time is $T_{\mathrm{sim}} = 120\,\mathrm{s}$, control cycle time is set to $\tau_{\mathrm{c}} = 30\,\mathrm{ms}$, and the (x^1, x^2) coordinates are illustrated in cm. Again, the symbol ○ denotes the anchor agent, × the leader agent and followers are indicated as ●.

Figure 6.24 shows the simulations results. It can be observed that the applied feedback control strategy is able to stabilise the transition between the two configured steady states. Moreover, comparing the current final profile pictured in Figure 6.25 and the previous one in Figure 6.17 it is clear that the additional coupling by means of the convection term together with the reduced number of agents leads to a stretched or clinched profile in the directions x^1 and x^2. Interestingly convection terms lead to similar examinations in context with uncoupled systems, see Remark 3.13 for details. However, recalling (3.50) and the associated discussion for coupled systems in Appendix B.1 the length of the domain always influences the complete shape of the steady states in all directions. Since $\ell_{\mathrm{N}} = \mathrm{N} - 1$ the reduced number of agents changes the image of the final profile as well.

Motion Planning and Feedforward Control

The following two figures picture important results regarding the FF part of the 2DOF controller. Particularly Figure 6.26a and Figure 6.26b show the desired trajectories of the corresponding state from a linear deployment to some combination of sin− and cos−functions.



(a) Starting line profile.

(b) Final shape.

Figure 6.25: Steady state profiles in the (x^1, x^2)-plane: (a) A line as starting deployment and (b) an *open ellipse* as final formation. Colour grey (●) denotes desired values and black symbols (●) stand for simulated values.

The choice of the Gevrey functions for the flat output $\varPhi_{\tau_{\boldsymbol{y}},\omega_{\boldsymbol{y}}}$, and the parameters $\varPhi_{\tau_B,\omega_B}$ and, especially for this case, $\varPhi_{\tau_C,\omega_C}$ remain as in the preceding scenario. The same applies to the parameter setting $\alpha_{\boldsymbol{y}} = \alpha_B = \alpha_C = 1.05$ and $\tau_{\boldsymbol{y}} = \tau_B = \tau_C = 80\,\mathrm{s}$. However, here it has to be pointed out that in this case the complete algorithm (4.18) with both integrals is required for the computation of the desired trajectories. Therefore it is fair to say that especially this scenario verifies the flatness-based motion planning approach which is developed in Chapter 4 for coupled parabolic PDEs. For this, the plots in Figure 6.27a show the corresponding desired trajectories $\boldsymbol{y}^m(t) = [y_0^m(t),\, y_1^m(t)]^T$ of the flat output $\boldsymbol{y}$ and Figure 6.27b pictures the associated FF control inputs.

Measurement-based Tracking Control utilising Non-Commutative Compensation

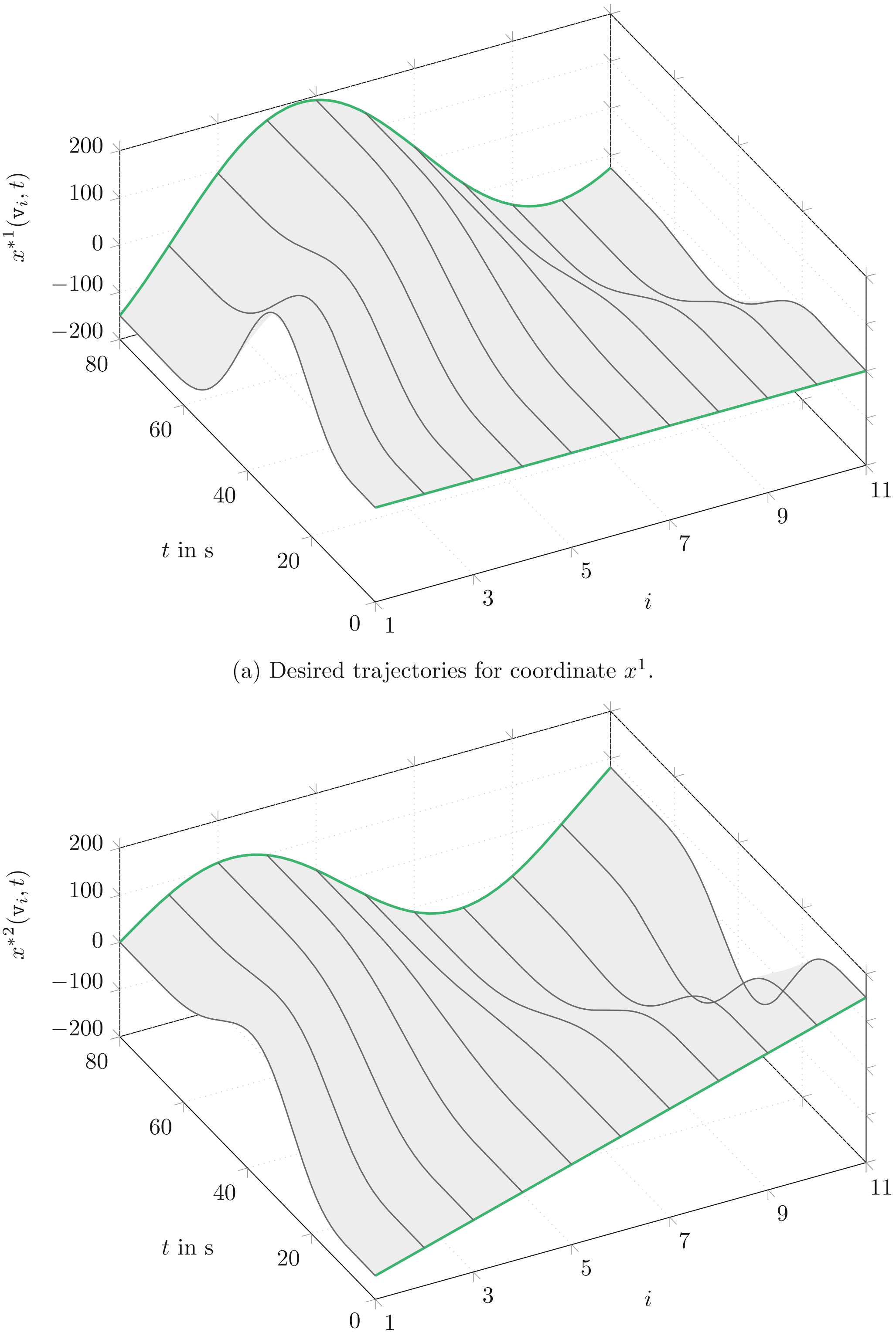
The presence of the convection term as well as the *non-commutative compensation* strategy requires M to be set to

$$\begin{aligned} M(z,t) &= 2\,(A - aI_2)\,\frac{\mathrm{d}}{\mathrm{d}z}K^{\mathrm{d}}(z,z,t) \\ &\quad + \big(K^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(t) - B^{\mathrm{c}}(t)K^{\mathrm{d}}(z,z,t)\big) + C^{\mathrm{c}}(t)\,. \end{aligned} \tag{6.17}$$

Note, since B^* is chosen to be symmetric and taking Remark 5.17 into account the term

$$K^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(t) - B^{\mathrm{c}}(t)K^{\mathrm{d}}(z,z,t) = -\frac{1}{3\ell_{\mathrm{N}}}\left(k^{11}(z,z,t) - k^{22}(z,z,t)\right)\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{6.18}$$

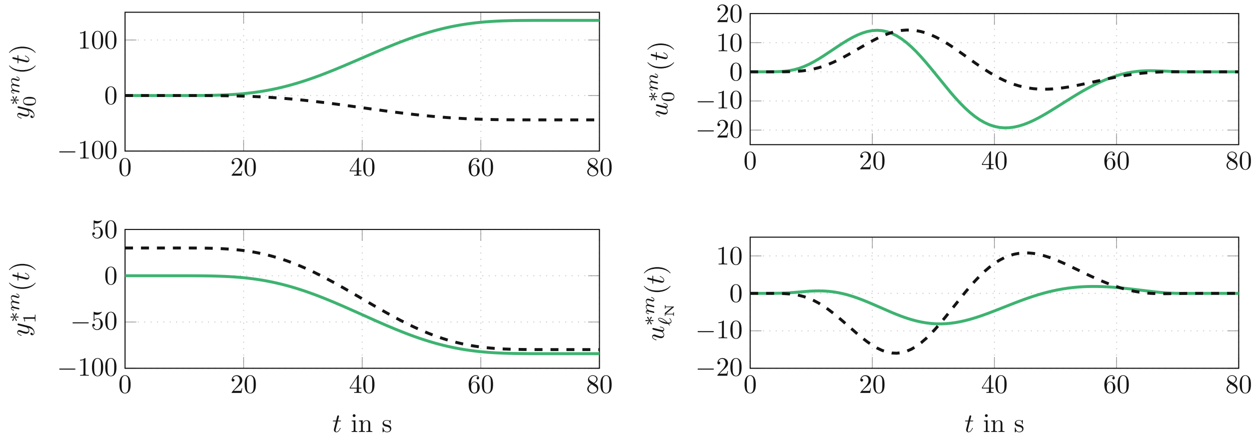
is certainly antisymmetric for this scenario. Referring to (5.51) and applying the condition $d^{11}(t) + c^{11}(t) = d^{22}(t) + c^{22}(t)$ for the design matrix $D(t)$ then even the entire part induced



(a) Desired trajectories for coordinate x^1.

(b) Desired trajectories for coordinate x^2.

Figure 6.26: Desired trajectories $\boldsymbol{x}^*$ for the planning period $t \in [0, 80]\,\mathrm{s}$ computed by means of formal state parametrisation of a coupled DRS, see Section 4.1. The grey lines (—) show the nominal trajectories for 11 agents, the green lines (—) shall symbolise the steady states situations at $t = \bar{t} = \{0\,\mathrm{s}, 80\,\mathrm{s}\}$ according to the applied motion planning procedure.



(a) Trajectories of the flat outputs.

(b) Feed forward control inputs.

Figure 6.27: Desired and planned trajectories for the motion planning period $t \in [0, 80]$ s: (a) Desired trajectories $\{y_0^{*m}, y_1^{*m}\}$ of the components $\boldsymbol{y}^m$ of the flat output; (b) planned FF control inputs $\{u_0^{*m}, u_{\ell_{\mathrm{N}}}^{*m}\}$ for dynamic BCs. Green solid lines (—) stand for coordinate $m = 1$ and black dashed lines (--) for coordinate $m = 2$.

by convection completely disappears. Now, with setting $a := \min\{a^{11},\, a^{22}\}$ the compensation matrix $M(z,t)$ is simplified to

$$M(z,t) := M(t) = -\left(A/a_{\min} - I_2\right)\big(D(t) + C^{\mathrm{d}}(t)\big) + C^{\mathrm{c}}(t) \tag{6.19}$$

whereas the first term, i.e. the diagonal part, is clearly negative semi-definite due to the choice of $a = a_{\min}$. Moreover, obviously it holds $\partial_z M(z,t) = 0_{2,2}$.

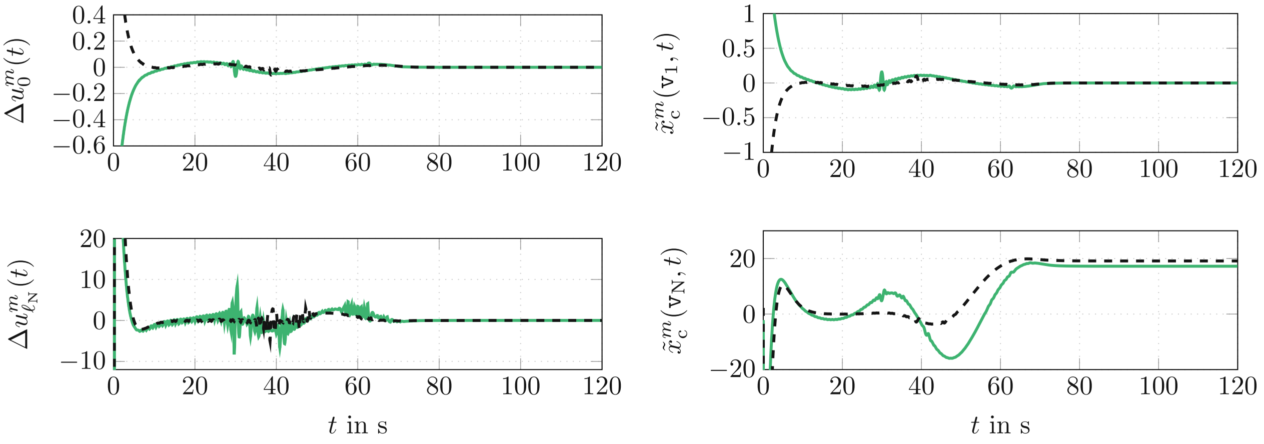
Now paying attention to the compensation kernel H and taking the current analysis into account, then (5.47) can be written as

$$\begin{aligned} H(z,s,t) = \Delta A \partial_z^2 K(z,s,t) - \partial_s^2 K(z,s,t)\Delta A - \Delta B(t)\partial_z K(z,s,t) \\ - \partial_s K(z,s,t)\Delta B(t) + 2\Delta A \frac{\mathrm{d}}{\mathrm{d}z} K^{\mathrm{d}}(z,z,t) K(z,s,t)\,. \end{aligned} \tag{6.20}$$

For this no other obvious assumption may be introduced apart from the setting $\Delta B(t) = B(t)$ since B has zero diagonal. However, similarly to the previous scenario the overall parameter configuration at $t = \tau$ is considered as the *worst case setting*. Thus, again the design matrix is configured with constant diagonal elements which fulfil $d^{11} + c^{11}(\tau) = d^{22} + c^{22}(\tau)$ and consequently $\partial_z M(z,t) = 0_{2,2}$ for $t \geq \tau$. Then with $\varphi_{\mathrm{B}}(t) = \max_{[0,\ell_{\mathrm{N}}]} \big\| B(z,t) \big\|_2$ the conditional inequality (5.94) for design purposes may be simplified to

$$d_{\min} \geq \lambda_{\max}\left(M^{\mathrm{s}}(\tau)\right) + \frac{3}{4a_{\min}} \varphi_{\mathrm{B}}^2(\tau) + \varphi_{\mathrm{H}}^2(\tau)\left(1 + \varphi_{\mathrm{G}}(t)\frac{\ell_{\mathrm{N}}}{\sqrt{2}}\right)^2 \frac{3{\ell_{\mathrm{N}}}^4}{2a_{\min}} + \lambda_{\min}(t)\,. \tag{6.21}$$

Since (5.94) can not be solved directly and it is (only) sufficient but it does not not say anything about necessity the configuration for this scenario uses a kind of *trial and error* approach. More



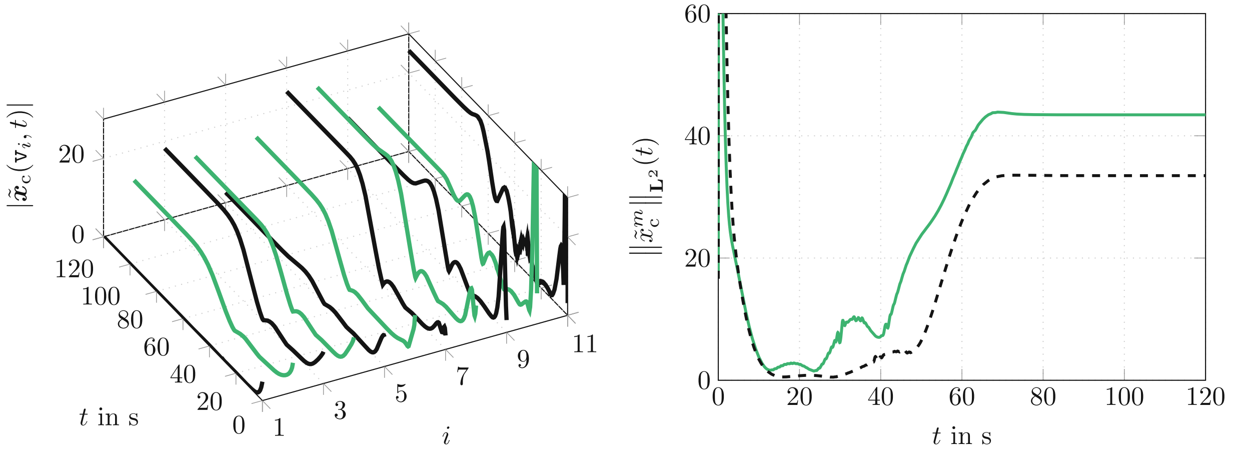
(a) Tracking control inputs.

(b) Control error of the active agents.

Figure 6.28: Feedback control inputs and errors for the entire simulation period $t \in [0, 120]$ s: (a) Tracking or feedback control inputs $\Delta u_0^m(t)$ of the anchor and $\Delta u_{\ell_N}^m(t)$ of the leader; (b) coordinate-wise position errors $\tilde{x}_c^m(v_1, t)$ of the anchor and $\tilde{x}_c^m(v_N, t)$ of the leader. As mentioned above, green solid lines (—) refer to coordinate $m = 1$ and black dashed lines (– –) to coordinate $m = 2$.

precisely the setting $d_{\min} := 0.40 + 3\varphi_B^2(\tau)/(4a_{\min})$ obviously leads to positive convergence rate for this scenario and stabilises the tracking error of the transition. The relatively high setting compared to the previous scenarios may explain the oscillating feedback control inputs of the leader agent illustrated in Figure 6.28a. Apart from that it can be said that the randomly induced initial error challenges the control algorithm most in terms of absolute figures. Since this is of less importance for the analysis Figure 6.28 cuts off the initial values. Moreover, since the starting profile, i.e., the line configuration, is stable a time varying gain with $D(0) = 0_{2,2}$ would allow to reduce the control value at the beginning of the simulation. Furthermore Figure 6.28a indicates the exponential stability of the anchor agent. The control error of the leader agent remains in the range as experienced with the previous studies. The overall control performance can be assessed by looking at Figure 6.29. Pondering over the reduced number of agents, the relatively high setting of the design parameter, the P-control nature of backstepping, and the additional coupling due to convection it is an acceptable result. On the one hand the remaining control error would benefit from higher design parameters. However, on the other hand control input signals may tend even more oscillations.

For completeness the next two figures show all involved controller gains. Figure 6.30 summaries the gains which are responsible to determine the feedback control signals $\Delta u_{\{0,\ell_N\}}^1(t)$. Recalling (6.5b) the computation of the feedback control inputs can separated into two parts, one considering the control errors of x^1 and the other the control errors of x^2. With this, obviously the controller gains plotted in the Figures 6.30a and 6.30b amplify the control errors of coordinate 1 and the gains in Figure 6.30c and Figure 6.30d refer to $\tilde{x}_c^2$. This can be equivalently analysed for Figure 6.31. The determination of the feedback signals $\Delta u_{\{0,\ell_N\}}^2(t)$ is a combination of amplifying $\tilde{x}_c^1$ by gains illustrated in Figure 6.31a and Figure 6.31b and taking the control error of



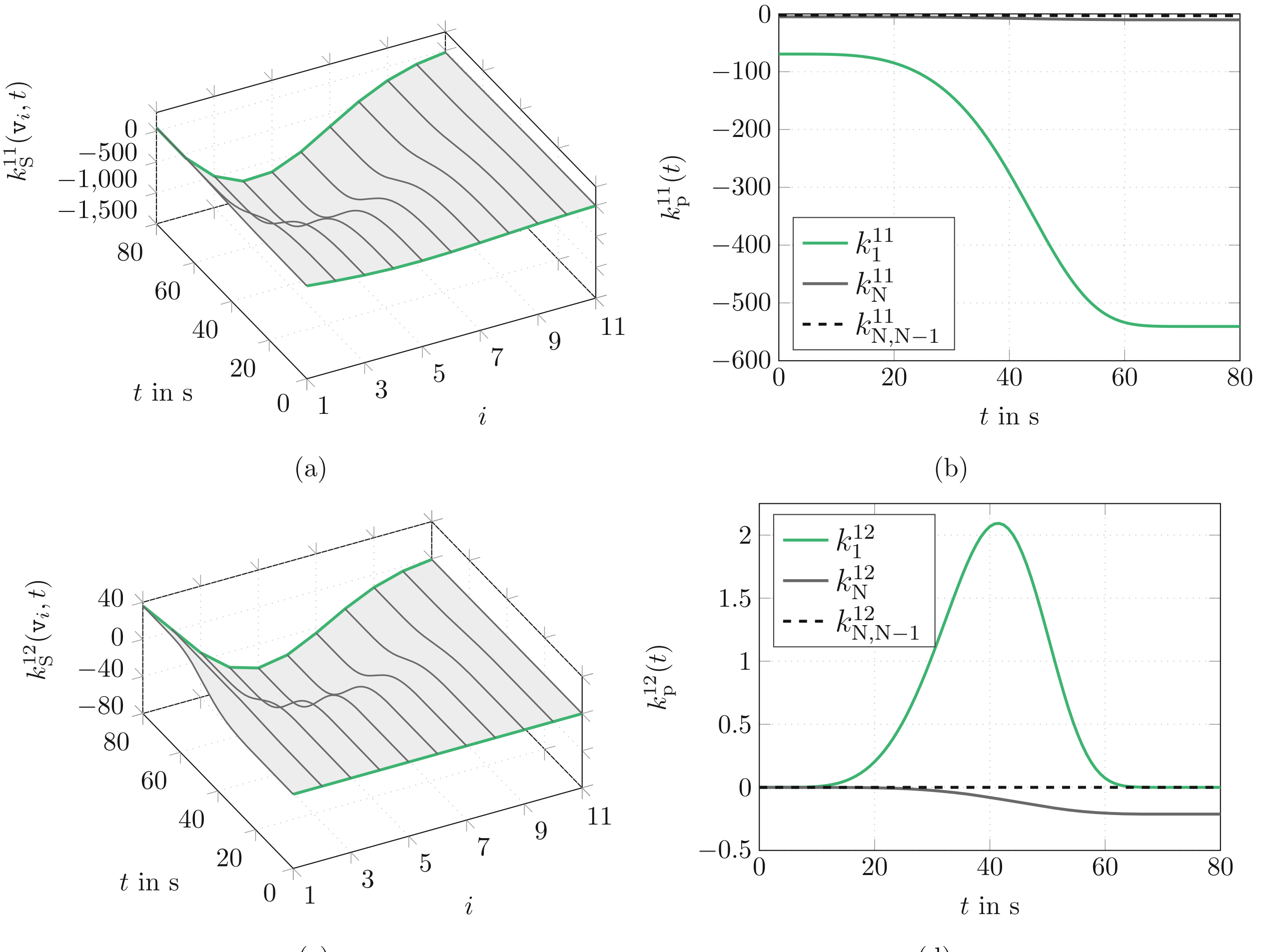
(a) Distance error of all N agents.

