Reverse Stackelberg Games:
Theory and
Applications in Traffic Control

Noortje Groot

Cover design: Noortje Groot, illustrated by Lucas Rozenboom.

Head: von Stackelberg; proverb: 'to dangle a carrot in front of ...'; pyramids as symbols of hierarchy; chess pieces as a symbol of game theory.

Pas d'art sans désordre
No science without a quest for order
———————————— +
Science if and only if art

Reverse Stackelberg Games:
Theory and
Applications in Traffic Control

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 12 november 2013 om 12:30 uur
door

Nore Berta GROOT

Master of Science in Econometrics and Operations Research,
Universiteit Maastricht,
geboren te Woerden.

Dit proefschrift is goedgekeurd door de promotoren:
Prof. dr. ir. B. De Schutter
Prof. dr. ir. J. Hellendoorn

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr. ir. B. De Schutter	Technische Universiteit Delft, promotor
Prof. dr. ir. J. Hellendoorn	Technische Universiteit Delft, promotor
Prof. dr. G.J. Olsder (emeritus)	Technische Universiteit Delft
Prof. dr. ir. S.P. Hoogendoorn	Technische Universiteit Delft
Dr. J.C. Engwerda	Tilburg University
Dr. D. Bauso	Università Degli Studi Di Palermo
Prof. G. Zaccour	HEC Montréal, Université de Montréal

This dissertation has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate studies. The support of the European Union COST Actions TU1102 and TU0702 is gratefully acknowledged, as well as the support of the European Union 7th Framework Network of Excellence "Highly-complex and networked control systems (HYCON2)" program and the BSIK project "Next Generation Infrastructures (NGI)".

Published and distributed by: Noortje Groot
E-mail: postbusnoor@gmail.com

ISBN 978-94-6108-524-5

Keywords: Multilevel decision making, noncooperative hierarchical game theory, optimization, incentives, route guidance, model predictive traffic control.

Printed in the Netherlands, Gildeprint Drukkerijen

Acknowledgements

Het is ons maar gegund
De vele mooie dingen
Ons onbetwistbaar eigendom
Zijn de herinneringen[1]
- In memory of my mom Béke -

Not quite aware of what to really expect from a Ph.D. trajectory, I have learned a lot from a new area of research, working amidst an abundance of possible interesting research topics and available papers, struggles with Matlab, etcetera. Fortunately, most drawbacks turned out to be challenges. I hope you will enjoy the results of these years as accumulated in this thesis as well as in the propositions.

When starting four years ago, I *did* have a clear set of requirements in mind for my thesis, e.g., the acknowledgements should not exceed the limit of one page – *one should not exaggerate after all* – and articles should be written solely in the indirect speech. My supervisor and promotor Bart De Schutter succeeded, dealing with some stubbornness, in modifying the writing part, and life took care of the rest.

For this Ph.D. experience, I would like to first of all thank my main supervisor Bart for his frequent feedback training us to work accurately and for accepting some returning '*yes, but*'s. I especially enjoyed brainstorming about special cases that could apply in higher dimensions, where my imaginative powers fell short.

I would also like to thank my second promotor Hans Hellendoorn; I appreciated discussions and insights on e.g., the Ph.D. candidates within the department, the faculty, and developments regarding the graduate school. To follow your motto 'Whenever there is an open loop, there is work for us,' I do hope that the work needed before an open loop can even be considered will also be acknowledged, and that it will indeed result in a closed loop some day.

To both I am grateful for leaving the space for developing skills in supervision and as Ph.D. representative, as well as for supporting short visits to LCCC Lund and Inria Grenoble.

Further, I am honored for dr. Dario Bauso, dr. Jacob Engwerda, prof. Serge Hoogendoorn, prof. Geert-Jan Olsder, and prof. Georges Zaccour to be part of the defense committee. In particular, prof. Olsder, it was a pleasure to talk about the relevance of research. Dr. Engwerda and Dr. Bauso; I have enjoyed the encounters at conferences and much appreciate your helpful feedback on this thesis.

[1] *"We are lucky to be granted, those many nice things, yet what we possess no matter what, is in the memories."*

A special thanks goes to Kateřina Staňková; without your Ph.D. work I would not have entered the area of reverse Stackelberg games in the first place. Thank you for all feedback and thoughts on various topics, as well as for your pleasant enthusiasm.

As the work of a Ph.D. candidate oftentimes takes place individually at the desk, I have greatly appreciated you Dang, Hildo, Jia, Renshi, Sadegh, Sachin, Subramanya, Mohammad, and Yue, as enthusiastic (badminton and volleybal) companions.

Further, I much enjoyed to share office with you, Samira, for most of my time at DCSC, and Kim, I do not think I have talked more with anyone else at DCSC!

Also, I have good memories of the conference visits I could share with you Alfredo, Kateřina, Mohammad, Samira, Yihui, and Zhe, and the colleagues at other universities that I met on the way.

Here I would also like to thank our secretariat, Esther, Kitty, Marieke, Saskia, as well as Imas for being helpful and supportive.

Having had so many colleagues that cannot all be mentioned but that I nonetheless appreciate for creating a nice working environment, I would like to thank Aleksandar, Alina, Amir, Amol, Andrea, Anna, Arturo, Bart, Dieky, Hans, Ilya, Jacopo, Marco, Mernout, and Rudy, for at least 15 reasons.

In addition I would like to thank:

Anil, Farzaneh, Patricio, and *Yashar* for your company;

Arne, Edwin, Hildo, and *Yue* for having fun during the social event preparations;

Bart, Iris, and *Mayke* for keeping the UCM spirit alive together;

Diederik for many nice dinners;

Eki and *Zul* for your hospitality;

Esther for your friendship;

Gijs for sharing your Ph.D. experiences;

Ivo for being a great colleague from our starting day at DCSC;

Jonas for keeping a sharp mind, or for keeping the mind sharp;

Pieter for exploring Rotterdam;

Yu Hu for your hospitality and Chinese dinners.

Finally, I appreciated you ladies of our 'Delft Women in Science' (DEWIS) intervision group for sharing experiences of work and life.

One of the best parts of this written thesis is the opportunity to acknowledge and thank the friends and family that most supported me at different times.

First of all I would like to thank my father Jan, amongst other things for helping us to achieve a new balance, for keeping a healthy relativist perspective needed especially when things seem not to work out, and for investing the time to help making my apartment a true home.

My aunts Anne, Ida, and Joke, thank you for showing interest in our life developments. The same holds for 'Werner *en* Carla' and Geert-Jan; you have been present at important moments and I appreciate and learned from our discussions.

Last but not least I would like to thank my paranymphs, my friend Marie-Claire and my sister Harmke, for being, next to generous, always available 'with ear and shoulder' when needed for advise and support.

Noortje Groot Rotterdam, September 30th, 2013

Table of Contents

Chapter 1

Introduction and Motivation

The main focus of this dissertation is on decision making among multiple parties according to a particular hierarchical or leader-follower game, the so-called *reverse Stackelberg game*, which will be properly defined in Chapter 2. The main concept around which this game evolves is the so-called *leader function* that embodies a mapping of the follower's decision space into the leader's decision space. By first presenting this leader function, the leader's actual decision follows directly from the choice of the follower's decision. Considering the objectives of leader and follower, mathematical solutions for determining optimal strategies or actions are desired.

Before motivating our interest in this particular game, we first provide a brief outline of game theory. At the end of this introductory chapter, the structure of this thesis as well as our contributions will be presented.

1.1 A Brief Introduction to Game Theory

For an intuitive description of a game and of its main concepts, we cite [156, p. 2]:

> "A *player* may be interpreted as an individual or as a group of individuals making a decision. A *game* is a description of strategic interaction that includes the constraints on the actions that the players *can* take and the players' interests, but does not specify the actions that the players *do* take. A *solution* is a systematic description of the outcomes that may emerge in a family of games. *Game theory* suggests reasonable solutions for classes of games and examines their properties."

Within game theory, many classes of games can be specified. A game should first of all be placed in the context of *cooperative* or *noncooperative* game theory. While in noncooperative games players take individual decisions according to their individual objective functions, in cooperative games players form coalitions and act according to what is optimal for the coalition, after which some division of the utility that is achieved may be made amongst the individual players. In both cooperative and noncooperative games, players can have objective functions that are conflicting. Whereas *zero-sum* games assume that the gain for one player implies a loss to another, yielding the sum of objective function values to be zero, in a *nonzero-sum* game there are no such constraints on the objective function values of the players.

Another important distinction is between the representation of a game, which can be either in *strategic or explicit* or in *extensive form*. In the former class of

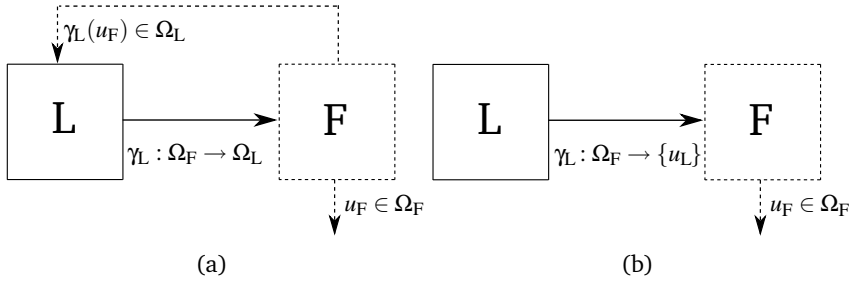


Figure 1.1: *(a) Schematic representation of the basic reverse Stackelberg game. The leader function and the decision spaces are denoted by γ_L and Ω_L, Ω_F, respectively. (b) The Stackelberg game as a special case.*

games, players decide upon their strategy at one time instant and make their decisions simultaneously. In the latter class, however, strategy preparation and decision making can occur at different stages. While extensive-form games are often depicted by a decision tree, a *normal form (matrix)* game representation summarizes all strategies of an explicit game in a matrix format. Finally, in a game setting the players can be perfectly, i.e., fully, or imperfectly informed of each others' decisions and characteristics.

For more information on the broad field of game theory, the reader is referred to [15, 65, 156, 194].

In this dissertation, we focus solely on a noncooperative, nonzero-sum game in which leader and follower players act sequentially and under the assumption of complete information. The decisions or actions involve the choice of a – we here assume, real-valued – decision variable that contains a finite number of elements. This game will be defined in the following chapter, together with a description of what constitutes a solution for this category of games. For clarity, a schematic representation is provided in Fig. 1.1. Upon the announcement of the leader function $\gamma_L : \Omega_F \to \Omega_L$, where Ω_L, Ω_F represent the decision spaces of respectively the leader (L) and the follower (F) player, the follower will take a decision, which implies the action of the leader. The leader function can thus been regarded as a *strategy*, which will lead to an *action* when it is applied.

A Comparison of Game Theory to Optimization and Control Theory

An obvious specification of any game concerns the number of players and the time-dependency of the actions that are taken.

While, in general, game theory can be perceived as a branch of applied mathematics, one can put game theory in the perspective of other disciplines: in [15], dynamic game theory is perceived as a "child of the parents game theory and optimal control theory" [15, p. 2], where the child is said "to be as old as one of his parents – optimal control theory" [15, p. 2] when referring to the years in which research in these areas manifested itself. Table 1.1 summarizes this relationship. To elaborate,

Table 1.1: *The place of game theory [15].*

	One player	Many players
Static	Mathematical programming	(Static) game theory
Dynamic	Optimal control theory	Dynamic (and/or differential) game theory

optimization and mathematical programming are present in many fields, including optimization-based control theory [7, 64]. Especially during the recent years, game theory has been considered in relation to control more often, e.g., in applying solutions of noncooperative games in networked control or consensus theory [6, 137]. Other fields in which games have often been applied are, e.g., economics [71, 194], finance [9, 124, 211], supply-chain management [19, 91, 203], and biology [176].

Here it should be noted that in [15], a game is called *dynamic* "if the order in which the decisions are made is important'. We perceive a game as dynamic if it is *multi-stage*, in particular in the sense that the action of at least one player is dependent on some (player's) action of a previous stage. At the same time, in the literature the 'open-loop (dynamic)' game has also been categorized as a dynamic game. There, players' actions are not dependent on actions at previous stages, but actions are simply made at subsequent time instants, possibly dependent on the current state of the system; such a game could therefore also be perceived as 'multi-stage static'.

We perceive the reverse Stackelberg game as outlined in the beginning of this chapter and as depicted in Fig. 1.1, as static. The reason is that only the announcement of the leader's strategy or *leader function* is made prior to the choice of actions. Subsequently, the leader and follower actions are basically taken simultaneously, once the leader is informed of the follower's decision. More information on static versus dynamic games can be found in Chapter 2 (Section 2.4.1).

1.2 Motivation and Aim

Now, *what do we strive to achieve by using game theory*? First of all, we consider the reverse Stackelberg game as a decision making structure that can be imposed on a setting in which multiple decision makers operate, where one can distinguish between:

- interaction between strictly rational controller units that seek the optima of their objective functions according to a given framework of cost functions and decision spaces;
- interaction between human parties that have conflicting objectives that can be translated into a mathematical optimization problem.

In the former setting, introducing reverse Stackelberg game elements can be perceived as a top-down approach to attempt to structure a (large) optimization

problem and thereby making it easier to handle. Differently, in the latter setting a natural hierarchical element can already be detected, making a reverse Stackelberg implementation of the problem a natural choice of a solution framework. An example of such a situation can be found in power or more general transportation networks in which the control of the physical network occurs at the lower levels with at the other end of the spectrum a higher-level operator that decides upon set-points for the lower levels [143, 149]. Moreover, in addition to noise or uncertainties that can disturb the optimal behavior of the players, *bounded rationality* plays a role here, i.e., the fact that players, due to limited abilities, may simply not be able to arrive at those decisions that are optimal according to a (complex) mathematical optimization problem [156]. As regards the contents of this dissertation, a noise-free, deterministic setting is mostly considered, which better connects to the former framework of *fully rational* controller units or agents. In this case without uncertainty, a rational decision for a player with a functional $f : \Omega \to \mathbb{R}$ as a preference relation on the set Ω of actions is simply a decision that is optimal amongst the feasible actions [156].

Another important facet of the reverse Stackelberg game as compared to the original Stackelberg game [195, 196] is in the link between the leader and the follower, represented by the leader function. In the original Stackelberg game this link is missing and leader and follower players make their decisions sequentially, which is why the game can be described as hierarchical. A clear reason to adopt reverse Stackelberg games as opposed to such purely hierarchical games is the fact that in the original Stackelberg game, the leader cannot control the follower's decision in case of a nonunique follower response. This can be circumvented with a leader function that gives the leader additional capacity to influence the follower.

To summarize, while in general our aim is to *apply elements of the reverse Stackelberg game in order to structure large-scale optimization or control problems* as described above, the focus of the current dissertation is mainly on the reverse Stackelberg game itself, i.e., ***to develop a structured, systematic solution approach for the general reverse Stackelberg game***.

In particular, given the initial outline of the game at the beginning of this chapter, two obvious but relevant questions that will be discussed in this dissertation are:

1. Does a leader function exist such that the leader can influence the follower to behave as desired?

2. What leader function should the leader adopt to optimize her[1] objective function?

The contributions regarding these questions that are presented in this thesis can be found in more detail in Section 1.3 below.

[1] In this dissertation, the leader and follower players are addressed with a feminine respectively masculine pronoun, for which the author finds support in '*... in adherence to a historical custom, we refer to the principal with the feminine pronoun 'she' and to the agent with the masculine 'he'.* [135, p. 4]'

1.3 Scope and Contributions

The main contributions of the work presented in this dissertation can be found in Chapters 2–5 and in the independent Appendix B and are listed as follows:

Chapter 2: A comprehensive overview is provided of research conducted in the area of reverse Stackelberg games since the 1970s, categorized in several topics that are relevant to the game. Second, an elaborate enumeration of open issues is presented as identified from the available literature.

The contents of Chapter 2 are based on [81, 82].

Chapter 3: As a first step towards a systematic solution approach for the reverse Stackelberg game, leader functions of an affine structure are considered, where the contribution is twofold:

- Necessary and sufficient conditions are provided for the existence of an optimal affine leader function in the general static, deterministic single leader-single follower game under mild constraints on convexity of the decision spaces and connectedness of the follower sublevel set. These conditions enable the application of easy-to-derive optimal affine leader functions in games that are not restricted to be of a linear-quadratic type.
- The full set of optimal affine leader functions in an unconstrained decision space is characterized and an analysis is provided on how to obtain the subset of optimal affine leader functions in case of a constrained decision space. This set can subsequently be used for further optimization, e.g., in a sensitivity analysis on the deviation of a follower from his optimal response.

The contents of Chapter 3 can be mainly found in [83] and have been partially presented in [77, 78] as well as during the 12th Viennese workshop on Dynamic Games, Optimal Control and Nonlinear Dynamics, May 30th-June 2nd 2012, Wien, Austria.

Chapter 4: Methods are provided for systematically deriving optimal leader functions of a nonlinear structure and they are compared with the heuristic evolutionary approaches that have been suggested in the literature, i.e., the genetic algorithm as well as neural network approach, w.r.t. computation time and optimality. More specifically, the following approaches are proposed:

- A *continuous multilevel* optimization program is considered that leads to an optimal leader function represented by a linear combination of a set of basis functions.
- The multilevel optimization program is also considered in a variant where the follower's decision space is *discretized in grid points*. Since optimality of the leader function cannot be guaranteed, an adaptive gridding approach is proposed based on this single-level program.

– Heuristic *interpolating spline methods* available in the literature can be adopted, especially in the special case the decision spaces of leader and follower together are of at most dimension three.

The contents of Chapter 4 can be mainly found in [86] and have been partially presented in [85].

Chapter 5: New applications of the reverse Stackelberg game are introduced in the area of route guidance and traffic control. Here, the road authority and groups of homogeneous drivers are represented by a leader respectively by follower players. Suggestions for real-life implementation are provided and a case study on a real-life ringroad network has been conducted. In particular, the following games have been proposed, where a distinction is made based on the domain of the leader functions and the type of traffic network, viz.:

– *Route choice in freeway networks*:
  The followers' *expected travel times* are mapped to monetary incentives on the basis of which dynamic routing is applied in the context of freeway networks in order to reach a system-optimal traffic distribution with respect to, e.g., the total travel time of the drivers.
– *Route choice in freeway networks*:
  The followers' *route splitting rates* are mapped to monetary incentives, again in order to reach a system-optimal traffic distribution in the context of freeway networks.
– *Single-corridor urban networks*:
  The followers' expected travel times are mapped to monetary incentives, now under the aim to reduce vehicular emissions while taking into account the urgency and desired travel time of the mainstream drivers in an urban traffic setting. In this setting, traffic signals for crossing, joining, and mainstream traffic are adopted as the road authority's control measure.

The contents of Chapter 5 can be mainly found in [87] and have been partially presented in [79, 80].

Appendix B: A procedure for approximating a nonlinear function by a piecewise affine function is proposed for the nonlinear METANET traffic flow model and the VT-macro vehicular emissions and fuel consumption model. Subsequently, model predictive control (MPC) is applied to the resulting mixed-logical dynamic model, leading to a mixed-integer linear programming (MILP) problem instead of an MPC problem that is based on nonconvex optimization. While both approaches are NP-hard, efficient MILP solvers are available; in a case study the trade-off between computational requirements and approximation inaccuracies is studied. This procedure is applied to a freeway traffic network where the aim is to reduce the total time spent by vehicles in the system as well as the vehicular emissions by means of variable speed limits and on-ramp metering.

The contents of Appendix B can be mainly found in [84] and have been partially presented in [75, 76].

1.4 Structure of the Thesis

The structure of this dissertation is straightforward: Chapters 3-5 could each be read independently after the introduction of the reverse Stackelberg game in Chapter 2 (Section 2.1). While the focus of Chapters 3 and 4 is on the theory of reverse Stackelberg games, in Chapter 5 possible applications in traffic control are presented. In Appendix A the model adopted in both Chapter 5 and in the independent Appendix B is presented. Chapter 6 concludes the thesis with a summary of the findings and directions for further research.

Fig. 1.2 clarifies the connections between the chapters.

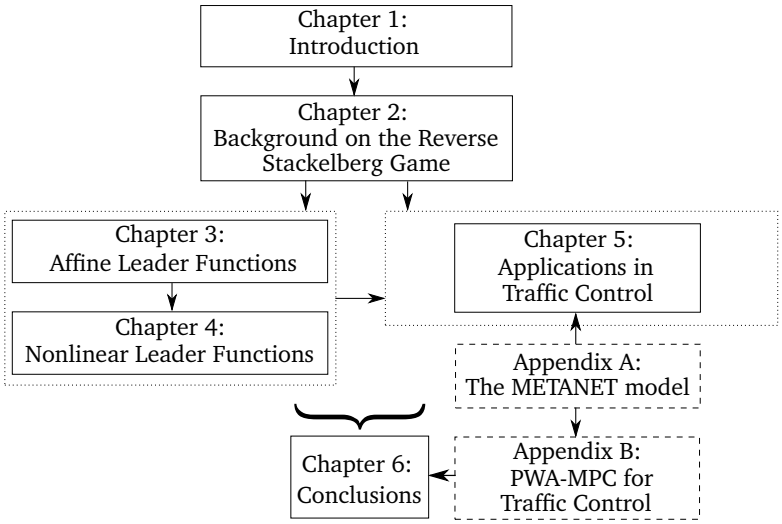


Figure 1.2: *Schematic representation of the connections between the chapters.*

Chapter 2

Background on the Reverse Stackelberg Game

In the current chapter, an overview will be presented to clarify the concept of the reverse Stackelberg game within several research areas as well as to emphasize its potential for application, while taking into account the computational complexity of the game. Further, an overview of the main results in the literature concerning reverse Stackelberg games will be provided as well as an analysis of open issues for further research.