(b) $\mathbf{L}^2-$ norm of the control error.

Figure 6.29: Absolute position errors and component-wise error norms in cm according to (6.9): (a) Temporal evolution of the agent's position error $|\tilde{\boldsymbol{x}}_{\mathrm{c}}(\mathrm{v}_i, t)| = \|\boldsymbol{x}(\mathrm{v}_i, t) - \boldsymbol{x}^*(\mathrm{v}_i, t)\|_2$ as euclidean distance in the $(x^1, x^2)-$plane; (b) discrete $\mathbf{L}^2-$norm of the error $\tilde{x}_{\mathrm{c}}^m(\mathrm{v}_i, t)$ between simulated state and desired state. The $\mathbf{L}^2-$ norm is computed for each coordinate $m = \{1, 2\}$ individually. For this, the green solid line (—) refers to coordinate 1 and the black dashed line (– –) to coordinate 2.

coordinate 2 into account by means of the gains plotted in Figure 6.31c and Figure 6.31d. In general it has to be said, that the *self-induced* gains k^{11} and k^{22} are much larger compared to the *cross-gains* k^{12} and k^{21}. This clearly can be explained by the configured parameter setting. Apart from that it is interesting to see that for the individual gains the consideration of the anchor agents, i.e., k_1^{11} and k_1^{22} weigh most, but the control error of the anchor agent is actually always the smallest due to its individual exponential convergence, see (6.5a).

This finalises the examination and analysis of the presented modelling concept and the developed control approach by means of simulation studies. In detail, four scenarios, whereas two system descriptions governed from uncoupled PDEs and two from coupled PDEs, verify the approach. Moreover the studies show the power of the discussed backstepping control approach for coupled systems. Next a real-time experiment with a swarm of robots shall validate and enhance the verification of the presented ideas.



(a) (b)

(c) (d)

Figure 6.30: Controller gains for coordinate 1: (a) and (b) consider the control error $\tilde{\boldsymbol{x}}_{\mathrm{c}}^1$ and (c) and (d) amplify the control error $\tilde{\boldsymbol{x}}_{\mathrm{c}}^2$.

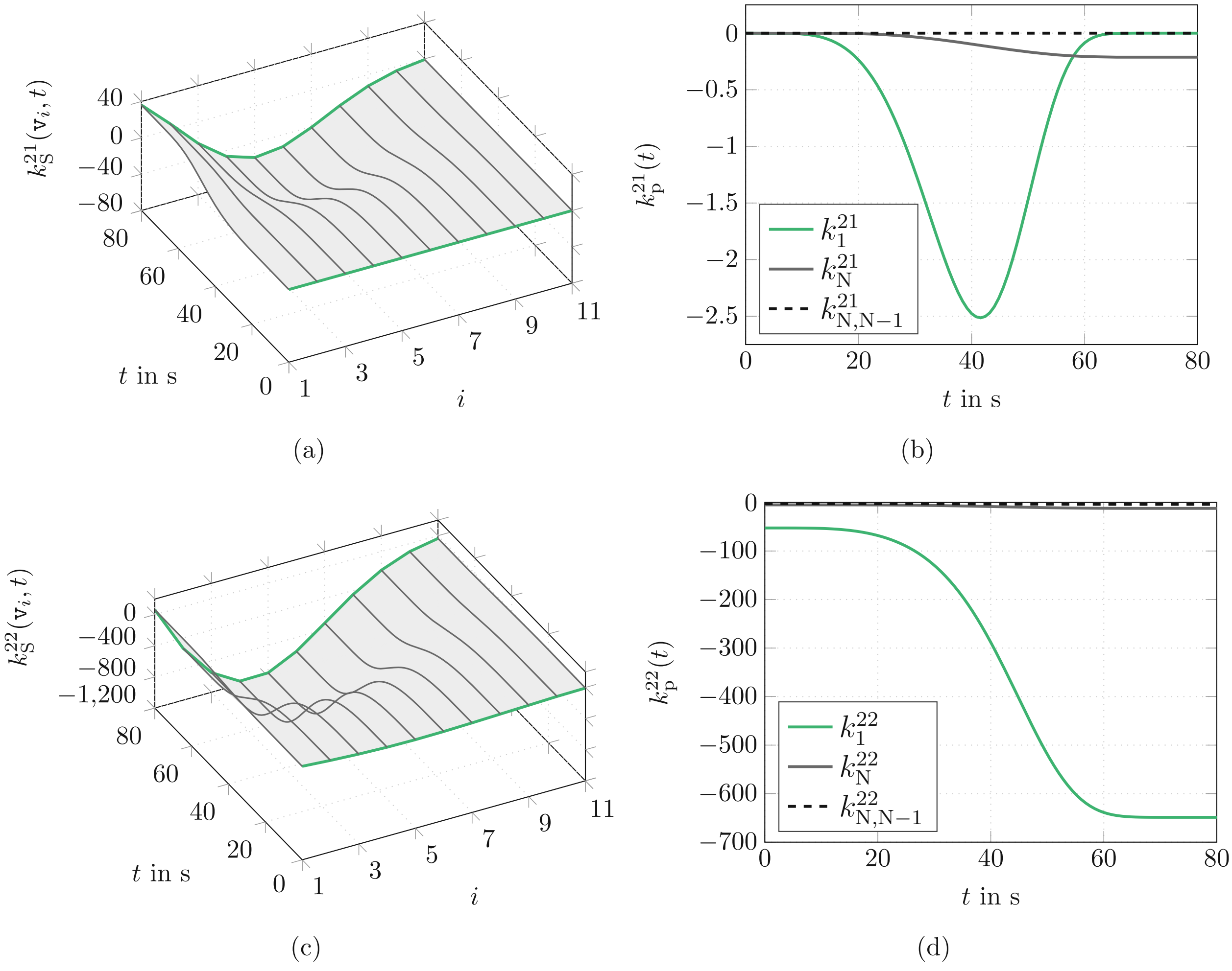


Figure 6.31: Controller gains for coordinate 2: (a) and (b) consider the control error $\tilde{\boldsymbol{x}}_{\mathrm{c}}^{1}$ and (c) and (d) amplify the control error $\tilde{\boldsymbol{x}}_{\mathrm{c}}^{2}$.

6.3 Application Results in a Real Time Test Bed Environment

In the following experimental results are presented for the proposed trajectory planning and tracking control scheme for the formation control of a MAS. In general the results and analysis is based on the publication [25]. It is a fair ambition to mention that to the best knowledge of the author this represents the first real-time implementation of the backstepping methodology for formation control of MASs using parabolic continuum models. The tests were performed by means a laboratory test rig at the Chair of Automatic Control, Kiel University. The author of this work was in charge of the analysis, design, and establishment of the entire test bed. The setup of the experimental scenario basically matches the configuration of the simulation study presented in Section 6.2.1. However, in order to explicitly conduct the transitions experimentally it was necessary to shift the starting line configuration from 0 cm to 150 cm in coordinate 1. Thus, for completeness the next section briefly discusses an approach where this kind of relocation of formation profiles is achieved without adding other theoretical complexity.

6.3.1 Relocating Formation Profiles

Typically formation profiles of DCRSs are arranged around some centre point in the (x^1, x^2)-plane by construction, mostly about the origin. In order to achieve a relocation of the profile it is proposed to add the state $\boldsymbol{x}_\mathrm{e}$ of a so-called *exogenous system* to the state $\boldsymbol{x}$. Therefore, modelling the *exogenous system* in terms of the uncoupled diffusion system (heat equation)

$$\begin{aligned}
\partial_t \boldsymbol{x}_\mathrm{e}(z,t) &= A_\mathrm{e} \partial_z^2 \boldsymbol{x}_\mathrm{e}(z,t)\,,\\
\partial_t \boldsymbol{x}_\mathrm{e}(0,t) &= \boldsymbol{w}_0(t), \quad \partial_t \boldsymbol{x}_\mathrm{e}(\ell_\mathrm{N},t) = \boldsymbol{w}_{\ell_\mathrm{N}}(t)\,,\\
\boldsymbol{x}_\mathrm{e}(z,t_0) &= \boldsymbol{x}_{\mathrm{e},0}(z)\,,
\end{aligned} \tag{6.22}$$

with diagonal and positive definite A_e, allows to introduce the shifted state $\boldsymbol{x}_\mathrm{s}(z,t) = \boldsymbol{x}_\mathrm{e}(z,t) + \boldsymbol{x}(z,t)$. With this corresponding formation profiles can be formulated by means of the steady states $\bar{\boldsymbol{x}}_\mathrm{s}(z) = \bar{\boldsymbol{x}}_\mathrm{e}(z) + \bar{\boldsymbol{x}}(z)$. Note that the steady states $\bar{\boldsymbol{x}}_\mathrm{e}(z)$ for (6.22) are given by $\bar{\boldsymbol{x}}_\mathrm{e}(z) = \boldsymbol{p}_0 + \boldsymbol{p}_1 z$ with $\boldsymbol{p}_0 = \bar{\boldsymbol{x}}_\mathrm{e}(0)$ and $\boldsymbol{p}_1 = (\bar{\boldsymbol{x}}_\mathrm{e}(\ell_\mathrm{N}) - \bar{\boldsymbol{x}}_\mathrm{e}(0))/\ell_\mathrm{N}$ with the values $\bar{x}_\mathrm{e}^m(0)$, $\bar{x}_\mathrm{e}^m(\ell_\mathrm{N})$ being freely assigned [23, 25]. With this, the dynamics of the shifted state directly follows from inserting $\boldsymbol{x} = \boldsymbol{x}_\mathrm{s} - \boldsymbol{x}_\mathrm{e}$ into the corresponding continuum model equation, e.g. considering a DCRS it leads to

$$\begin{aligned}
\partial_t \boldsymbol{x}_\mathrm{s}(z,t) &= A \partial_z^2 \boldsymbol{x}_\mathrm{s}(z,t) + \Delta A \partial_z^2 \boldsymbol{x}_\mathrm{e}(z,t) - B(z,t)\left(\partial_z \boldsymbol{x}_\mathrm{s}(z,t) - \partial_z \boldsymbol{x}_\mathrm{e}(z,t)\right)\\
&\quad + C(z,t)\left(\boldsymbol{x}_\mathrm{s}(z,t) - \boldsymbol{x}_\mathrm{e}(z,t)\right)\\
\partial_t \boldsymbol{x}_\mathrm{s}(0,t) &= \boldsymbol{u}_0^*(t) + \boldsymbol{w}_0(t) + \Delta \boldsymbol{u}_0(t)\,,\\
\partial_t \boldsymbol{x}_\mathrm{s}(\ell_\mathrm{N},t) &= \boldsymbol{u}_{\ell_\mathrm{N}}^*(t) + \boldsymbol{w}_{\ell_\mathrm{N}}(t) + \Delta \boldsymbol{u}_{\ell_\mathrm{N}}(t)\,,\\
\boldsymbol{x}_\mathrm{s}(z,t_0) &= \boldsymbol{x}_0(z) + \boldsymbol{x}_{\mathrm{e},0}(z)\,,
\end{aligned} \tag{6.23}$$

with diagonal $\Delta A = A_\mathrm{e} - A$. In view of motion planning the strategy discussed in Chapter 4 can be deduced in a straightforward way. With this, finite time transition between steady state solutions of (6.22) can be realized by interpreting the boundary values $\boldsymbol{w}_0(t) = \boldsymbol{w}_0^*(t)$ and $\boldsymbol{w}_{\ell_\mathrm{N}}(t) = \boldsymbol{w}_{\ell_\mathrm{N}}^*(t)$ as FF control signals and assigning their temporal path in suitable fashion, e.g., again by exploiting the flatness property of the heat equation (6.22). With this it is fair to say that the FF controls $\boldsymbol{w}_0^*(t)$ and $\boldsymbol{w}_{\ell_\mathrm{N}}^*(t)$ provide the open-loop state evolution $\boldsymbol{x}_\mathrm{e}(z,t) = \boldsymbol{x}_\mathrm{e}^*(z,t)$. Consequently the shifted tracking error fulfils

$$\begin{aligned}
\tilde{\boldsymbol{x}}_\mathrm{c}(z,t) &= \boldsymbol{x}(z,t) - \boldsymbol{x}^*(z,t)\\
&= \boldsymbol{x}(z,t) + \boldsymbol{x}_\mathrm{e}(z,t) - (\boldsymbol{x}^*(z,t) + \boldsymbol{x}_\mathrm{e}^*(z,t))\\
&= \boldsymbol{x}_s(z,t) - \boldsymbol{x}_s^*(z,t).
\end{aligned}$$

As a result, the measurement-based feedback control design applies in the relocated setting without any modification. Hence, subsequently no distinction is made between the shifted state and the state defined originally, meaning redefining $\boldsymbol{x}_s := \boldsymbol{x}$ for simplification purposes. The original state $\boldsymbol{x}$ is obtained by the simple setting $\boldsymbol{x}_\mathrm{e}(z,t) = \mathbf{0}_n$.

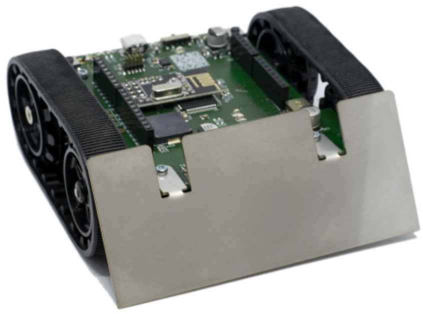
Remark 6.27. *As described in Remark 6.25, similar thoughts regarding the distribution of time-dependent parameter values can be transferred to the current discussion. By adding the* exogenous system (6.22) *a decentralised distribution of the relocation profile is allowed since the stable heat equation does not necessarily need a FB control part. As a consequence this again preserves the overall network topology. The dynamics of* (6.22) *can be adjusted by setting the parameter values for* A_e *appropriately since they directly influence the eigenvalues of the system. With this, the state of any agent associated with the vertex* $\mathtt{v}_i$, $i \in \{1,2,\dots,\mathrm{N}\}$, *is described in terms of the four states, i.e.,*$[x^1, x_\mathrm{e}^1, x^2, x_\mathrm{e}^2](\mathtt{v}_i,t)$, *when planar motion in the* (x^1,x^2)*-domain is taken into account.*

Next, applying the 2-step transformation to (6.23), i.e. FDM and the appropriate mapping to the set of vertices of the underlying subgraph, leads to the discrete MAS scheme

$$\begin{aligned}
\partial_t\boldsymbol{x}(\mathtt{v}_i,t) &= A\left(\boldsymbol{x}(\mathtt{v}_{i+1},t) - 2\boldsymbol{x}(\mathtt{v}_i,t) + \boldsymbol{x}(\mathtt{v}_{i-1},t)\right)\\
&\quad - \frac{B(\mathtt{v}_i,t)}{2}\left((\boldsymbol{x}(\mathtt{v}_{i+1},t) - \boldsymbol{x}(\mathtt{v}_{i-1},t)) - (\boldsymbol{x}_\mathrm{e}(\mathtt{v}_{i+1},t) - \boldsymbol{x}_\mathrm{e}(\mathtt{v}_{i-1},t))\right)\\
&\quad + C(\mathtt{v}_i,t)\left(\boldsymbol{x}(\mathtt{v}_i,t) - \boldsymbol{x}_\mathrm{e}(\mathtt{v}_i,t)\right) + \Delta A\left(\boldsymbol{x}_\mathrm{e}(\mathtt{v}_{i+1},t) - 2\boldsymbol{x}_\mathrm{e}(\mathtt{v}_i,t) + \boldsymbol{x}_\mathrm{e}(\mathtt{v}_{i-1},t)\right) \qquad (6.24)\\
\partial_t\boldsymbol{x}(\mathtt{v}_1,t) &= \boldsymbol{u}_0^*(t) + \boldsymbol{w}_0^*(t) + \Delta\boldsymbol{u}_0(t)\,,\\
\partial_t\boldsymbol{x}(\mathtt{v}_\mathrm{N},t) &= \boldsymbol{u}_{\ell_\mathrm{N}}^*(t) + \boldsymbol{w}_{\ell_\mathrm{N}}^*(t) + \Delta\boldsymbol{u}_{\ell_\mathrm{N}}(t)\,,\\
\boldsymbol{x}(\mathtt{v}_i,t_0) &= \boldsymbol{x}_0(\mathtt{v}_i) + \boldsymbol{x}_{\mathrm{e},0}(\mathtt{v}_i)\,.
\end{aligned}$$

From (6.24) it is clear that the agents need to share the extended state information $[\boldsymbol{x},\, \boldsymbol{x}_\mathrm{e}]^T$. However, for the special case $A = A_\mathrm{e}\,, B(\mathtt{v}_i,t) = 0_{n,n}$ the agents only require $\boldsymbol{x}$ from their network neighbours to compute the protocol.

Before actual results are discussed in detail the next section gives an overview of the test bed developed for the validation of suitable MAS control concepts.



Type	Features
Processor	ARM Cortex-M4F 120MHz
RAM	2 × 64kB
Flash	512kB
Communication	USB, nRF24, Bluetooth
Motors	2 DC motors with 75 : 1 gearbox
Periphery	6 axis IMU, 2 LEDs, Buzzer, Magnetic quadrature encoders, IR sensors, Arduino header, etc.
Dimensions	approx. 10cm × 10cm × 4cm

Figure 6.32: The agent: a caterpillar robot [85].

6.3.2 Multi Robot Test Rig and Software System

Basically the environment for testing MASs is built upon small caterpillar robots which are shown Figure 6.32. Additionally, next to the picture the basic features of the robot are listed in order to give an impression of the available computational power and memory capacity, the equipped actuators and sensors, and the featured network communication. The robot was developed by Prof. Styger and his group at Lucerne University of Applied Sciences and Arts.

Remark 6.28. *It should be pointed that the operated caterpillar robot represents a non-holonomic system. More precisely speaking, its three degrees of freedom (3DOF) kinematic model has the form*

$$\begin{aligned} \dot{x}_r^1 &= v_r \cos(\phi_r)\,, \\ \dot{x}_r^2 &= v_r \sin(\phi_r)\,, \\ \dot{\phi}_r &= \omega_r\,, \end{aligned}$$

with the two inputs v_r and ω_r. Here v_r stands for the translational velocity and ω_r denotes the angular velocity which defines the orientation of the robot ϕ_r in the 2D plane. Clearly the kinematic model has to satisfy the constraint $\dot{x}_r^2 = \dot{x}_r^1 \tan(\phi_r)$. This non-holonomic behaviour somehow stands in contrast to the modelling assumptions, where the agents are in principle represented as ideal geometric points. Moreover, the robot itself has an underlying kinematic control loop since the actual physical inputs are the angular velocities of the left and right belt drive $(\omega_\mathrm{L}\,, \omega_\mathrm{R})$ and in further consequence the two DC motors. Obviously the physical inputs can be set by

$$\begin{aligned} \omega_\mathrm{R} &= \frac{2v_r + L\omega_r}{D_\mathrm{w}}\,, \\ \omega_\mathrm{R} &= \frac{2v_r - L\omega_r}{D_\mathrm{w}} \end{aligned}$$

with D_w as the diameter of the wheels for the drive belt and L is the shortest distance between the left and right wheels.

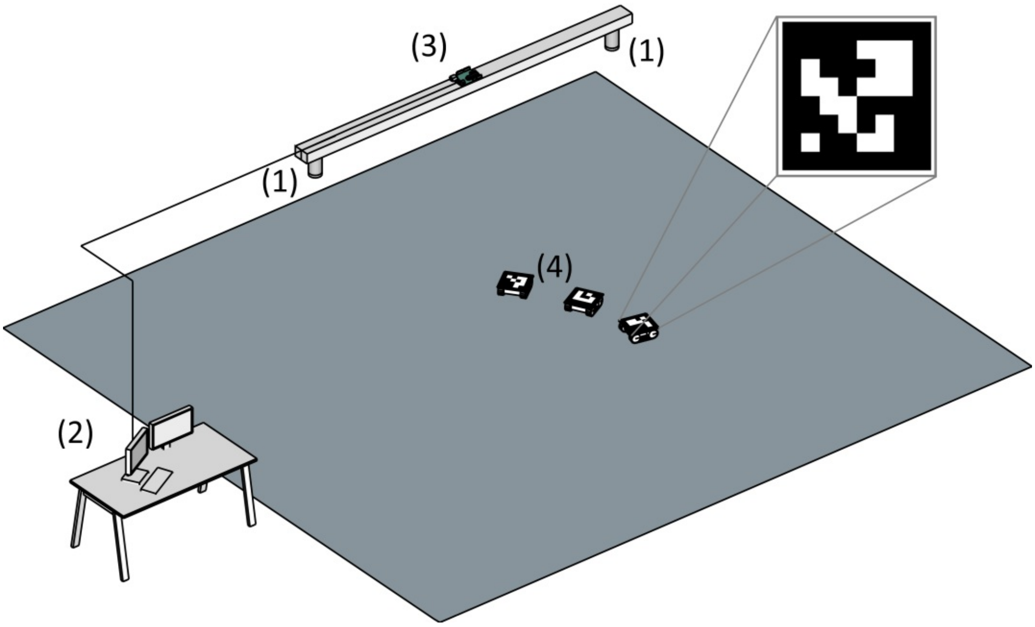


Figure 6.33: Basic scheme of the mobile robot test rig.

As a consequence this provides a significant challenge for the developed 2DOF controllers to compensate these deviations from ideal behaviour, hence imposing a benchmark for robustness analysis.

The real-time implementation of the measurement-based 2DOF control concept as introduced in Chapter 4 and Chapter 5 requires the access of the state vector $\boldsymbol{x}$ of each robot agent. Here, the state vector represents the euclidean coordinates in the 2D space (x^1, x^2). A suitable hard- and software environment has been set up to address the requirements and which allows to perform controlled transitions between different formations including their relocation from the origin. The employed environment is schematically depicted in Figure 6.33 and basically consists of the following four main subsystems:

(1) Top-mounted camera system for optical position detection.

(2) Computing unit for OpenCV application [9].

(3) Development board with a radio module.

(4) Caterpillar robots equipped with Augmented Reality Codes, University of Córdoba (ArUco Codes) [31].

In the test environment a single camera mounted on the ceiling is utilised for position measurement. The camera is connected to a work station which runs an image processing application. The developed software benefits from the OpenCV and the so-called ArUco C++ libraries [64]. The latter is used to detect the individual AruCo codes which are fixed on top of each agent

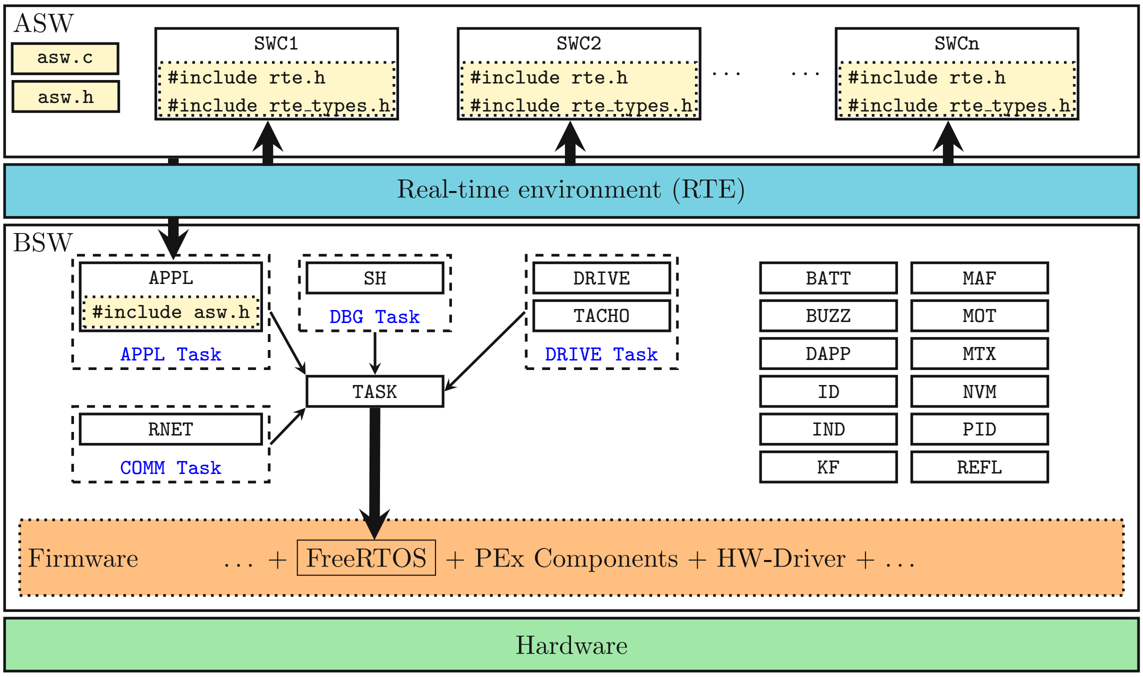


Figure 6.34: Overview of the reworked software architecture.

[31]. The image data is processed and is sent via a serial interface to an electronic development board which is equipped with a radio module. The electronic board runs a software application which broadcasts messages containing the position information of all agents via radio to the caterpillar robots. The robots, which can either act as leader or as follower agents, are equipped with a radio module as well. This allows them to receive the messages over the air. From the broadcast each robot only extracts the specific position information which is demanded due to the topology induced by the right hand side (6.24). In other words, this scheme emulates the topology of the wireless communication which is required to establish the measurement-based tracking control procedure. The representation of the underlying subgraph is pictured in Figure 6.1c.

Apart from the hardware installation it is essential for a satisfying test bed performance to develop a well-conceived software system and consequently fast and efficient system components. Basically three software packages were established:

- A C++ OpenCV application for the computing unit which computes local 2D coordinates and the orientation angle from the ArUco Codes fixed on top of the agents.
- An embedded software project for the development board which receives the position data via serial interface and broadcasts it over the air, and
- a rather complex embedded software project for the caterpillar robot.

For the latter a software project for basic functions was already developed by Prof. Styger and his group at Lucerne University. However, with the given structure and software architecture it was not possible to simply extend some functionality which allows to manoeuvrer a swarm of robots. Therefore a major architectural rework and the development of some additional basic software components were necessary in order to gain a certain structure and enough flexibility within the software project of the caterpillar robot. As a consequence, the reworked software architecture resulted in the block diagram sketched in Figure 6.34. It consists of the three main blocks, basic software (BSW), runtime environment (RTE), and application software (ASW). This basic structure is part of the standardized software framework called classic Automotive Open System Architecture (AUTOSAR) which is successfully implemented in various control units in Automotive Industry. Here, the RTE is basically a bunch of access functions and it completely decouples the BSW from the actual application. In principle, this allows to design ASW just by using the RTE-layer and without knowing the hardware underneath. Apart from the ASW and RTE, some BSW-components were developed from scratch, e.g.

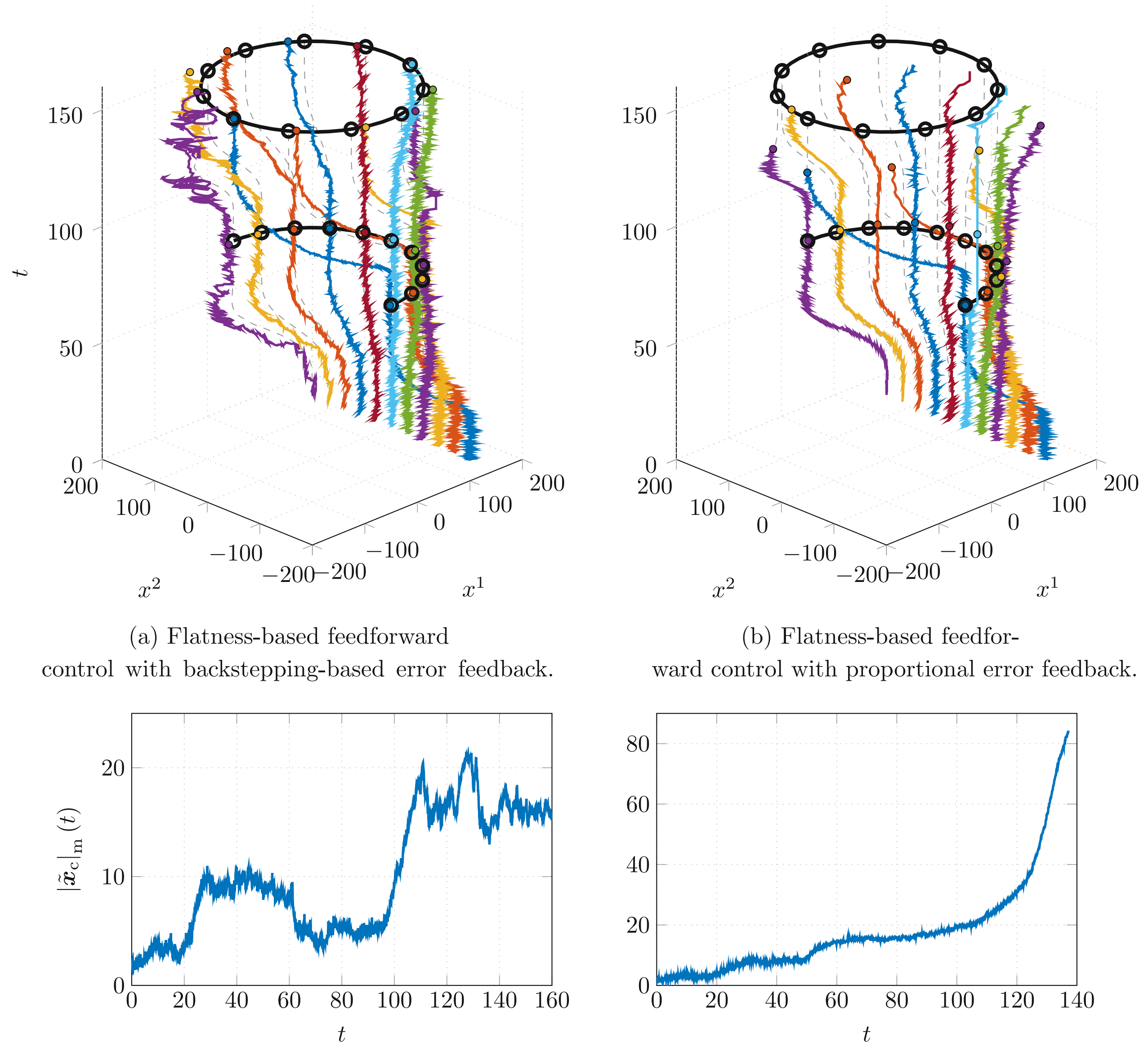
- application (APPL) – running the ASW,
- demo application (DAPP) – for writing small test applications,
- matrix (MTX) – for matrix manipulations,
- Kalman filter (KF) – fast and efficient implementation of a Kalman filter using $UD-$filter methods [34],
- task (TASK) – configures all tasks and acts as the interface to the real-time operation system FreeRTOS.

Detailed documentation pages, references and an API Manual of the all BSW-components can be found on `https://sumo-bsw.readthedocs.io/`.

In the next paragraphs experimental results are presented for a 2-transition scenario which allows a comparison between real-time behaviour and experience gained from the simulation cases above.

6.3.3 Experimental Results

The experimental results for N = 11 caterpillar robots are based on the parameter set summarised in Table 6.1. The formal setup is exactly the same as for the simulation scenario in Section 6.2.1. The experiment implements the follower protocol (6.2) which is imposed by the discretization process as described in Section 6.1. Since the parameter setting in Table 6.1 describes a uncoupled DRS the protocol simplifies with $B(\mathtt{v}_i, t) = 0_{n,n}$ and $C(\mathtt{v}_i, t)$ may be considered as diagonal and only evolves in time. For the ease of implementation the time-variant reaction term $c^m(\mathtt{v}_i, t) = c^m(t)$, $m = 1, 2$, $i = 1, \ldots, 11$ is configured off-line for each agent. However, the synchronisation of the temporal evolution of the parameter for all agents is reached through a trigger signal, which is broadcast via radio. The implementation of the



(a) Flatness-based feedforward control with backstepping-based error feedback.

(b) Flatness-based feedforward control with proportional error feedback.

(c) Mean distance error $|\tilde{\boldsymbol{x}}_\mathrm{c}|_\mathrm{m}(t)$ for (a).

(d) Mean distance error $|\tilde{\boldsymbol{x}}_\mathrm{c}|_\mathrm{m}(t)$ for (b).

Figure 6.35: Experimental results for the test scenario: (a) shows the spatial-temporal evolution of the transition with 2DOF controller including a backstepping-based feedback term; for (b) the 2DOF controller is evaluated with a proportional error feedback; (c) and (d) show the mean distance error computed according to (6.25).

leader protocol involves the 2DOF controller consisting of a flatness-based FF term and a measurement-based backstepping controller according to (6.5) and (6.6), respectively. Again, all matrices are considered as diagonal and $M(\mathrm{v}_i, t) = 0_{n,n}$. Moreover, all time-dependent system and design parameters, the FF term and the backstepping gains are configured offline and are mapped in time online using linear interpolation. It is worth to mention that exactly equal parameter values are used as they have been derived for the simulation scenario in Section 6.2.1. Then the parameter values are directly ported to appropriate data types and flashed hard-coded

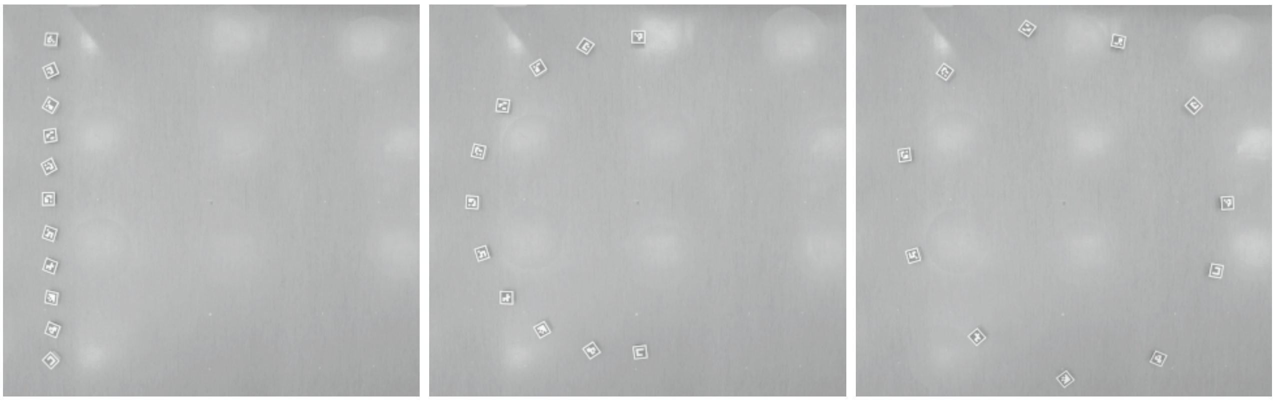


Figure 6.36: Snapshots of the transition using the top-mounted camera system.

with the entire software onto the microcontroller of the caterpillar robots. With this, the obtained experimental results are shown in Figure 6.35a for the twofold transition: first from the initial line configuration passing through the point $(-150, 0)$ cm to an intermediate half circle formation of radius $R = 150$ cm within the time interval $t \in [0, 80]$ s and secondly to a desired circle formation of radius R within the time interval $t \in (80, 160]$ s. For a striking view of the steady state situations see also Figure 6.36 which shows snapshots of the camera system mounted on top of the test bed. For comparison reasons and to illustrate the performance of the 2DOF control concept in Figure 6.35b additional experimental results are provided which use a combination of the flatness-based FF control and a proportional error control for both leader agents and as written in (6.5a). Since the scenario is uncoupled the proportional error control algorithm can be simplified to $\Delta\boldsymbol{u}_0(t) = -D\tilde{\boldsymbol{x}}_{\mathrm{c}}(0, t)$ and $\Delta\boldsymbol{u}_{\ell_{\mathrm{N}}}(t) = -D\tilde{\boldsymbol{x}}_{\mathrm{c}}(\ell, t)$ with $D = \mathrm{diag}\,[0.15,\ 0.15]$. Furthermore, the mean distance error

$$|\tilde{\boldsymbol{x}}_{\mathrm{c}}|_{\mathrm{m}}(t) = \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} |\tilde{\boldsymbol{x}}_{\mathrm{c}}(\mathrm{v}_i, t)| \ , \tag{6.25}$$

with $|\tilde{\boldsymbol{x}}_{\mathrm{c}}(\mathrm{v}_i, t)| = \|\boldsymbol{x}(\mathrm{v}_i, t) - \boldsymbol{x}^*(\mathrm{v}_i, t)\|_2$, between desired and measured position values is shown in Figure 6.35c and Figure 6.35d. Analysing the results of Figure 6.35 gives a clear picture. The 2DOF controller including the backstepping-based error feedback is able to stabilise the transitions while the simple proportional error feedback fails. The latter cannot keep track of all agents and consequently the desired formation profile falls apart. When backstepping tracking control is in place the mean error distance remains in the same order as it is for the associated simulation scenario, see Figure 6.7a for comparison with the individual distance errors. However, when only the individual proportional error control is applied to the leader agents the mean distance error clearly diverges with time.

This closes the analysis of the simulation studies and the test-bed experiment. The next chapter gives a comprehensive retrospective view of the entire thesis and the author presumes to state possible future topics for the crossover of MASs and modelling and control of DPSs.

Chapter 7

Conclusion and Outlook

For the retrospect let us summaries the presented work from a slightly wider perspective. After a brief introduction to the topic the thesis mainly consists of three parts. The first part starting with Chapter 2 deals with questions regarding the relationship between a discrete formulation of the MAS dynamics and a continuous description using DPSs governed by PDEs. Here, an appropriate scheme is worked out which supports the transformation in both directions. More precisely speaking, it results in a discretisation and subsequent mapping process from a discrete to a continuous problem formulation and in a reverse mapping and a Taylor series expansion for the opposite direction. Here the crucial message is that the PDE serves as an approximating model for the ordinary discrete set-up of the MAS dynamics in terms of ODEs. This is an alternative perspective compared to the traditional approach when dealing with DPS. Another important result refers to the varying spatial domain which depends on the number of agents by $\ell_{\mathrm{N}} = \mathrm{N} - 1$. This implicitly sets the discretisation step for FDM to $\Delta z = 1$. As a result the dynamics of a single agent is independent of the number of agents when FDM is applied. In other words, this preserves the agent dynamics but the dynamic behaviour of the entire MAS varies with $1/\mathrm{N}^2$ in case the continuous system formulation is of parabolic type. In contrast to that a fixed spatial domain, i.e. $\ell_{\mathrm{N}} = \ell = const.$ preserves the MAS dynamics and the discretised agent dynamics depends on N^2. Moreover, the discussion motivates the so-called *inverse design approach* where the dynamic behaviour of the MAS is imposed directly by a PDE. With this, Chapter 3 formulates the investigated theoretical problems of the thesis. It starts with a very general PDE formulation. However, eventually the elaboration comes down to two systems, the coupled DCRS and the coupled MVBE whereas both are defined with temporal and spatial varying parameters.

From this, the second part copes with the controller and observer synthesis. A 2DOF is developed consisting of a flatness-based FF term and FB part for tracking error control. The latter and the observer design is based in the *backstepping* technique. In Chapter 4 a motion planning concept is presented which allows to steer the MAS in open loop from a starting formation profile to a target deployment. Here, the mentioned formation profiles refer to solutions of the steady state problem of the underlying PDE. The approach allows to connect different steady states from different families of PDE by utilising their flatness-based properties. Applying tracking error control on the one hand permits the modification of the convergence properties of the swarm dynamics and on the other hand it allows to stabilise and to transit to formation profiles which are inherently unstable. Here, the backstepping technique is applied to both, to the controller as well as to the observer design. In this context a framework is formulated in Chapter 5 which is an alternative approach to the concepts currently found in literature. The presented concept simplifies the coupled conditional equations which are required to determine the controller and observer gains. Basically, the complexity is shifted from the kernel equations to the stability proof. This has the consequence that the resulting equations are similar or equal to the equations of the uncoupled case. The latter are well researched and documented over the last years and therefore their handling is considered as state of the art.

For a concluding statement on the presented work let us recall the overall goal of the thesis. In Section 1.3 the first aim noted is the verification that DPSs governed by PDEs are appropriate models for MAS dynamics. As a second, it shall be validated that PDE theories for controller and observer design can be utilised in order to establish comprehensive formation control algorithms. At this stage one inevitable referred to Chapter 6 which is the last part of the thesis. There, an extensive analysis is conducted on four simulations studies and one real-time application. Without hesitation it can be said that the successful application of the 2DOF control approach on a swarm consisting of 11 caterpillar robots demonstrates the huge potential of the approach and answers both questions in a very positive way. Moreover, the demonstration on the embedded real-time system validates the flatness-based as well as the backstepping theory for PDEs in general. Even though the theory has been well researched over the last two decades this is still a very rare case for this academic area. Furthermore, the theoretical work of the first two mentioned parts above and the five investigations in Chapter 6 clearly meet and fulfil the requirements of the self-imposed objectives backing the overall goal of the thesis.

At this stage it is pointed out that the satisfying outcome of this thesis shall encourage and incentive for further investigations on several topics. Clearly there exist many ideas related to the continuous modelling approach for MAS and the associated control strategies. Possible scenarios, extensions and modifications to the current set-up are:

- Adding a backstepping-based Luenberger-type observer to real-time experiment.
- Adding an exogenous system for the distribution of system parameters to the real-time experiment.
- Changing the number of agents online, i.e. add an agent to, or remove an agent from the MAS and validate this scenario by means of the real-time experiment.
- Extending the *point-to-point relocation* of system state to a *point-to-line expansion* or *deformation* by using stable transitions of the exogenous state; validate this scenario by means of the real-time experiment.

- Changing the order of agent in their network topology online; verify and validate that the backstepping-basedtracking controller can stabilise this scenario.