In Section 2.1, the background necessary to understand the concept of the reverse Stackelberg game is provided. In Section 2.2 the reverse Stackelberg game is subsequently positioned amongst related fields of research and in Section 2.3 a brief summary of solution approaches is given. The section thereafter includes a survey of different topics that have been considered in previously conducted research on reverse Stackelberg games. In particular, in Section 2.4 a classification is made of main results in the current literature along several axes of characteristics that are inherent to the definition of a reverse Stackelberg game, i.e., contributions are considered that (1) involve either static or dynamic cases; (2) look into continuous-time differential games; (3) deal with stochastic scenarios; (4) consider partial, nonnested information; (5) perform a sensitivity analysis; and that (6) consider multilevel games with multiple players on each layer. Several areas of application are considered in Section 2.5. The chapter concludes in Section 2.6 with a list of problems that have not yet been fully considered or solved. As explained in Chapter 1, the research presented in the following chapters of this dissertation is aimed towards addressing some of these open issues. A selection of topics for further research will finally be elaborated upon in Chapter 6.

2.1 Definition of the Reverse Stackelberg Game

2.1.1 The Stackelberg Game

A brief introduction to (noncooperative) game theory has been provided in the previous Chapter 1. We here define the Stackelberg game and equilibrium, which is the sequential counterpart of a noncooperative game in which players act simultaneously. For the latter game, the well-known Nash equilibrium is adopted as the

solution concept, which we will also define for the sake of clarity.

Definition 2.1 Stackelberg game, Stackelberg equilibrium [156, 195, 196]
A *Stackelberg game* is described as a two-player extensive game with perfect information in which a 'leader' chooses an action from a set Ω_L and a 'follower', informed of the leader's choice, subsequently chooses an action from a set Ω_F. Some equilibria $\left(u_L^s, u_F^s\right)$ of a Stackelberg game correspond to the following solutions:

$$(u_L^s, u_F^s) \in \left\{ \min_{(u_L, u_F) \in \Omega_L \times \Omega_F} \jmath_L(u_L, u_F) : u_F \in \arg\min_{u_F' \in \Omega_F} \jmath_F(u_L, u_F') \right\}, \tag{2.1}$$

where $\jmath_L : \Omega_L \times \Omega_F \to \mathbb{R}$, $\jmath_F : \Omega_L \times \Omega_F \to \mathbb{R}$ denote the leader respectively the follower cost function and where $\jmath_L(u_L^s, u_F^s)$ represents an optimal leader cost. Due to possible nonuniqueness of the optimal follower response to $u_L^s \in \Omega_L$, the game can result in a suboptimal leader cost function value $\jmath_L(u_L^s, u_F^{\text{sub}}) > \jmath_L(u_L^s, u_F^s)$. Therefore, alternative solutions consist in those where the leader adopts a decision $u_L \neq u_L^s$ that does not minimize $\jmath_L(\cdot)$ as in (2.1) but that yield better objective function values than $\jmath_L(u_L^s, u_F^{\text{sub}})$. ◇

If the decision spaces Ω_L, Ω_F are closed and bounded (compact), $\Omega_L \subset \mathbb{R}^{n_L}, \Omega_F \subset \mathbb{R}^{n_F}$ and $\jmath_L, \jmath_F$ are real-valued continuous functionals on $\Omega_L \times \Omega_F$, a Stackelberg equilibrium exists [173].

Since the Stackelberg game was introduced in 1934, multiplayer extensions and games with incomplete information have been considered too. More information on Stackelberg games can be found in [15, 91, 174, 175]. An important implication of adopting a Stackelberg game is that the leader cannot control the decisions of the follower in case his response is nonunique. This matter is a principal reason for studying the more general reverse Stackelberg game.

When players act simultaneously rather than sequentially, the following equilibrium concept applies:

Definition 2.2 Nash equilibrium [148]
A *Nash equilibrium* is a set of strategies (decisions) under the application of which no player can obtain a better objective function value by unilaterally deviating from these strategies.

Formally, this means that a strategy profile $u^* := (u_1^*, \ldots, u_n^*) \in S$ with $S := S_1 \times \cdots \times S_n$ the set of possible profiles is a Nash equilibrium if for every player $i \in \{1, \ldots, n\}$, the following inequality holds for $u_i \in S_i$, $u_i^* \neq u_i$, where $\jmath_i$ denotes the cost function of player i: $\jmath_i\left(u_1^*, \ldots u_{i-1}^*, u_i^*, u_{i+1}^*, \ldots u_n^*\right) \leq \jmath_i\left(u_1^*, \ldots u_{i-1}^*, u_i, u_{i+1}^*, \ldots u_n^*\right)$. ◇

Considering possible improvements or deterioration in the objective function value of any player in a cooperative or noncooperative game, the concept of Pareto efficiency or optimality can be applied:

Definition 2.3 Pareto optimality, Pareto efficiency [65, 156]
No player in a *Pareto-optimal* or *Pareto-efficient* equilibrium is able to unilaterally deviate from a Pareto-optimal decision without making another player worse

off, i.e., a strategy profile $u^* := (u_1^*, \ldots, u_n^*) \in S$ with $S := S_1 \times \cdots \times S_n$ the set of possible profiles is Pareto optimal if for all $i \in \{1, \ldots, n\}$ there does not exist a $u_i \in S_i$, $u_i^* \neq u_i$, such that for some player $j \in \{1, \ldots, n\}$: $\jmath_j\left(u_1^*, \ldots u_{i-1}^*, u_i, u_{i+1}^*, \ldots u_n^*\right) \leq \jmath_j\left(u_1^*, \ldots u_{i-1}^*, u_i^*, u_{i+1}^*, \ldots u_n^*\right)$, where $\jmath_i$ denotes the cost function of player i.

In cooperative games, a Pareto-optimal solution corresponds to the case in which, in addition, no *joint* decisions of players can lead to an improved performance of at least one player, without resulting in a deterioration in the other players' performance. ◇

2.1.2 The Basic Reverse Stackelberg Game

The basic single leader-single follower, static, deterministic *reverse* Stackelberg game [97, 99, 153] can be defined through the leader respectively follower objective (cost) functions as in Definition 2.1. For the leader and follower decision variables we here assume $u_\mathrm{L} \in \Omega_\mathrm{L} \subseteq \mathbb{R}^{n_\mathrm{L}}$, $u_\mathrm{F} \in \Omega_\mathrm{F} \subseteq \mathbb{R}^{n_\mathrm{F}}$, with $n_\mathrm{L}, n_\mathrm{F} \in \mathbb{N}$.

The leader player acts first by announcing a leader function $\gamma_\mathrm{L} : \Omega_\mathrm{F} \to \Omega_\mathrm{L}$, that can be restricted to belong to a particular class of function structures, denoted by Γ_L. In the so-called *direct approach* to the game, one simultaneously aims to find a leader function $\gamma_\mathrm{L}(\cdot)$ and a corresponding solution point $(\gamma_\mathrm{L}(u_\mathrm{F}), u_\mathrm{F})$ that is associated with a leader objective function value that cannot be improved under any leader function, while taking into account the follower's response to the given leader function.

This problem can be summarized by the following indirect formulation with composed functions (adapted from [153]):

Direct Reverse Stackelberg Game Formulation

$$\begin{aligned} \text{To find:} \quad & \gamma_\mathrm{L}^* && \in \arg\min_{\gamma_\mathrm{L} \in \Gamma_\mathrm{L}} \jmath_\mathrm{L}\left(\gamma_\mathrm{L}\left(u_\mathrm{F}^*(\gamma_\mathrm{L}(\cdot))\right), u_\mathrm{F}^*(\gamma_\mathrm{L}(\cdot))\right), \\ \text{such that} \quad & u_\mathrm{F}^*(\gamma_\mathrm{L}(\cdot)) && \in \arg\min_{u_\mathrm{F} \in \Omega_\mathrm{F}} \jmath_\mathrm{F}\left(\gamma_\mathrm{L}^*(u_\mathrm{F}), u_\mathrm{F}\right), \end{aligned} \tag{2.2}$$

where, in order for the problem to be well-defined, we assume that an optimal[1] leader function $\gamma_\mathrm{L}^*(\cdot)$ is constructed such that the optimal follower response $u_\mathrm{F}^*(\gamma_\mathrm{L}(\cdot))$ is unique. Further, note that for feasibility of the leader function in the game setting, (i) it is necessary that all elements of Ω_F are mapped to an element of Ω_L and (ii) not all elements of Ω_L need to be included in the mapping, while (iii) different elements of Ω_F are allowed to be mapped to the same element of Ω_L. Hence, the mapping is allowed to be *non-injective* as well as *non-surjective*.

In order for the leader to be able to solve this game, we make the following assumption as regards the availability of information:

Assumption 2.4 In the current thesis, we assume the leader to have complete knowledge of the follower's objective function $\jmath_\mathrm{F}(\cdot)$, of the decision space Ω_F, and of the follower decision once made. Here, the follower's objective function can be known either as an analytical expression, or as a black box. Furthermore, we assume the follower to behave *fully rationally* (see Chapter 1, p. 4), unless stated otherwise. ◇

[1] Optimal values or, more general, optimal strategies are often indicated by a superscribed asterisk.

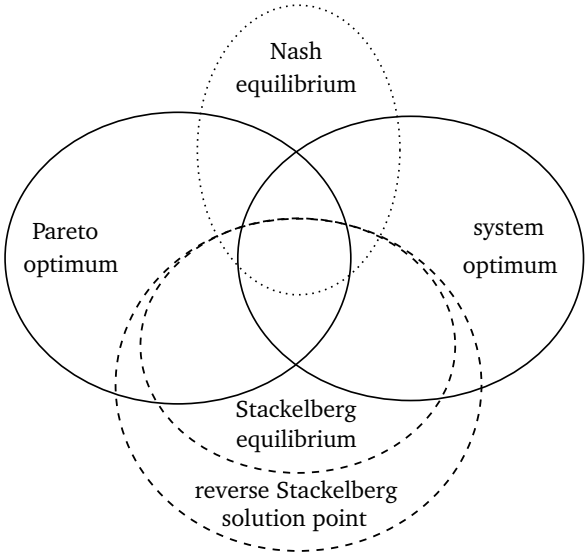


Figure 2.1: *Venn diagram of several types of equilibria and solutions.*

Remark 2.5 In order for the leader function $\gamma_L : \Omega_F \to \Omega_L$ to influence the follower's behavior, it is important that the follower's cost function is indeed a function of u_L. In order for the reverse Stackelberg game to be well-defined it is, however, not strictly necessary for $J_F(\cdot)$ to be directly dependent on u_F, nor is it necessary for $J_L(\cdot)$ to be directly dependent on u_L. ◇

Remark 2.6 [Relations between solutions and equilibria in game theory]

The Venn diagram of Fig. 2.1 shows the relationships between major classes of game solutions and equilibria. A *Stackelberg equilibrium* can be obtained as a special case of the *reverse Stackelberg game* and therefore consists of a subset of possible solution points to the reverse Stackelberg game. As shown in [173], a Stackelberg solution is at least as good for the leader player as a *Nash equilibrium*, while the follower may or may not be as least as good off when playing sequentially instead of simultaneously. In general, existence of a Stackelberg solution does not imply the existence of a Nash equilibrium, and vice versa [173]. In the Nash equilibrium and in case of *optimal* (reverse) Stackelberg solutions, the players cannot perform better by changing their individual strategies unilaterally. In addition, in a *Pareto-optimal* solution, any improvement in the objective function value of a certain player will result in a deterioration of the objective function value of at least one player. The concept of *system optimality* is not a game-theoretical concept; it is included to distinguish those solutions that cannot be improved for the one particular player that acts according to the objectives of the system. A system-optimal solution can thus be equal to a Pareto-optimal solution.

In this thesis we focus in particular on solutions to the reverse Stackelberg game that correspond to a leader optimum without paying special attention to possible improvements in the follower's function value. ◇

2.1.3 Computational Complexity

Even the single leader-single follower static reverse Stackelberg problem is complex and in general difficult to solve analytically due to the composed functions appearing in the problem formulated in (2.2) as well as due to the possible existence of multiple global optima that could be selected as desired equilibria for the leader player, and a nonunique follower response to the corresponding desired decision variables [13, 153, 154, 180].

Before we analyze the complexity of the general reverse Stackelberg game in Theorem 2.7, a brief summary of complexity theory is provided next [68].

First of all, in case one can prove that in general no algorithm exists that can solve an instance of the problem with finite termination, the given problem is said to be *undecidable*. Here, a problem can either be a *search problem* that is solved in case a solution is found or in case it is concluded that no solution exists, or a *decision problem* that can only have the answers 'yes' and 'no'. A problem belongs to the class P of *polynomial-time* solvable problems if there exists an algorithm that can solve the problem within a time that is bounded from above by a polynomial function of the parameter that indicates the size of a problem instance. If a *decision problem* belongs to the class NP of *nondeterministic polynomial-time* solvable problems, there exists a (nondeterministic) polynomial-time solvable algorithm that can verify whether a given certificate or claim on the answer of a decision problem is correct or not. The class NP includes next to the class P also the subclass of NP-complete problems. A *decision* problem NPC is NP-*complete* if it belongs to the class NP and if all other decision problems in the class NP can be transformed to the given problem NPC with a polynomial transformation. In order to prove NP-completeness of a decision problem, it therefore has to be shown that the problem is in NP and that a known NP-complete problem can be transformed to the given problem in polynomial time. Further, a problem is *strongly* NP-complete if the modified problem in which any numerical (integer) parameter of the original problem is bounded by a polynomial, is still NP-complete.

A last important complexity category we mention here is the class of NP-hard problems. A search problem NPH that is NP-*hard* is 'at least as hard' as any NP-complete problem, i.e., it is NP-hard if there exists an NP-complete problem that can be (Turing) reduced to the given problem NPH in polynomial time. Such a polynomial-time reduction of problem Π to problem Π' is an algorithm A that solves Π with a hypothetical subroutine or 'oracle' for solving Π such that if this subroutine were a polynomial time algorithm for Π', then A would be a polynomial time algorithm for Π.

In particular, if a decision problem associated with a search problem is NP-complete and a polynomial-time algorithm for the search problem could be adopted to solve the associated decision problem in polynomial time, this implies that the search problem is NP-hard. Hence, a (search) problem is *strongly NP-hard* if there exists a strongly NP-complete problem that can be reduced to it in polynomial time.

Finally, it should be noted that it is generally accepted that P is a proper subclass of NP, but this has not yet been proven in a generally accepted manner. If P=NP would hold, all problems in the class of NP-complete problems, for which

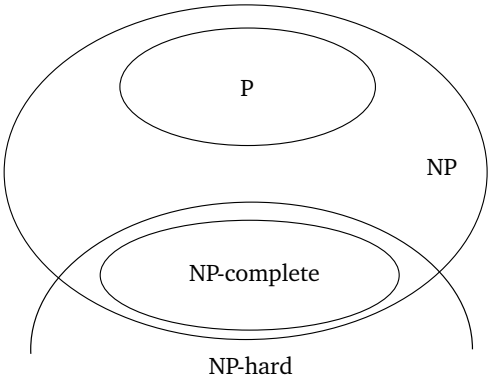


Figure 2.2: *Schematic representation of the relation between several computational complexity classes.*

no polynomial-time solvable algorithms have been found thus far, are solvable in polynomial time.

A scheme of the relation between important computational complexity classes can be found in Fig. 2.2, where we assume that P $\neq$ NP.

Theorem 2.7 *The reverse Stackelberg game as formulated in (2.2) is strongly NP-hard.*

Proof: The original Stackelberg game is a special case of (2.2), i.e., for $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \rightarrow \left\{u_{\mathrm{L}}^{\mathrm{d}}\right\}$, with $u_{\mathrm{L}}^{\mathrm{d}} \in \Omega_{\mathrm{L}}$ a free variable, (2.2) can be written as:

$$(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \in \arg \min_{u_{\mathrm{L}} \in \Omega_{\mathrm{L}}, u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} \left\{ \jmath_{\mathrm{L}}(u_{\mathrm{L}}, u_{\mathrm{F}}) : u_{\mathrm{F}} \in \arg \min_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} \{\jmath_{\mathrm{F}}(u_{\mathrm{L}}, u_{\mathrm{F}})\} \right\}, \tag{2.3}$$

from which a suitable, explicit value $u_{\mathrm{L}}^{\mathrm{d}}$ for the specification of $\gamma_{\mathrm{L}}(\cdot)$ follows.

Moreover, the Stackelberg game (2.3) is equivalent [39, 192] to the bilevel programming problem that can be written:

$$\min_{x \in X} \left\{ F(x, \tilde{y}) : G(x, \tilde{y}) \leq 0, \tilde{y} \in \arg\min_{y \in Y} \{ f(x, y) : g(x, y) \leq 0 \} \right\}, \tag{2.4}$$

for general cost functions $F(\cdot), f(\cdot)$ and constraint functions $G(\cdot), g(\cdot)$. The linear bilevel programming problem is proven to be NP-hard [102] and later strongly NP-hard [89].

Hence, the reverse Stackelberg game can be transformed to the strongly NP-hard bilevel optimization problem, and therefore belongs at least to this complexity class. □

2.1.4 An Indirect Game Formulation

A commonly adopted simplifying approach to the reverse Stackelberg problem is for the leader player to first determine a particular desired optimum $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ that

she seeks to achieve [16, 97, 98]. A natural choice would be a global optimum of the leader: $(u_L^d, u_F^d) \in \arg\min_{u_L \in \Omega_L, u_F \in \Omega_F} J_L(u_L, u_F)$. This global optimum is often referred to as '*team optimum*' according to the theory of teams [88, 138] where it refers to the best the leader can achieve if the other players would support her. The term team optimum is therefore in a sense misleading as a substitute for the leader's, i.e., not the follower's, global optimum. Here it should be noted that it may be difficult to compute such a globally optimal equilibrium point in the case of incomplete information on, e.g., the follower's decision space. Instead of through optimization, depending on the problem setting in which a reverse Stackelberg game is applied, a desired pair of decision variables may also be determined based on past experience, i.e., on historical data, or it may be a particular outcome measured on an ordinal scale [45].

Given such a desired point (u_L^d, u_F^d), the remaining problem can be written as follows:

Indirect Reverse Stackelberg Game Formulation

$$\text{To find:} \quad \gamma_L \in \Gamma_L, \tag{2.5}$$

$$\text{such that} \quad \arg\min_{u_F \in \Omega_F} J_F(\gamma_L(u_F), u_F) = u_F^d, \tag{2.6}$$

$$\gamma_L(u_F^d) = u_L^d. \tag{2.7}$$

Given this indirect formulation of the reverse Stackelberg game, the problem is reduced to finding a leader function that solves the game to optimality, i.e., that leads to the desired leader equilibrium point. The constraints (2.6)–(2.7) imply that the leader should construct her leader function such that it passes through (u_L^d, u_F^d), but such that for all $u_F \in \Omega_F \setminus \{u_F^d\}$, $(\gamma_L(u_F), u_F)$ remains outside of the follower sublevel set:

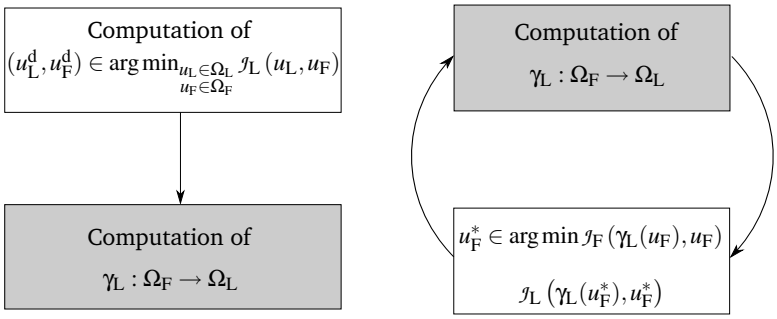
Sublevel Set

$$\Lambda_d := \left\{ (u_L, u_F) \in \Omega_L \times \Omega_F : J_F(u_L, u_F) \leq J_F(u_L^d, u_F^d) \right\}, \tag{2.8}$$

since then, the optimal follower response coincides with the desired decision variable value u_F^d. Based on the concept of this set Λ_d – which we will refer to as '*the sublevel set*' in the remainder of this dissertation, omitting reference to the follower to whom this set applies – one can adopt a geometric approach for the derivation of a suitable leader function.

A schematic representation of the indirect versus the direct approach for solving a reverse Stackelberg game is presented in Fig. 2.3.

We now define the concepts of an equilibrium and of an optimal solution that will be adopted in this thesis. For clarity, first the general game theoretical definition of a solution is described, after which the specific use of this concept throughout this dissertation is clarified.



(a) Indirect Approach (b) Direct Approach

Figure 2.3: *Schematic representation of the direct and indirect solution approaches.*

Definition 2.8 Solution, equilibrium (game theory) [108, 156]
In general game theory, a *solution* can be described as "a systematic description of the outcomes that may emerge in a family of games," [156, p. 2]. In noncooperative game theory, an *equilibrium* can be seen as one of those outcomes, which can either be strategically stable (self-enforcing) or unstable [108], in which case a player is able to deviate (unilaterally or simultaneously with other players) to a strategy that is at least as good as the current strategy, or that causes a better strategy for at least one of the players. ◇

Definition 2.9 Optimal solution, equilibrium (this thesis)

In this dissertation,

- a *(desired) equilibrium (point)* refers to the tuple of decision variables (u_L^d, u_F^d) that is desired by the leader player;
- an *optimal solution* of the reverse Stackelberg game refers to a leader function $\gamma_L : \Omega_F \to \Omega_L$ that – under the assumption of a fully rational follower player – leads to the desired equilibrium point that is specified by the leader player. ◇

In other words, we adopt the terminology of an *equilibrium point*, even if a suboptimal leader function is applied that leads to an undesired follower response $u_F \neq u_F^d$ and the corresponding leader decision, whereas technically, for the point (u_L^d, u_F^d) to be an equilibrium in the general game-theoretical sense, both players should adopt the desired decision variable values.