- Rotating the target state online; verify and validate that the MAS rotates accordingly and the backstepping-basedtracking controller can stabilise this scenario.

- Use periodic boundary conditions, e.g. $\boldsymbol{x}(0,t) = \boldsymbol{x}(\ell_{\mathrm{N}},t)$, $\partial_z \boldsymbol{x}(0,t) = \partial_z \boldsymbol{x}(\ell_{\mathrm{N}},t)$. Without proof, applying FDM should result in a N$-$cycle graph.

- Use other discretisation patterns, either higher order FDM or other concepts as finite volume method (FVM) or finite element method (FEM); verify and validate the given scenario with the resulting communication topology in simulation and practical application.

- Use higher dimensional spatial domains to induce more complex communication topologies; verify and validate the PDE based formation control appraoch of MAS for the resulting problem formulation in simulation and experiment.

- Develop strategies for collision avoidance using the continuous formulation approach.

Apart from the MAS related topics the introduced concept for backstepping concerning coupled PDEs contains great potential for further investigations. An apparent extension is the formulation for Neumann and Robin BCs and the corresponding stability proofs in the $\mathbf{L}^2-$space. Moreover, the presented sufficient inequalities for exponential stability invites one to investigate the impact of the symmetric and ant-symmetric part of the coefficient matrices as well as the effect of spatial dependency of the parameter matrices. On top of that one might be eager to formulate necessary inequalities or conditions.

Obviously there is no limiting factor for future work. From the author's point of view this thesis is just the beginning of many more research papers regarding MAS, continuous versus discrete problem formulation and their relationship, and the presented backstepping concept for coupled PDEs.

Appendix A

Finite Difference Method

The following tables list the coefficients of the FDM for the discrete approximation of first, second and fourth derivative. Moreover, for each order of derivative the table states the order of accuracy in terms of the approximation error.

Table A.1: Coefficients for the central finite difference quotients.

order of derivative	order of accuracy	z_{i-3}	z_{i-2}	z_{i-1}	z_i	z_{i+1}	z_{i+2}	z_{i+3}
1st	$\mathcal{O}\left(\Delta z^2\right)$			$-1/2$	0	$1/2$		
	$\mathcal{O}\left(\Delta z^4\right)$		$1/12$	$-2/3$	0	$2/3$	$-1/12$	
2nd	$\mathcal{O}\left(\Delta z^2\right)$			1	-2	1		
	$\mathcal{O}\left(\Delta z^4\right)$		$-1/12$	$4/3$	$-5/2$	$4/3$	$-1/12$	
4th	$\mathcal{O}\left(\Delta z^2\right)$		1	-4	6	-4	1	
	$\mathcal{O}\left(\Delta z^4\right)$	$-1/6$	2	$-13/2$	$28/3$	$-13/2$	2	$-1/6$

Table A.2: Coefficients for the forward and backward finite difference quotients. Swap signs of odd derivatives for the backward approximations.

order of derivative	order of accuracy	z_i	$z_{i\pm1}$	$z_{i\pm2}$	$z_{i\pm3}$	$z_{i\pm4}$	$z_{i\pm5}$	$z_{i\pm6}$	$z_{i\pm7}$
1st	$\mathcal{O}\left(\Delta z\right)$	-1	1						
	$\mathcal{O}\left(\Delta z^2\right)$	$-3/2$	2	$-1/2$					
	$\mathcal{O}\left(\Delta z^4\right)$	$-25/12$	4	-3	$4/3$	$-1/4$			
2nd	$\mathcal{O}\left(\Delta z\right)$	1	-2	1					
	$\mathcal{O}\left(\Delta z^2\right)$	2	-5	4	-1				
	$\mathcal{O}\left(\Delta z^4\right)$	$15/4$	$-77/6$	$107/6$	-13	$61/12$	$-5/6$		
4th	$\mathcal{O}\left(\Delta z\right)$	1	-4	6	-4	1			
	$\mathcal{O}\left(\Delta z^2\right)$	3	-14	26	-24	11	-2		
	$\mathcal{O}\left(\Delta z^4\right)$	$28/3$	$-111/2$	142	$-1219/6$	176	$-185/2$	$82/3$	$-7/2$

At this stage it has to be clarified that Table A.1 and Table A.2 are listed as a work of reference for the interested reader and for the sake of completeness, respectively. Therefore note the following examples.

Example 1.1. *First order derivative approximated by the central finite difference quotient with fourth order of accuracy.*

$$\partial_z \boldsymbol{x}_i \approx \frac{-\boldsymbol{x}_{i+2} + 8\boldsymbol{x}_{i+1} - 8\boldsymbol{x}_{i-1} + \boldsymbol{x}_{i-2}}{12\Delta z}$$

Example 1.2. *Second order derivative approximated by the central finite difference quotient with second order of accuracy.*

$$\partial_z^2 \boldsymbol{x}_i \approx \frac{\boldsymbol{x}_{i+1} - 2\boldsymbol{x}_i + \boldsymbol{x}_{i-1}}{\Delta z^2}$$

Example 1.3. *Fourth order derivative approximated by the forward finite difference quotient with forth order of accuracy.*

$$\partial_z^4 \boldsymbol{x}_i \approx \frac{56\boldsymbol{x}_i - 333\boldsymbol{x}_{i+1} + 852\boldsymbol{x}_{i+2} - 1219\boldsymbol{x}_{i+3} + 1056\boldsymbol{x}_{i+4} - 555\boldsymbol{x}_{i+5} + 164\boldsymbol{x}_{i+6} - 21\boldsymbol{x}_{i+7}}{6\Delta z^4}$$

Example 1.4. *First order derivative approximated by the forward finite difference quotient with first order of accuracy. Signs are swapped with respect to the coefficients listed in Table A.2*

$$\partial_z \boldsymbol{x}_i \approx \frac{\boldsymbol{x}_i - \boldsymbol{x}_{i-1}}{\Delta z}$$

Appendix B

Analysis of Steady States

This chapter extends the evaluation of steady states from Section 3.2 with explicit determination of the coefficient vectors $\boldsymbol{\kappa}_j$, $j = 1, \cdots, 4$ in (3.45) for the coupled DCRS as well as k_c and k_s, or k_0 and k_1, respectively in (3.57)-(3.59) for the uncoupled case. Furthermore, for the latter it features some more examples of 2D stationary formation profiles.

B.1 Steady States of the Coupled DCRS

Let us recall the discussion in Section 3.2 where the boundary value problem

$$\begin{aligned} &A\partial_z^2\bar{\boldsymbol{x}}(z) - \bar{B}\partial_z\bar{\boldsymbol{x}}(z) + \bar{C}\bar{\boldsymbol{x}}(z) = \boldsymbol{0}_2\,, \\ &\bar{\boldsymbol{x}}(0) = \bar{\boldsymbol{x}}_\mathtt{a}\,, \quad \bar{\boldsymbol{x}}(\ell_\mathrm{N}) = \bar{\boldsymbol{x}}_1 \end{aligned} \tag{B.1}$$

leads to the underlying eigenvalue problem (3.38). With this, assuming two pairs of conjugated complex valued eigenvalues $\nu_j \in \mathbb{C}$, $j = 1, \cdots, 4$ then the ansatz

$$\bar{\boldsymbol{x}}(z) = \sum_{j=1}^{4} \boldsymbol{\kappa}_j \exp(\nu_j z)\,. \tag{B.2}$$

for the solution of (B.1) can be combined by pairs as

$$\bar{\boldsymbol{x}}(z) = \sum_{j=1}^{2} \exp(\sigma_j z)\left(\boldsymbol{k}_{\mathrm{c},j}\cos(\theta_j z) + \boldsymbol{k}_{\mathrm{s},j}\sin(\theta_j z)\right) \tag{B.3}$$

with $\nu_{1,2} = \sigma_1 \pm \imath\theta_1$, $\nu_{3,4} = \sigma_2 \pm \imath\theta_2$, and the definition of the complex constants as $\boldsymbol{\kappa}_1 = (\boldsymbol{k}_{\mathrm{c},1} - \imath\boldsymbol{k}_{\mathrm{s},1})/2$, $\boldsymbol{\kappa}_3 = (\boldsymbol{k}_{\mathrm{c},2} - \imath\boldsymbol{k}_{\mathrm{s},2})/2$ and $\boldsymbol{\kappa}_2 = \boldsymbol{\kappa}_1^*$, $\boldsymbol{\kappa}_4 = \boldsymbol{\kappa}_3^*$ respectively. Equivalently, four real valued solutions $\nu_j \in \mathbb{R}$, $j = 1, \cdots, 4$ can be expressed as $\nu_{1,2} = \sigma_1 \pm \theta_1$ and $\nu_{3,4} = \sigma_2 \pm \theta_2$. Then making use of the abbreviation of the real-valued constants $\boldsymbol{\kappa}_1 = (\boldsymbol{k}_{\mathrm{c},1} + \boldsymbol{k}_{\mathrm{s},1})/2$, $\boldsymbol{\kappa}_2 = (\boldsymbol{k}_{\mathrm{c},1} - \boldsymbol{k}_{\mathrm{s},1})/2$, $\boldsymbol{\kappa}_3 = (\boldsymbol{k}_{\mathrm{c},2} + \boldsymbol{k}_{\mathrm{s},2})/2$, and $\boldsymbol{\kappa}_4 = (\boldsymbol{k}_{\mathrm{c},2} - \boldsymbol{k}_{\mathrm{s},2})/2$ the procedure yields

$$\bar{\boldsymbol{x}}(z) = \sum_{j=1}^{2} \exp(\sigma_j z)\left(\boldsymbol{k}_{\mathrm{c},j}\cosh(\theta_j z) + \boldsymbol{k}_{\mathrm{s},j}\sinh(\theta_j z)\right). \tag{B.4}$$

For a combination of real-valued solutions $\nu_{1,2} \in \mathbb{R}$ and conjugated complex solutions $\nu_{3,4} \in \mathbb{C}$ of the characteristic equation, it yields a combination of (B.3) and (B.4), which may be written as

$$\begin{aligned}\bar{\boldsymbol{x}}(z) = \exp(\sigma_1 z)\left(\boldsymbol{k}_{\mathrm{c},1}\cos(\theta_1 z) + \boldsymbol{k}_{\mathrm{s},1}\sin(\theta_2 z)\right)\\ + \exp(\sigma_2 z)\left(\boldsymbol{k}_{\mathrm{c},2}\cosh(\theta_2 z) + \boldsymbol{k}_{\mathrm{s},2}\sinh(\theta_2 z)\right).\end{aligned} \quad \text{(B.5)}$$

Next, the discussion deals with special constraints on the coefficient matrices (3.35).

- Assuming a DRS with $\bar{B} = 0_{2,2}$ the coefficients of odd order of the characteristic equation simplify to $p_3 = p_1 = 0$. Consequently (3.42) is reduced to

  $$p_4\nu^4 + p_2\nu^2 + p_0 = 0 \quad \text{(B.6)}$$

  Obviously, intermediate solutions for ν^2 follow from

  $$(\nu^2)_{1,2} = \frac{-p_2 \pm \sqrt{p_2^2 - 4p_4p_0}}{2p_4}\,. \quad \text{(B.7)}$$

  With the abbreviations $\zeta = -p_2/(2p_4)$ and $\eta = \sqrt{|p_2^2 - 4p_4p_0|}/2p_4$ this results in $\nu_j = \pm\sqrt{\zeta \pm \imath\eta}$ in case $(\nu^2)_{1,2} \in \mathbb{C}$, and $\nu_j = \pm\sqrt{\zeta \pm \eta}$ if $(\nu^2)_{1,2} \in \mathbb{R}$. Similarly to the discussion above a steady state solution according to (B.3), (B.4), or(B.5) depends on whether the eigenvalues ν_j of the matrix $\bar{M}$ in (3.37) are real or complex valued.

  At this stage it has to be pointed out that with reference to (3.46) it holds $f(\nu_1) = f(\nu_2)$ and $f(\nu_3) = f(\nu_4)$ since $\nu_1^2 = \nu_2^2 = (\nu^2)_1$ and $\nu_3^2 = \nu_4^2 = (\nu^2)_2$. The same applies to $g(\nu_j)$. However the rank of matrix $\mathcal{M}$ in (3.50) remains 4 since $\exp(\nu_1\ell_\mathrm{N}) \neq \exp(\nu_2\ell_\mathrm{N}) \neq \exp(\nu_3\ell_\mathrm{N}) \neq \exp(\nu_4\ell_\mathrm{N})$.

- Furthermore, let us presume a cascaded system with upper or lower triangular coefficient matrices $\bar{B}$ and $\bar{C}$, meaning $b^{12} = c^{12} = 0$ or $b^{21} = c^{21} = 0$. From (3.43) directly follows the characteristic equation

  $$(a^{11}\nu^2 - b^{11}\nu + c^{11})(a^{22}\nu^2 - b^{22}\nu + c^{22}) = 0\,. \quad \text{(B.8)}$$

  With this, obviously the eigenvalues are given by

  $$\nu_{1,2} = \frac{b^{11} \pm \sqrt{(b^{11})^2 - 4a^{11}c^{11}}}{2a^{11}} \quad \text{and} \quad \nu_{3,4} = \frac{b^{22} \pm \sqrt{(b^{22})^2 - 4a^{22}c^{22}}}{2a^{22}}\,. \quad \text{(B.9)}$$

  Depending on whether the coefficient matrices are chosen as upper or lower triangular (3.46) or (3.47) is applicable.

B.2 Steady States of the Uncoupled DCRS

Again, as derived in Section 3.2 the examination of the stationary case for the uncoupled DCRS leads to the boundary value problem

$$a\partial_z^2\bar{x}(z) - \bar{b}\partial_z\bar{x}(z) + \bar{c}\bar{x}(z) = 0\,, \qquad \bar{x}(0) = \bar{x}_{\mathtt{a}}\,, \quad \bar{x}(\ell_{\mathrm{N}}) = \bar{x}_{\mathtt{1}}\,. \tag{B.10}$$

Then, depending on the given parameter set the generic analytic solution

$$\bar{x}(z) = \kappa_1 \exp(\nu_1 z) + \kappa_2 \exp(\nu_2 z) \tag{B.11}$$

leads to different specific representations. Thus, note the discussion for the following cases:

(i) A parameter configuration in (B.10) which fulfils the inequality $\bar{b}^2 - 4a\bar{c} < 0$ renders the steady state solution to

$$\bar{x}(z) = \exp\left(\sigma z\right)\left(k_{\mathrm{c}}\cos(\mu z) + k_{\mathrm{s}}\sin(\mu z)\right)\,. \tag{B.12}$$

The coefficients k_{c} and k_{s} may be evaluated depending on the value of $\mu = \sqrt{\left|\bar{b}^2 - 4a\bar{c}\right|/(2a)}$ and the boundary values $\bar{x}_{\{\mathtt{a},\mathtt{1}\}}$ defined in (B.10). With $l \in \mathbb{N}$ and

- $\underline{\mu \neq l\pi/\ell_{\mathrm{N}}}$ the coefficients follow to

$$k_{\mathrm{c}} = \bar{x}_{\mathtt{a}}\,, \quad \text{and} \quad k_{\mathrm{s}} = \frac{\exp\left(-\sigma\ell_{\mathrm{N}}\right)\bar{x}_{\mathtt{1}} - \bar{x}_{\mathtt{a}}\cos(\mu\ell_{\mathrm{N}})}{\sin(\mu\ell_{\mathrm{N}})}. \tag{B.13}$$

The special case $\bar{x}_{\mathtt{a}} = \bar{x}_{\mathtt{1}} = 0$ and consequently $k_{\mathrm{c}} = k_{\mathrm{s}} = 0$ and leads to the trivial solution $\bar{x}(z) = 0$.

- $\underline{\mu = l\pi/\ell_{\mathrm{N}}}$ the boundary values (3.53) have to fulfil $\bar{x}_{\mathtt{1}} = (-1)^l \exp\left(\sigma\ell_{\mathrm{N}}\right)\bar{x}_{\mathtt{a}}$ and the coefficients follow to

$$k_{\mathrm{c}} = \bar{x}_{\mathtt{a}}\,, \quad \text{and} \quad k_{\mathrm{s}} = k \tag{B.14}$$

with arbitrary amplitude k.

(ii) A positive expression $\bar{b}^2 - 4a\bar{c} > 0$ leads to a stationary profile expressed by the equation

$$\bar{x}(z) = \exp\left(\sigma z\right)\left(k_{\mathrm{c}}\cosh(\mu z) + k_{\mathrm{s}}\sinh(\mu z)\right) \tag{B.15}$$

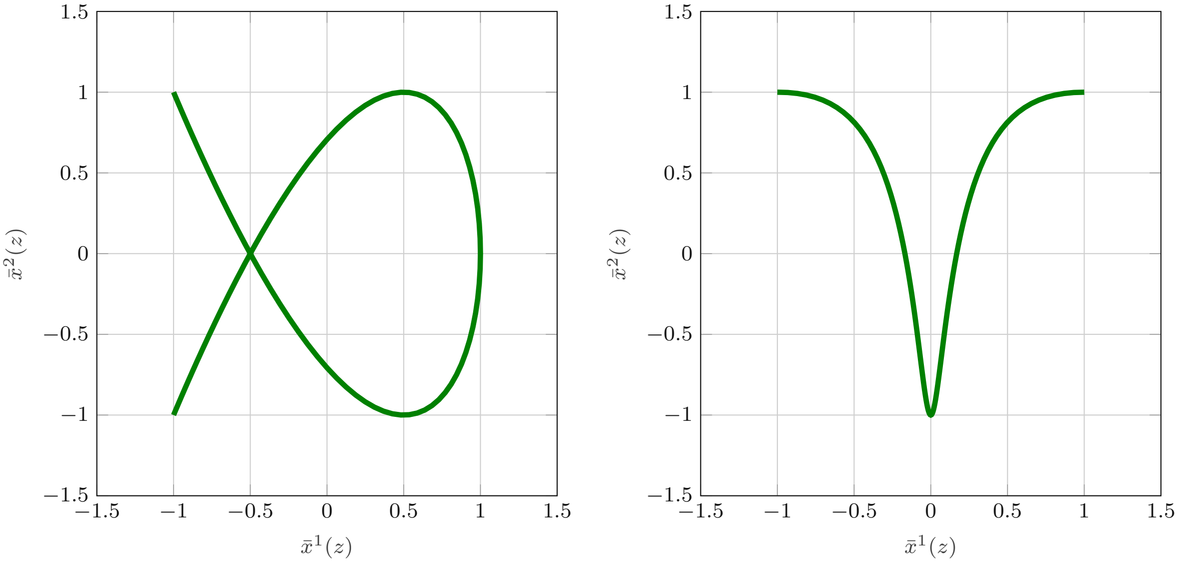
and the parameters k_{c} and k_{s} can be evaluated to

$$\begin{aligned} k_{\mathrm{c}} &= \bar{x}_{\mathtt{a}}\,, \quad \text{and} \\ k_{\mathrm{s}} &= \frac{\exp\left(-\sigma\ell_{\mathrm{N}}\right)\bar{x}_{\mathtt{1}} - \bar{x}_{\mathtt{a}}\cosh(\mu\ell_{\mathrm{N}})}{\sinh(\mu\ell_{\mathrm{N}})}\,. \end{aligned} \tag{B.16}$$

(iii) For the special case $\bar{b}^2 - 4a\bar{c} = 0$ the simplified linear solution

$$\bar{x}(z) = \exp\left(\sigma z\right)\left(k_0 + k_1 z\right) \tag{B.17}$$

is applicable. The constant coefficients follow to $k_0 = \bar{x}_{\mathtt{a}}$ and $k_1 = \left(\exp\left(-\sigma\ell_{\mathrm{N}}\right)\bar{x}_{\mathtt{1}} - \bar{x}_{\mathtt{a}}\right)/\ell_{\mathrm{N}}$.



(a) Fish formation.

(b) Gull formation.

Figure B.1: Illustration of steady state solutions of the uncoupled DRS with a fish-shaped formation profile in (a) and a gull-like formation profile in (b).

Table B.1: Parameters of the formation profiles plotted in Figure B.1 with $\ell_{\mathrm{N}} = 1$.

Profile	Coord.	a	$\bar{b}$	$\bar{c}$	$\bar{x}_{\mathtt{a}}$	$\bar{x}_{\mathtt{1}}$
Fish	(1) :	1	0	$4\pi^2$	-1	-1
	(2) :	1	0	$9\pi^2$	1	-1
Gull	(1) :	1	0	-49	-1	1
	(2) :	1	0	$4\pi^2$	1	1

Figure B.1 illustrates a deployment associated with the first case (i) in the subplot (a) and associated with the first (i) and second case (ii) in the subplot (b). Obviously the last case (iii) covers straight lines or lines distorted by the exp −term. The Table B.1 summarises the parameter configuration for the two plots shown in Figure B.1.

Appendix C

Mathematical Background

C.1 Norms

The convergence analysis performed in Chapter 4 for flatness-based trajectory planning and the determination of integral kernels in Chapter 5 is based on appropriate norms of functions, either over the time domain $t \in \mathbb{R}^+_{t_0}$, or over the entire domain $(z, t) \in \Omega\left(\bar{\ell}_\mathrm{N}, t_0\right)$ with $\Omega\left(\bar{\ell}_\mathrm{N}, t_0\right) := [0, \ell_\mathrm{N}] \times \mathbb{R}^+_{t_0}$. Therefore consider the following notation of norms for the state trajectories $\boldsymbol{x}$ which are defined in the function space $\mathbf{C}^{k,l}(\Omega\left(\bar{\ell}_\mathrm{N}, t_0\right); \mathbb{R}^n)$ with $k, l \in \mathbb{N}_0$. In the following the arguments $(\bar{\ell}_\mathrm{N}, t_0)$ are neglected for simplicity.

Definition 3.6. *Let $x \in \mathbf{C}^{k,l}(\Omega; \mathbb{R})$, then*

$$\sup\nolimits_\Omega |x| := \sup_{(z,t)\in\Omega} |x(z,t)| \,, \tag{C.1}$$

$$\sup\nolimits_{\mathbb{R}^+_{t_0}} |x|(z) := \sup_{t\in\mathbb{R}^+_{t_0}} |x(z,t)| \,. \tag{C.2}$$

Note, that (C.2) actually defines a function in the coordinate z. However, it renders a norm with respect to a fixed spatial position $z \in [0, \ell_\mathrm{N}]$. Coupled distributed parameter systems deal with multi-dimensional state trajectories $\boldsymbol{x}$ and therefore the norm definitions are extended as following.

Definition 3.7. *Let $\boldsymbol{x} \in \mathbf{C}^{k,l}(\Omega; \mathbb{R}^n)$ with $n \in \mathbb{N}$, and consider the subsequent definitions of norms*

$$\left\|\sup\nolimits_\Omega \boldsymbol{x}\right\|_\infty = \left\|\sup\nolimits_\Omega \boldsymbol{x}(z,t)\right\|_\infty := \max_{1\le i\le n}\left(\sup_{(z,t)\in\Omega}\left|x^i(z,t)\right|\right) \,, \tag{C.3}$$

$$\left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty(z) = \left\|\sup\nolimits_{\mathbb{R}^+_{t_0}} \boldsymbol{x}(z,t)\right\|_\infty := \max_{1\le i\le n}\left(\sup_{t\in\mathbb{R}^+_{t_0}}\left|x^i(z,t)\right|\right) \,. \tag{C.4}$$

Note that (C.4) *is a function in z with $\left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty(z) \in \mathbf{C}^0([0, \ell_\mathrm{N}]; \mathbb{R})$ but implements a norm for every fixed $z \in [0, \ell_\mathrm{N}]$.*

Basically, these norms in Definition 3.7 are based on Definition 3.6, but overlaid with the maximum norm for vectors. Obviously (C.3) and (C.4) satisfy the inequality

$$0 \leq \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty (z) \leq \left\|\sup_\Omega \boldsymbol{x}\right\|_\infty . \tag{C.5}$$

Moreover, the definition of norms for appropriate matrix-valued functions is followed by means of an equivalent approach. Instead of the maximum norm for vectors the induced operator max −norm, or maximum absolute row sum of the matrix, is considered. Hence notice the modification of the introduced norms for matrices.

Definition 3.8. *Let $F(z,t) \in \mathbf{C}^{k,l}\left(\Omega; \mathbb{R}^{m\times n}\right)$, $m,n \in \mathbb{N}$, and consider the subsequent definitions of norms*

$$\left\|\sup_\Omega F\right\|_\infty = \left\|\sup_\Omega F(z,t)\right\|_\infty := \max_{i=1,\ldots,m} \sum_{j=1}^{n} \left(\sup_{(z,t)\in\Omega} \left|f_{ij}(z,t)\right| \right) , \tag{C.6}$$

$$\left\|\sup_{\mathbb{R}^+_{t_0}} F\right\|_\infty (z) = \left\|\sup_{\mathbb{R}^+_{t_0}} F(z,t)\right\|_\infty := \max_{i=1,\ldots,m} \sum_{j=1}^{n} \left(\sup_{t\in\mathbb{R}^+_{t_0}} \left|f_{ij}(z,t)\right| \right) . \tag{C.7}$$

Note that (C.7) *is a function in z with $\left\|\sup_{\mathbb{R}^+_{t_0}} F\right\|_\infty (z) \in \mathbf{C}^0\left([0,\ell_\mathrm{N}];\mathbb{R}\right)$. Moreover, the induced operator* max −*norm is sub-multiplicative. Consequently, it holds $\left\|\sup_\Omega FH\right\|_\infty \leq \left\|\sup_\Omega F\right\|_\infty \left\|\sup_\Omega H\right\|_\infty$ and $\left\|\sup_{\mathbb{R}^+_{t_0}} FH\right\|_\infty (z) \leq \left\|\sup_{\mathbb{R}^+_{t_0}} F\right\|_\infty (z) \left\|\sup_{\mathbb{R}^+_{t_0}} H\right\|_\infty (z)$.*

With this and the property

$$0 \leq \left\|\sup_{\mathbb{R}^+_{t_0}} F\right\|_\infty (z) \leq \left\|\sup_\Omega F\right\|_\infty \tag{C.8}$$

it is possible to deduce the useful inequality

$$\left\|\sup_{\mathbb{R}^+_{t_0}} F\boldsymbol{x}\right\|_\infty (z) \leq \left\|\sup_{\mathbb{R}^+_{t_0}} F\right\|_\infty (z) \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty (z) \leq \left\|\sup_\Omega F\right\|_\infty \left\|\sup_{\mathbb{R}^+_{t_0}} \boldsymbol{x}\right\|_\infty (z) . \tag{C.9}$$

Definition 3.9. *Let the spatial domain $\mathfrak{D}(\ell_\mathrm{N})$ be defined in two variables and let its area depend on ℓ_N. Moreover, consider the matrix valued functions $F(z,s,t) \in \mathbf{CG}^{k,l,\alpha}\left(\mathfrak{D}(\ell_\mathrm{N}) \times \mathbb{R}^+_{t_0}; \mathbb{R}^{m\times n}\right)$ and $G(z,t) \in \mathbf{CG}^{k,\alpha}\left(\Omega\left(\overline{\ell}_\mathrm{N}, t_0\right) \times \mathbb{R}^+_{t_0}; \mathbb{R}^{m\times n}\right)$, with $m,n \in \mathbb{N}$. With this, note the definition of norms*

$$\max_{\mathfrak{D}(\ell_\mathrm{N})} \left\|F\right\|_2 (t) = \max_{\mathfrak{D}(\ell_\mathrm{N})} \left\|F(z,s,t)\right\|_2 := \sqrt{\max_{(z,s)\in\mathfrak{D}(\ell_\mathrm{N})} \lambda(F^T(z,s,t)F(z,s,t))} , \tag{C.10}$$

$$\max_{[0,\ell_\mathrm{N}]} \left\|G\right\|_2 (t) = \max_{[0,\ell_\mathrm{N}]} \left\|G(z,t)\right\|_2 := \sqrt{\max_{z\in[0,\ell_\mathrm{N}]} \lambda(G^T(z,t)G(z,t))} . \tag{C.11}$$

This represents the largest singular value of the matrix $F(z,s,t)$ or $G(z,t)$, respectively. Note, in order to be exact λ is a function in t for both (C.10) *and* (C.11) *with $\lambda(t) \in \mathbf{C}^0\left(\mathbb{R}^+_{t_0}; \mathbb{R}\right)$. However, λ is the eigenvalue of the matrix $F^T(z,s,t)F(z,s,t)$ or $G^T(z,t)G(z,t)$, respectively. Therefore the definitions render the spectral norm for matrices for every fixed $t \in \mathbb{R}^+_{t_0}$ and consequently it is sub-multiplicative.*

C.2 Functions

The so-called Gevrey class functions are essential for the assignment of admissible trajectories [55]. These allow the design of feedforward controlled transitions between stationary formation profiles of MAS. Let the set Λ define an open subset of $\mathbb{R}$ and let $\alpha \geq 1$.

Definition 3.10 (Gevrey class [74])**.** *The function $f : \Lambda \to X \subseteq \mathbb{R}$ is in $\mathbf{G}_{\alpha}^{\delta}(\Lambda; X)$, the Gevrey class of order α, if $f \in \mathbf{C}^{\infty}(\Lambda; X)$ and for every compact subset $\bar{\Lambda}$ of Λ, there exists a positive constant δ such that*

$$\sup_{t\in\bar{\Lambda}} \left|\partial_t^k f(t)\right| \leq \delta^{k+1}(k!)^{\alpha} \tag{C.12}$$

$\forall k \in \mathbb{N}_0$.

Two examples of Gevrey functions $\varPhi_{\tau,\omega} : \mathbb{R} \to [0,1]$ according to Definition 3.10 are well documented in literature. For the following let $\tau > 0$.

Example 3.5 (Gevrey function using bell-shaped functions [1])**.** *This first example is given by*

$$\varPhi_{\tau,\omega}(t) = \begin{cases} 0 & t \leq 0 \\ \dfrac{\int_0^t \phi_{\tau,\omega}(\nu)\mathrm{d}\nu}{\int_0^\tau \phi_{\tau,\omega}(\nu)\mathrm{d}\nu} & t \in (0,\tau) \\ 1 & t \geq \tau \end{cases} \tag{C.13}$$

with $\phi_{\tau,\omega}(t)$ defined as

$$\phi_{\tau,\omega}(t) = \begin{cases} \exp\left(-\dfrac{1}{\left[\left(1-\frac{\nu}{\tau}\right)\frac{\nu}{\tau}\right]^{\omega}}\right) & t \in (0,\tau) \\ 0 & t \notin (0,\tau)\ . \end{cases} \tag{C.14}$$

Example 3.6 (Gevrey function using hyperbolic functions [77])**.** *As a second, here the Gevrey function is written as*

$$\varPhi_{\tau,\omega}(t) = \begin{cases} 0 & t \leq 0 \\ \dfrac{1}{2} + \dfrac{1}{2}\tanh\left(\dfrac{2\left(2\frac{t}{\tau}-1\right)}{\left(\frac{4t}{\tau}\left(1-\frac{t}{\tau}\right)\right)^{\omega}}\right) & t \in (0,\tau) \\ 1 & t \geq \tau\,. \end{cases} \tag{C.15}$$

Both examples are in the Gevrey class of order $\alpha = 1 + 1/\omega$. For further details see, e.g., [37] for (C.13) and [76] for (C.15).

In general for matrix (or vector) valued functions $F \in \mathbf{C}^{\infty}(\Lambda; X^{m\times n})$, i.e. $F : \Lambda \to X^{m\times n} \subseteq \mathbb{R}^{m\times n}$, each element $f_{ij}(t), 1 \leq i \leq m\,, 1 \leq j \leq n$ of the matrix-valued function can be treated independently and in equivalent fashion as introduced in Definition 3.10 for scalar functions. However, this would lead to a function space of $m \times n$ different Gevrey classes of the form $\mathbf{G}_{\alpha_{ij}}^{\delta_{ij}}(\Lambda; X)$. Thus, the fact that $\mathbf{G}_{\alpha}^{\delta}(\Lambda; X) \subseteq \mathbf{G}_{\beta}^{\delta}(\Lambda; X)$ for $\alpha \leq \beta$ allows to define a common Gevrey class for vector and matrix valued functions accordingly.

Definition 3.11 (Class of vector- and matrix-valued Gevrey functions). *The function $F : \Lambda \to X^{m\times n} \subseteq \mathbb{R}^{m\times n}$ is in $\mathbf{G}^{\delta}_{\alpha}(\Lambda; X^{m\times n})$, $m, n \in \mathbb{N}$, the $m \times n-$matrix-valued Gevrey class of order α, if $F \in \mathbf{C}^{\infty}(\Lambda; X^{m\times n})$ and the elements f_{ij} are in $\mathbf{G}^{\delta_{ij}}_{\alpha_{ij}}(\Lambda; X)$ according to Definition 3.10. Then for every compact subset $\bar{\Lambda}$ of Λ the $m \times n-$valued function F satisfies*

$$\left\|\sup\nolimits_{t\in\bar{\Lambda}} \left(\partial_t^k F(t)\right)\right\|_{\infty} \leq \delta^{k+1}(k!)^{\alpha}\,, \tag{C.16}$$

$\forall k \in \mathbb{N}_0$, and with $\delta = \max_{1\leq i\leq m}\left(\sum_{j=1}^{n}\delta_{ij}\right)$ and $\alpha = \max_{\substack{1\leq i\leq m\\ 1\leq j\leq n}} \alpha_{ij}$. Here, the notation $\|\cdot\|_{\infty}$ stands for the induced operator $\max -$norm.

Definition 3.12 (Functions with two variables). *Assume $k, l \in \mathbb{N}$ and $\alpha \in \mathbb{R}$. Then the notation $\mathbf{CG}^{k,\alpha}(\Omega\times\Lambda; X)$ stands for the class of functions $f : \Omega\times\Lambda \to X \subseteq \mathbb{R}$ such that $f(\cdot, t) \in \mathbf{C}^{k}(\Omega; X)$ for every fixed $t \in \Lambda$ and $f(z, \cdot) \in \mathbf{G}^{\delta}_{\alpha}(\Lambda; X)$ for every fixed $z \in \Omega$. Consequently, by $\mathbf{C}^{k,l}(\Omega_1 \times \Omega_2; X)$ the class of functions $g : \Omega_1 \times \Omega_2 \to X \subseteq \mathbb{R}$ is characterised, s.t. $g(\cdot, s) \in \mathbf{C}^{k}(\Omega_1; X)$ for every fixed $s \in \Omega_2$ and $g(z, \cdot) \in \mathbf{C}^{l}(\Omega_2; X)$ for every fixed $z \in \Omega_1$.*

Definition 3.13 (Functions with three variables). *Assume $k, l \in \mathbb{N}$ and $\alpha \in \mathbb{R}$. Then by $\mathbf{CG}^{k,l,\alpha}(\Omega_1 \times \Omega_2 \times \Lambda; X)$ the class of functions $f : \Omega_1 \times \Omega_2 \times \Lambda \to X \subseteq \mathbb{R}$ is denoted s.t. $f(\cdot, s, t) \in \mathbf{C}^{k}(\Omega; X)$ for every fixed $s \in \Omega_2$ and $t \in \Lambda$; $f(z, \cdot, t) \in \mathbf{C}^{l}(\Omega; X)$ for every fixed $z \in \Omega_1$ and $t \in \Lambda$; $f(z, s, \cdot) \in \mathbf{G}^{\delta}_{\alpha}(\Lambda; X)$ for every fixed $z \in \Omega_1$ and $s \in \Omega_2$.*

C.3 Useful Theorems and Lemmas

The following Lemmas summarise some intermediate results and support the convergence analysis in Chapter 4.

Theorem 3.16 (Extreme value theorem [36]). *Let $\bar{\Lambda}$ be a compact set and let $f : \bar{\Lambda} \to \mathbb{R}$ be a continuous function. Then f has a maximum (minimum) on $\bar{\Lambda}$, that is, there exists a $\nu \in \bar{\Lambda}$ such that $f(\lambda) \leq f(\nu)$ $(f(\lambda) \geq f(\nu))$ $\forall \lambda \in \bar{\Lambda}$.*

Theorem 3.17 (Upper bound for functions in two variables involving Gevrey classes). *Let $f(z, t)$ be continuous in $z \in [z_{\mathrm{l}}, z_{\mathrm{r}}]$ and be in Gevrey class of order α with respect to t and be defined on the open subset Λ. Then for every compact subset $\bar{\Lambda}$ of Λ the function f can be bounded by*

$$\sup_{(z,t)\in\mathfrak{D}} \left|\partial_t^k f(z, t)\right| \leq \delta^{k+1}(k!)^{\alpha} \tag{C.17}$$

with $\mathfrak{D} = [z_{\mathrm{l}}, z_{\mathrm{r}}] \times \bar{\Lambda}$.

Proof. Assuming $f(z, t)$ being in $\mathbf{CG}^{0,\alpha}([z_{\mathrm{l}}, z_{\mathrm{r}}] \times \Lambda; X)$ with $X \subseteq \mathbb{R}$ implies $f(z, t) \in \mathbf{C}^{0,\infty}([z_{\mathrm{l}}, z_{\mathrm{r}}] \times \Lambda; X)$. Furthermore, since the interval $[z_{\mathrm{l}}, z_{\mathrm{r}}]$ and $\bar{\Lambda}$ are individually compact the entire domain $[z_{\mathrm{l}}, z_{\mathrm{r}}] \times \bar{\Lambda}$ is compact. With this, considering the extreme value theorem yields that $\left|\partial_t^k f(z, t)\right|$ is bounded on every compact subset $[z_{\mathrm{l}}, z_{\mathrm{r}}] \times \bar{\Lambda}$, i.e., there exist a pair $(z_k, t_k) \in [z_\mathrm{l}, z_\mathrm{r}] \times \bar{\Lambda}$, $\forall k \in \mathbb{N}_0$, such that $\sup_{(z,t)\in[z_\mathrm{l},z_\mathrm{r}]\times\bar{\Lambda}} \left|\partial_t^k f(z,t)\right| \leq \left|\partial_t^k f(z_k,t_k)\right| = m_k$. Thus, taking Definition 3.10 into account since $f(z,t)$ is in t element of the Gevrey class of order α, then there exists a (sufficiently large) constant δ such that the kth−derivative of f with respect to t can be bounded by (C.17). More precisely δ has to satisfy

$$\sqrt[k+1]{\frac{m_k}{(k!)^\alpha}} \leq \delta < \infty \qquad \forall k \in \mathbb{N}_0 \,. \tag{C.18}$$

□

Corollary 3.5 (Upper bound for matrix-valued functions in two variables involving Gevrey classes)**.** *Let the matrix-valued function* $F(z,t)$ *be in* $\mathbf{CG}^{0,\alpha}\left([z_\mathrm{l}, z_\mathrm{r}] \times \Lambda; X^{m\times n}\right)$ *with elements* $f_{ij}(z,t) \in \mathbf{CG}^{0,\alpha_{ij}}\left([z_\mathrm{l}, z_\mathrm{r}] \times \Lambda; X\right)$ *and considering Definition 3.11. Then, with this and Theorem 3.17 the norm* (C.6) *of* F *can be bounded by*

$$\left\| \sup_{(z,t)\in\mathfrak{D}} \partial_t^k F(z,t) \right\|_\infty \leq \delta^{k+1}(k!)^\alpha \tag{C.19}$$

with $\mathfrak{D} = [z_\mathrm{l}, z_\mathrm{r}] \times \bar{\Lambda}$, $\delta = \max_{1\leq i\leq m}\left(\sum_{j=1}^n \delta_{ij}\right)$, *and* $\alpha = \max_{\substack{1\leq i\leq m\\ 1\leq j\leq n}} \alpha_{ij}$.

Lemma 3.18 (Leibnitz rule [10])**.** *Assuming* $f,\, g \in \mathbf{C}^l(\mathbb{R};\mathbb{R})$, *then for* $k,\, l \in \mathbb{N}$ *and* $t \in \mathbb{R}$ *it holds*

$$\partial_t^k\left[f(t)g(t)\right] = \sum_{l=0}^{k} \binom{k}{l} \partial_t^{k-l} f(t) \partial_t^l g(t) \,. \tag{C.20}$$

Lemma 3.19. *Combining the identity [52]*

$$\sum_{j=0}^{k} \binom{k}{j} (k-j+n)!(j+m)! = \frac{n!m!(k+n+m+1)!}{(n+m+1)!} \tag{C.21}$$

for $j,\, k,\, m,\, n \in \mathbb{N}$ *and the inequality [33]*

$$\sum_j a_j^\alpha \leq \left(\sum_j a_j\right)^\alpha \tag{C.22}$$

for $\alpha \geq 1$, $a_j \geq 0$ *allows to derive the upper bound estimation*

$$\sum_{j=0}^{k} \binom{k}{j} \left((k-j+n)!(j+m)!\right)^\alpha \leq \left(\frac{n!m!(k+n+m+1)!}{(n+m+1)!}\right)^\alpha \,. \tag{C.23}$$

Lemma 3.20 (Leibnitz integral rule [10])**.** *If the function* $f \in \mathbf{C}^{0,0}\left([z_\mathrm{l}, z_\mathrm{r}] \times [t_\mathrm{l}, t_\mathrm{r}]\,;\mathbb{R}\right)$ *and* $\partial_z f(z,t)$ *exists, and the functions* $a,\, b \in \mathbf{C}^1\left([z_\mathrm{l}, z_\mathrm{r}]\,;\mathbb{R}\right)$ *remain in the interval* $t_\mathrm{l} \leq t = a(z)$, $t = b(z) \leq t_\mathrm{r}$, *then for* $z_\mathrm{l} \leq z \leq z_\mathrm{r}$ *the differentiation under the integral sign applies*

$$\frac{\mathrm{d}}{\mathrm{d}z}\int_{a(z)}^{b(z)} f(z,t)\,\mathrm{d}t = f(z,b(z))\frac{\mathrm{d}}{\mathrm{d}z}b(z) - f(z,a(z))\frac{\mathrm{d}}{\mathrm{d}z}a(z) + \int_{a(z)}^{b(z)} \partial_z f(z,t)\,\mathrm{d}t \,. \tag{C.24}$$

Lemma 3.21 (Integral lemmas)**.** *Consider $z, \eta, \xi \in \mathbb{R}$ and $k \in \mathbb{N}_0$ then the identities*

$$\left| \int_{\xi}^{z} \int_{\xi}^{\eta} \frac{|\sigma - \xi|^k}{k!} \,\mathrm{d}\sigma \,\mathrm{d}\eta \right| = \frac{|z - \xi|^{k+2}}{(k+2)!} \tag{C.25}$$

$$\left| \int_{\xi}^{z} \frac{|\sigma - \xi|^k}{k!} \,\mathrm{d}\sigma \right| = \frac{|z - \xi|^{k+1}}{(k+1)!} \tag{C.26}$$

hold true.