When assuming a particular parametrized leader function structure, the problem can be further reduced to finding coefficient values for which the parametrized leader function is optimal. Different from the direct approach, it is possible that for a given desired equilibrium point, no leader function exists that can induce the follower player to act according to the desired decisions. An obvious case is the situation in which the desired equilibrium is a boundary point of a follower sublevel

set that intersects with the boundary of the decision space Ω_F in more points than solely (u_L^d, u_F^d) (see Fig. 2.4(e)). The property of a particular desired leader equilibrium to be feasible for an instance of the reverse Stackelberg game is known as *incentive controllability* in the literature [98, 209]. This term stems from the concept *incentive compatibility* as adopted in the theory of incentives (refer to Section 2.2.2 below), where it is used to indicate whether a game or strategy is strategy-proof, i.e., whether the follower can be induced to act truthfully and to reveal his true information in spite of asymmetric information in which the leader is unable to observe the follower's actions [120, 135]. It should thus be noted that this concept is used differently in the context of reverse Stackelberg games, where it refers to the existence of an optimal leader function in cases where the leader may have full information concerning the follower.

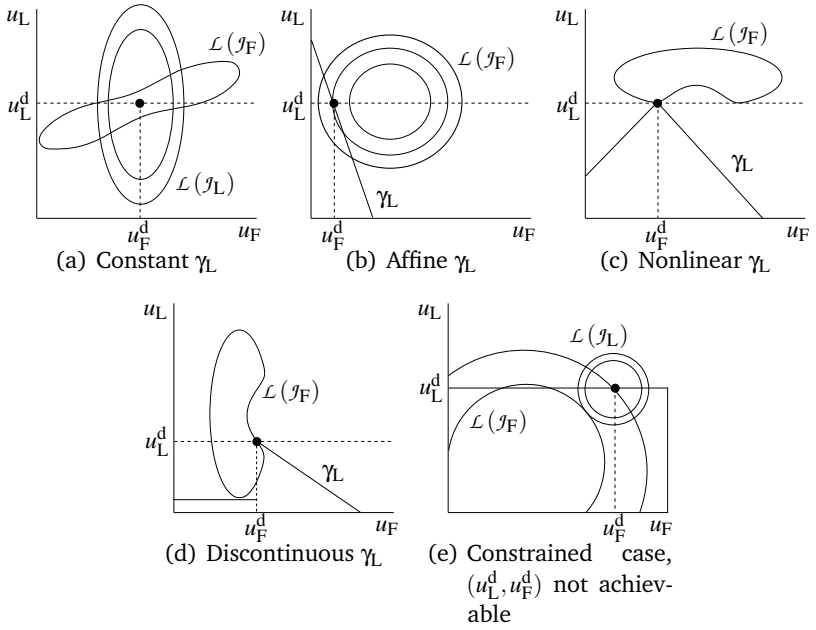
Different types of incentive controllability are illustrated in Fig. 2.4 where the leader's desired equilibrium point (u_L^d, u_F^d) is depicted together with one or several contour lines for the follower or the leader, as denoted by respectively $\mathcal{L}_c(J_F)$ and $\mathcal{L}_c(J_F)$ for a given value $c \in \mathbb{R}$. In Fig. 2.4 the subscript representing the corresponding value is omitted as it is not relevant in this context.

In Fig. 2.4(a) the reverse Stackelberg game reduces to the original Stackelberg framework: the follower's optimal response to u_L^d is the singleton $\{u_F^d\}$; in Fig. 2.4(b) an affine leader function suffices to induce the follower to adopt the desired value u_F^d with the associated $u_L^d = \gamma_L(u_F)$; in Fig. 2.4(c) no affine leader function exists that intersects with the level curve $\mathcal{L}_{J_F(u_L^d, u_F^d)}(J_F)$ or more generally with the sublevel set Λ_d solely in $\{(u_L^d, u_F^d)\}$; in Fig. 2.4(d) a case is depicted in which no optimal continuous explicit leader function exists; in Fig. 2.4(e) the follower sublevel set for (u_L^d, u_F^d) intersects with both upper and lower bound of the leader's constrained decision space, excluding any optimal leader function for which $\gamma_L(\Omega_F) \subseteq \Omega_L$.

In general, the computational complexity of this indirect formulation is still NP-hard, since the minimization of the generally nonlinear, nonconvex functional $J_F(\cdot)$ in the left-hand side of constraint (2.6) is a nonconvex problem, and a similar statement holds for the computation of (u_L^d, u_F^d) as a global optimum with respect to $J_L(\cdot)$. However, we aim to tackle the general problem by evaluating subclasses of solutions or function structures of $\gamma_L(\cdot)$. Moreover, as will be shown in Chapter 3 and Chapter 4, the focus on the sublevel set Λ_d according to (2.8) instead of on the optimization according to (2.6) as applies in the direct formulation (2.2) can ease the computation of a solution. The research presented in this dissertation is focused on the indirect variant (2.5)–(2.7) of the reverse Stackelberg game. However, in Chapter 4 the algorithms devised for the indirect version of the game are compared to a direct solution approach according to the game (2.2).

2.2 The Reverse Stackelberg Game in Different Fields

Since several groups of researchers have considered the reverse Stackelberg game or a similar concept independently while using different terms, for clarity a summary is given and differences and similarities are discussed of the branches of generalized and more recently of inverse Stackelberg games, viz. the theory of incentives



(a) Constant γ_L (b) Affine γ_L (c) Nonlinear γ_L

(d) Discontinuous γ_L (e) Constrained case, (u_L^d, u_F^d) not achievable

Figure 2.4: *Incentive controllability: Reachability of the desired equilibrium point.*

(economics), incentive strategies (control), and of the related field of bilevel programming.

2.2.1 Generalized and Inverse Stackelberg Games

The first step towards the reverse Stackelberg game formulation may be found in [123] where a generalized strategy is introduced that leads to the best solution the leader can achieve in case the follower's response to the original Stackelberg decision u_L^d is nonunique. In the original Stackelberg game, uniqueness of the follower response is usually assumed at a loss of generality, in order to simplify the problem. Formally, if for every $u_L \in \Omega_L$ it holds that

$$\Omega_F^*(u_L) := \{u_F^* \in \Omega_F \,|\, \mathcal{J}_F(u_L, u_F^*) \leq \mathcal{J}_F(u_L, u_F), \forall u_F \in \Omega_F\} \neq \varnothing, \tag{2.9}$$

and if there exists a $u_{Lgen}^* \in \Omega_F$ such that

$$\sup_{u_F \in \Omega_F^*(u_{Lgen}^*)} \mathcal{J}_L(u_{Lgen}^*, u_F) = \min_{u_L \in \Omega_L} \sup_{u_F \in \Omega_F^*(u_L)} \mathcal{J}_L(u_L, u_F) = \mathcal{J}_L^*, \tag{2.10}$$

then u_{Lgen}^* is called a *generalized Stackelberg strategy* for the leader, i.e., it leads to the least upper bound on $\mathcal{J}_L(\cdot)$ [123].

It should be noted that this generalized strategy basically results in a reduced set of possible Stackelberg solutions for the leader that constitute an upper bound to her objective function value. The generalized strategy thus deals with the problem of a nonunique follower response by accepting a solution that leads to a reduced

performance for the leader. In contrast, the reverse Stackelberg game deals with a nonunique response by substituting the leader strategy u_{L} with a more involved function $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$.

The term *reverse Stackelberg game* first appeared[2] in [97], where it was chosen to illustrate the order of first announcing the leader strategy $\gamma_{\mathrm{L}}(\cdot)$ (rather than her action $u_{\mathrm{L}} \in \Omega_{\mathrm{L}}$ as in the original Stackelberg formulation), followed by the follower's actual action or decision $u_{\mathrm{F}} \in \Omega_{\mathrm{F}}$, from which u_{L} follows. Instead of approaching only an upper bound on $J_{\mathrm{L}}(\cdot)$ by using the generalized strategy, the leader may then in fact be able to reach exactly her desired equilibrium value.

As additional reasons for adopting a reverse structure, it is mentioned in [97] that (i) the leader may infer information on the system state from knowing the follower's decision first, especially in a stochastic setting in which the leader does not possess the follower's full information, and that (ii) the follower's decision may directly affect the leader's objective function value. However, to the latter argument it should be added that also in the original Stackelberg game $J_{\mathrm{L}}(\cdot)$ is dependent on $u_{\mathrm{F}} \in \Omega_{\mathrm{F}}$. The reverse structure does provide more 'power' to the leader to influence the follower and enforce her desired solution, as compared to when providing only a decision $u_{\mathrm{L}} \in \Omega_{\mathrm{L}}$. More formally, in terms of the reaction sets $\mathcal{R}_{\mathrm{SG}}, \mathcal{R}_{\mathrm{RSG}}$ of the follower player for the original Stackelberg and reverse Stackelberg game, respectively:

$$\mathcal{R}_{\mathrm{SG}} := \left\{ u_{\mathrm{F}} : u_{\mathrm{F}} \in \arg\min_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} J_{\mathrm{F}}\left(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}\right) \right\}, \tag{2.11}$$

$$\mathcal{R}_{\mathrm{RSG}} := \left\{ u_{\mathrm{F}} : u_{\mathrm{F}} \in \arg\min_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} J_{\mathrm{F}}\left(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}}\right) \right\}, \tag{2.12}$$

it holds that $\left\{J_{\mathrm{F}}\left(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}\right)\right\}_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} \subseteq \left\{J_{\mathrm{F}}\left(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}}\right)\right\}_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}}$ for $\gamma_{\mathrm{L}}(\cdot)$ such that $\gamma_{\mathrm{L}}(u_{\mathrm{F}}^{\mathrm{d}}) = u_{\mathrm{L}}^{\mathrm{d}}$.

Most recently, the game in which the leader announces a strategy as a mapping $\Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$ has been studied as the *inverse Stackelberg game* [153, 154, 178, 180]. There, several problem instances are investigated to show the difficulties in solving this game, both for the static [153] and the dynamic [154] case, and for cases with multiple leaders or followers [153]. Nonetheless, over the years several conditions have been developed for the existence of an optimal solution of a particular (affine) structure, especially within the research on incentives strategies [132, 209].

2.2.2 Theory of Incentives

Theory of incentives, also known as contract theory, involves so-called *principal-agent* problems in which some quantity produced by the agent or follower is exchanged for a (monetary) transfer by the principal or leader. A new element of information is considered here, i.e., the so-called type of an agent that refers to, e.g., skills or opportunity cost. The agent may not reveal his type to the principal or he may even provide false characteristics. Therefore, an aspect of paramount importance within this area is uncertainty due to a lack of information. The three main

[2] To be precise, in [97] the term 'reversed Stackelberg problem' was adopted.

types of principle-agent problems are moral hazard, adverse selection, and signaling. Here, the agent has either (i) private information concerning actions that occur after the signing of a contract, (ii) private information concerning his type before the composition of the contract, or (iii) the ability to send information to the principal during the game [120, 135]. Although controller agents are usually assumed to provide their available information truthfully, results from the theory of incentives concerning incomplete information can provide useful insight to the reverse Stackelberg game formulation in control settings [99].

Another important part of the problem definition in incentives theory is the participation constraint or bail-out option of the follower, which allows him to withdraw from participating in the game in case the leader proposes a contract that leaves the follower with an insufficient performance. This constraint does not directly appear in the reverse Stackelberg game formulations (2.9)–(2.10) and (2.2) or (2.5)–(2.7) mentioned in respectively Sections 2.2.1 and 2.1.2, nor in the related problem descriptions of Section 2.2.3 and Section 2.2.4.

2.2.3 Incentive Strategies

From a control-theoretic rather than an economic perspective, the leader strategy is often referred to as 'incentive' [98, 99]; as in Section 2.2.2, the term is chosen to indicate the problem of how the leader can incentivize the follower to perform as desired. Different from the leader function as described in Section 2.1.2, the incentive strategy is not always a mapping $\Omega_{\mathrm{F}} \rightarrow \Omega_{\mathrm{L}}$; some authors define the incentive strategy more generally as a function of the available information [99], or solely of the system state variables as will be introduced in Section 2.4.1 [98, 125, 184]. In fact, in [125] the use of state feedback is motivated by the argument that it is unrealistic to have access to the follower's decision variables in a real-life dynamic setting. At the same time, some authors consider such state-dependent leader function as a regular (feedback or closed-loop) Stackelberg strategy without mentioning the concept of incentives [16, 44, 175].

The incentives information structure has also been considered as a fourth alternative along with the open-loop, closed-loop, and feedback information structure in a multistage context [55]. Although the last three patterns are indeed only relevant in a dynamic framework (see Section 2.4.1), the reverse Stackelberg game or incentives structure with $u_{\mathrm{L}} = \gamma_{\mathrm{L}}(u_{\mathrm{F}})$ can, however, very well occur in a single-stage context without the presence of a state variable.

A link has also been made between incentives and *social choice theory* in [98]. In social choice theory, agents need to propose an ordering of preferences (e.g., in the voting for elections) based on which a final listing (the solution or election outcome) is developed, depending on a predetermined choice rule [5]. In order to make people reveal their true preferences, the choice rule should be strategy-proof. In [98] the equivalence is stated between a leader function of a reverse Stackelberg game and a social choice rule that allocates a final ordering (solution) to a preference ordering that represents the decision variables of the agents in a strategy-proof manner. However, there is no desired election order as an outcome that the leader strives after in social choice theory, as opposed to in the reverse Stackelberg game

where the leader optimizes $J_L(\cdot)$, which is directly dependent on the follower decision u_F. Therefore, the proposed resemblance with a reverse Stackelberg game does not completely fit. Finally, also the related field of *mechanism design* deals with finding strategies that induce players to convey their true information, which can be useful to study in the context of reverse Stackelberg games with incomplete information [65, 121].

In order to put the different terms in perspective, we may conclude that the incentive problem of determining the leader function $\gamma_L(\cdot)$ to induce the follower to behave as desired can be seen as a part of the overall reverse Stackelberg game, whereas the design of more general incentive strategies is also present in a broader class of problems.

2.2.4 Bilevel Programming

Finally, the Stackelberg game can be rewritten as a bilevel programming problem, in which the follower's lower-level optimization problem is considered as a constraint to the higher-level optimization problem [39, 192]. Different from the perspective of the Stackelberg game, bilevel programming is focused rather on the *computation* of a Stackelberg solution, where the sequential nature of the game is translated into constraints. Whereas cases with incomplete information can apply in the game, in a translation to a multilevel mathematical program, perfect information is assumed [54]. While the resemblance with the original Stackelberg game is often mentioned in the literature on multilevel programming, a link with the reverse game does not appear. Nonetheless, the reverse game is subject to the same hierarchical structure, where in addition the relation between u_L and u_F is captured by $\gamma_L(\cdot)$. In other words, the Stackelberg game is a special case of the reverse game: for a relation $\gamma_L : \Omega_F \rightarrow \{u_L^d\}$, the reverse game reduces to a Stackelberg game. Hence, results on multilevel programming [39, 52] could prove useful for the analysis of the reverse Stackelberg game. In general, linear bilevel, hence multilevel, programming is proven NP-hard [102] and later strongly NP-hard [89]. A more elaborate (complexity) analysis of multilevel programming can be found in [36, 89].

2.3 Solution Approaches

2.3.1 Analytic Solution Approaches

In order to ease the solvability of the reverse Stackelberg game, an indirect approach can be adopted as explained in Section 2.1.4 above. If the leader is able to induce the follower to arrive at the desired equilibrium point (u_L^d, u_F^d) by the application of an affine leader function $\gamma_L(\cdot)$, the problem can been called linearly incentive controllable [98, 209]. Except for the derivation of such leader functions that can be applied in some special linear-quadratic cases of the – in that case linearly incentive controllable – reverse Stackelberg game [34, 56, 139, 209], by the author's best knowledge no analytic approaches have been presented thus far. We will elaborate and extend results on this analytic solution approach in Chapter 3.

2.3.2 Numerical Solution Methods

While research on the reverse Stackelberg game from a game-theoretical or even control-theoretical perspective aims mostly towards obtaining analytical solutions or leader functions, for available numerical solution methods, inspiration can be gained from multilevel programming. Available solution methods for such programming problems can be categorized as extreme point algorithms, branch-and-bound algorithms, complementary pivot algorithms, descent methods, and penalty function methods [192]. More references to algorithms for multilevel programming can be found in [131].

Alternatively, evolutionary programming methods have been suggested for solving Stackelberg games in [186] and [179], as will be mentioned in Section 2.4.4 for applications in which incomplete information applies, and as will be further elaborated upon in Chapter 4. An overview with references on genetic algorithms applied to multilevel programming problems can be found in [131]. There, a genetic algorithm was developed for general multilevel Stackelberg games in which players on a single level play a noncooperative simultaneous game for which the Nash equilibrium concept is adopted, without assumptions regarding linearity, convexity, continuity, or differentiability. On the other hand, the follower's response is assumed to be a singleton or the leader is assumed to be indifferent amongst the follower's optimal decisions in case these are nonunique. The relaxation of these assumptions leads exactly to the need for a reverse Stackelberg formulation as explained in Section 2.1. Although the method is able to find a global optimum for the general game, the approach is computationally still rather inefficient.

A more recent development involves using multiparametric programming methods for multilevel optimization [59], in which each subproblem is stated as a multiparametric programming problem with parameters linked to other subproblems. The complexity of the overall problem is thus broken down to the computation of the reaction set of optimal, parametrized solutions for each subproblem. When studying the particular linear quadratic case, this approach results in a single-level convex optimization problem. However, efficient methods for general nonlinear and nonconvex multilevel problems are far from widespread.

In Chapter 4 we will discuss new solution approaches for the indirect reverse Stackelberg game formulation and compare these with the evolutionary approaches devised to solve the direct game variant.

2.4 A Thematic Overview of Results

A reverse Stackelberg game includes with the basic description as provided in Section 2.1 a specification of (1) time elements, specifying amongst other aspects the duration of the game, (2) leveling, and (3) information and uncertainty [98, 99]. In the current section, an overview of contributions in the area of reverse Stackelberg games is provided, categorized into these several aspects as is also depicted in Fig. 2.5, viz.:

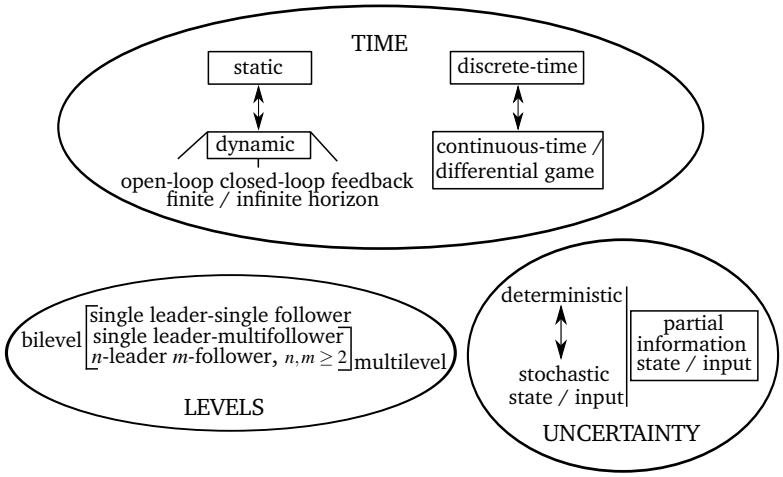


Figure 2.5: *Overview of characteristics within the reverse Stackelberg game.*

2.4.1 static versus dynamic problems;

2.4.2 continuous-time differential problems;

2.4.3 deterministic versus stochastic problems;

2.4.4 problems with partial, nonnested information;

2.4.5 performance of a sensitivity analysis;

2.4.6 multilevel, multiplayer problems.

2.4.1 Static Versus Dynamic Problems

There are not so many results from a control-theoretic perspective that consider the static reverse Stackelberg game. As is also mentioned in [154], a legitimate reason for studying the dynamic case is that it more often occurs in real-life settings. On the contrary, results within the theory of incentives and especially multilevel programming are often based on single-stage problems [39, 135].

Nonetheless, an introduction to the static reverse Stackelberg game can be found in [98], as well as in [167], where a static affine leader function is presented, motivated by a situation with the regulating government as a leader that strives to achieve Pareto optimality while multiple followers (firms) play according to their Nash equilibrium strategies. Since objective functions in [167] are assumed to be quadratic and strictly convex, an optimal affine strategy can be computed.

When considering a dynamic game, in addition, often a state variable $x \in X \subseteq \mathbb{R}^{n_x}, n_x \in \mathbb{N}$ is considered to define the global system with an associated update equation, i.e., in discrete time:

$$x(k+1) = A_L(k)u_L(k) + A_F(k)u_F(k) + B(k)x(k), \tag{2.13}$$

with $x_0 = x(0)$ the initial state and with matrices $A_{\mathrm{L}}, A_{\mathrm{F}}, B$ of appropriate sizes, with $k \in \{1,\ldots,K\}$ where K denotes the number of stages. The continuous-time equivalent, i.e., the differential game, will be described in Section 2.4.2. One can also distinguish between the state space of the leader and the follower, i.e., $\mathcal{X} := \mathcal{X}_{\mathrm{L}} \times \mathcal{X}_{\mathrm{F}}$. This state variable can be integrated in the objective function $\mathcal{J}_p : \Omega_{\mathrm{L}} \times \Omega_{\mathrm{F}} \times \mathcal{X} \rightarrow \mathbb{R}, p \in \{\mathrm{L},\mathrm{F}\}$.

The general game where the aspects depicted in Fig. 2.5 are specified can now be denoted by a tuple

$$\left(\mathcal{P}, K, \{\mathcal{J}_i\}_{i\in\mathcal{N}}, \{(\Omega_i)^K\}_{i\in\mathcal{N}}, \mathcal{X}^K, f, x_0, (\mathcal{I}_i)^K\}_{i\in\mathcal{N}}, \{(\Gamma_i)^K\}_{i\in\mathcal{N}}\right), \tag{2.14}$$

where

- $\mathcal{P} : \mathcal{N} \rightarrow \{\mathrm{L},\mathrm{F}\}$ indicates the type of player associated with each of the players $i \in \mathcal{N} = \{1,...,N\}$.
- $\mathcal{I}_i \subseteq \mathbb{R}^{n_i \times N}$ denotes the information space of player i.
- $\Gamma_i \subseteq \{\gamma_i | \gamma_i : \mathcal{I}_i \rightarrow \Omega_i\}$ denotes the strategy space.

To elaborate, $\iota_i(k) \in \mathcal{I}_i$ captures the knowledge that is available to player i at stage k: it can contain information of the other players' decision space and objective function, state, etc. In the leader's case this includes the follower's decisions. Finally, the strategy space refers to the permissible mappings $\gamma_i(k) : \mathcal{I}_i \rightarrow \Omega_i$ that lead to a decision $u_i(k)$. These mappings can be announced several stages ahead.

Depending on the time indices of the variables on which a leader function is dependent, a leader strategy in a dynamic game can be:

- an *open-loop strategy*: these are a function of the time step and of the initial conditions only, where the leader function in the reverse Stackelberg game is in addition dependent on the follower's decision: $\gamma_{\mathrm{L}}(k) = g_{\mathrm{ol}}(x_0, u_{\mathrm{F}}(k), k)$.
- a *closed-loop strategy*: here, decisions can be defined to be a function of the time step and of the current and some α previous-stage state and decision variables $x(k-\alpha),...,x(k), u_{\mathrm{F}}(k-\alpha),...,u_{\mathrm{F}}(k)$.
- a *feedback strategy*: state *feedback* strategies can be seen as a special type of closed-loop decisions with one-step memory, i.e., the strategies depend on the previous state and decision variables, with $\gamma_{\mathrm{L}}(k) = g_{\mathrm{fb}}(x(k-1), u_{\mathrm{F}}(k-1), k)$ (a feedback Stackelberg strategy), or $\gamma_{\mathrm{L}}(k) = g_{\mathrm{fb}}(x(k-1), u_{\mathrm{F}}(k-1), u_{\mathrm{F}}(k), k)$ (a feedback reverse Stackelberg strategy with a leader function directly dependent on the current follower decision variable).

Solving a closed-loop or feedback game requires different solution methods from those applicable to the static game. However, the open-loop game is a straightforward extension of the static game. In other words, at the start of the discrete-time dynamic game the optimal open-loop values $(u_{\mathrm{L}}^{\mathrm{d}}(k), u_{\mathrm{F}}^{\mathrm{d}}(k))$ for the leader can be computed for time steps $k \in \mathcal{K}$, taking into account the predicted values of the state variables. Then, the mappings $\gamma_{\mathrm{L}}^k : \Omega_{\mathrm{F}} \rightarrow \Omega_{\mathrm{L}}$ can be computed as is done for

the static case, now for each $k \in \mathcal{K}$. Since we evaluate the performance measures $\jmath_p^k(\cdot), p \in \{\mathrm{L},\mathrm{F}\}$ – that can be dependent on $k \in \mathcal{K}$ in addition to $u_p \in \Omega_p$ – in open loop with $\mathcal{K} = \{1,2,...,K\}$ in discrete time, the game basically results in a series of static optimization problems, i.e., where the desired equilibrium and the follower response can be written:

$$(u_{\mathrm{L}}^{\mathrm{d}}(k), u_{\mathrm{F}}^{\mathrm{d}}(k)) \in \arg\min_{\substack{u_{\mathrm{L}}\in\Omega_{\mathrm{L}} \\ u_{\mathrm{F}}\in\Omega_{\mathrm{F}}}} \jmath_{\mathrm{L}}^k(u_{\mathrm{L}}, u_{\mathrm{F}}); \quad u_{\mathrm{F}}^*(k) \in \arg\min_{u_{\mathrm{F}}\in\Omega_{\mathrm{F}}} \jmath_{\mathrm{F}}^k\left(\gamma_{\mathrm{L}}^k(u_{\mathrm{F}}), u_{\mathrm{F}}\right). \tag{2.15}$$

Note that in this formulation, the objective functions as well as the state and decision spaces could be considered time-variant without consequences for the results presented in this dissertation. In order for conciseness, in this dissertation no state variables will be adopted as arguments of the leader function.

Finally, note that *repeated games* can be regarded as open-loop dynamic games without the specification of a state variable and in which the players' final cost is a weighted average over the stages [65]. In terms of so-called *tit for tat* strategies in repeated games, in which a player is 'punished' for a given number of successive stages if (s)he does not collaborate [65], a leader function in a (static) reverse Stackelberg game already incorporates one-stage 'punishment' by associating leader decisions that are less desirable for the follower to every follower decision that is undesirable for the leader. In cases where the follower may behave suboptimally, the leader however cannot completely control the follower's decision, which makes it impossible to adopt such 'punishment' for multiple stages in a repeated reverse Stackelberg game, even if the follower does behave as desired. If the leader wants to 'punish' a follower for an undesired, suboptimal behavior for multiple stages, she has to adopt suboptimal leader functions that do not result in the leader's desired equilibrium.

Although the static leader functions in [98] are of the form $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$, in the multistage case the follower decision variable is replaced by a state-dependent leader function, which is only indirectly dependent on the follower input. Also in [184] derivations of such state-dependent leader strategies that are nonetheless called incentive strategies can be found. As has just been explained, such strategies could be seen as closed-loop or feedback strategies of the original Stackelberg game, i.e., they are different from the leader function as defined in the reverse Stackelberg game, in which the leader's decision at stage k is dependent on $u_{\mathrm{F}}(k)$.

For the static and dynamic open-loop case, a sufficient condition was derived in [209] for the existence of an optimal affine leader function

$$u_{\mathrm{L}} = \gamma_{\mathrm{L}}(u_{\mathrm{F}}) = u_{\mathrm{L}}^{\mathrm{d}} + B(u_{\mathrm{F}} - u_{\mathrm{F}}^{\mathrm{d}}), \tag{2.16}$$

where the decision spaces are assumed to be Hilbert spaces, with $B : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$ a linear operator, i.e.: $B^*\nabla_{u_{\mathrm{L}}}\jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) = -\nabla_{u_{\mathrm{F}}}\jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$, with $B^* : \Omega_{\mathrm{L}} \to \Omega_{\mathrm{F}}$ the adjoint of B. Such an operator B now exists if $\nabla_{u_{\mathrm{L}}}\jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \neq 0$. However, these results are restricted to games where $\jmath_{\mathrm{F}}(\cdot)$ is convex and locally strictly convex as well as twice continuously differentiable. This result will be elaborated upon in Chapter 3 (Section 3.4.1).

Further, in [132] it is proven that the optimal affine leader function is unique for linear-quadratic (LQ) dynamic games with quadratic cost functions and linear state update equations for the case with a scalar follower decision variable. For $n_{\mathrm{F}} > 1$ a unique affine strategy can still be found if in addition to the conditions of [209], additional conditions regarding the system matrices are satisfied, for which the reader is referred to [132].

More recently, in [125] the analysis of the affine incentive structure for a linear-quadratic discrete-time system has been continued (both for the finite and infinite horizon case) but instead of using a leader function that is in fact dependent on u_{F}, also there, state feedback is applied for both players. We therefore do not consider this case as a truly *reverse* Stackelberg game. Such dynamic state-feedback strategies in LQ problems have also been studied before in continuous time in, e.g., [168].

It should be noted that the dynamic game with a linear state equation and quadratic cost functions is widely used as an illustrative example, e.g., amongst several other references stated in this survey section, in [125, 167, 184]. Moreover, existence results for an optimal (affine) leader function also rely on this specific LQ case, as in [209].

2.4.2 Continuous-Time Differential Problems

While so far the discrete-time reverse Stackelberg game has been considered in this survey, the results can also be extended to the continuous-time differential game. Also in (Stackelberg) differential games, the LQ problem structure is a popular one [56, 58, 139]; the LQ continuous-time structure of state update equation and objective functions for a reverse Stackelberg can be written as follows:

$$\dot{x}(t) = A(t)x(t) + B_{\mathrm{L}}(t)u_{\mathrm{L}}(t) + B_{\mathrm{F}}(t)u_{\mathrm{F}}(t), \quad x(t_0) = x_0, \tag{2.17}$$

$$\begin{aligned} \jmath_i(u_{\mathrm{L}}, u_{\mathrm{F}}) = \frac{1}{2}x^{\mathrm{T}}(t_{\mathrm{f}})Q_{i,t_{\mathrm{f}}}x(t_{\mathrm{f}}) + \frac{1}{2}\int_{t_0}^{t_{\mathrm{f}}} (x^{\mathrm{T}}(t)Q_i x(t), \\ + u_i^{\mathrm{T}}(t)R_{ii}u_i(t) + u_j^{\mathrm{T}}(t)R_{ij}u_j(t)\big)\,\mathrm{d}t, \end{aligned} \tag{2.18}$$

$i, j \in \{\mathrm{L}, \mathrm{F}\}, i \neq j$, $t \in [t_0, t_{\mathrm{f}}]$, where the matrices are of appropriate dimension, and $Q_{i,t_{\mathrm{f}}} \geq 0, Q_i \geq 0, R_{ij} \geq 0, R_{ii} > 0$.

In [210] conditions are developed under which the reverse Stackelberg game with memory from stage $\tau \in [t_0, t]$ at stage $t \in [t_0, t_{\mathrm{f}}]$, has the following optimal solution:

$$u_{\mathrm{L}}(t) = \gamma_{\mathrm{L}}(u_{\mathrm{F}})(t) = u_{\mathrm{L}}^{\mathrm{d}}(t) + \int_{t_0}^{t} R(t,\tau)(u_{\mathrm{F}}(\tau) - u_{\mathrm{F}}^{\mathrm{d}}(\tau))\mathrm{d}\tau, \tag{2.19}$$

which is the continuous-time equivalent to the affine function (2.16). However, in order to find a matrix $||R|| < \infty$ as required for a leader function, a necessary condition is that $||\nabla_{u_{\mathrm{F}}} \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})(t)|| < c(t_{\mathrm{f}} - t), t \to t_{\mathrm{f}}$, with c a constant [210].

In order to relax this requirement, in [56] the leader strategy with memory is instead defined by using the Lebesgue-Stieltjes integral:

$$u_{\mathrm{L}}(t) = \gamma_{\mathrm{L}}(u_{\mathrm{F}})(t) = u_{\mathrm{L}}^{\mathrm{d}}(t) + \int_{t_0}^{t} [d_\theta \eta(t,\theta)](u_{\mathrm{F}}(\theta) - u_{\mathrm{F}}^{\mathrm{d}}(\theta)), \tag{2.20}$$

with $\theta \in [t_0, t], t \in [t_0, t_f]$ and where $\eta : [t_0, t_f] \times \mathbb{R} \to \mathbb{R}^{n_x+n_F}$ represents the vector of available information. Analytically solvable necessary and sufficient conditions are obtained for the optimality of this strategy in case of convex cost functions. Different representations of $\eta(t,\theta)$ are considered in [56]; however, they are dependent on the initial state x_0 rather than explicitly on $u_F(t)$, for which the authors state that stringent conditions on the game parameters would be needed. Likewise, in [159] strategies dependent on $u_F(t)$ and past values of x are suggested. However, the dependence on $u_F(t)$ is not explicitly adopted in the derivation of (sufficient) conditions of [159].

In [56] also time-delay strategies are considered in which $\gamma_L(t)$ depends linearly on $u_F(\theta)$ for $t-\delta \leq \theta \leq t-\sigma, \delta \leq \sigma$, with $\delta > 0, \sigma \geq 0$ constants. The necessary and sufficient existence conditions on optimal leader functions are also shown to extend to this time-delay case, where a distinction is made for the strategy $\gamma_L(t)$ between the cases $t \leq t_0 + \delta$ and $t > t + \delta$.

Also in [139] conditions are developed for the existence of an optimal affine strategy in a continuous-time LQ game. There, not only the leader function is taken to be dependent on u_F; the same strategy applies to the second player in the game:

$$\gamma_i(u_j) = u_j^d + D_j(u_i - u_i^d), i, j = 1,2, i \neq j, \tag{2.21}$$

with $D_j : \Omega_i \to \Omega_j$, $D_j \neq 0, j = 1,2$ for scalar variables u_1, u_2. Although the derivation of this strategy is useful for the leader in the reverse Stackelberg game, it does not follow a true Stackelberg setting but rather a cooperative game with equivalent players.

2.4.3 Deterministic versus Stochastic Problems

As a stochastic reverse Stackelberg game usually the case is considered in which the state variable of the game includes random components; in general, the state variable is assumed to have a known distribution, often Gaussian with zero mean [34, 97].

In [97] the two-player, static reverse Stackelberg case has been analyzed with a randomly (Gaussian) distributed cost function parameter ξ. The leader's cost function is now defined as $J_L = \mathbb{E}[L_L(\gamma_L, u_F, \xi)]$, with $L_L : \Omega_L \times \Omega_F \times \Xi \to \mathbb{R}$, with $\mathbb{E}[\cdot]$ the expected value operator; $J_F(\cdot)$ is defined accordingly.[3]

In particular, the static LQ-Gaussian (LQG) case is examined and sufficient conditions for the existence of an optimal affine leader function are obtained under several additional assumptions on the game parameters, for which we refer to [97]. In the analysis of how the results translate to a multistage version of the game, however, a state feedback strategy is adopted, i.e., the leader's strategy is no longer formulated as a function of the follower's decision as is applicable in a reverse Stackelberg game.

In [34] a stochastic closed-loop reverse Stackelberg game is considered with a leader function that is directly dependent on the current or previous-stage value

[3] It should be noted that ξ is called a state vector in [97], defined to represent some unknown elements of the game in both the static and the dynamic case. This vector ξ should therefore not be confused with the system state variable $x \in X$ we use in the dynamic, multistage game.

of the follower decision variable, or on the (partial) information that the follower conveys to the leader. It is shown that the three problems can be solved in a similar way, and they lead to an optimal solution in case a LQG problem is considered.

Several results on stochastic cases also consider incomplete information, as will be briefly discussed in Section 2.4.4 below.

2.4.4 Problems with Partial, Nonnested Information

Another variant of uncertainty is in the lack of complete information; in particular within the theory of incentives, different types of information asymmetry are studied.

In [97, 98] both a nested and nonnested stochastic reverse Stackelberg game is considered, where the random variable ξ as introduced in Section 2.4.3 is assumed to follow a Gaussian distribution. Whereas in the nested case, a follower's information is a subset of the information that the leader possesses, in the nonnested case, the leader does not have access to all knowledge relevant to the leader about the followers. In the latter case, the leader is generally unable to compute her globally optimal solution. In order to arrive at a feasible desired equilibrium, the restrictive assumption is made that L_L is in fact independent of u_L [98]. Assuming that Ω_F is known, the desired leader solution is now defined as a $\gamma_\mathrm{L} : \Omega_\mathrm{F} \to \Omega_\mathrm{L}$ such that

$$\arg \min_{u_\mathrm{F} \in \Omega_\mathrm{F}} \mathbb{E}[L_\mathrm{F}(\gamma_\mathrm{L}(u_\mathrm{F}), u_\mathrm{F}, \xi)] = \arg \min_{u_\mathrm{F} \in \Omega_\mathrm{F}} \mathbb{E}[L_\mathrm{L}(u_\mathrm{F}, \xi)]. \tag{2.22}$$

Also in [99] an overview is provided of possible incomplete information structures in a stochastic setting, both for a static and a dynamic game. There, the leader strategy is taken to be a function of the available information, which does not in all cases include $u_\mathrm{F} \in \Omega_\mathrm{F}$. In other words, the case is analyzed in which the leader cannot observe the follower's decision. As in the theory of incentives, the focus of [99] is on the follower not acting truthfully in the case of incomplete information.

In a setting with multiple followers, a stochastic, random state reverse Stackelberg game is considered in [31] where the leader has access to a linear combination of the followers' actions. Here, the followers' cost functions are strictly convex and continuously differentiable. An affine leader function is computed that is based on this random linear combination; it is shown that the performance obtained by the leader is equivalent to the performance that applies in case she would be able to observe the followers' individual actions. In [14] this result is expanded to deal with more than two levels of hierarchy, according to the technique from [98] described in Section 2.4.6 below.

In case no knowledge is available of $\jmath_\mathrm{F}(\cdot)$ or of the follower's reaction curve that can be directly derived from $\jmath_\mathrm{F}(\cdot)$, an iterative learning procedure may be adopted to arrive at a close-to optimal leader decision [187] or a leader function in the reverse case [186]. For this purpose the use of a genetic algorithm is proposed and compared with a standard gradient approach for off-line computation of an incentive strategy in [186]. It should be noted that, when adopting such iterative procedure, the game would have to be repeated until convergence is reached and the resulting strategy yields a sufficient performance for the leader. This requires a setting in which a start-up period with suboptimal policies for the leader would be possible. Alternative to

performing several off-line iterations of the game in order to make up for the missing information, methods could be investigated for enforcing the follower to (truthfully) communicate $J_F(\cdot)$ to the leader. Finally, in general when using the genetic approach, no analytical convergence guarantees or suboptimality bounds can be obtained for the leader's neither for the the follower's performance.

2.4.5 Sensitivity Analysis

Since the set of possible optimal (affine) leader functions is often nonunique, a minimum sensitivity approach to incentive strategies is developed in [29]. In case the leader does not have full information on $J_F(\cdot)$, i.e., it is assumed that the value of one or several of the parameters of a known parametrization of $J_F(\cdot)$ is unknown to the leader but the range of the parameter(s) is known, a robust leader function can be computed based on nominal values of the unknown parameter(s). Here, robustness of a leader function refers to the degree of sensitivity to deviation of the parameters from their nominal values. Based on the results of [209], it is known that for each possible value of the unknown parameters, there exists an optimal affine leader function under the assumption of strictly convex cost functions and rational follower behavior. Next to the proposed affine leader function, additional degrees of freedom in the reduction of sensitivity can be introduced by incorporating also nonlinear terms in the strategy [29]. The least-sensitive optimal strategy is taken to belong to the twice continuously differentiable functions with bounded first and second derivatives with respect to $u_F \in \Omega_F$. However, it should be noted that in [29] no explicit analysis is provided of the performance of the leader in case the assumed nominal values characterizing $J_F(\cdot)$ are incorrect.

In [30] the work of [29] is extended to include stochastic incentive schemes, where again some parameters characterizing the unknown part of $J_F(\cdot)$ vary around some nominal value and where in addition the state is a random variable. A smooth strategy is found that results in the desired leader solution, which solution is again based on the assumed nominal values under the assumption of strictly convex and twice continuously differentiable cost functions. Compared to the deterministic case, in the stochastic setting the follower's optimal response is proven to be minimally or even completely insensitive to variations in the unknown parameter values with respect to the nominal values. It should be noted though that this result is only possible in case the leader is assumed to have full access to the follower's information.

2.4.6 Multilevel, Multiplayer Problems

The static reverse Stackelberg game with multiple leaders or followers is considered in [153], where it is noted that not much theory is available with respect to multiplayer extensions to the reverse Stackelberg game. Indeed, most cases with multiple followers assume that these play a noncooperative simultaneous Nash game amongst themselves and act as one follower group in response to the single leader [167], where the leader strategy is of the form $\gamma_L : \Omega_{F,1} \times \cdots \times \Omega_{F,n} \to \Omega_L$ for n followers [99, 103]. This setting is also considered in [153], where in case of multiple leaders, also these are assumed to announce their leader function simultaneously

according to their Nash equilibrium.

In [103] an alternative is presented to multiple followers playing a simultaneous Nash game. There, it is shown that a single leader function $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F},1} \times \cdots \times \Omega_{\mathrm{F},n} \to \Omega_{\mathrm{L}}$ can prove sufficient to reach the leader's desired equilibrium, since the n followers' objective functions become identical except for a constant; the problem then reduces to a single leader-single follower game. The results are, however, based on the assumption of continuous differentiability of $\jmath_{\mathrm{F},i}$ on $\Omega_{\mathrm{L}} \times_{i=1,\ldots,n} \Omega_{\mathrm{F},i}$ for all $i = 1,\ldots,n$ and on the existence of some k such that $\partial \jmath_{\mathrm{F},i}/\partial u_{\mathrm{F},k} \neq \partial \jmath_{\mathrm{F},j}/\partial u_{\mathrm{F},k}$ $i \neq j$, $i, j = 1,\ldots,n$, as well as on the convexity of $\jmath_{\mathrm{F},i}$ on $\Omega_{\mathrm{L},-k} \times_{i=1,\ldots,n} \Omega_{\mathrm{F},i}$ and strict convexity at $(u^{\mathrm{d}}_{\mathrm{L},-k}, u_{\mathrm{F},1}, \ldots, u_{\mathrm{F},N})$ with $u_{\mathrm{L},-k} = (u_{\mathrm{L},1}, \ldots, u_{\mathrm{L},k-1}, u_{\mathrm{L},k+1}, \ldots, u_{\mathrm{L},n_{\mathrm{L}}}) \in \Omega_{\mathrm{L},-k} \subseteq \mathbb{R}^{n_{\mathrm{L}}-1}$ [103].

The same idea of realizing similar follower cost functions has been discussed by means of a specific numerical example in earlier work [98]. There, the leader adopts one distinct decision variable for each of the n followers, and proposes for each follower $i = 1,\ldots,n$ a different leader function $\gamma_{\mathrm{L},i} : \Omega_{\mathrm{F},1} \times \cdots \times \Omega_{\mathrm{F},n} \to \Omega_{\mathrm{L},i}$.