Proof. In (C.25) the outer absolute value function applied to the integrals does make the absolute value of the integrands redundant. The proof is trivial for k even, or $z, \eta \geq \xi$, respectively. For k odd and $z, \eta \leq \xi$ note

$$\left| \int_{\xi}^{z} \int_{\xi}^{\eta} \frac{|\sigma - \xi|^k}{k!} \,\mathrm{d}\sigma \,\mathrm{d}\eta \right| = \left| \int_{\xi}^{z} \int_{\xi}^{\eta} \frac{(\xi - \sigma)^k}{k!} \,\mathrm{d}\sigma \,\mathrm{d}\eta \right| = \left| \frac{(\xi - z)^{k+2}}{(k+2)!} \right| = \frac{|z - \xi|^{k+2}}{(k+2)!} \,.$$

Note, that the same procedure may be applied to (C.26). □

Theorem 3.22 (Cauchy-Hadamard theorem [50])**.** *Let $\sum_{m=0}^{\infty} p_m z^m$ be a power series, and let ρ be its radius of convergence. Then ρ is given by*

$$\rho = \frac{1}{\limsup_{m\to\infty} |p_m|^{1/m}} \,. \tag{C.27}$$

If $\rho = 0$, i.e., $\limsup_{m\to\infty} |p_m|^{1/m} = \infty$, this is to be interpreted that the series is divergent. In case $\limsup_{m\to\infty} |p_m|^{1/m} = 0$, and hence $\rho = \infty$ the series converges absolutely for all z. For $0 < \rho < \infty$ the series converges absolutely for $|z| < \rho$.

Lemma 3.23 (d'Alembert ratio test [36])**.** *Let the coefficients p_m of the infinite power series $\sum_{m=0}^{\infty} p_m z^m$ satisfy $p_m \neq 0, \forall m \geq m_0 \in \mathbb{N}_0$ and let the following limit exist. Then the convergence radius ρ of the power series is given by*

$$\rho = \lim_{m\to\infty} \left| \frac{p_m}{p_{m+1}} \right| \,. \tag{C.28}$$

Theorem 3.24 (Cauchy Product of absolutely convergent series of functions [50])**.** *Let the power series $a(z) = \sum_{m=0}^{\infty} \alpha_m z^m$, $b(z) = \sum_{m=0}^{\infty} \beta_m z^m$ and the function series $f(z,t) = \sum_{m=0}^{\infty} \phi_m(z,t)$, $g(z,t) = \sum_{m=0}^{\infty} \psi_m(z,t)$ with $\sup_{t\in\mathbb{R}_{t_0}^{+}} |\phi_m(z,t)| \leq |\alpha_m| \, |z|^m$, $\sup_{t\in\mathbb{R}_{t_0}^{+}} |\psi_m(z,t)| \leq |\beta_m| \, |z|^m$ converge absolutely in z on the interval $(0,\rho)$, then the Cauchy product $h(z,t) = \sum_{m=0}^{\infty} \chi_m(z,t)$ with $\chi_m(z,t) = \sum_{l=0}^{m} \phi_m(z,t)\psi_{m-l}(z,t)$ also converges absolutely in z with convergence radius ρ, and we have*

$$f(z,t)g(z,t) = \sum_{m=0}^{\infty} \phi_m(z,t) \sum_{m=0}^{\infty} \psi_m(z,t) = \sum_{m=0}^{\infty} \sum_{l=0}^{m} \phi_l(z,t)\psi_{m-l}(z,t) = h(z,t) \tag{C.29}$$

Proof. The result can be verified in two steps. First, absolute convergence has to be shown. Since $a(z)$ and $b(z)$ are absolutely convergent there exists a constant c with $\alpha_m \leq c/\left|s\right|^m$ and $\beta_m \leq c/\left|s\right|^m$ for $0 < |s| < \rho$. With this consider the upper bound estimation

$$\begin{aligned}\frac{\sup_{t\in\mathbb{R}^+_{t_0}}|\chi_m(s,t)|}{|s|^m} &\leq \frac{1}{|s|^m}\sum_{l=0}^{m}\sup_{t\in\mathbb{R}^+_{t_0}}|\phi_l(s,t)|\sup_{t\in\mathbb{R}^+_{t_0}}|\psi_{m-l}(s,t)| \\ &\leq \sum_{l=0}^{m}|\alpha_l|\,|\beta_{m-l}| \leq \frac{(m+1)c^2}{|s|^m}.\end{aligned} \tag{C.30}$$

With this and Theorem 3.22 it follows that

$$\lim_{m\to\infty}\sup\frac{\sup_{t\in\mathbb{R}^+_{t_0}}|\chi_m(s,t)|^{1/m}}{|s|} \leq \lim_{m\to\infty}\sup\frac{(m+1)^{1/m}c^{2/m}}{|s|} \leq 1/\left|s\right| \leq 1/\rho \tag{C.31}$$

which proofs that the series of the Cauchy product $\sum_{l=0}^{m}\phi_m(z,t)\psi_{m-l}(z,t)$ converges absolutely in z on the interval $(0,\rho)$. Consequently, there exists the upper bound estimation $\sup_{t\in\mathbb{R}^+_{t_0}}|\chi_m(z,t)| \leq |\gamma_m|\,|z|^m$.

Second, it needs to be proven that the Cauchy product of the function series converges uniformly, i.e., $h(z,t) \to f(z,t)g(z,t)$. However in this case it is more convenient to proof the opposite direction, that fg uniformly approaches h. Thus, consider the partial sums

$$f_M(z,t) = \sum_{m=0}^{M}\phi_m(z,t) \quad \text{and} \quad g_M(z,t) = \sum_{m=0}^{M}\psi_m(z,t)\,, \tag{C.32}$$

then for given $\epsilon > 0$ there exists $m_0 < \infty$ and it holds

$$\begin{aligned}&\sup_{t\in\mathbb{R}^+_{t_0}}|h(z,t) - f_M(z,t)g_M(z,t)| \\ &\qquad = \sup_{t\in\mathbb{R}^+_{t_0}}\left|\sum_{m=0}^{\infty}\sum_{l=0}^{m}\phi_m(z,t)\psi_{m-l}(z,t) - \sum_{m=0}^{M}\phi_m(z,t)\sum_{m=0}^{M}\psi_m(z,t)\right| \\ &\qquad = \sup_{t\in\mathbb{R}^+_{t_0}}\left|\sum_{m=0}^{\infty}\sum_{l=0}^{m}\phi_m(z,t)\psi_{m-l}(z,t) - \sum_{m=0}^{M}\sum_{l=0}^{m}\psi_m(z,t)\psi_{m-l}(z,t)\right| \\ &\qquad = \sup_{t\in\mathbb{R}^+_{t_0}}\left|\sum_{m=M+1}^{\infty}\chi_m(z,t)\right| \leq \sum_{m=M+1}^{\infty}\sup_{t\in\mathbb{R}^+_{t_0}}|\chi_m(z,t)| \leq \sum_{m=M+1}^{\infty}|\gamma_m|\,|z|^m < \epsilon\,.\end{aligned} \tag{C.33}$$

With the given premises $\sum_{m=M+1}^{\infty}|\gamma_m|\,|z|^m$ gets arbitrary small for sufficiently large $M \geq m_0$, and hence

$$f(z,t)g(z,t) = \lim_{M\to\infty}f_M(z,t)g_M(z,t) = h(z,t)\,. \tag{C.34}$$

□

C.4 Important Inequalities

Please note [45] for the following discussion on inequalities.

Definition 3.14 (Young's inequality in its most elementary vectorial version)**.**

$$\boldsymbol{u}^T\boldsymbol{v} \leq \frac{1}{2}\boldsymbol{u}^T P\boldsymbol{u} + \frac{1}{2}\boldsymbol{v}^T P^{-1}\boldsymbol{v} \quad \forall P > 0\,. \tag{C.35}$$

Proof. Assume $Q = \operatorname{diag}[q_{11}, q_{22}, \ldots, q_{nn}] \in \mathbb{R}^{n\times n}$ and be positive definite, then

$$\begin{aligned}\left\|Q\boldsymbol{u} - Q^{-1}\boldsymbol{v}\right\|_2^2 &= \left(Q\boldsymbol{u} - Q^{-1}\boldsymbol{v}\right)^T \left(Q\boldsymbol{u} - Q^{-1}\boldsymbol{v}\right) = \\ &= \boldsymbol{u}^T Q^2\boldsymbol{u} - 2\boldsymbol{u}^T\boldsymbol{v} + \boldsymbol{v}^T \left(Q^2\right)^{-1} \boldsymbol{v} \geq 0\end{aligned} \tag{C.36}$$

With this and $P = Q^2 > 0$ directly follows (C.35). □

Definition 3.15 (Cauchy-Schwarz inequality for standard scalar product)**.** *Assume the real vectors* $\boldsymbol{f}$, $\boldsymbol{g} \in \mathbb{R}^n$*, then*

$$\left|\boldsymbol{f}^T\boldsymbol{g}\right| \leq \|\boldsymbol{f}\|_2 \, \|\boldsymbol{g}\|_2 \tag{C.37}$$

Definition 3.16 (Integral formulation of the Cauchy-Schwarz inequality)**.** *Assume the real functions* $\boldsymbol{f}$, $\boldsymbol{g} \in \mathbf{L}^2\left([0,\ell]\,;\mathbb{R}^n\right)$*, then*

$$\left|\int_0^\ell \boldsymbol{f}^T(z)\boldsymbol{g}(z)\mathrm{d}z\right| \leq \left(\int_0^\ell \|\boldsymbol{f}(z)\|_2^2 \,\mathrm{d}z\right)^{\frac{1}{2}} \left(\int_0^\ell \|\boldsymbol{g}(z)\|_2^2 \,\mathrm{d}z\right)^{\frac{1}{2}} \tag{C.38}$$

Lemma 3.25 (Poincaré inequality on a real interval)**.** *Let* $\boldsymbol{x}(z) \in \mathbf{H}^1\left([0,\ell]\,;\mathbb{R}^n\right)$ *and the weight* $P \in \mathbb{R}^{n\times n}$ *be diagonal and positive definite. Then the following inequalities hold*

$$\int_0^\ell \boldsymbol{x}^T(z)P\boldsymbol{x}(z)\mathrm{d}z \leq \frac{r}{r-1}\ell\boldsymbol{x}^T(0)P\boldsymbol{x}(0) + \frac{r^2}{r-1}\ell^2 \int_0^\ell \partial_z\boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z\,, \tag{C.39}$$

$$\int_0^\ell \boldsymbol{x}^T(z)P\boldsymbol{x}(z)\mathrm{d}z \leq \frac{r}{r-1}\ell\boldsymbol{x}^T(\ell)P\boldsymbol{x}(\ell) + \frac{r^2}{r-1}\ell^2 \int_0^\ell \partial_z\boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z\,. \tag{C.40}$$

for $r > 1$*.*

Proof. The lemma can be verified by considering the partial integration

$$\int_0^\ell 1\cdot\boldsymbol{x}^T(z)P\boldsymbol{x}(z)\mathrm{d}z = \left[z\boldsymbol{x}^T(z)P\boldsymbol{x}(z)\right]_0^\ell - 2\int_0^\ell z\boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z\,. \tag{C.41}$$

Due to

$$\begin{aligned}0 \leq \left\|\sqrt{P/r}\boldsymbol{x}(z) + z\sqrt{Pr}\partial_z\boldsymbol{x}(z)\right\|_2^2 \leq \frac{1}{r}\boldsymbol{x}(z)^T P\boldsymbol{x}(z) + 2z\boldsymbol{x}(z)^T P\partial_z\boldsymbol{x}(z) \\ + rz^2\partial_z\boldsymbol{x}(z)^T P\partial_z\boldsymbol{x}(z)\,,\end{aligned}$$

or

$$
\begin{aligned}
-2\int_0^\ell z\boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z &\leq \frac{1}{r}\int_0^\ell \boldsymbol{x}(z)^T P\boldsymbol{x}(z)\mathrm{d}z + r\int_0^\ell z^2\partial_z\boldsymbol{x}(z)^T P\partial_z\boldsymbol{x}(z)\mathrm{d}z \\
&\leq \frac{1}{r}\int_0^\ell \boldsymbol{x}(z)^T P\boldsymbol{x}(z)\mathrm{d}z + r\ell^2\int_0^\ell \partial_z\boldsymbol{x}(z)^T P\partial_z\boldsymbol{x}(z)\mathrm{d}z \,,
\end{aligned}
$$

respectively, follows from (C.41) that

$$
\left(1-\frac{1}{r}\right)\int_0^\ell \boldsymbol{x}^T(z)P\boldsymbol{x}(z)\mathrm{d}z \leq \ell\boldsymbol{x}^T(\ell)P\boldsymbol{x}(\ell) + r\ell^2\int_0^\ell \partial_z\boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z \tag{C.42}
$$

which confirms (C.40). For the verification of (C.39) consider the identity $\left[\boldsymbol{x}^T(z)P\boldsymbol{x}(z)\right]_0^\ell = 2\int_0^\ell \boldsymbol{x}^T(z)P\partial_z\boldsymbol{x}(z)\mathrm{d}z$. This directly allows to exchange z with $(z-\ell)$ in (C.41) and the following estimations and as a result verifies the first inequality. □

Lemma 3.26 (Agmon's inequality on a real interval)**.** *For the vector-valued function* $\boldsymbol{x}(z) \in \mathbf{H}^1\left([0,\ell]\,;\mathbb{R}^n\right)$ *and* $\mathbf{X} = \left(\mathbf{L}^2\left([0,\ell]\,;\mathbb{R}^n\right)\right.$ *the following inequalities hold*

$$
\max_{z\in[0,\ell]} \|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{x}(0)\|_2^2 + \|\boldsymbol{x}(z)\|_{\mathbf{X}}^2 \|\partial_z\boldsymbol{x}(z)\|_{\mathbf{X}}^2 \,, \tag{C.43}
$$

$$
\max_{z\in[0,\ell]} \|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{x}(\ell)\|_2^2 + 2\|\boldsymbol{x}(z)\|_{\mathbf{X}}^2 \|\partial_z\boldsymbol{x}(z)\|_{\mathbf{X}}^2 \,. \tag{C.44}
$$

Proof. Again let us start with

$$
\|\boldsymbol{x}(z)\|^2 = \|\boldsymbol{x}(0)\|^2 + 2\int_0^z \boldsymbol{x}^T(s)\partial_s\boldsymbol{x}(s)\mathrm{d}s \leq \|\boldsymbol{x}(0)\|^2 + 2\int_0^\ell \left|\boldsymbol{x}^T(s)\partial_s\boldsymbol{x}(s)\right|\mathrm{d}s \,. \tag{C.45}
$$

Obviously the right hand side does not depend on z, therefore

$$
\max_{z\in[0,\ell]} \|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{x}(0)\|^2 + 2\int_0^\ell \left|\boldsymbol{x}^T(z)\partial_z\boldsymbol{x}(z)\right|\mathrm{d}z \,. \tag{C.46}
$$

Utilising the Cauchy-Schwartz inequality yields (C.43). The second inequality can be derived in similar fashion. □

Appendix D

Auxiliary Notes for Backstepping based Control of Coupled Parabolic PDEs

D.1 Kernel Equations for the Inverse Backstepping Transformation

The upcoming section deals with the determination of the integral kernel G of the inverse backstepping transformation

$$\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) = \boldsymbol{w}(z,t) + \int_0^z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\,. \tag{D.1}$$

for control design purposes. Therefore let us note down its temporal and spatial derivatives as

$$\partial_t\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) = \partial_t\boldsymbol{w}(z,t) + \int_0^z \partial_t G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s + \int_0^z G(z,s,t)\partial_t\boldsymbol{w}(s,t)\,\mathrm{d}s\,, \tag{D.2}$$

$$\partial_z\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) = \partial_z\boldsymbol{w}(z,t) + G(z,z,t)\boldsymbol{w}(z,t) + \int_0^z \partial_z G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\,, \tag{D.3}$$

$$\begin{aligned}\partial_z^2\tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) &= \partial_z^2\boldsymbol{w}(z,t) + \frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t)\boldsymbol{w}(z,t) + G(z,z,t)\partial_z\boldsymbol{w}(z,t)\\ &\qquad + \partial_z G(z,z,t)\boldsymbol{w}(z,t) + \int_0^z \partial_z^2 G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\,.\end{aligned} \tag{D.4}$$

Inserting (D.1)-(D.4) into the PIDE (5.12a) of the target state $\boldsymbol{w}$, then this leads to

$$\begin{aligned}\int_0^z \partial_t G(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s + \int_0^z G(z,s,t)\partial_t\boldsymbol{w}(s,t)\,\mathrm{d}s &= \Big[A\frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t)\\ + A\partial_z G(z,z,t) - B(z,t)G(z,z,t) + D(t) - M(z,t) + C(z,t)\Big]\boldsymbol{w}(z,t)\\ + AG(z,z,t)\partial_z\boldsymbol{w}(z,t) - F(z,t)\boldsymbol{w}(0,t) - \int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s\\ + \int_0^z \Big[A\partial_z^2 K(z,s,t) - B(z,t)\partial_z G(z,s,t) + C(z,t)G(z,s,t)\Big]\boldsymbol{w}(s,t)\,\mathrm{d}s\,.\end{aligned} \tag{D.5}$$

With this, utilising (5.12a) again, and rearranging and collecting the terms on the right hand side gives

$$
\begin{aligned}
\mathbf{0} = &-\int_0^z G(z,s,t)\Big[A\partial_s^2\boldsymbol{w}(s,t) - B(s,t)\partial_s\boldsymbol{w}(s,t) - \left(D(t) - M(s,t)\right)\boldsymbol{w}(s,t)\Big]\,\mathrm{d}s\\
&+\Big[A\frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t) + A\partial_z G(z,z,t) - B(z,t)G(z,z,t) + D(t) - M(z,t) + C(z,t)\Big]\boldsymbol{w}(z,t)\\
&+AG(z,z,t)\partial_z\boldsymbol{w}(z,t) - \left[\int_0^z G(z,s,t)F(s,t)\,\mathrm{d}s + F(z,t)\right]\boldsymbol{w}(0,t)\\
&-\int_0^z\Big[\partial_t G(z,s,t) - A\partial_z^2 G(z,s,t) + B(z,t)\partial_z G(z,s,t) - C(z,t)G(z,s,t)\Big]\boldsymbol{w}(s,t)\,\mathrm{d}s\\
&-\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^z\int_0^s G(z,s,t)H(s,p,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(p,t)\mathrm{d}p\,\mathrm{d}s\,.
\end{aligned}
\tag{D.6}
$$

Similarly, as conducted in Section 5.1.2 the expressions with $\partial_s^2\boldsymbol{w}$ and $\partial_s\boldsymbol{w}$ require partial integration which can be stated as

$$
\begin{aligned}
\int_0^z G(z,s,t)A\partial_s^2\boldsymbol{w}(s,t)\,\mathrm{d}s = &\Big[G(z,s,t)A\partial_s\boldsymbol{w}(s,t)\Big]_{s=0}^{s=z}\\
&-\Big[\partial_s G(z,s,t)A\boldsymbol{w}(s,t)\Big]_{s=0}^{s=z} + \int_0^z \partial_s^2 G(z,s,t)A\boldsymbol{w}(s,t)\,\mathrm{d}s\,,
\end{aligned}
\tag{D.7}
$$

and

$$
\begin{aligned}
\int_0^z G(z,s,t)B(s,t)\partial_s\boldsymbol{w}(s,t)\,\mathrm{d}s = &\Big[G(z,s,t)B(s,t)\boldsymbol{w}(s,t)\Big]_{s=0}^{s=z}\\
&-\int_0^z\Big[\partial_s G(z,s,t)B(s,t) + G(z,s,t)\partial_s B(s,t)\Big]\boldsymbol{w}(s,t)\,\mathrm{d}s\,.
\end{aligned}
\tag{D.8}
$$

Acknowledging these results in (D.6) finally leads to

$$
\begin{aligned}
\mathbf{0} = &\Big[A\frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t) + A\partial_z G(z,z,t) + \partial_s G(z,z,t)A - B(z,t)G(z,z,t)\\
&+G(z,z,t)B(z,t) + D(t) - M(z,t) + C(z,t)\Big]\boldsymbol{w}(z,t) + G(z,0,t)A\partial_z\boldsymbol{w}(0,t)\\
&-\Big[\int_0^z G(z,s,t)F(s,t)\,\mathrm{d}s + F(z,t) + \partial_s G(z,0,t)A + G(z,0,t)B(0,t)\Big]\boldsymbol{w}(0,t)\\
&-\Big[G(z,z,t)A - AG(z,z,t)\Big]\partial_z\boldsymbol{w}(z,t) - \int_0^z\Big[\partial_t G(z,s,t) - A\partial_z^2 G(z,s,t)\\
&+\partial_s^2 G(z,s,t)A + B(z,t)\partial_z G(z,s,t) + \partial_s G(z,s,t)B(s,t) - C(z,t)G(z,s,t)\\
&-G(z,s,t)\left(D(t) - M(s,t) - \partial_s B(s,t)\right)\Big]\boldsymbol{w}(s,t)\mathrm{d}s\\
&-\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^z\int_0^s G(z,s,t)H(s,p,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(p,t)\mathrm{d}p\,\mathrm{d}s\,.
\end{aligned}
\tag{D.9}
$$

The derived equation has to be satisfied for the entire triangular spatial domain $(z,s) \in \mathfrak{D}_{\mathrm{G}}(\ell_{\mathrm{N}}) = \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) := \{(z,s) \in \mathbb{R}^2 \,|\, z \in [0,\ell_{\mathrm{N}}]\,,\, s \in [0,z]\}$ and for all $t \in \mathbb{R}_{t_0}^+$. With the introduction of the compensation matrix $J(z,s,t)$, as a counterpart to $H(z,s,t)$, the kernel equations of the inverse transformation can be written as the following PDE

$$\begin{aligned}\partial_t G(z,s,t) = A\partial_z^2 G(z,s,t) - \partial_s^2 G(z,s,t)A - B(z,t)\partial_z G(z,s,t)\\ - \partial_s G(z,s,t)B(s,t) + C(z,t)G(z,s,t)\\ + G(z,s,t)\left(D(t) - M(s,t) - \partial_s B(s,t)\right) - J(z,s,t)\,,\end{aligned} \tag{D.10}$$

with the boundary conditions

$$\begin{aligned}A\frac{\mathrm{d}}{\mathrm{d}z}G(z,z,t) + A\partial_z G(z,z,t) + \partial_s G(z,z,t)A = B(z,t)G(z,z,t)\\ - G(z,z,t)B(z,t) - D(t) + M(z,t) - C(z,t)\,,\\ G(z,z,t)A = AG(z,z,t)\,,\\ G(z,0,t) = 0_{n,n}\end{aligned} \tag{D.11}$$

at $z = s$ and $s = 0$ and the two further conditions

$$\int_0^z G(z,s,t)F(s,t)\,\mathrm{d}s + F(z,t) + \partial_s G(z,0,t)A = 0_{n,n}\,, \tag{D.12}$$

$$\begin{aligned}\int_0^z H(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s - \int_0^z J(z,s,t)\boldsymbol{w}(s,t)\,\mathrm{d}s\\ + \int_0^z\int_0^s G(z,s,t)H(s,p,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(p,t)\mathrm{d}p\,\mathrm{d}s = \boldsymbol{0}_n\,.\end{aligned} \tag{D.13}$$

Obviously, the equations (D.10) and (D.11) match exactly the statements of Remark 5.19. This leaves behind the two latter conditions which are discussed in the following.