True multihierarchy settings in which players perform as a leader and follower simultaneously with respect to the upper and lower levels, respectively have also been briefly studied in [98]. Sufficient conditions are stated such that the lower-level players are induced to perform as desired for the higher-level players, by successively substituting the leader functions in the order of announcement. For a three-level system with player 1 being the upper level leader and with $\gamma_i : \times_{i=1}^{3} \Omega_i \to \mathbb{R}, i = 1,2,3$ [98]:

$$\left(u_1^{\mathrm{d},1}, u_2^{\mathrm{d},1}, u_3^{\mathrm{d},1}\right) = \arg \min_{(u_1,u_2,u_3) \in \Omega_1 \times \Omega_2 \times \Omega_3} \jmath_1(u_1,u_2,u_3), \tag{2.23}$$

$$\left[u_2^{\mathrm{d},2}(\gamma_1), u_3^{\mathrm{d},2}(\gamma_1)\right] = \arg \min_{(u_2,u_3) \in \Omega_2 \times \Omega_3} \jmath_2(u_2,u_3;\gamma_1), \text{ and} \tag{2.24}$$

$$u_3^{\mathrm{d},3}(\gamma_2;\gamma_1) = \arg \min_{u_3 \in \Omega_3} \jmath_3(u_3;\gamma_2,\gamma_1), \tag{2.25}$$

where $u_1 = \gamma_1(u_2,u_3), u_2 = \gamma_2(u_3)$.

However, to the author's best knowledge, cases with multiple leaders and multiple followers simultaneously where the Nash concept is not adopted, have not been looked into in the available literature.

2.5 Areas of Application

Hierarchical control can be roughly divided in cases in which a natural division in multiple levels exists and those where a hierarchical structure is adopted as an alternative to strictly distributed optimization in order to facilitate the problem solving [170]. The latter approach arises in large-scale control problems where information is not automatically available to all controllers and difficult or costly to communicate. Hence, the overall problem is decomposed where the higher-level controller acts as a coordinator. However, in Stackelberg games, usually applications that embody a natural hierarchy are considered.

Looking into more specific applications of the reverse Stackelberg game, next to contracting and pricing problems as studied in the area of incentives and management science [57] (see also Section 2.2.2), the following application areas have been considered:

- ***Network pricing:***

  In [172] a reverse Stackelberg game is used to model a situation in which an Internet service provider is considered as a leader that sets a price for the bandwidth used by followers, where the price is dependent on the actual bandwidth used. Both complete and incomplete information on the type of users as denoted by a constant parameter $w \in \mathbb{R}_0^+$ in the users' cost functions is studied. For both cases, ε-optimal nonlinear strategies are obtained, with ε an arbitrarily small constant. Such strategy will lead to an equilibrium that is arbitrarily close to the leader's optimal solution. The ε-optimal solutions are obtained by making a small deviation in the leader function from the desired but unattainable optimum, i.e., by substituting $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ by $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}} - \varepsilon)$, which becomes the new equilibrium point.

  For the rather specific instance of the problem with $u_{\mathrm{L}}, u_{\mathrm{F}}$ scalar and $J_{\mathrm{F}} = w\log(1+u_{\mathrm{F}}) - \frac{1}{1-u_{\mathrm{F}}} - u_{\mathrm{L}}$, it is shown in [172] that such approximate solutions can always be found, e.g., for a function $\gamma_{\mathrm{L}}(u_{\mathrm{F}}) = a_1 u_{\mathrm{F}} + a_2 u_{\mathrm{F}}^2$, $a_1 \in \mathbb{R}, a_2 \in \mathbb{R}$ constants.

- ***Road tolling*:**

  In [180] a dynamic toll design problem is considered on a three-link highway network where one of the links is not subject to a toll. In order to minimize the total travel time and thus to reduce congestion, a toll proportional to the traffic flow is proposed. It is shown that this Stackelberg game formulation indeed yields a better performance for the road authority compared to when adopting a constant or time-varying toll. However, the three links have the same origin and destination and no overlap of roads applies; hence, the route choice of the homogeneous traffic considered is solely based on the tolls of each independent link. In Chapter 5 the use of reverse Stackelberg games in traffic control settings will be elaborated upon.

- ***Electricity pricing:***

  In [134] reverse Stackelberg games are used to model the pricing of electricity consumption in a stochastic peak load problem where, instead of adopting an open-loop formulation, a closed-loop 'load adaptive pricing' policy is adopted where the leader proposes an electricity price p consisting of a fixed charge c and a unit price c^{var} that is based on the varying demand d that represents the follower's decision, i.e., $p := u_{\mathrm{L}} = \gamma_{\mathrm{L}}(u_{\mathrm{F}}) = c^{\mathrm{var}} u_{\mathrm{F}} + c$. This game is played for n time-periods or cycles of a fixed duration, each consisting of an m-stage game with a price associated with each stage. The values of c_{nm}^{var} and c_{nm} are computed as functions of the values of d, p, and a randomly distributed state variable of previous stages in the current cycle, or in the previous cycle. When the load adaptive pricing strategy is adopted within a dynamic infinite-cycle

setting, stability of the system to small disturbances in the decision variables u_L and u_F as well as the convergence of the coefficients that determine the price p is proven.

It should be noted that a large number of publications is available that consider a standard hierarchical game, i.e., that adopt the original Stackelberg structure. An overview of applications of Stackelberg (differential) games in supply chain management and marketing settings can be found in [91]. Also the areas enumerated above of road pricing [119], electricity markets [100, 178], and more general network pricing [188] are considered in an original Stackelberg setting.

Overall, while the use of Stackelberg games in large-scale control systems is proposed [16, 44], not many results actually consider such problem statements; applications are rather adopted as a means to illustrate the Stackelberg concept. Nonetheless, as indicated in [170] hierarchical, multilevel methods can indeed pose a viable approach to structure large-scale control problems. We therefore see potential in applying the reverse Stackelberg game also in those areas to which the original Stackelberg game is applied, while real-life application remains a challenge for both the Stackelberg and the reverse Stackelberg framework.

2.6 Open Problems

In the previous section an overview has been presented to integrate results on reverse Stackelberg games from its origin in the 1970s with more recent contributions to the field. While the reverse Stackelberg game can be adapted with respect to several aspects as depicted in Fig. 2.5, which allows it to be flexible in various settings, there are still many unresolved problems. In general, these problems stem from the fact that the game is difficult to solve analytically (especially if asymmetric and imperfect information applies) as well as numerically. In the following, open issues are enumerated to emphasize the potential of the reverse Stackelberg for further research and application within the field of control.

- ***Convexity assumptions:***

  There is a large body of literature available on dynamic reverse Stackelberg games with linear state equations and quadratic cost functions, for which a leader function of an affine structure has been proven to solve the game to optimality according to Definition 2.9 [34, 56, 139, 209]. Although this result is said to be applicable to a 'sufficiently large' number of cases [209], a (strictly) convex and differentiable cost function and linear constraints will not generally be found in possible real-life applications. A relaxation of these assumptions should therefore be made.

- ***Computational tractability and optimality bounds:***

  Another opportunity may be found in the lack of focus on numerical tractability of reverse Stackelberg problems, which is important especially in the context of real-life control or optimization applications, given that large-scale control

or optimization problems are often mentioned as a reason for studying Stackelberg strategies [16, 44]. Especially in the largely untouched case of nonconvex cost functions, approximate or suboptimal solutions may be required. Moreover, there is little focus on establishing bounds on the leader objective function value associated with suboptimal solutions; similarly, the associated objective function value for the follower is not taken into account, whereas bounds on the performance of lower-level controllers would be a relevant addition.

- ***Robustness:***

  Even in case a simple (affine) leader function suffices for attaining the leader's desired equilibrium, it would be interesting to investigate how sensitive a given leader function is to changes in the game parameters. As mentioned in Section 2.4.5, in [29] nonuniqueness of the optimal leader function is used to consider the minimization of the deviation from an estimated nominal parameter value as a secondary objective. Similarly, robustness to unintentional deviations of the follower to the optimal response could be considered, as discussed in [184] for the closed-loop Stackelberg game. There, the problem is addressed by adopting discontinuous state-dependent closed-loop strategies that are, however, developed for very particular numerical problem instances; also, it is not clear how much worse off the leader is by punishing the follower from a deviation from his optimal response. Further, the stability of (ε-optimal) Stackelberg solutions has been considered in [127]; however, no similar results can currently be found in the context of reverse Stackelberg games.

- ***Desired leader solution*:**

  Generally, it is assumed that the leader strives after obtaining a unique global optimum. Irrespective of whether this solution can be obtained by a particular leader function, cases could be investigated in which the leader strives after a broader set of possible solutions. This becomes even more relevant in case the leader solves a multiobjective problem and thus has to find a trade-off between several optima. Similarly, instead of cardinal solutions that follow from optimization of a real-valued objective function, discrete orderings of preferred solutions could be considered. Such ordinal solutions for regular Stackelberg games have also been advocated in [45].

- ***Stability:***

  A stability analysis has been performed for the continuous-time [168] and later for the discrete-time [125] LQ Stackelberg game with no-memory state feedback. In case of time-invariant weighting matrices that occur in the cost functions and state equation of the Stackelberg game, sufficient conditions are developed for a leader function that leads to an asymptotically stable system for the infinite-time game. However, it is not guaranteed that such a leader function exists, nor is a direct approach available for the computation of this function due to the complexity of the problem. Moreover, similar guarantees on the system stability for the dynamic reverse Stackelberg game have not yet been investigated.

- ***Leader-follower role:***

  In most cases the positions of leader and follower are taken to be known in advance and fixed. While recent research in Stackelberg games allocates more flexibility to this role [12, 17, 152, 173], no similar results can currently be found for the reverse Stackelberg game, to which this flexible role should also be applicable.

  In the original Stackelberg framework, the leader-follower role has been analyzed in [12]. There, it is shown that if two players act sequentially rather than simultaneously as in a Nash game, both players may obtain better, but not worse, results. Note that this provides another benefit of adopting a hierarchical design for decision making as opposed to the alternative solution concept of the Nash equilibrium. Further, in [173] the leader role is said to be preferred to the follower role for players in a closed-loop Stackelberg game, which is in general, however, not the case if feedback information applies.

  In [152] a discrete-time dynamic Stackelberg game is considered where players in one of two groups constituting a leader or follower take the position of the leader in turns. There, players are allocated to one of the two fixed groups that take a switching position of leader respectively follower at each stage of the game. Finally, in [17] an open-loop differential Stackelberg game is considered with mixed leadership, meaning that a player can be both leader and follower at the same time, depending on the subset of control variables that are associated with a particular role *a priori*. After announcement of the leader decisions, the optimal follower responses are determined simultaneously, i.e., here, the Nash equilibrium concept is adopted.

- ***Nonlinear leader functions:***

  Whereas several options are given for possible leader function structures in relation to incentive controllability [98], nonlinear function structures are hardly considered in the reverse Stackelberg game, except from those that occur in specific numerical examples [153, 154, 172, 180]. In [184], discontinuous state-dependent closed-loop Stackelberg strategies are considered. In case the follower played suboptimally during the previous stage, a leader strategy is adopted that differs from the default affine leader function in the sense that a leader strategy is adopted under which the follower performs worse. However, it is not analyzed how much the leader's performance is reduced in case of such a 'punishment' strategy that follows irrational follower behavior. Moreover, although it is considered an incentive strategy, the leader strategies in [184] are not directly dependent on the follower's decisions as is required for the reverse Stackelberg game.

- ***Approximate solutions:***

  Most research is focused on achieving the leader's desired (global) optimum, for which often those cases are considered in which an optimal affine leader function can be derived. On the other hand, the case in which this is not possible has not much been studied. In particular, suboptimal or ε-optimal

strategies could be investigated, as is done for specific numerical examples in [153, 154].

- ***Applications:***

  As has been mentioned in Section 2.5 above, several applications have been considered for the original Stackelberg game. However, similar to the flexible leader-follower role, it should be no problem to also extend the reverse game to these problems. Moreover, as has been mentioned in Section 2.2, when adopting a – more general – reverse Stackelberg approach, the leader player may be able to achieve a better performance than in the original case.

- ***Constraints:***

  Most results on reverse Stackelberg or incentive strategies do not take into account constraints on decision and state spaces, except for the origin and nonnegativity constraints $\gamma_L(\cdot) \geq 0$ and $\gamma_L(0) = 0$ [153]. As this prevents from considering general applications, more general constraints should therefore be considered as well. In Chapter 3 it is explained how considering a constrained leader and follower decision space influences the set of feasible solutions as composed for the unconstrained case.

In the following chapters, first the convexity and differentiability assumption will be relaxed in conditions on the existence and in a characterization of an optimal affine leader function as will be discussed in Chapter 3, where also a constrained decision space will be considered. Solution methods for deriving optimal nonlinear leader functions will subsequently be proposed in Chapter 4 and applications will be considered in Chapter 5.

Chapter 3

On Optimal Affine Leader Functions

In this chapter, a first step is made towards developing a systematic solution approach that on the one hand eases the solution process of the generally complex single-leader single-follower reverse Stackelberg game, and that at the same time deals with a game setting in which assumptions that could restrict the application to certain problem settings are relaxed as much as possible. In order to solve the reverse Stackelberg game, an indirect approach will be adopted in which a specific structure of the leader function is considered, given a desired equilibrium that the leader strives to achieve. In particular, in this chapter a leader function of the affine type is analyzed in order to procure a systematic approach for solving the game to optimality. To this end, necessary and sufficient existence conditions for an optimal affine leader function are developed. Compared to earlier results reported in the literature, the differentiability requirement of the follower objective functional is relaxed and locally strict convexity of the follower's sublevel set at the desired reverse Stackelberg equilibrium is replaced by the more general property of an exposed point. Moreover, a full characterization of the set of affine leader functions that solve the game to optimality is derived. The parametrized characterization of such a set facilitates further optimization, e.g., when considering the sensitivity to deviations from the optimal follower response, as will be illustrated by a simple example. Moreover, the characterization can be used to verify the existence of an optimal affine leader function in a constrained decision space.

The research discussed in this chapter is based on [83], supported by the results presented in [77, 78].

3.1 Introduction

As has been explained in the literature survey of Chapter 2, much research on the reverse Stackelberg game is tailored to the specific case of a quadratic, (strictly) convex and differentiable cost functional and, in a dynamic game framework, of a linear state update equation. Since the aim of the work presented in this thesis is to expand the available solution methods of the reverse Stackelberg game to form a basis applicable to general problem structures, a first step is to generalize the analysis to

cover a broader class of problems including problem instances in which nonconvex sublevel sets for the follower objective function as well as nondifferentiable objective functions apply.

For this purpose, first the reachability of the leader's desired equilibrium is analyzed for a leader function belonging to the affine class of leader function structures, in the form of necessary and sufficient existence conditions. This is achieved by using a geometric approach similar to the approach adopted in [209], where sufficient conditions were developed for the existence of an optimal affine leader function in the special case of a convex and differentiable leader and follower objective function.

In addition, we focus on the actual derivation of the set of optimal affine leader functions by characterizing the parameter values of possible optimal leader functions. Here, the computational complexity of the proposed solution approach is considered and several illustrative examples are provided. The resulting set of possible solutions can be used to consider secondary objectives, e.g., related to a sensitivity analysis [29, 30], which is especially important if the full rationality assumption of the players is relaxed or dynamic, stochastic settings with uncertainty and noise are considered. Furthermore, the characterization can be used to verify the existence of an optimal affine leader function in a constrained decision space, in which case the derivation of existence conditions is a challenging task.

The remainder of this chapter is structured as follows. Section 3.2 includes a clarification of notation and assumptions adopted in this chapter. In Section 3.3, necessary and sufficient conditions are proven for the existence of an optimal affine leader function for the unconstrained case, considering separately the case of a scalar leader input and the cases in which the desired equilibrium is either an interior or a boundary point of the nonconvex and nonsmooth sublevel set. Then, in Section 3.4 we derive the full set of affine leader functions that solve the reverse Stackelberg game to optimality in the unconstrained case. These results are analyzed under the presence of constraints in Section 3.5 and simple examples are provided to show the relevance of the full characterization. A secondary objective like the minimization of the sensitivity of a leader function is considered as a further motivation of the characterization in Section 3.6. Conclusions are finally presented in Section 3.7.

3.2 Preliminaries

In the current chapter the indirect reverse Stackelberg game variant is adopted, i.e., an optimal leader function is aimed for given the leader's desired equilibrium point, as described in Chapter 2 (Section 2.1.4). In the section below we first elaborate on the particular affine parametrization of the leader function that is considered in this chapter.

3.2.1 Affine Incentive Controllability

In order to reduce the complexity of the general reverse Stackelberg game and to create a systematic approach towards solving the general game, in this chapter we focus on the affine structure of the leader function, i.e., we assume the set Γ_L to include only functions of the form

$$u_L := \gamma_L(u_F) = u_L^d + B(u_F - u_F^d), \tag{3.1}$$

where B denotes a linear operator mapping $\Omega_F \to \Omega_L$, represented by an $n_L \times n_F$ matrix in the finite-dimensional case we will consider according to assumption A.3 in Section 3.2.3 below. The property of a particular desired leader equilibrium to be feasible for an instance of the reverse Stackelberg game is known as incentive controllability in the literature [98, 209]. It will now be analyzed under what conditions an optimal affine leader function exists, meaning that the leader is able to induce the follower to choose the desired input u_F^d and thus reach her desired equilibrium. Here, the follower's sublevel set will play a crucial part; for clarity we recall its definition according to 2.8:

$$\Lambda_d := \left\{ (u_L, u_F) \in \Omega_L \times \Omega_F \,\middle|\, J_F(u_L, u_F) \leq J_F(u_L^d, u_F^d) \right\}. \tag{3.2}$$

From now on, we denote by $\mathcal{A}_L$ the set of affine relations through (u_L^d, u_F^d) that are defined as sets of dimension n_F in $\Omega_L \times \Omega_F$ and such that for $\alpha_L \in \mathcal{A}_L$, $\alpha_L \cap \Lambda_d = \{(u_L^d, u_F^d)\}$. Note that this construction is necessary in order to be able to work with the function $\gamma_L : \Omega_F \to \Omega_L$ as a set of points $\{(u_L, u_F) | u_F \in \Omega_F, u_L = \gamma_L(u_F)\}$:

Remark 3.1 Note that in this chapter the leader function γ_L is defined as a mapping $\Omega_F \to \Omega_L$ that can also be represented by the set of points $\{(u_L, u_F) | u_F \in \Omega_F, u_L = \gamma_L(u_F)\}$. In the following, both the mapping and the set representation of γ_L are adopted depending on the context. ◇

For $\alpha_L \in \mathcal{A}_L$, with $\alpha_L(\Omega_L) = \Omega_F$, a candidate leader function can now be characterized by $\gamma_L := (\alpha_L)^{-1}$. Finally, for a set X, let $\mathcal{A}_L^X := \{\alpha_L \in \mathcal{A}_L | \alpha_L \subseteq X\}$.

3.2.2 Notation and Definitions

The analysis of this chapter mostly relies on concepts from convex analysis and geometry, such as hyperplanes and strictly convex functions and sets (see e.g., [10, 164]). In addition, the following notation and definitions are adopted:

- By $f(\tilde{X})$ we denote the image of a function $f : X \to Y$ for a subset $\tilde{X} \subseteq X$, where the domain is denoted $X := \mathrm{dom}(f)$.
- Let $S = X \times Y$. The projection of a set $P \subseteq S$ onto the space X or Y is denoted by $\mathrm{proj}_{X \times Y \to X}(P)$, $\mathrm{proj}_{X \times Y \to Y}(P)$, which for ease of notation we shorten to $\mathrm{proj}_X(P)$ and $\mathrm{proj}_Y(P)$ respectively. For an element $p \in P, p = \left[x^T, y^T\right]^T$, with $x \in X, y \in Y$, we have $\mathrm{proj}_X(p) := x$ and similarly, $\mathrm{proj}_Y(p) := y$. Finally, $\mathrm{proj}_X(P) := \{\mathrm{proj}_X(p) | p \in P\}$ and $\mathrm{proj}_Y(P) := \{\mathrm{proj}_Y(p) | p \in P\}$.

- By $\{0\}^{n_L} \times \Omega_F$ we denote the $(n_L + n_F)$-dimensional decision space in which the leader components are taken to be zero.

While different definitions are adopted for the concepts of polyhedra and polytopes, we refer to [171]:

Definition 3.2 Polyhedron, polytope
A *(convex) polyhedron* in the Euclidean space $\mathbb{R}^n$ is the solution set of a finite system of linear inequalities, defined by the intersection of finitely many affine halfspaces. These halfspaces can be closed or open (i.e., $\{x \in \mathbb{R}^n : a^T x \leq b\}$ or $\{x \in \mathbb{R}^n : a^T x < b\}, a \in \mathbb{R}^n, b \in \mathbb{R}$ respectively.) A set P is a *polytope* if and only if P is a bounded polyhedron. ◇

Definition 3.3 Convex hull, (strict) convexity
The *convex hull* of a set X of vectors is the smallest convex set containing X:

$$\text{conv}(X) := \{\lambda_1 x_1 + \cdots + \lambda_t x_t | t \geq 1; x_1, \ldots, x_t \in X; \lambda_1, \ldots, \lambda_t \geq 0; \lambda_1 + \cdots + \lambda_t = 1\}.$$

A set C is *convex* if it satisfies: if $x, y \in C$ and $0 \leq \lambda \leq 1$, then $\lambda x + (1-\lambda)y \in C$. Further, if $S \subseteq \mathbb{R}^n$, a functional $f : S \to \mathbb{R}$ is *convex* if S is convex and if $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$ whenever $x, y \in S$ and $0 \leq \lambda \leq 1$. Such a function is *strictly convex* if the inequality strictly holds for $0 < \lambda < 1$.

A set X is *strictly convex* if there do not exist two different points on its boundary $\text{bd}(X)$ for which the line segment connecting the points is part of the boundary:

$$\forall y, z \in \text{bd}(X), y \neq z, \forall \lambda \in (0,1) : (1-\lambda)y + \lambda z \in \text{int}(X),$$

where $\text{int}(X)$ denotes the interior of X, i.e., $\text{int}(X) := \{x | \exists \varepsilon > 0, x + \varepsilon B \subsetneq X\}$, with B the Euclidean unit ball in $\mathbb{R}^n$: $B := \{x \,|\, |x|_2 \leq 1\}$ where $|x|_2 := \sqrt{x^T x}$ denotes the Euclidean norm of x.

A function f is *locally strictly convex* at a point $x \in \text{int}(\text{dom}(f))$ if there is a neighborhood of x such that the restriction of f to that neighborhood is strictly convex. A similar definition applies to a set X. ◇

Definition 3.4 Affine subspace
A set X is an *affine subspace* if $\alpha y + (1-\alpha)z \in X$ for all $y, z \in X, \alpha \in \mathbb{R}$. ◇

Definition 3.5 (Polyhedral) cone
A *(convex) cone* is a nonempty subset C of points in Euclidean space that satisfies: if $x, y \in C$ and $\lambda, \mu \in \mathbb{R}^+$, then $\lambda x + \mu y \in C$. A *polyhedral cone* is the set of solutions of a finite system of linear inequalities, i.e., $C := \{x | Ax \leq 0\}$ for some matrix A. The *cone generated* by the set of vectors $x_1, \ldots, x_m$ is the smallest convex cone constaining the set:

$$\text{cone}(x_1, \ldots, x_m) := \{\lambda_1 x_1 + \cdots + \lambda_t x_t | \lambda_1, \ldots, \lambda_t \geq 0\}.$$

A convex cone is polyhedral if and only if it is finitely generated. ◇

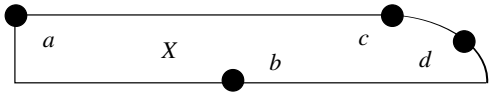


Figure 3.1: *The points a and d of this closed convex set X are exposed; points b and c are not.*

Definition 3.6 Supporting hyperplane
Let the polyhedron $\mathcal{P}$ be defined by $\mathcal{P} := \{x \in \mathbb{R}^n | Ax \geq b\}$. If $c \in \mathbb{R}^n$ with $c \neq 0$ and $\delta = \max\{c^{\mathrm{T}}x | Ax \geq b\}$, then the hyperplane $\{x | c^{\mathrm{T}}x = \delta\}$ is called a *supporting hyperplane* of $\mathcal{P}$, denoted by Π_X. In particular, we denote by $\Pi_X(x)$ a supporting hyperplane to the set X at the point $x \in X$.

The hyperplane is said to be *strictly supporting* if $c^{\mathrm{T}}x = \delta$ for one single point $x \in X$. ◇

Definition 3.7 Exposed point
An *exposed point* of a convex set X is defined as a point in its closure $\bar{X} := \bigcap\{X + \varepsilon B | \varepsilon > 0\}$ – with B the Euclidean unit ball as defined within Definition 3.3 – that intersects with a strictly supporting hyperplane to X.

Similarly, a point $\tilde{x}$ in the closure of a nonconvex set $\tilde{X}$ is an exposed point if there exists a neighborhood of $\tilde{x}$, $\mathcal{N}(\tilde{x})$, such that $\tilde{x}$ intersects with a strictly supporting hyperplane to $\mathcal{N}(\tilde{x})$. ◇

Definition 3.8 Generalized gradient, generalized normal
The *generalized gradient* denoted $\partial f(x)$ of a locally Lipschitz continuous function $f : \mathbb{R}^n \to \mathbb{R}$ at x is defined as follows:

$$\partial f(x) := \mathrm{conv}(\{\lim_{m\to\infty} \nabla f(x_m) | x_m \to x, x_m \in \mathrm{dom}(f) \setminus \Omega_f\}),$$

with Ω_f being the set of points where f is nondifferentiable [38]. By $\mathcal{V}(X,x)$ we denote the *generalized normal* to the set X at the point $x \in \bar{X}$, defined as the set of normal vectors to the possible tangent hyperspaces to X at x. Thus, in case Λ_{d} is smooth at $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$, $\mathcal{V}\left(\Lambda_{\mathrm{d}}, (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\right) = \nabla \mathcal{J}_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$. ◇

3.2.3 Assumptions

The following assumptions are adopted in the analysis of this chapter:

[A.1] Let $\Omega_{\mathrm{L}}, \Omega_{\mathrm{F}}$ be convex sets.

[A.2] Let Λ_{d} be a connected set.

[A.3] Let $n_{\mathrm{L}}, n_{\mathrm{F}}$ be finite.

[A.4] Let $\Lambda_{\mathrm{d}} \neq \{(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\}$.

The first assumption A.1 is taken from the literature on Stackelberg games, e.g., [15, 209] and is a necessary condition for convexity of $\jmath_{\mathrm{F}}(\cdot)$ and Λ_{d}. Assumption A.2 is a less restrictive case of taking $\jmath_{\mathrm{F}}(\cdot)$ and therefore also Λ_{d} to be strictly convex, as was done in [209]. Note that A.2 is automatically satisfied if it holds that $\jmath_{\mathrm{F}}(\cdot)$ is a convex or quasiconvex function. Further note that we do not require $\jmath_{\mathrm{F}}(\cdot)$ to be continuous. In the special case that $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \in \mathrm{bd}(\mathrm{conv}(\Lambda_{\mathrm{d}}))$ assumption A.2 can even be omitted altogether if Λ_{d} is substituted by $\mathrm{conv}(\Lambda_{\mathrm{d}})$ in the analysis below. Assumption A.3 is also accepted in many control applications [7, 64] and moreover it is necessary in order to be able to adopt the concept of a supporting hyperplane. Finally, the special case excluded by assumption A.4 presents the trivial situation in which $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ is automatically optimal for the follower as well.

3.3 Necessary and Sufficient Existence Conditions

In the current section, basic necessary and sufficient conditions for the existence of an optimal affine leader function are proposed for the most general case in which the sublevel set Λ_{d} is allowed to be nonconvex and nonsmooth, in an unconstrained decision space. These existence conditions form the basis for a characterization of the set of optimal affine solutions provided in Section 3.4. Here it should be noted that when relaxing the strict convexity of the follower objective function from the original results in [209], the desired leader equilibrium is not automatically a boundary point of the sublevel set. Exclusion of this case prevents the current theory that adopts such a setting [34, 56, 125, 139, 167, 184, 209] from being generally applicable.

For the sake of clarity and completeness, an overview of the results presented in the following sections can be found in the diagram below.

Table 3.1: *Overview of Results.*

Existence conditions – Section 3.3		
Basis	Lem. 3.9 Lem. 3.10 Lem. 3.11	Supporting hyperplane Lemmata
$n_{\mathrm{L}} = 1$ $n_{\mathrm{L}} > 1$	Prop. 3.12 Prop. 3.13 Prop. 3.14	 $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ exposed point $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \in \mathrm{conv}\,(\Lambda_{\mathrm{d}})$
Special cases	Rem. 3.15 Rem. 3.16	$(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ not exposed $n_{\mathrm{L}} = n_{\mathrm{F}} = 1$
Characterization – Section 3.4		
Basis	Lem. 3.18 Lem. 3.19	Conditions on R $R_{\mathrm{F}} = I_{n_{\mathrm{F}}}$
	Prop. 3.20 Prop. 3.21	Characterization of γ_{L} Characterization of $\mathcal{R}_{\mathrm{L}}$

In this section first some lemmata are provided that form basic elements for the proofs of the propositions in Section 3.3.2 and 3.3.3.

3.3.1 Supporting Hyperplane Lemmata

Lemma 3.9 follows straightforwardly from the supporting hyperplane theorem (e.g., Theorem 11.6 in [164]) and Definition 3.6 of a strictly supporting hyperplane.

Lemma 3.9 *Assume the set Λ_d defined through (3.2) to be convex. Let $\Omega_L = \mathbb{R}^{n_L}, \Omega_F = \mathbb{R}^{n_F}$ and let $\alpha_L \in \mathcal{A}_L$ be any affine function through (u_L^d, u_F^d) such that $\alpha_L \cap \Lambda_d = \{(u_L^d, u_F^d)\}$. Then α_L lies on a supporting hyperplane to Λ_d at (u_L^d, u_F^d).*

Lemma 3.10 *Let Λ_d be defined through (3.2). A supporting hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ exists at (u_L^d, u_F^d) if and only if $(u_L^d, u_F^d) \notin \text{int}(\text{conv}(\Lambda_d))$. Further, for an exposed point (u_L^d, u_F^d) of $\text{conv}(\Lambda_d)$, $\Pi_{\Lambda_d}(u_L^d, u_F^d) \cap \Lambda_d = \{(u_L^d, u_F^d)\}$.*

Proof: By definition of a convex hull, a supporting hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ exists if and only if there exists a supporting hyperplane $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$ to $\text{conv}(\Lambda_d)$ at (u_L^d, u_F^d). Further, a supporting hyperplane to $\text{conv}(\Lambda_d)$ exists at (u_L^d, u_F^d) if and only if (u_L^d, u_F^d) is a boundary point of $\text{conv}(\Lambda_d)$ and thus also of Λ_d ([164, Theorem 11.6]). Clearly, an exposed point of $\text{conv}(\Lambda_d)$ is such a boundary point. For the intersection of $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ with Λ_d solely to occur in the point (u_L^d, u_F^d), it is required that (u_L^d, u_F^d) is an exposed point of $\text{conv}(\Lambda_d)$. (Note that it is therefore sufficient for $\text{conv}(\Lambda_d)$ to be locally strictly convex at (u_L^d, u_F^d).) □

Lemma 3.11 *Assume that there exists a strictly supporting hyperplane $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$: $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d) \cap \Lambda_d = \{(u_L^d, u_F^d)\}$, with Λ_d defined by (3.2). Then an affine function $\alpha_L^{\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)} \in \mathcal{A}_L$ coincides with $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$ if and only if u_L is scalar ($n_L = 1$).*

Proof: Only in case of a scalar u_L the dimension of a hyperplane Π_{Λ_d}, i.e., $(n_L + n_F) - 1$, equals the number n_F of independent variables of an affine leader function $\alpha_L \in \mathcal{A}_L$, with $u_F \in \Omega_F \subseteq \mathbb{R}^{n_F}$. If there exists a strictly supporting hyperplane $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$, it follows that this plane coincides with $\alpha_L^{\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)}$. □

We can now proceed by stating the main propositions.

3.3.2 Case $n_L = 1$

We first need to consider the special case of $n_L = 1$; no optimal affine leader function then exists if, in addition, (u_L^d, u_F^d) is not an exposed point of $\text{conv}(\Lambda_d)$.

Proposition 3.12 *Let $\Omega_L = \mathbb{R}^{n_L}, \Omega_F = \mathbb{R}^{n_F}$ and assume that $n_L = 1$. Then the desired equilibrium (u_L^d, u_F^d) can be reached under an affine leader function $\gamma_L : \Omega_F \to \Omega_L$ if and only if it both holds that (u_L^d, u_F^d) is an exposed point of $\text{conv}(\Lambda_d)$ and that $\text{proj}_{\Omega_L}\left(\mathcal{V}\left(\text{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) \neq \{0\}$.*

Proof: First note that since $n_L = 1$, if and only if a strictly supporting hyperplane $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$ exists, it coincides with an affine $\alpha_L \in \mathcal{A}_L$, as was shown in Lemma

3.11. Note that a plane $\Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)$ is strictly supporting if and only if (u_L^d, u_F^d) is an exposed point of $\mathrm{conv}(\Lambda_d)$ – implying that $(u_L^d, u_F^d) \notin \mathrm{int}(\mathrm{conv}(\Lambda_d))$.

It remains to be shown that in addition to (u_L^d, u_F^d) being exposed, in order for $\alpha_L^{\Pi_{\Lambda_d}}(\Omega_L) = \Omega_F$ to hold, it is necessary and sufficient that there exists a vector $\nu \in \mathcal{V}\left(\Lambda_d, (u_L^d, u_F^d)\right) : \mathrm{proj}_{\Omega_L}(\nu) \neq \{0\}$, from which it follows that the projection $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right)$ should not include only the zero vector. In that case, no explicit description of a leader function exists. This sufficiency and necessity is proven next.

($\Rightarrow$) By contraposition: Suppose that $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) = \{0\}$. Then there exists a tangent plane $\Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)$ with a normal vector ν for which it holds that $\mathrm{proj}_{\Omega_L}(\nu) = \{0\}$. It follows that this normal vector defining the hyperplane $\Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)$ is parallel to the decision space Ω_F, i.e., this hyperplane is orthogonal to $\{0\}^{n_L} \times \Omega_F$. Therefore, $\mathrm{proj}_{\Omega_F}\left(\Pi_{\Lambda_d}(u_L^d, u_F^d)\right) \subsetneq \Omega_F$ and $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ will not include any elements $(u_L, u_F) \in \Omega_L \times (\Omega_F \setminus \{u_F^d\})$, which implies that $\alpha_L^{\Pi_{\Lambda_d}}(\Omega_L) \subsetneq \Omega_F$.

($\Leftarrow$) If $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) \neq \{0\}$, there exists a normal vector $\nu \in \mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)$ defining a hyperplane $\Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)$ that is not orthogonal to the decision space Ω_L. It follows that the hyperplane is not orthogonal to $\{0\}^{n_L} \times \Omega_F$, i.e., $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) = \Omega_F$.

Hence,

$$\forall u_F \in \Omega_F \, \exists u_L \in \Omega_L : (u_L, u_F) \in \Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d).$$

Thus, there exists an affine $\alpha_L^{\Pi_{\Lambda_d}}$ such that $\alpha_L^{\Pi_{\Lambda_d}}(\Omega_L) = \Omega_F$.

Under the use of a leader function $\gamma_L := \left(\alpha_L^{\Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)}\right)^{-1}$, by definition of the level set (3.2), the minimum of $\jmath_F(\cdot)$ will be obtained at (u_L^d, u_F^d), concluding the proof. □

An example of a case in which Λ_d is nonsmooth and no affine $\gamma_L(\cdot)$ exists is depicted in Fig. 3.2(a) : here, $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\Lambda_d, (u_L^d, u_F^d)\right)\right) = \{0\}$.

3.3.3 Case $n_L > 1$

Propositions 3.13 and 3.14 below consider respectively the case in which the desired leader equilibrium (u_L^d, u_F^d) is an exposed point of $\mathrm{conv}(\Lambda_d)$ or the case in which it is in the interior of $\mathrm{conv}(\Lambda_d)$ for $n_L \geq 1$ and $n_L > 1$.

Proposition 3.13 *Let $n_L \geq 1$ and assume that (u_L^d, u_F^d) is an exposed point of* $\mathrm{conv}(\Lambda_d)$. *Allow Λ_d to be nonsmooth at (u_L^d, u_F^d) and assume that $\Omega_L = \mathbb{R}^{n_L}, \Omega_F = \mathbb{R}^{n_F}$. Then the desired equilibrium (u_L^d, u_F^d) can be reached under an affine $\gamma_L : \Omega_F \to \Omega_L$ if and only if* $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) \neq \{0\}$.

Proof : The necessity and sufficiency of the condition $\mathrm{proj}_{\Omega_L}\left(\mathcal{V}\left(\mathrm{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) \neq \{0\}$ is proven in the previous Proposition 3.12 for $\gamma_L := \Pi_{\mathrm{conv}(\Lambda_d)}(u_L^d, u_F^d)$. In case

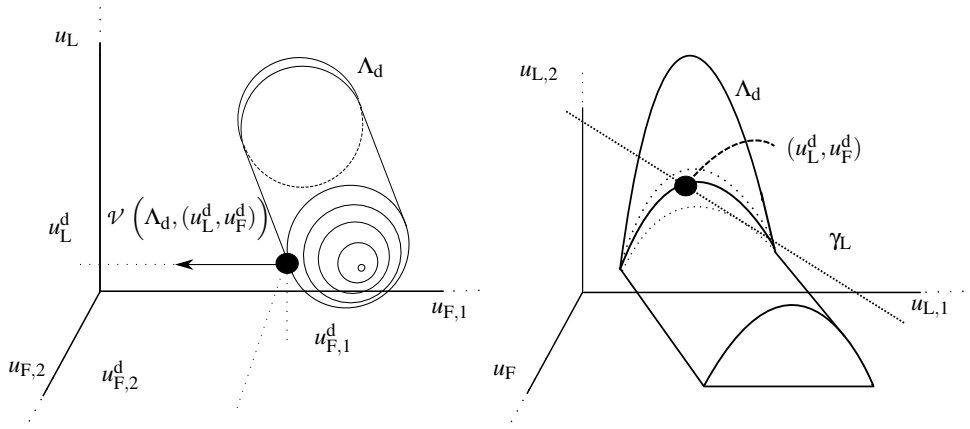


Figure 3.2: *(a) Example of a convex set Λ_{d} that is nonsmooth at $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$, for which no optimal affine leader function exists. (b) Example of an optimal affine leader function not lying on a supporting hyperplane $\Pi_{\Lambda_{\mathrm{d}}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ for $n_{\mathrm{L}}>1$, $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\in\mathrm{int}(\mathrm{conv}(\Lambda_{\mathrm{d}}))$.*

$n_L > 1$, it holds that $\gamma_{\mathrm{L}} \subsetneq \Pi_{\mathrm{conv}(\Lambda_{\mathrm{d}})}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$. Thus, there exists at least one leader function on the given hyperplane that satisfies the condition that $\alpha_{\mathrm{L}}^{\Pi_{\Lambda_{\mathrm{d}}}}(\Omega_{\mathrm{L}}) = \Omega_{\mathrm{F}}$. □

Finally, for $n_{\mathrm{L}} > 1$, requiring $\alpha_{\mathrm{L}} \in \mathcal{A}_{\mathrm{L}}$ to lie on a supporting hyperplane separating the full $(n_{\mathrm{F}}+n_{\mathrm{L}})$-dimensional decision space into subspaces is generally too restrictive for the existence of an optimal affine leader function. This applies to e.g., the general nonconvex case under a constrained decision space, and to the the case in which $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\in\mathrm{int}(\mathrm{conv}(\Lambda_{\mathrm{d}}))$ as depicted in Fig. 3.2(b). Hence, instead the tangent hyperplane concept is adopted in Proposition 3.14 below.

Proposition 3.14 *Let $n_L > 1$ and assume that $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\in\mathrm{int}(\mathrm{conv}(\Lambda_{\mathrm{d}}))$. Allow Λ_{d} to be nonsmooth at $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ and assume that $\Omega_{\mathrm{L}}=\mathbb{R}^{n_{\mathrm{L}}},\Omega_{\mathrm{F}}=\mathbb{R}^{n_{\mathrm{F}}}$. Then the desired equilibrium $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ can be reached under an affine $\gamma_{\mathrm{L}}:\Omega_{\mathrm{F}}\to\Omega_{\mathrm{L}}$ if and only if there exists an n_{F}-dimensional tangent, affine subspace $\Pi_{\mathrm{d}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ to Λ_{d} at $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ such that*

$$\Pi_{\mathrm{d}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\cap\Lambda_{\mathrm{d}}=\{(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\},$$

and such that

$$\mathrm{proj}_{\Omega_{\mathrm{L}}}\left(\mathcal{V}\left(\Lambda_{\mathrm{d}},(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\right)\right)\neq\{0\}.$$

Proof:

Since $\alpha_{\mathrm{L}}\in\mathcal{A}_{\mathrm{L}}$ is of the same dimension as a tangent, affine subspace $\Pi_{\mathrm{d}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$,

$$\exists\alpha_{\mathrm{L}}\in\mathcal{A}_{\mathrm{L}}:\alpha_{\mathrm{L}}\cap\Lambda_{\mathrm{d}}=\{(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\}$$

if and only if

$$\exists\Pi_{\mathrm{d}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}):\Pi_{\mathrm{d}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\cap\Lambda_{\mathrm{d}}=\{(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\}.$$

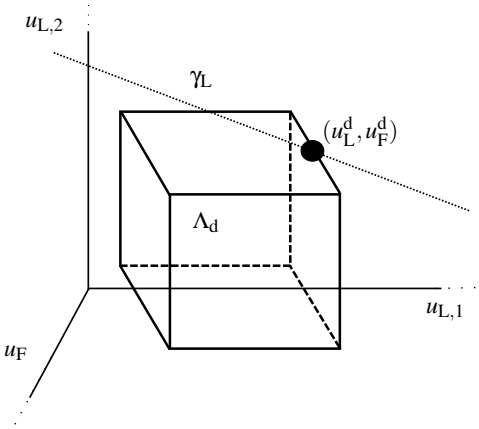


Figure 3.3: *Example of an affine leader function lying on a supporting hyperplane* $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ *that is not strictly supporting.*

In order for $\alpha_L \in \mathcal{A}_L^{\Pi_d^t(u_L^d, u_F^d)}$ to be a mapping $\Omega_L \to \Omega_F$ it is necessary and sufficient that $\text{proj}_{\Omega_L}\left(\mathcal{V}\left(\Lambda_d, (u_L^d, u_F^d)\right)\right) \neq \{0\}$ as was proven before in Proposition 3.12. □

It is important to note that an optimal affine leader function should satisfy (2.5) and (2.6). Constraint (2.7) is easily satisfied, namely for any affine function passing through the desired equilibrium. The main elements of the necessary and sufficient conditions with respect to (2.5) and (2.6) are therefore:

- $\text{dom}(\gamma_L) = \Omega_F$, i.e., it is not allowed for the function γ_L represented by a set of points to be perpendicular to the follower decision space. This orthogonality requirement is necessary and sufficient for the coverage of Ω_F by an affine function if $\Omega_L = \mathbb{R}^{n_L}, \Omega_F = \mathbb{R}^{n_F}$. Hence, $\text{proj}_{\Omega_L}\left(\mathcal{V}\left(\Lambda_d, (u_L^d, u_F^d)\right)\right) \neq \{0\}$ or $\text{proj}_{\Omega_L}\left(\mathcal{V}\left(\text{conv}\left(\Lambda_d\right), (u_L^d, u_F^d)\right)\right) \neq \{0\}$ is used in the above propositions.
- u_F^d should be the optimal follower response to $\gamma_L(\cdot)$, i.e., the set defined by γ_L should not intersect with the sublevel set Λ_d in any other point than (u_L^d, u_F^d). Hence, the concepts of an exposed point and of respectively a tangent hyperplane and an affine subspace are adopted in the above propositions.