Considering the setting $F(z,t) = -\partial_s K(z,0,t)A$ for dynamic BCs the first condition (D.12) leads to

$$\partial_s G(z,0,t) - \partial_s K(z,0,t) = \int_0^z G(z,p,t)\partial_s K(p,0,t)\mathrm{d}p\,. \tag{D.14}$$

This is fulfilled by all means and can be proven easily by utilising the spatial derivative of (5.11) with respect to s and subsequent determination of the resulting expression at $s = 0$. The second condition (D.13) requires a more profound examination which shall be the objective of the following section.

D.2 Transversal Condition of the Compensation Terms

As discussed in Section 5.1.1 the integral kernels K and G are connected by means of the Volterra integral transformation of the second kind

$$\boldsymbol{w}(z,t) = \tilde{\boldsymbol{x}}_{\mathrm{c}}(z,t) - \int_0^z K(z,s,t)\tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \tag{D.15}$$

or (D.1), respectively. However comparing the PDEs (5.29a) and (D.10) both contain the same compensation matrix M which depends on the corresponding kernel at the boundary $z = s$. More precisely speaking the setting of the matrix M has to fulfil

$$\begin{aligned} M(z,t) &= 2\Delta A \frac{\mathrm{d}}{\mathrm{d}z} G^{\mathrm{d}}(z,z,t) + G^{\mathrm{d}}(z,z,t) B^{\mathrm{c}}(z,t) - B^{\mathrm{c}}(z,t) G^{\mathrm{d}}(z,z,t) + C^{\mathrm{c}}(z,t) \\ &\overset{!}{=} 2\Delta A \frac{\mathrm{d}}{\mathrm{d}z} K^{\mathrm{d}}(z,z,t) + K^{\mathrm{d}}(z,z,t) B(z,t) - B(z,t) K^{\mathrm{d}}(z,z,t) + C^{\mathrm{c}}(z,t)\,, \end{aligned} \tag{D.16}$$

when the compensation strategy *non-commutative compensation* as discussed in Section 5.1.2 is applied. Note, that this covers the setting for the other strategy *coupling compensation* as well. Since $K^{\mathrm{d}}(z,z,t)$ and consequently $G^{\mathrm{d}}(z,z,t)$ can be directly determined by the parameter setting of ΔA, B, and C^{c} the condition is fulfilled by all means. An alternative way to proof (D.16) is to evaluate (5.10) and its spatial derivatives at $z = s$, i.e.,

$$\begin{aligned} &G(z,z,t) = K(z,z,t)\,, \\ &\partial_z G(z,z,t) - \partial_z K(z,z,t) = G(z,z,t)K(z,z,t) \\ &\partial_s G(z,z,t) - \partial_s K(z,z,t) = -G(z,z,t)K(z,z,t) \end{aligned} \tag{D.17}$$

and consequently

$$\frac{\mathrm{d}}{\mathrm{d}z} G(z,z,t) = \frac{\mathrm{d}}{\mathrm{d}z} K(z,z,t)\,. \tag{D.18}$$

At this stage it has to be pointed out that from a diagonal $K = K^{\mathrm{d}}$ it directly follows a diagonal $G = G^{\mathrm{d}}$.

Furthermore, the second condition (D.13) needs to be valid for the spatial domain $\mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}})$, defined as mentioned above, and $t \in \mathbb{R}^{+}_{t_0}$. Applying the backstepping transformation (D.15) to (D.13) it can be written as

$$\begin{aligned} &\int_0^z \left[J(z,s,t) - H(z,s,t)\right] \tilde{\boldsymbol{x}}_{\mathrm{c}}(s,t)\,\mathrm{d}s \\ &\quad= \int_0^z \int_0^s \left[J(z,s,t)K(s,p,t) + G(z,s,t)H(s,p,t)\right] \tilde{\boldsymbol{x}}_{\mathrm{c}}(p,t)\mathrm{d}p\,\mathrm{d}s\,. \end{aligned} \tag{D.19}$$

By changing the sequence of double integration one can easily derive the equation

$$J(z,s,t) - H(z,s,t) = \int_s^z \left[J(z,p,t)K(p,s,t) + G(z,p,t)H(p,s,t)\right] \mathrm{d}p\,. \tag{D.20}$$

In the following this condition shall be denoted as the *transversal compensation condition (TCC)*.

From this, it is referred to Page 96 and Page 98 where two different compensation strategies are introduced and the next paragraphs shall show that (D.20) holds true for both strategies. More precisely speaking, if a strategy is chosen, it always has to be applied to both kernel equations, i.e., for the PDEs of K and of G. Then it can be shown that the TCC applies. Therefore the following discussion starts with the verification of the TCC for the compensation strategy *coupling compensation*.

Due to linearity it is allowed to separate compensation terms $M(z,t) = M_B(z,t) + M_C(z,t)$, $H(z,s,t) = H_B(z,s,t) + H_C(z,s,t)$ and $J(z,s,t) = J_B(z,s,t) + J_C(z,s,t)$. With this, for simplicity the verification is firstly conducted for terms where the parameter matrix C is involved. It can simply be reached by the setting $B(z,t) := 0_{n,n}$. This leads to

$$\begin{aligned}
M_C(z,t) &= C^{\mathrm{c}}(z,t)\\
H_C(z,s,t) &= M_C(z,t)K(z,s,t) - K(z,s,t)C^{\mathrm{c}}(s,t)\\
J_C(z,s,t) &= C^{\mathrm{c}}(z,t)G(z,s,t) - G(z,s,t)M_C(s,t)
\end{aligned} \tag{D.21}$$

Inserting this into (D.20) gives

$$\begin{aligned}
&C^{\mathrm{c}}(z,t)\left(G(z,s,t) - K(z,s,t)\right) - \left(G(z,s,t) - K(z,s,t)\right)C^{\mathrm{c}}(s,t)\\
&\quad= \int_s^z \Big[\left(C^{\mathrm{c}}(z,t)G(z,p,t) - G(z,p,t)M_C(p,t)\right)K(p,s,t)\\
&\qquad + G(z,p,t)\left(M_C(p,t)K(p,s,t) - K(p,s,t)C^{\mathrm{c}}(s,t)\right)\Big]\mathrm{d}p\\
&\quad= \int_s^z \Big[C^{\mathrm{c}}(z,t)G(z,p,t)K(p,s,t) - G(z,p,t)K(p,s,t)C^{\mathrm{c}}(s,t)\Big]\mathrm{d}p\,.
\end{aligned} \tag{D.22}$$

Just by rearranging the terms (D.22) can be simplified to

$$\begin{aligned}
&C^{\mathrm{c}}(z,t)\left(G(z,s,t) - K(z,s,t) - \int_s^z G(z,p,t)K(p,s,t)\mathrm{d}p\right)\\
&\quad= \left(G(z,s,t) - K(z,s,t) - \int_s^z G(z,p,t)K(p,s,t)\mathrm{d}p\right)C^{\mathrm{c}}(s,t)\,,
\end{aligned} \tag{D.23}$$

which is valid for all $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0}$ since (5.11) has to be applicable by all means. Next, the same procedure is applied for terms where the convection parameter matrix B is place. For this proof it is necessary to compute the spatial derivatives of (5.11), in the following written as

$$\begin{aligned}
\partial_z G(z,s,t) - \partial_z K(z,s,t) &= G(z,z,t)K(z,s,t) + \int_s^z \partial_z G(z,p,t)K(p,s,t)\,\mathrm{d}p\\
\partial_s G(z,s,t) - \partial_s K(z,s,t) &= -G(z,s,t)K(s,s,t) + \int_s^z G(z,p,t)\partial_s K(p,s,t)\,\mathrm{d}p\,.
\end{aligned} \tag{D.24}$$

Moreover, for this case the setting $C := 0_{n,n}$ leads to the compensation terms

$$\begin{aligned}
M_B(z,t) &= G^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(z,t) - B^{\mathrm{c}}(z,t)G^{\mathrm{d}}(z,z,t)\\
&= K^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(z,t) - B^{\mathrm{c}}(z,t)K^{\mathrm{d}}(z,z,t)\\
H_B(z,s,t) &= -B^{\mathrm{c}}(z,t)\partial_z K(z,s,t) - \partial_s K(z,s,t)B^{\mathrm{c}}(s,t)\\
&\quad + M_B^{\mathrm{c}}(z,t)K(z,s,t) - K(z,s,t)\partial_s B^{\mathrm{c}}(s,t)\\
J_B(z,s,t) &= -B^{\mathrm{c}}(z,t)\partial_z G(z,s,t) - \partial_s G(z,s,t)B^{\mathrm{c}}(s,t)\\
&\quad - G(z,s,t)\left(M_B^{\mathrm{c}}(s,t) + \partial_s B^{\mathrm{c}}(s,t)\right)\,.
\end{aligned} \tag{D.25}$$

First, the evaluation of the left hand side of (D.20) gives

$$
\begin{aligned}
J_B(z,s,t) - H_B(z,s,t) &= B^{\mathrm{c}}(z,t)\left(\partial_z K(z,s,t) - \partial_z G(z,s,t)\right) \\
&\quad + \left(\partial_s K(z,s,t) - \partial_s G(z,s,t)\right) B^{\mathrm{c}}(s,t) + \left(K(z,s,t) - G(z,s,t)\right)\partial_s B^{\mathrm{c}}(s,t) \\
&\quad - M_B^{\mathrm{c}}(z,t)K(z,s,t) - G(z,s,t)M_B^{\mathrm{c}}(s,t) \\
&= G(z,s,t)B^{\mathrm{c}}(s,t)K^{\mathrm{d}}(s,s,t) - G^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(z,t)K(z,s,t) \\
&\quad + B^{\mathrm{c}}(z,t)\big(\partial_z K(z,s,t) - \partial_z G(z,s,t) + G^{\mathrm{d}}(z,z,t)K(z,s,t)\big) \\
&\quad + \big(\partial_s K(z,s,t) - \partial_s G(z,s,t) - G(z,s,t)K^{\mathrm{d}}(s,s,t)\big) B^{\mathrm{c}}(s,t) \\
&\quad + \left(K(z,s,t) - G(z,s,t)\right)\partial_s B^{\mathrm{c}}(s,t)
\end{aligned}
\tag{D.26}
$$

and second, the right hand side leads to the expressions

$$
\begin{aligned}
&\int_s^z \left[J_B(z,p,t)K(p,s,t) + G(z,p,t)H_B(p,s,t)\right]\mathrm{d}p \\
&\quad = -\int_s^z \Big[B^{\mathrm{c}}(z,t)\partial_z G(z,p,t) + \partial_p G(z,p,t)B^{\mathrm{c}}(p,t) \\
&\qquad - G(z,p,t)\left(M_B^{\mathrm{c}}(p,t) + \partial_p B^{\mathrm{c}}(p,t)\right)\Big]K(p,s,t)\mathrm{d}p \\
&\qquad - \int_s^z G(z,p,t)\Big[B^{\mathrm{c}}(p,t)\partial_p K(p,s,t) + \partial_s K(p,s,t)B^{\mathrm{c}}(s,t) \\
&\qquad - M_B^{\mathrm{c}}(p,t)K(p,s,t) + K(p,s,t)\partial_s B^{\mathrm{c}}(s,t)\Big]\mathrm{d}p = \\
&\quad = -B^{\mathrm{c}}(z,t)\int_s^z \partial_z G(z,p,t)K(p,s,t)\mathrm{d}p - \int_s^z G(z,p,t)\partial_s K(p,s,t)\mathrm{d}p\, B^{\mathrm{c}}(s,t) \\
&\qquad - \int_s^z G(z,p,t)K(p,s,t)\mathrm{d}p\, \partial_s B^{\mathrm{c}}(s,t) - \int_s^z \Big[G(z,p,t)\partial_p B^{\mathrm{c}}(p,t)K(p,s,t) \\
&\qquad + \partial_p G(z,p,t)B^{\mathrm{c}}(p,t)K(p,s,t) + G(z,p,t)B^{\mathrm{c}}(p,t)\partial_p K(p,s,t)\Big]\mathrm{d}p\,.
\end{aligned}
\tag{D.27}
$$

By combing (D.26) and (D.27) according to (D.20), and taking (5.11) and the spatial derivatives (D.24) most of the terms can be compensated. The remaining parts are collected by

$$
\begin{aligned}
&G^{\mathrm{d}}(z,z,t)B^{\mathrm{c}}(z,t)K(z,s,t) - G(z,s,t)B^{\mathrm{c}}(s,t)K^{\mathrm{d}}(s,s,t) \\
&\quad = \int_s^z \Big[\partial_p G(z,p,t)B^{\mathrm{c}}(p,t)K(p,s,t) + G(z,p,t)\partial_p B^{\mathrm{c}}(p,t)K(p,s,t) \\
&\qquad + G(z,p,t)B^{\mathrm{c}}(p,t)\partial_p K(p,s,t)\Big]\mathrm{d}p\,.
\end{aligned}
\tag{D.28}
$$

which is satisfied for all $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0}$. Here the evidence is obvious, e.g., assume $F(z,p,s,t) = G(z,p,t)B^{\mathrm{c}}(p,t)K(p,s,t)$, then clearly by utilising the fundamental theorem of calculus

$$
F(z,z,s,t) - F(z,s,s,t) = \int_s^z \partial_p F(z,p,s,t)\mathrm{d}p
\tag{D.29}
$$

and consequently (D.28) holds true. This finalises the verification and gives evidence for the following statement. If the *coupling compensation* strategy is applied for the computation of the backstepping kernel K and for the calculation of the kernel G, then the compensation terms H and J fulfil the TCC. Now, the same procedure needs to be performed for the compensation strategy *non-commutative compensation*. Therefore, note the following discussion.

Again, the linear properties of the problem allow the verification of the TCC for terms which deal with diffusion and convection independently. In fact, the discussion for convection can be omitted since it is identical to the (D.26)-(D.28) when B is exchanged by ΔB. This only leaves the verification for the diffusion terms involving ΔA. For this, on top of (D.17), (D.18), and (D.24) further reflections on (5.11) are necessary, i.e. the second spatial derivatives yield

$$\begin{aligned}
\partial_z^2 G(z,s,t) - \partial_z^2 K(z,s,t) &= \frac{\mathrm{d}}{\mathrm{d}z} G(z,z,t)K(z,s,t) + G(z,z,t)\partial_z K(z,s,t) \\
&\quad + \partial_z G(z,z,t)K(z,s,t) + \int_s^z \partial_z^2 G(z,p,t)K(p,s,t)\,\mathrm{d}p\,, \\
\partial_s^2 G(z,s,t) - \partial_s^2 K(z,s,t) &= -\partial_s G(z,s,t)K(s,s,t) - G(z,s,t)\frac{\mathrm{d}}{\mathrm{d}s}K(s,s,t) \\
&\quad - G(z,s,t)\partial_s K(s,s,t) + \int_s^z G(z,p,t)\partial_s^2 K(p,s,t)\,\mathrm{d}p\,.
\end{aligned} \tag{D.30}$$

For this case the compensation terms can be derived from Section 5.1.2 as

$$\begin{aligned}
M_A(z,t) &= \Delta A \frac{\mathrm{d}}{\mathrm{d}z} K^{\mathrm{d}}(z,z,t) + \Delta A \partial_z K^{\mathrm{d}}(z,z,t) + \partial_s K^{\mathrm{d}}(z,z,t)\Delta A \\
&= \Delta A \frac{\mathrm{d}}{\mathrm{d}z} G^{\mathrm{d}}(z,z,t) + \Delta A \left(\partial_z G^{\mathrm{d}}(z,z,t) + K^{\mathrm{d}}(z,z,t)G^{\mathrm{d}}(z,z,t)\right) \\
&\quad + \left(\partial_s G^{\mathrm{d}}(z,z,t) - K^{\mathrm{d}}(z,z,t)G^{\mathrm{d}}(z,z,t)\right)\Delta A \\
&= \Delta A \frac{\mathrm{d}}{\mathrm{d}z} G^{\mathrm{d}}(z,z,t) + \Delta A \partial_z G^{\mathrm{d}}(z,z,t) + \partial_s G^{\mathrm{d}}(z,z,t)\Delta A\,, \\
H_A(z,s,t) &= \Delta A \partial_z^2 K(z,s,t) - \partial_s^2 K(z,s,t)\Delta A + M_A(z,t)K(z,s,t)\,, \\
J_A(z,s,t) &= \Delta A \partial_z^2 G(z,s,t) - \partial_s^2 G(z,s,t)\Delta A - G(z,s,t)M_A(s,t)\,.
\end{aligned} \tag{D.31}$$

First, considering this on the left hand side of (D.20) leads to the expression

$$\begin{aligned}
J_A(z,s,t) - H_A(z,s,t) &= \Delta A\Big(\partial_z^2 G(z,s,t) - \partial_z^2 K(z,s,t) - \frac{\mathrm{d}}{\mathrm{d}z} G^{\mathrm{d}}(z,z,t)K(z,s,t) \\
&\quad - \partial_z G^{\mathrm{d}}(z,z,t)K(z,s,t)\Big) - \Big(\partial_s^2 G(z,s,t) - \partial_s^2 K(z,s,t) \\
&\quad + G(z,s,t)\frac{\mathrm{d}}{\mathrm{d}s} K^{\mathrm{d}}(s,s,t) + G(z,s,t)\partial_s K^{\mathrm{d}}(s,s,t)\Big)\Delta A \\
&\quad - \partial_s G^{\mathrm{d}}(z,z,t)\Delta A K(z,s,t) - G(z,s,t)\Delta A \partial_z K^{\mathrm{d}}(s,s,t)
\end{aligned} \tag{D.32}$$

and second, applying the terms to the right hand side of (D.20) as well gives the integral equation

$$\begin{aligned}
&\int_s^z \left[J_A(z,p,t)K(p,s,t) + G(z,p,t)H_A(p,s,t)\right]\mathrm{d}p \\
&\quad = \int_s^z \left[\Delta A \partial_z^2 G(z,p,t) - \partial_p^2 G(z,p,t)\Delta A\right] K(p,s,t)\mathrm{d}p \\
&\qquad + \int_s^z G(z,p,t)\left[\Delta A \partial_p^2 K(p,s,t) - \partial_s^2 K(p,s,t)\Delta A\right]\mathrm{d}p
\end{aligned} \tag{D.33}$$

Combing the last two expressions and moving all terms to the right hand side the TCC evaluates to

$$
\begin{aligned}
0_{n,n} = \Delta A \Bigg(& \partial_z^2 G(z,s,t) - \partial_z^2 K(z,s,t) - \frac{\mathrm{d}}{\mathrm{d}z} G^{\mathrm{d}}(z,z,t) K(z,s,t) \\
& - \partial_z G^{\mathrm{d}}(z,z,t) K(z,s,t) - \int_s^z \partial_z^2 G(z,p,t) K(p,s,t) \mathrm{d}p \Bigg) \\
& - \Bigg(\partial_s^2 G(z,s,t) - \partial_s^2 K(z,s,t) + G(z,s,t) \frac{\mathrm{d}}{\mathrm{d}s} K^{\mathrm{d}}(s,s,t) \\
& + G(z,s,t) \partial_s K^{\mathrm{d}}(s,s,t) + \int_s^z G(z,p,t) \partial_s^2 K(p,s,t) \mathrm{d}p \Bigg) \Delta A \\
& - \partial_s G^{\mathrm{d}}(z,z,t) \Delta A K(z,s,t) - G(z,s,t) \Delta A \partial_z K^{\mathrm{d}}(s,s,t) \\
& + \int_s^z \partial_p^2 G(z,p,t) \Delta A K(p,s,t) - G(z,p,t) \Delta A \partial_p^2 K(p,s,t) \mathrm{d}p \,.
\end{aligned}
\tag{D.34}
$$

By considering (D.30) the terms in the parenthesis can be simplified to $\Delta A G^{\mathrm{d}}(z,z,t) \partial_z K(z,s,t) + \partial_s G(z,s,t) K^{\mathrm{d}}(s,s,t) \Delta A$. With this, and taking into account that ΔA is diagonal (D.34) can be written as

$$
\begin{aligned}
& \partial_s G^{\mathrm{d}}(z,z,t) \Delta A K(z,s,t) + G(z,s,t) \Delta A \partial_z K^{\mathrm{d}}(s,s,t) \\
& \qquad - \partial_s G(z,s,t) \Delta A K^{\mathrm{d}}(s,s,t) - G^{\mathrm{d}}(z,z,t) \Delta A \partial_z K(z,s,t) \\
& \quad = \int_s^z \partial_p^2 G(z,p,t) \Delta A K(p,s,t) - G(z,p,t) \Delta A \partial_p^2 K(p,s,t) \mathrm{d}p \,.
\end{aligned}
\tag{D.35}
$$

Then, extending the integrand with $0_{n,n} = \partial_p G(z,p,t) \Delta A \partial_p K(p,s,t) - \partial_p G(z,p,t) \Delta A \partial_p K(p,s,t)$ allows to split the remaining condition (D.35) into the two identities

$$
\begin{aligned}
& \partial_s G^{\mathrm{d}}(z,z,t) \Delta A K(z,s,t) - \partial_s G(z,s,t) \Delta A K^{\mathrm{d}}(s,s,t) \\
& \quad = \int_s^z \partial_p^2 G(z,p,t) \Delta A K(p,s,t) + \partial_p G(z,p,t) \Delta A \partial_p K(p,s,t) \mathrm{d}p \,,
\end{aligned}
\tag{D.36}
$$

$$
\begin{aligned}
& G^{\mathrm{d}}(z,z,t) \Delta A \partial_z K(z,s,t) - G(z,s,t) \Delta A \partial_z K^{\mathrm{d}}(s,s,t) \\
& \quad = \int_s^z G(z,p,t) \Delta A \partial_p^2 K(p,s,t) + \partial_p G(z,p,t) \Delta A \partial_p K(p,s,t) \mathrm{d}p \,,
\end{aligned}
\tag{D.37}
$$

which are obviously fulfilled for all $(z,s,t) \in \mathfrak{D}_{\mathrm{K}}(\ell_{\mathrm{N}}) \times \mathbb{R}^+_{t_0}$ by applying the fundamental theorem of calculus again, c.f. (D.28) and (D.29).