Remark 3.15 Consider the case with $(u_L^d, u_F^d) \in \text{bd}(\text{conv}(\Lambda_d))$ but where (u_L^d, u_F^d) is not an exposed point, i.e., no supporting hyperplane $\Pi_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$ exists that intersects with $\text{conv}(\Lambda_d)$ solely in the point (u_L^d, u_F^d). It follows directly that neither a hyperplane exists that intersects with Λ_d solely in (u_L^d, u_F^d). However, by definition of the convex hull, there does exist a supporting hyperplane $\tilde{\Pi}_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d)$ for which it consequently holds that $\tilde{\Pi}_{\text{conv}(\Lambda_d)}(u_L^d, u_F^d) \cap \Lambda_d \setminus \{(u_L^d, u_F^d)\} \neq \varnothing$. For this case not captured by Proposition 3.13 an optimal affine γ_L may still exist for $n_L > 1$, as depicted in Fig. 3.3. ◇

Remark 3.16 In the special case with (u_L^d, u_F^d) exposed and with scalar decision variables ($n_F = 1, n_L = 1$), an affine $\gamma_L : \Omega_F \to \Omega_L$ leading to (u_L^d, u_F^d) automatically

exists.

Since Λ_d is nonsmooth at (u_L^d, u_F^d), a supporting hyperplane to Λ_d will not be a unique (tangent) hyperplane. By both the convexity of Λ_d and by (u_L^d, u_F^d) being an exposed point, we know that there does *not* exist a leader decision $u_L \in \Omega_L \setminus \{u_L^d\}$: $\{(u_L, u_F^d)\} \in \Lambda_d$. Therefore, there must exist an alternative normal vector defining the hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ that is not orthogonal to $\{0\}^{n_L} \times \Omega_F$. For such a vector, $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ and therefore $\alpha_L^{\Pi_{\Lambda_d}(u_L^d, u_F^d)}$ will cover Ω_F, i.e., $\mathrm{dom}(\gamma_L) = \Omega_F$. ◇

3.4 Characterization of an Optimal Affine Leader Function

In the remainder of this chapter, the full set of affine leader functions will be derived under which the leader is able to induce the follower to choose the input u_F^d and thereby to reach the desired solution point. In Section 3.4.1 first the case considered in the literature is summarized, after which we deal with the more general case in Section 3.4.2.

3.4.1 Under Differentiability Assumptions

A characterization of an optimal affine leader function (3.1), which reduces to the computation of an $n_L \times n_F$ matrix B, was first derived in [209] in case $J_F(\cdot)$ is differentiable in (u_L^d, u_F^d).

In order to make sure that B exists as defined next, it is assumed in [209] that Ω_L, Ω_F are Hilbert spaces[1] and that $J_F(\cdot)$ is Fréchet differentiable[2] on $\Omega_L \times \Omega_F$. Additionally, $J_F(\cdot)$ is assumed to be strictly convex in $\Omega_L \times \Omega_F$. It is known that for Ω_L, Ω_F Banach spaces[3] there exists a continuous linear operator B such that $Bu_F = u_L$, $u_F \neq 0$ [209]. Then, for n_L, n_F finite – a similar analysis is applicable for the infinite case – B should satisfy

$$[\nabla_{u_L} J_F(u_L^d, u_F^d)]^T B = [\nabla_{u_F} J_F(u_L^d, u_F^d)]^T, \tag{3.3}$$

which holds under the assumption that $\nabla_{u_L} J_F(u_L^d, u_F^d) \neq 0$ (as follows from the conditions for the existence of an optimal affine leader function) and which can be verified by taking the inner product of the expression $u_L = 0$ according to (3.1) and $[\nabla_{u_L} J_F(u_L^d, u_F^d)]^T$. This product

$$0 = [\nabla_{u_L} J_F(u_L^d, u_F^d)]^T [(u_L^d - u_L) + B(u_F^d - u_F)] \tag{3.4}$$

$$= [\nabla_{u_L} J_F(u_L^d, u_F^d)]^T (u_L^d - u_L) + [\nabla_{u_L} J_F(u_L^d, u_F^d)]^T B(u_F^d - u_F) \tag{3.5}$$

$$= [\nabla_{u_L} J_F(u_L^d, u_F^d)]^T (u_L^d - u_L) + [\nabla_{u_F} J_F(u_L^d, u_F^d)]^T (u_F^d - u_F) \tag{3.6}$$

[1] A Hilbert space $\mathcal{H}$ is a vector space with an inner product $\langle f, g \rangle$ such that the norm $\|f\|_2 = \sqrt{(f,f)}$ makes $\mathcal{H}$ a complete inner product space, i.e, for every Cauchy sequence $x^j \in \mathcal{H}$ ($\|x^j - x^k\| \to 0$ as $j,k \to \infty$) there is some $x \in \mathcal{H}$ such that $\|x - x^j\| \to 0$ as $j \to \infty$ [126].

[2] A function $f : X \to Y$ is Fréchet differentiable at $x \in X$ if for each $a \in X$ there exists a bounded linear operator $\delta f(x;a) \in Y$ linear and continuous with respect to a such that $\lim_{\|a\|_2 \to 0} \frac{\|f(x+a) - f(x) - \delta f(x;a)\|_2}{\|a\|_2} = 0$. In finite-dimensional spaces, it is represented by the Jacobian matrix [133].

[3] A Banach space is a complete normed linear vector space. A Hilbert space is special case where the norm $\|\cdot\|_2$ is an inner product [133].

corresponds exactly to the expression of a tangent hyperplane $\Pi^{\mathrm{t}}_{\Lambda_\mathrm{d}}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$ to Λ_d at $(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$, from which it is concluded that if (3.3) holds, the affine function γ_L indeed lies on the hyperplane $\Pi^{\mathrm{t}}_{\Lambda_\mathrm{d}}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$.

Under the condition $\nabla_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) \neq 0$ the following expression is mentioned in [209]:

$$B = \nabla_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) \nabla^\mathrm{T}_{u_\mathrm{F}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) / ||\nabla^\mathrm{T}_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})||^2. \tag{3.7}$$

Note that this is only one of many possible expressions for $n_\mathrm{L} > 1$. Moreover, in some constrained cases this expression does not yield an optimal leader function, while an alternative, optimal affine solution does exist as will be illustrated by Example 3.17 below. A generalized characterization of the optimal affine leader function is therefore developed in Section 3.4.2 below.

Example 3.17

We now provide a situation in which the specific expression of B proposed in [209] for $\jmath_\mathrm{F}(\cdot)$ differentiable at $(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$ does not yield a feasible leader function in the constrained case, but in which an optimal leader function does exist.

Let

$$\jmath_\mathrm{F}(u_\mathrm{L}, u_\mathrm{F}) = (u_\mathrm{F} - 6)^2 + (u_{\mathrm{L},1} - 1)^2 + (u_{\mathrm{L},2} - 5)^2,$$

and let $(u^\mathrm{d}_{\mathrm{L},1}, u^\mathrm{d}_{\mathrm{L},2}, u^\mathrm{d}_\mathrm{F}) = (0.5, 6, 4)$.

Then, $\nabla_{u_\mathrm{F}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) = 2u_F - 12$, $\nabla_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) = \begin{bmatrix} 2u_{L,1} - 2 \\ 2u_{L,2} - 10 \end{bmatrix}$, leading to

$$\begin{aligned} B &:= \frac{\nabla_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F}) \nabla^\mathrm{T}_{u_\mathrm{F}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})}{||\nabla^\mathrm{T}_{u_\mathrm{L}} \jmath_\mathrm{F}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})||^2} \\ &= \left(\begin{bmatrix} -1 \\ 2 \end{bmatrix} \cdot (-4) \right) / \left(\begin{bmatrix} 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} 4/5 \\ -8/5 \end{bmatrix}. \end{aligned} \tag{3.8}$$

As can be seen in Fig. 3.4 a mapping γ_L as defined through (3.1) with B as expressed in (3.8) does not return values for all $u_\mathrm{F} \in \Omega_\mathrm{F}$ within the bounding box imposed by the constraints

$$u_{\mathrm{L},1} \in [0, 16], u_{\mathrm{L},2} \in [0, 5], u_\mathrm{F} \in [2, 8].$$

Thus, a parametrization B as defined by (3.7) does not belong to the characterization of an optimal leader function in this constrained case.

However, there do exist optimal affine mappings $\Omega_\mathrm{F} \to \Omega_\mathrm{L}$ through $(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$ that lie on $\Pi_{\Lambda_\mathrm{d}}(u^\mathrm{d}_\mathrm{L}, u^\mathrm{d}_\mathrm{F})$. As also plotted in Fig. 3.4, a suitable leader function that also lies on the tangent hyperplane defined by the relation

$$-u_{\mathrm{L},1} + 1/2 \cdot u_{\mathrm{L},2} + 2 \cdot u_\mathrm{F} - 9/4 = 0,$$

would be:

$$u_\mathrm{L} = \widetilde{\gamma}_\mathrm{L}(u_\mathrm{F}) = \begin{bmatrix} 6 \\ 1/2 \end{bmatrix} + \begin{bmatrix} (-9/4 - 6)/4 \\ -1/8 \end{bmatrix} (4 - u_\mathrm{F}). \qquad \diamond$$

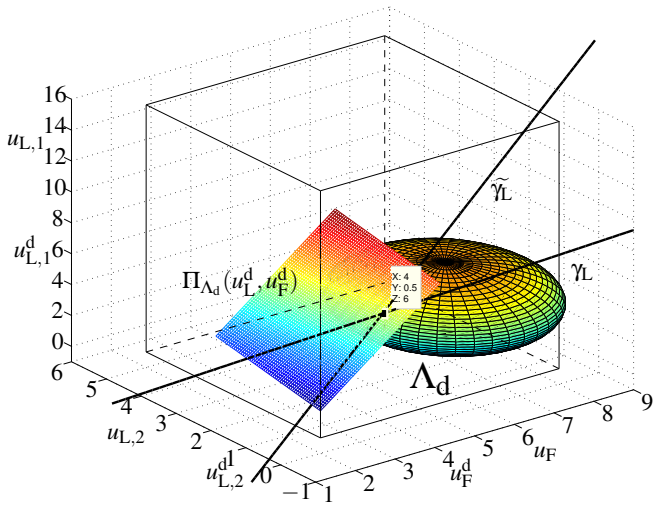


Figure 3.4: *Situation with a supporting hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ that is unique due to the differentiability of $\jmath_F(\cdot)$ at (u_L^d, u_F^d). The bounds of the decision space are indicated by a box.*

3.4.2 The General Case

As the previously presented result only captures the case in which $\jmath_F(\cdot)$ is differentiable and moreover as only one particular solution is specified, we now provide a characterization of the full set of possible leader functions with an affine structure that are optimal in an unconstrained decision space for the cases in which $\jmath_F(\cdot)$ is not required to be differentiable. Based on such a characterization one can deal with constraints on the decision space as well as apply further, secondary selection criteria like the minimization of the sensitivity to deviations from the optimal response as will be shown in respectively Section 3.5 and Section 3.6 below.

In the following, we characterize γ_L as a linear combination of the columns of the matrices $R = \begin{bmatrix} R_L^T & R_F^T \end{bmatrix}^T$, $R \in \mathbb{R}^{(n_L+n_F)\times n_F}, R_L \in \mathbb{R}^{n_L \times n_F}, R_F \in \mathbb{R}^{n_F \times n_F}$, and an offset $\begin{bmatrix} (u_L^d)^T & (u_F^d)^T \end{bmatrix}^T$, i.e.,

$$\gamma_L : \begin{bmatrix} u_L \\ u_F \end{bmatrix} = \begin{bmatrix} u_L^d \\ u_F^d \end{bmatrix} + \begin{bmatrix} R_L \\ R_F \end{bmatrix} \cdot s, \tag{3.9}$$

where $s \in \mathbb{R}^{n_F}$ represents the free parameters of the affine function. Now, for R_F invertible – which automatically follows from the necessary conditions as will be proven in Lemma 3.18 below – it follows that:

$$u_F = u_F^d + R_F \cdot s \quad \Rightarrow \quad s = R_F^{-1}(u_F - u_F^d), \tag{3.10}$$

$$u_{\mathrm{L}} = u_{\mathrm{L}}^{\mathrm{d}} + \underbrace{R_{\mathrm{L}} R_{\mathrm{F}}^{-1}}_{B}(u_{\mathrm{F}} - u_{\mathrm{F}}^{\mathrm{d}}), \tag{3.11}$$

i.e., one arrives at the explicit form of leader function (3.1). The problem left in order to arrive at a full characterization of an optimal affine leader function $\gamma_{\mathrm{L}}(\cdot)$ is to determine the set of possible basis vectors that represent the columns of R.

Lemma 3.18 *In order for a leader function γ_{L} characterized by (3.9) to be optimal, for $R = \begin{bmatrix} R_{\mathrm{L}}^{\mathrm{T}} & R_{\mathrm{F}}^{\mathrm{T}} \end{bmatrix}$ the following should hold:*

1. *There exists a vector $\nu \in \mathcal{V}(X,(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$ such that $\nu^{\mathrm{T}} R = 0^{\mathrm{T}}$, with $\mathcal{V}(X,(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$ the generalized normal to X at $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ where $X = \mathrm{conv}(\Lambda_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$ in case $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ is exposed with respect to $\mathrm{conv}(\Lambda_{\mathrm{d}})$ (i.e., Proposition 3.13 applies) or $X = \Lambda_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ in case $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}) \in \mathrm{int}(\mathrm{conv})(\Lambda_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$ (i.e., Proposition 3.14 applies).*

2. *The columns of R_{F} should be a basis for Ω_{F}, i.e., R_{F} should be of full rank n_{F} and thus invertible.*

Proof:

1. By definition of a tangent hyperplane $\Pi_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ to a set X at $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$, it holds that $\Pi_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}) \perp \nu$ for some $\nu \in \mathcal{V}(X,(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$. Since we require each optimal leader function $\gamma_{\mathrm{L}}(\cdot)$ characterized by (3.9) to lie on $\Pi_{\mathrm{d}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$, it follows that it is needed that also each column of R is orthogonal to ν, i.e., $\nu^{\mathrm{T}} R = 0^{\mathrm{T}}$.

   Note that in case $\jmath_{\mathrm{F}}(\cdot)$ is differentiable, i.e., $\mathcal{V}(X,(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})) = \left\{\nu = \begin{bmatrix} \nu_{\mathrm{L}}^{\mathrm{T}} & \nu_{\mathrm{F}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}\right.$ with $\nu_{\mathrm{L}} = \nabla_{u_{\mathrm{L}}} \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$ and $\nu_{\mathrm{F}} = \nabla_{u_{\mathrm{F}}} \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\left.\right\}$, this condition is equivalent to the expression of a tangent hyperplane:

$$\Pi_{\Lambda_{\mathrm{d}}}^{\mathrm{t}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}) : [\nabla_{u_{\mathrm{L}}} \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})]^{\mathrm{T}}(u_{\mathrm{L}}^{\mathrm{d}} - u_{\mathrm{L}}) + [\nabla_{u_{\mathrm{F}}} \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})]^{\mathrm{T}}(u_{\mathrm{F}}^{\mathrm{d}} - u_{\mathrm{F}}) = 0. \tag{3.12}$$

2. For an optimal affine leader function $\gamma_{\mathrm{L}}(\cdot)$ characterized by (3.9) to satisfy $\mathrm{dom}(\gamma_{\mathrm{F}}) = \Omega_{\mathrm{F}}$, it is required that the n_{F} columns of R_{F} are independent basis vectors spanning Ω_{F}. Thus, R_{F} is of full rank and hence invertible. □

In fact, we can select w.l.o.g. $R_{\mathrm{F}} := I_{n_{\mathrm{F}}} = \begin{bmatrix} e_1 & , & \dots & , & e_{n_{\mathrm{F}}} \end{bmatrix}$ as shown in Lemma 3.19.

Lemma 3.19 *If there exists an optimal affine γ_{L} characterized by (3.9), one can select w.l.o.g. $R_{\mathrm{F}} = I_{n_{\mathrm{F}}}$.*

Proof: Consider

$$\mathcal{S} := \left\{ \gamma_{\mathrm{L}} \;\middle|\; \gamma_{\mathrm{L}} : \begin{bmatrix} u_{\mathrm{L}} \\ u_{\mathrm{F}} \end{bmatrix} = \begin{bmatrix} u_{\mathrm{L}}^{\mathrm{d}} \\ u_{\mathrm{F}}^{\mathrm{d}} \end{bmatrix} + \begin{bmatrix} R_{\mathrm{L}} \\ I_{n_{\mathrm{F}}} \end{bmatrix} \cdot s, \text{ with } s \in \mathbb{R}^{n_{\mathrm{F}}} \text{ and with } R_{\mathrm{L}}, R_{\mathrm{F}} = I_{n_{\mathrm{F}}} \right. \\ \left. \text{satisfying conditions 1) and 2) of Lemma 3.18} \right\}, \tag{3.13}$$

$$\tilde{\mathcal{S}} := \left\{ \gamma_{\mathrm{L}} \;\middle|\; \gamma_{\mathrm{L}} : \begin{bmatrix} u_{\mathrm{L}} \\ u_{\mathrm{F}} \end{bmatrix} = \begin{bmatrix} u_{\mathrm{L}}^{\mathrm{d}} \\ u_{\mathrm{F}}^{\mathrm{d}} \end{bmatrix} + \begin{bmatrix} \tilde{R}_{\mathrm{L}} \\ \tilde{R}_{\mathrm{F}} \end{bmatrix} \cdot \tilde{s}, \text{ with } \tilde{s} \in \mathbb{R}^{n_{\mathrm{F}}} \text{ and with } \tilde{R}_{\mathrm{L}}, \tilde{R}_{\mathrm{F}} \right. \\ \left. \text{satisfying conditions 1) and 2) of Lemma 3.18 with R substituted by } \tilde{R} \right\}. \tag{3.14}$$

To prove that $\mathcal{S} \equiv \tilde{\mathcal{S}}$ we will show that for each possible triple $(s, I_{n_{\mathrm{F}}}, R_{\mathrm{L}})$ according to (3.9) with $\nu^{\mathrm{T}} \begin{bmatrix} R_{\mathrm{L}}^{\mathrm{T}} & I_{n_{\mathrm{F}}} \end{bmatrix}^{\mathrm{T}} = 0^{\mathrm{T}}$ that yields some $u_{\mathrm{L}}, u_{\mathrm{F}}$, one can find an equivalent triple $(\tilde{s}, \tilde{R}_{\mathrm{F}}, \tilde{R}_{\mathrm{L}})$, with $\tilde{s} \in \mathbb{R}^{n_{\mathrm{F}}}$ for which additionally it holds that $\nu^{\mathrm{T}} \begin{bmatrix} \tilde{R}_{\mathrm{L}}^{\mathrm{T}} & \tilde{R}_{\mathrm{F}} \end{bmatrix}^{\mathrm{T}} = 0^{\mathrm{T}}$, yielding the same values $u_{\mathrm{L}}, u_{\mathrm{F}}$.

It can be easily seen that the expression $u_{\mathrm{F}} = u_{\mathrm{F}}^{\mathrm{d}} + I_{n_{\mathrm{F}}} \cdot s$ is equivalent to $u_{\mathrm{F}} = u_{\mathrm{F}}^{\mathrm{d}} + \tilde{R}_{\mathrm{F}} \cdot \tilde{s}$ with $s = \tilde{R}_F \cdot \tilde{s}$: as shown in Lemma 3.18 it follows from the existence of an optimal affine γ_{L} that R_{F} is invertible. Then, for a given s there exists a unique $\tilde{s}$ and vice versa. From $B = \tilde{R}_{\mathrm{L}} \tilde{R}_{\mathrm{F}}^{-1}$ according to (3.10) and from the substitution to $B = R_{\mathrm{L}} I_{n_{\mathrm{F}}}$, for equivalence it should hold that $R_{\mathrm{L}} = \tilde{R}_{\mathrm{L}} \tilde{R}_{\mathrm{F}}^{-1}$. Finally, we have that

$$\nu_{\mathrm{L}}^{\mathrm{T}} R_{\mathrm{L}} + \nu_{\mathrm{F}}^{\mathrm{T}} = 0 \quad \Leftrightarrow \quad \nu_{\mathrm{L}}^{\mathrm{T}} \tilde{R}_{\mathrm{L}} \tilde{R}_{\mathrm{F}}^{-1} + \nu_{\mathrm{F}}^{\mathrm{T}} = 0 \quad \Leftrightarrow \quad \nu_{\mathrm{L}}^{\mathrm{T}} \tilde{R}_{\mathrm{L}} + \nu_{\mathrm{F}}^{\mathrm{T}} \tilde{R}_{\mathrm{F}} = 0. \tag{3.15}$$

Hence, $\mathcal{S} = \tilde{\mathcal{S}}$. □

Now, given $R_{\mathrm{F}} := I_{n_{\mathrm{F}}}$ we still need to identify the set of matrices R_{L} that satisfy $\nu^{\mathrm{T}} R = 0^{\mathrm{T}}$ for some normal vector ν, which reduces to $\begin{bmatrix} \nu_{\mathrm{L}}^{\mathrm{T}} & \nu_{\mathrm{F}}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} R_{\mathrm{L}} \\ I_{n_{\mathrm{F}}} \end{bmatrix} = 0^{\mathrm{T}}$ or equivalently $\nu_{\mathrm{L}}^{\mathrm{T}} R_{\mathrm{L}} = -\nu_{\mathrm{F}}^{\mathrm{T}}$. Due to the necessary condition $\mathrm{proj}_{\Omega_{\mathrm{L}}} \left(\mathcal{V} \left(\mathrm{conv} \left(\Lambda_{\mathrm{d}} \right), (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \right) \right) \neq \{0\}$ (Proposition 3.13) or $\mathrm{proj}_{\Omega_{\mathrm{L}}} \left(\mathcal{V} \left(\Lambda_{\mathrm{d}}, (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \right) \right) \neq \{0\}$ (Proposition 3.14), $\nu_{\mathrm{L}}^{\mathrm{T}}$ must contain at least one nonzero entry. Hence, the equations $\nu_{\mathrm{L}}^{\mathrm{T}} R_{\mathrm{L},j} = -\nu_{\mathrm{F},j}, j = 1, \ldots, n_{\mathrm{F}}$ can indeed be solved. Proposition 3.21 below provides a parametrized characterization of this problem that will be needed for further optimization.