Again, with this it can be concluded that if the strategy *non-commutative compensation* is applied for the computation of the backstepping kernel K and for the calculation of the kernel G then the compensation terms H and J fulfil the TCC.

Now, in summary the evidence of the TCC allows to make further conclusions, i.e., with TCC satisfied (5.10) or (5.11), respectively, are applicable by all means. This further ensures that the backstepping transformation (5.6) between the target state $\boldsymbol{w}$ and the error state $\tilde{\boldsymbol{x}}_{\mathrm{c}}$ together with its inverse (5.7) are still valid for both compensation strategies. Moreover, if one of the kernels is diagonal, then the corresponding kernel of the inverse transformation is diagonal as well. Finally, it has to be pointed out that the same conclusion applies to the strategies for the observer design developed in Section 5.2.

Bibliography

1. Aoustin Y, Fliess M, Mounier H, Rouchon P, Rudolph J (1997) Theory and Practice in the Motion Planning and Control of a Flexible Robot Arm Using Mikusiński Operators. IFAC Proceedings Volumes 30(20):267–273, DOI 10.1016/S1474-6670(17)44275-3

2. Baccoli A, Pisano A (2015) Anticollocated Backstepping Observer Design for a Class of Coupled Reaction-Diffusion PDEs. Journal of Control Science and Engineering 2015:e164,274, DOI 10.1155/2015/164274

3. Baccoli A, Orlov Y, Pisano A (2014) On the boundary control of coupled reaction-diffusion equations having the same diffusivity parameters. In: 53rd IEEE Conference on Decision and Control, pp 5222–5228, DOI 10.1109/CDC.2014.7040205

4. Baccoli A, Pisano A, Orlov Y (2015) Boundary control of coupled reaction–diffusion processes with constant parameters. Automatica 54:80–90, DOI 10.1016/j.automatica.2015.01.032

5. Bajec IL, Heppner FH (2009) Organized flight in birds. Animal Behaviour 78(4):777–789, DOI 10.1016/j.anbehav.2009.07.007

6. Balakrishnan R, Ranganathan K (2012) A Textbook of Graph Theory. Springer Science & Business Media, google-Books-ID: mpgu6wgnZgYC

7. Balch T, Arkin RC (1998) Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation 14(6):926–939, DOI 10.1109/70.736776

8. Boonkumkrong N, Kuntanapreeda S (2014) Backstepping boundary control: An application to rod temperature control with Neumann boundary condition:. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering DOI 10.1177/0959651813520146, publisher: SAGE PublicationsSage UK: London, England

9. Bradski G (2000) The OpenCV Library. Dr Dobb's Journal of Software Tools

10. Bronstejn IN, Semendjaev KA, Musiol G (2001) Taschenbuch der Mathematik, 5th edn. Deutsch, Thun

11. Bullo F (2020) Lectures on Network Systems, 1st edn. Kindle Direct Publishing

12. Bullo F, Cortés J, Martínez S (2009) Distributed Control of Robotic Networks. Applied Mathematics Series, Princeton University Press

13. Cardaliaguet P, Porretta A, Salvarani F (eds) (2018) PDE Models for Multi-Agent Phenomena. Springer INdAM Series, Springer International Publishing, DOI 10.1007/978-3-030-01947-1

14. Chen S, Vazquez R, Krstic M (2019) Folding Bilateral Backstepping Output-Feedback Control Design For an Unstable Parabolic PDE. arXiv:190605434 ArXiv: 1906.05434

15. Chung FRK (1995) Eigenvalues of Graphs. In: Chatterji SD (ed) Proceedings of the International Congress of Mathematicians, Birkhäuser, Basel, pp 1333–1342, DOI 10.1007/978-3-0348-9078-6_128

16. Cui Z, Shi Z (2009) Boid particle swarm optimisation. International Journal of Innovative Computing and Applications 2(2):77–85, DOI 10.1504/IJICA.2009.031778

17. Deutscher J, Kerschbaum S (2018) Backstepping Control of Coupled Linear Parabolic PIDEs with Spatially-Varying Coefficients. IEEE Transactions on Automatic Control pp 1–1, DOI 10.1109/TAC.2018.2802422

18. Deutscher J, Kerschbaum S (2019) Robust Output Regulation by State Feedback Control for Coupled Linear Parabolic PIDEs. IEEE Transactions on Automatic Control pp 1–1, DOI 10.1109/TAC.2019.2938329, conference Name: IEEE Transactions on Automatic Control

19. Dörfler F, Bullo F (2014) Synchronization in complex networks of phase oscillators: A survey. Automatica 50(6):1539–1564, DOI 10.1016/j.automatica.2014.04.012

20. Easley D, Kleinberg J (2010) Networks, Crowds, and Markets: Reasoning about a Highly Connected World. Cambridge University Press, Cambridge, DOI 10.1017/CBO9780511761942

21. Ferrari-Trecate G, Buffa A, Gati M (2006) Analysis of coordination in multi-agent systems through partial difference equations. IEEE Transactions on Automatic Control 51(6):1058–1063, DOI 10.1109/TAC.2006.876805, conference Name: IEEE Transactions on Automatic Control

22. Fliess M, Lévine J, Martin P, Rouchon P (1995) Flatness and defect of non-linear systems: introductory theory and examples. International Journal of Control 61(6):1327–1361, DOI 10.1080/00207179508921959

23. Freudenthaler G, Meurer T (2016) PDE-based tracking control for multi-agent deployment. IFAC-PapersOnLine 49(18):582–587, DOI 10.1016/j.ifacol.2016.10.228

24. Freudenthaler G, Meurer T (2017) Ein verteilt-parametrischer Zugang zur Regelung von Multi-Agentensystemen. at-Automatisierungstechnik 65(8):574–585, DOI 10.1515/auto-2017-0018

25. Freudenthaler G, Meurer T (2020) PDE-based multi-agent formation control using flatness and backstepping: Analysis, design and robot experiments. Automatica 115:108,897, DOI 10.1016/j.automatica.2020.108897

26. Freudenthaler G, Göttsch F, Meurer T (2017) Backsteppingbased extended Luenberger observer design for a Burgers-type PDE for multi-agent deployment. IFAC-PapersOnLine 50(1):6780–6785, DOI 10.1016/j.ifacol.2017.08.1196

27. Friedman J, Tillich JP (2004) Calculus on Graphs. arXiv:cs/0408028 ArXiv: cs/0408028

28. Friedman J, Tillich JP (2004) Wave equations for graphs and the edge-based Laplacian. Pacific Journal of Mathematics 216(2):229–266

29. Frihauf P, Krstic M (2010) Multi-agent deployment to a family of planar Arcs. In: Proceedings of the 2010 American Control Conference, pp 4109–4114, DOI 10.1109/ACC.2010.5530623, iSSN: 2378-5861

30. Frihauf P, Krstic M (2011) Leader-Enabled Deployment Onto Planar Curves: A PDE-Based Approach. IEEE Transactions on Automatic Control 56(8):1791–1806, DOI 10.1109/TAC.2010.2092210

31. Garrido-Jurado S, Muñoz-Salinas R, Madrid-Cuevas F, Marín-Jiménez M (2014) Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognition 47(6):2280–2292, DOI 10.1016/j.patcog.2014.01.005

32. Ge F, Meurer T, Chen Y (2018) Mittag-Leffler convergent backstepping observers for coupled semilinear subdiffusion systems with spatially varying parameters. Systems & Control Letters 122:86–92, DOI 10.1016/j.sysconle.2018.10.009

33. Gevrey M (1918) Sur la nature analytique des solutions des équations aux dérivées partielles. Premier mémoire. Annales scientifiques de l'École Normale Supérieure 35:129–190

34. Grewal MS, Andrews AP (2014) Kalman Filtering: Theory and Practice with MATLAB, 4th Edition | Wiley

35. Grossmann C, Roos HG, Stynes MM (2007) Numerical treatment of partial differential equations. Berlin ; New York : Springer

36. Heuser H (2001) Lehrbuch der Analysis : 1. Mit 805 Aufgaben, zum Teil mit Lösungen, 14th edn. Teubner, Stuttgart

37. Hua C, Rodino L (1996) General theory of PDE and Gevrey classes. In: Min–You Q, Rodino L (eds) General Theory of PDEs and Microlocal Analysis, Addison Wesley, pp 6–81

38. Jadachowski L, Meurer T, Kugi A (2012) An Efficient Implementation of Backstepping Observers for Time-Varying Parabolic PDEs. IFAC Proceedings Volumes 45(2):798–803, DOI 10.3182/20120215-3-AT-3016.00141

39. Jadachowski L, Meurer T, Kugi A (2013) State estimation for parabolic PDEs with reactive-convective non-linearities. In: 2013 European Control Conference (ECC), pp 1603–1608

40. Kerschbaum S (2021) Backstepping Control of Coupled Parabolic Systems with Varying Parameters. PhD thesis, DOI 10.25593/978-3-96147-391-5

41. Kerschbaum S, Deutscher J (2019) Backstepping Control of Coupled Linear Parabolic PDEs with Space and Time Dependent Coefficients. IEEE Transactions on Automatic Control pp 1–1, DOI 10.1109/TAC.2019.2944918, conference Name: IEEE Transactions on Automatic Control

42. Khosroushahi RB, Marquez HJ (2015) PDE Backstepping Boundary Observer Design for Microfluidic Systems. IEEE Transactions on Control Systems Technology 23(1):380–388, DOI 10.1109/TCST.2014.2320859, conference Name: IEEE Transactions on Control Systems Technology

43. Klingenberg W (1978) A Course in Differential Geometry. Graduate Texts in Mathematics, Springer-Verlag, New York, DOI 10.1007/978-1-4612-9923-3

44. Krstić M, Kanellakopoulos I, Kokotović PV (1995) Nonlinear and adaptive control design. Wiley

45. Krstic M (2009) Delay Compensation for Nonlinear, Adaptive, and PDE Systems. Systems & Control: Foundations & Applications, Birkhäuser Basel, DOI 10.1007/978-0-8176-4877-0

46. Kubera Y, Mathieu P, Picault S (2010) Everything can be agent! In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1 - Volume 1, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, AAMAS '10, pp 1547–1548

47. Kuchment P (2004) Quantum graphs: I. Some basic structures. Waves in Random Media 14(1):S107–S128, DOI 10.1088/0959-7174/14/1/014

48. Kuchment P (2005) Quantum graphs: II. Some spectral properties of quantum and combinatorial graphs. Journal of Physics A: Mathematical and General 38(22):4887, DOI 10.1088/0305-4470/38/22/013

49. Kuchment P (2012) Quantum graphs: An introduction and a brief survey. Computers & Structures 104-105:13–20, DOI 10.1016/j.compstruc.2012.03.001, arXiv: 0802.3442

50. Lang S, Shakarchi R (1999) Complex analysis, 4th edn. Graduate texts in mathematics, Springer, New York, NY

51. Leonard NE, Fiorelli E (2001) Virtual leaders, artificial potentials and coordinated control of groups. In: Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228), vol 3, pp 2968–2973 vol.3, DOI 10.1109/.2001.980728

52. Lynch AF, Rudolph J (2002) Flatness-based boundary control of a class of quasilinear parabolic distributed parameter systems. International Journal of Control 75(15):1219–1230, DOI 10.1080/00207170210163640

53. Mesbahi M, Egerstedt M (2010) Graph Theoretic Methods in Multiagent Networks, student edition edn. Princeton University Press

54. Meurer T (2005) Feedforward and Feedback Tracking Control of Diffusion–Convection–Reaction Sys\-tems Using Summability Methods. Fortschr.–Ber. VDI Reihe 8 Nr. 1081, VDI Verlag, Düsseldorf

55. Meurer T (2013) Control of Higher–Dimensional PDEs: Flatness and Backstepping Designs. Communications and Control Engineering, Springer-Verlag, Berlin Heidelberg, DOI 10.1007/978-3-642-30015-8

56. Meurer T (2013) On the Extended Luenberger-Type Observer for Semilinear Distributed-Parameter Systems. IEEE Transactions on Automatic Control 58(7):1732–1743, DOI 10.1109/TAC.2013.2243312

57. Meurer T (2014) Lecture notes in Control of distributed-parameter systems. Kiel University, Chair of Automatic Control

58. Meurer T (2017) Some perspectives in PDE control. In: Proc. 20th IFAC World Congress, Toulouse (F)

59. Meurer T, Krstic M (2011) Finite-time multi-agent deployment: A nonlinear PDE motion planning approach. Automatica 47(11):2534–2542, DOI 10.1016/j.automatica.2011.08.045

60. Meurer T, Kugi A (2007) Tracking control for a diffusion-convection-reaction system: combining flatness and backstepping. IFAC Proceedings Volumes 40(12):140–145

61. Meurer T, Kugi A (2009) Tracking control for boundary controlled parabolic PDEs with varying parameters: Combining backstepping and differential flatness. Automatica 45(5):1182–1194

62. Meurer T, Kugi A (2009) Trajectory Planning for Boundary Controlled Parabolic PDEs With Varying Parameters on Higher-Dimensional Spatial Domains. IEEE Transactions on Automatic Control 54(8):1854–1868, DOI 10.1109/TAC.2009.2024572

63. Moere A (2004) Time-Varying Data Visualization Using Information Flocking Boids. In: IEEE Symposium on Information Visualization, pp 97–104, DOI 10.1109/INFVIS.2004.65, iSSN: 1522-404X

64. Munoz-Salinas R, Garrido-Jurado S (2017) AruCo Library. URL `http://sourceforge.net/projects/aruco`

65. Murray RM (2007) Recent Research in Cooperative Control of Multivehicle Systems. Journal of Dynamic Systems, Measurement, and Control 129(5):571–583, DOI 10.1115/1.2766721

66. Olfati-Saber R (2006) Flocking for multi-agent dynamic systems: algorithms and theory. IEEE Transactions on Automatic Control 51(3):401–420, DOI 10.1109/TAC.2005.864190

67. Olfati-Saber R, Fax JA, Murray RM (2007) Consensus and Cooperation in Networked Multi-Agent Systems. Proceedings of the IEEE 95(1):215–233, DOI 10.1109/JPROC.2006.887293

68. Panait L, Luke S (2005) Cooperative Multi-Agent Learning: The State of the Art. Autonomous Agents and Multi-Agent Systems 11(3):387–434, DOI 10.1007/s10458-005-2631-2

69. Pilloni A, Pisano A, Orlov Y, Usai E (2016) Consensus-Based Control for a Network of Diffusion PDEs With Boundary Local Interaction. IEEE Transactions on Automatic Control 61(9):2708–2713, DOI 10.1109/TAC.2015.2506990

70. Pisano A, Baccoli A, Orlov Y, Usai E (2016) Boundary control of coupled reaction-advection-diffusion equations having the same diffusivity parameter. IFAC-PapersOnLine 49(8):86–91, DOI 10.1016/j.ifacol.2016.07.423

71. Qi J, Vazquez R, Krstic M (2015) Multi-Agent Deployment in 3-D via PDE Control. IEEE Transactions on Automatic Control 60(4):891–906

72. Qi J, Tang SX, Wang C (2019) Parabolic PDE-based multi-agent formation control on a cylindrical surface. International Journal of Control 92(1):77–99, DOI 10.1080/00207179.2017.1308556

73. Reynolds CW (1987) Flocks, Herds and Schools: A Distributed Behavioral Model. In: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, ACM, New York, NY, USA, SIGGRAPH '87, pp 25–34, DOI 10.1145/37401.37406

74. Rodino L (1993) Gevrey functions and ultradistributions. In: Linear Partial Differential Operators in Gevrey Spaces, WORLD SCIENTIFIC, pp 5–59

75. Rouchon P (2001) Motion planning, equivalence, infinite dimensional systems. International Journal of Applied Mathematics and Computer Science 11(1):165–188

76. Rudolph J (2003) Beiträge zur flachheitsbasierten Folgeregelung linearer und nichtlinearer Systeme endlicher und unendlicher Dimension. Berichte aus der Steuerungs- und Regelungstechnik, Shaker, Aachen

77. Rudolph J (2003) Flatness based control of distributed parameter systems. Aachen, Germany : Shaker

78. Saska M, Vonásek V, Krajník T, Pv(z,t)reuv(z,t)cil L (2012) Coordination and navigation of heterogeneous UAVs-UGVs teams localized by a hawk-eye approach. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 2166–2171, DOI 10.1109/IROS.2012.6385517, iSSN: 2153-0866

79. Silvester JR (2000) Determinants of Block Matrices. The Mathematical Gazette 84(501):460, DOI 10.2307/3620776

80. Smyshlyaev A, Krstic M (2004) Closed-form boundary state feedbacks for a class of 1-D partial integro-differential equations. IEEE Transactions on Automatic Control 49(12):2185–2202

81. Smyshlyaev A, Krstic M (2005) Backstepping observers for a class of parabolic PDEs. Systems & Control Letters 54(7):613–625, DOI 10.1016/j.sysconle.2004.11.001

82. Smyshlyaev A, Krstic M (2005) On control design for PDEs with space-dependent diffusivity or time-dependent reactivity. Automatica 41(9):1601–1608

83. Solomon J (2015) PDE Approaches to Graph Analysis. arXiv:150500185 [cs, math] ArXiv: 1505.00185

84. Soon-Yeong C, Yun-Sung C, Jong-Ho K (2007) Diffusion and Elastic Equations on Networks. Publ Res Inst Math Sci 43(3):699–726

85. Styger E (2016) Caterpillar Robot. Personal communication

86. Vazquez R, Krstic M (2015) Explicit boundary control of reaction-diffusion PDEs on arbitrary-dimensional balls. In: Control Conference (ECC), 2015 European, pp 879–884

87. Vazquez R, Krstic M (2016) Bilateral boundary control of one-dimensional first- and second-order PDEs using infinite-dimensional backstepping. In: 2016 IEEE 55th Conference on Decision and Control (CDC), pp 537–542, DOI 10.1109/CDC.2016.7798324

88. Vazquez R, Krstic M (2016) Boundary control of coupled reaction-diffusion systems with spatially-varying reaction. IFAC-PapersOnLine 49(8):222–227, DOI 10.1016/j.ifacol.2016.07.445

89. Vazquez R, Krstic M (2017) Boundary Control of Coupled Reaction-Advection-Diffusion Systems With Spatially-Varying Coefficients. IEEE Transactions on Automatic Control 62(4):2026–2033, DOI 10.1109/TAC.2016.2590506

90. Wang S, Woittennek F (2013) Backstepping-method for parabolic systems with in-domain actuation. IFAC Proceedings Volumes 46(26):43–48, DOI 10.3182/20130925-3-FR-4043.00049

91. Wei J, Fridman E, Selivanov A, Johansson KH (2019) Multi-agent deployment under the leader displacement measurement: a PDE-based approach. In: 2019 18th European Control Conference (ECC), pp 2424–2429, DOI 10.23919/ECC.2019.8796132, iSSN: null

92. Woittennek F, Wang S, Knüppel T (2014) Backstepping design for parabolic systems with in-domain actuation and Robin boundary conditions. IFAC Proceedings Volumes 47(3):5175–5180, DOI 10.3182/20140824-6-ZA-1003.02285

93. Wooldridge M (2009) An Introduction to MultiAgent Systems, 2nd edn. John Wiley & Sons

94. Zeitz M (1987) The extended Luenberger observer for nonlinear systems. Systems & Control Letters 9(2):149–156, DOI 10.1016/0167-6911(87)90021-1