From the previous derivations, the following theorem automatically follows.

Theorem 3.20 *Let $\Omega_{\mathrm{L}} = \mathbb{R}^{n_{\mathrm{L}}}, \Omega_{\mathrm{F}} = \mathbb{R}^{n_{\mathrm{F}}}$. Assume that the conditions in Proposition 3.13 or 3.14 are satisfied and that therefore an optimal affine leader function of the form (3.1) exists. Then, the set $\Gamma_{\mathrm{L}}^* := \{\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}} | \gamma_{\mathrm{L}}$ according to (3.1) satisfying (2.6)-(2.7), (3.9) $\}$ contains optimal affine solutions that can be characterized by $B := R_{\mathrm{L}} I_{n_{\mathrm{F}}}$, with $\nu_{\mathrm{L}}^{\mathrm{T}} R_{\mathrm{L}} = -\nu_{\mathrm{F}}^{\mathrm{T}}$, for some $\nu \in \mathcal{V} \left(\mathrm{conv} \left(\Lambda_{\mathrm{d}} \right), (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \right)$ in case $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ is exposed with respect to $\mathrm{conv}(\Lambda_{\mathrm{d}})$ (Proposition 3.13 applies) or for some some $\nu \in \mathcal{V} \left(\Lambda_{\mathrm{d}}, (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \right)$ in case $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \in \mathrm{int} \left(\mathrm{conv} \left(\Lambda_{\mathrm{d}} (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}) \right) \right)$ (Proposition 3.14 applies).*

For the sake of conciseness, in the remainder of this section we will assume $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ to be an exposed point of $\mathrm{conv}(\Lambda_{\mathrm{d}})$. As a result, we consider the case in

which Proposition 3.13 is satisfied rather than Proposition 3.14, i.e., we consider the generalized normal $\mathcal{V}\left(\text{conv}\left(\Lambda_\text{d}\right),(u_\text{L}^\text{d},u_\text{F}^\text{d})\right)$. For the case in which Proposition 3.14 applies, the generalized normal should be substituted by $\mathcal{V}\left(\Lambda_\text{d},(u_\text{L}^\text{d},u_\text{F}^\text{d})\right)$ in the following.

In order to be able to optimize over the set of possible leader functions and to select a function that is optimal with respect to some criteria, Proposition 3.21 now provides a parametrized characterization of the set of optimal affine leader functions.

Proposition 3.21 *Let*

$$\Gamma_\text{L}^* := \{\gamma_\text{L} : \Omega_\text{F} \to \Omega_\text{L} | \gamma_\text{L} \textit{ satisfies (2.6)-(2.7), (3.9)}\}. \tag{3.16}$$

1. *For $\jmath_\text{F}(\cdot)$ nondifferentiable at $(u_\text{L}^\text{d},u_\text{F}^\text{d})$, the possible realizations of $R_\text{L}\in\mathbb{R}^{n_\text{L}\times n_\text{F}}$ can be written:*

$$R_\text{L}\in\mathcal{R}_\text{L} := \left\{ \begin{bmatrix} R_{\text{L},1} & \dots & R_{\text{L},n_\text{F}}\end{bmatrix} \middle| \, R_{\text{L},j}\in\mathcal{R}_j, j=1,\dots,n_\text{F}\right\},$$

with the set of possible columns of R_L characterized by

$$\begin{aligned}\mathcal{R}_{\text{L},j} := &\left\{ Q\cdot W\cdot p_j^+ \,\middle|\, \begin{array}{l} p_j^+ := \left(\sum_{i=1}^{N_{+,j}^\text{f}} \alpha_{i,j}^+\beta_{i,+,j}^\text{f} + \sum_{i=1}^{N_{+,j}^\text{e}} \mu_{i,j}^+\beta_{i,+,j}^\text{e}\right), \\ \sum_i \alpha_{i,j}^+ = 1, \alpha_{i,j}^+\in\mathbb{R}^+, \mu_{i,j}^+\in\mathbb{R}^+ \end{array}\right\} \\ &\bigcup \left\{ Q\cdot(-W)\cdot p_j^- \,\middle|\, \begin{array}{l} p_j^- := \left(\sum_{i=1}^{N_{-,j}^\text{f}} \alpha_{i,j}^-\beta_{i,-,j}^\text{f} + \sum_{i=1}^{N_{-,j}^\text{e}} \mu_{i,j}^-\beta_{i,-,j}^\text{e}\right), \\ \sum_i \alpha_{i,j}^- = 1, \alpha_{i,j}^-\in\mathbb{R}^+, \mu_{i,j}^-\in\mathbb{R}^+ \end{array}\right\},\end{aligned} \tag{3.17}$$

for $j=1,\dots,n_\text{F}$, with $Q := \begin{bmatrix} I_{n_\text{L}} & 0_{n_\text{L}\times n_\text{F}}\end{bmatrix}$ and with $W=\begin{bmatrix} w_1 & \dots & w_m\end{bmatrix}$, where $\{w_i\}_{i=1}^m$, $m\in\mathbb{N}\cup\{\infty\}$ with $w_i\in\mathbb{R}^{n_\text{L}+n_\text{F}}$ is the set of generators of $\bar{\mathcal{V}}\left(\text{conv}\left(\Lambda_\text{d}\right),(u_\text{L}^\text{d},u_\text{F}^\text{d})\right)$ such that

$$\bar{\mathcal{V}}\left(\text{conv}\left(\Lambda_\text{d}\right),(u_\text{L}^\text{d},u_\text{F}^\text{d})\right) := \left\{\sum_{i=1}^m \beta_i w_i | \beta_i\in\mathbb{R}^+\right\}\bigcup\left\{\sum_{i=1}^m \beta_i(-w_i) | \beta_i\in\mathbb{R}^+\right\}.$$

Further, $\{\beta_{i,s,j}^\text{f}\}_{i=1}^{N_{s,j}^\text{f}}$ and $\{\beta_{i,s,j}^\text{e}\}_{i=1}^{N_{s,j}^\text{e}}$, $s\in\{+,-\}$ are the sets of finite vertices and extreme rays respectively, of the polyhedra $\mathcal{P}_j^+ = \left\{\beta | PW\beta = e_j, \beta\in(\mathbb{R}^+)^m\right\}$ and $\mathcal{P}_j^- = \left\{\beta | P(-W)\beta = e_j, \beta\in(\mathbb{R}^+)^m\right\}$.

2. *For $\jmath_\text{F}(\cdot)$ differentiable at $(u_\text{L}^\text{d},u_\text{F}^\text{d})$, R_L belongs to the affine space of the form*

$$\mathcal{R}_\text{L} := \left\{ R_\text{L} \middle| R_\text{L} = R_\text{L}^0 + \mathcal{B}_N\cdot T, T\in\mathbb{R}^{\dim(N)\times n_\text{F}}\right\}, \tag{3.18}$$

with R_L^0 a particular solution of $\nabla_{u_\text{L}}^\text{T}\jmath_\text{F}(u_\text{L}^\text{d},u_\text{F}^\text{d})R_\text{L} = \nabla_{u_\text{F}}\jmath_\text{F}(u_\text{L}^\text{d},u_\text{F}^\text{d})$ and with $\mathcal{B}_N$ a basis of $N := \text{null}\left(\nabla_{u_\text{L}}^\text{T}\jmath_\text{F}(u_\text{L}^\text{d},u_\text{F}^\text{d})\right)$.

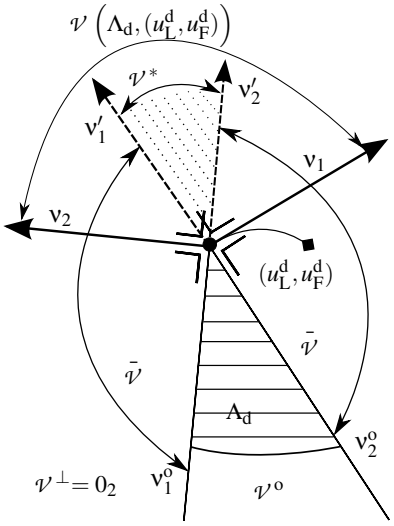


Figure 3.5: *The finitely generated normal cone* $\mathcal{V}\left(\Lambda_d,(u_L^d,u_F^d)\right)$ *and the associated cone* $\bar{\mathcal{V}}\left(\Lambda_d,(u_L^d,u_F^d)\right)=\left(\mathbb{R}^{n_L+n_F}\setminus\left(\mathcal{V}^0\cup\mathcal{V}^*\right)\right)\cup\mathcal{V}^\perp$.

Proof:

1. For Λ_d nonsmooth at (u_L^d,u_F^d), the generalized normal $\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)$ is by definition a convex pointed cone, i.e., $a_1\nu_1+a_2\nu_2\in\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)$ for any $a_1,a_2\in\mathbb{R}^+,\nu_1,\nu_2\in\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)$. In fact, $\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)$ is the normal cone [164] defined by the set of normal vectors to Λ_d at (u_L^d,u_F^d), which is generated by $n\in\mathbb{N}\cup\{\infty\}$ generators:

$$\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right):=\left\{\sum_{i=1}^{n}\alpha_i\nu_i\,\middle|\,\alpha_i\in\mathbb{R}^+,\nu_i\in\mathbb{R}^{n_L+n_F}\right.$$
$$\left.\text{a generator of }\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)\right\}. \tag{3.19}$$

The polar and dual cone of $\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)$ and its orthogonal complement are denoted respectively by[4]:

$$\mathcal{V}^0=\left\{r\in\mathbb{R}^{n_L+n_F}\,\middle|\,r^T\cdot\nu\le 0\quad\forall\nu\in\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)\right\},$$
$$\mathcal{V}^*=\left\{r\in\mathbb{R}^{n_L+n_F}\,\middle|\,r^T\cdot\nu\ge 0\quad\forall\nu\in\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)\right\},$$
$$\mathcal{V}^\perp=\left\{r\in\mathbb{R}^{n_L+n_F}\,\middle|\,r^T\cdot\nu=0\quad\forall\nu\in\mathcal{V}\left(\text{conv}\left(\Lambda_d\right),(u_L^d,u_F^d)\right)\right\}.$$

It now follows from condition 1) of Lemma 3.18 that the set of possible columns

[4]The arguments $\text{conv}\left(\Lambda_d\right)$ and (u_L^d,u_F^d) are omitted for the sake of conciseness.

$R_j, j = 1,...,n_\mathrm{F}$ of R can be represented by:

$$\bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right) := \left\{ r \in \mathbb{R}^{n_\mathrm{L}+n_\mathrm{F}} \,\middle|\, \exists \mathrm{v} \in \mathcal{V}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right) : r^\mathrm{T}\cdot\mathrm{v}=0 \right\}$$
$$= \left(\mathbb{R}^{n_\mathrm{L}+n_\mathrm{F}} \setminus \left(\mathcal{V}^0 \cup \mathcal{V}^*\right)\right) \cup \mathcal{V}^\perp. \tag{3.20}$$

This last expression is illustrated in Fig. 3.5 where $\bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right)$ corresponds to the union of the areas between v_1^o and v_1' and between the vectors v_2^o and v_2', where v_1' and v_2' are perpendicular to v_1 and v_2. The expression (3.20) follows from the fact that the (closure of the) complement of a cone is again a cone [90]; hence, the complement of the double cone $\mathcal{V}^0 \cup \mathcal{V}^*$ [69] that consists of the union of two apex-to-apex placed pointed cones, i.e., $\mathbb{R}^{n_\mathrm{L}+n_\mathrm{F}} \setminus \left(\mathcal{V}^0 \cup \mathcal{V}^*\right)$, embodies a cone. Finally, the null vector is included again in order to yield a pointed cone. The set $\bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right)$ can therefore be written as a linear combination of generators w_i from the set $\{\pm w_i\}_{i=1}^m$ as in (3.19), now also considering the negatives $-w_i$:

$$\bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right) := \left\{ \sum_{i=1}^m \beta_i w_i \,\middle|\, \beta_i \in \mathbb{R}^+, w_i \in \mathbb{R}^{n_\mathrm{L}+n_\mathrm{F}} \right\}$$
$$\bigcup \left\{ \sum_{i=1}^m \beta_i (-w_i) \,\middle|\, \beta_i \in \mathbb{R}^+, w_i \in \mathbb{R}^{n_\mathrm{L}+n_\mathrm{F}} \right\}.$$

Finally, since $R = \begin{bmatrix} R_\mathrm{L}^\mathrm{T} & I_{n_\mathrm{F}} \end{bmatrix}^\mathrm{T}$ by Lemma 3.18 and 3.19, we need to select from $\bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right)$ those vectors r such that for $j = 1,...,n_\mathrm{F}$:

$$\mathcal{R}_j := \left\{ r \in \bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right) : \mathrm{proj}_{\Omega_\mathrm{F}}(r) = e_j \right\},$$

i.e., approved as a possible j-th column of R are those vectors $r = W\cdot\beta \in \bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right)$ as well as vectors $r = -W\cdot\beta \in \bar{\mathcal{V}}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right),\left(u_\mathrm{L}^\mathrm{d},u_\mathrm{F}^\mathrm{d}\right)\right)$ with $W = \begin{bmatrix} w_1 & \ldots & w_m \end{bmatrix}$, $\beta = \begin{bmatrix} \beta_1 & \ldots & \beta_m \end{bmatrix}^\mathrm{T}$, such that for $P := \begin{bmatrix} 0_{n_\mathrm{F}\times n_\mathrm{L}} & I_{n_\mathrm{F}} \end{bmatrix}$, we have

$$P\cdot W\cdot\beta = e_j \text{ respectively } P\cdot(-W)\cdot\beta = e_j.$$

The solutions to these two equations, where it should be noted that $\beta^+ \in (\mathbb{R}^+)^m$, $\beta^- \in (\mathbb{R}^+)^m$, can be computed using e.g., the double-description method [147] for the polyhedra $\mathcal{P}_j^+ = \left\{\beta^+ \,\middle|\, PW\beta^+ = e_j, \beta^+ \geq 0\right\}$ and $\mathcal{P}_j^- = \left\{\beta^- \,\middle|\, P(-W)\beta^- = e_j, \beta^- \geq 0\right\}$, i.e.,

$$\beta_j^s = \sum_{i=1}^{N_{s,j}^\mathrm{f}} \alpha_{i,j}^s \beta_{i,s,j}^\mathrm{f} + \sum_{i=1}^{N_{s,j}^\mathrm{e}} \mu_{i,j}^s \beta_{i,s,j}^\mathrm{e}, s \in \{+,-\},$$

with $\sum_i \alpha_{i,j}^s = 1, \alpha_{i,j}^s \in \mathbb{R}^+, \mu_{i,j}^k \in \mathbb{R}^+$, and with $\left\{ \beta_{i,s,j}^\mathrm{f} \,\middle|\, i = 1,...,N_{s,j}^\mathrm{f} \right\}$ the set of finite vertices of $\mathcal{P}_j^s$, and with $\left\{ \beta_{i,s,j}^\mathrm{e} \,\middle|\, i = 1,...,N_{k,s}^\mathrm{e} \right\}$ the set of extreme rays of $\mathcal{P}_j^s$, for $s \in \{+,-\}$.

2. For $\jmath_\mathrm{F}(\cdot)$ differentiable at $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$, $\mathcal{V}\left(\mathrm{conv}\left(\Lambda_\mathrm{d}\right), (u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})\right)$ is uniquely defined by $\nu = \nabla \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$ with $\nabla_{u_\mathrm{L}} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}) \neq 0$ (by Proposition 3.13 and 3.14) and therefore $\nu_\mathrm{L}^\mathrm{T} R_{\mathrm{L},j} = -\nu_{\mathrm{F},j}, j = 1, \ldots, n_\mathrm{F}$ can be solved as a simple system of equalities. Here, for each zero element $\nu_{\mathrm{L},i}$, the corresponding entry $R_{\mathrm{L},0,j}$ of R_L is free. Therefore, the possible solutions can be written as

$$\begin{aligned} &\nabla_{u_\mathrm{L}}^\mathrm{T} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}) R_{\mathrm{L},j} = \nabla_{u_{\mathrm{F},j}} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}) \\ &\Rightarrow R_{\mathrm{L},j} = R_{\mathrm{L},j}^0 + \underbrace{\mathcal{B}\left(\mathrm{null}\left(\nabla_{u_\mathrm{L}}^\mathrm{T} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})\right)\right)}_{\mathcal{B}_N} \cdot t, t \in \mathbb{R}^{\dim(N)}, \end{aligned} \tag{3.21}$$

where $R_{\mathrm{L},j}^0$ denotes a particular solution to (3.21) and $\mathcal{B}_N \cdot t$, with $t \in \mathbb{R}^{\dim(N)}$ is a homogeneous solution to (3.21). Note that a basis of the null space of $\nabla_{u_\mathrm{L}}^\mathrm{T} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}))$, $\mathcal{B}\left(\mathrm{null}(\nabla_{u_\mathrm{L}}^\mathrm{T} \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}))\right)$ as well as a particular solution $R_{\mathrm{L},j}^0$ can be computed with a singular value decomposition (SVD) or QR decomposition (see e.g., [74]). □

Remark 3.22 So far a static, single-stage reverse Stackelberg game has been considered. Whereas this basic case serves for developing the conditions summarized in Section 3.3 and the characterization of the present section, real-life control settings will often have a dynamic, multi-stage nature [154], as has also been elaborated upon in Chapter 2. As it is also done in e.g., [209], the current static results can be simply applied to the dynamic case with open-loop information. In other words, at the start of the game the open-loop values $(u_\mathrm{L}^\mathrm{d}(k), u_\mathrm{F}^\mathrm{d}(k))$ are computed, for which the mappings $\gamma_\mathrm{L}(u_\mathrm{F}(k), k)$ can be computed as done in the static case, for each (discrete) time step $k \in \mathcal{K}$. ◇

3.4.3 Computation and Complexity

While the characterization of this section is aimed to provide a structured method to solve the reverse Stackelberg game with an affine leader function, the computational efficiency of testing the several conditions should be kept into account. The following observations can be made:

- Determining a *global optimum* to represent the desired leader equilibrium $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$ is in general a constrained nonconvex nonlinear programming problem; this subproblem has to be solved in any solution approach. Alternatively, a desired equilibrium that is not directly derived from a leader objective functional, or a series of such points, may be provided *a priori*.

- Determining the *convex hull* of Λ_d is only required in case $\jmath_\mathrm{F}(\cdot)$ is both nonconvex and nondifferentiable at $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$. The exposed points of Λ_d can then be determined if $\jmath_\mathrm{F}(\cdot)$ is of the particular type of a piecewise affine function. For a polyhedron [171] with n vertices as input points, computing the convex hull can be done with a worst-case complexity of $\mathcal{O}(n \log p)$ for $n_\mathrm{L} + n_\mathrm{F} \leq 3$ and $\mathcal{O}(n \cdot f_p / p)$ for $n_\mathrm{L} + n_\mathrm{F} \geq 4$, where p points are actually on the hull and f_p denotes the maximum number of facets for p vertices [18].

- Verifying whether the *projection* of the generalized normal onto the leader's decision space is nonzero relies on simple inner vector products that can be obtained in time $o(n)$ for vectors of dimension n.
- Computing *particular and homogeneous solutions* to (3.21) with a singular value decomposition (SVD) leads to a numerically reliable solution due to its ability to deal with rank-deficient matrices; however, no finite termination can be guaranteed for the computation of the SVD. The iterative Golub-Reinsch algorithm for determining the SVD of a matrix can however be terminated when a sufficiently precise solution is obtained, which leads to a practical overall complexity of $o(n^3)$ floating-point operations, for an $m \times n$ matrix in case $m \approx n$ [74]. The alternative of using the QR decomposition technique does not have such numerical reliability properties but it does have finite termination; the complexity of the QR algorithms discussed in [74] are also of the order $o(n^3)$ in case $m \approx n$.

In the following numerical example, the computation of optimal affine leader functions is illustrated.

Example 3.23

The nonconvex Rosenbrock function [165] is often used to illustrate the performance of optimization algorithms and it is defined as:

$$f(x_1,x_2) = (1-x_1)^2 + 100(x_2 - x_1^2)^2, \tag{3.22}$$

as depicted in Fig. 3.6 together with several level curves. If we adopt this function for $\jmath_{\mathrm{F}}(\cdot)$, it can be inferred from these contour lines that several desired leader equilibria cannot be obtained under an affine leader function, i.e., those in the valleys of the upper part of the level curves that are associated with increasing objective function values when considering increasing values of u_{L}.

In order to illustrate the approach in higher dimensions, we adopt the extended Rosenbrock function [49], written in general for n dimensions as :

$$f_{\mathrm{e}}(x_1,\ldots,x_n) = \sum_{i=1}^{n-1} [(1-x_i)^2 + 100(x_{i+1} - x_i^2)^2]. \tag{3.23}$$

Let $\Omega_{\mathrm{L}} = \mathbb{R}$ and $\Omega_{\mathrm{F}} = \mathbb{R}^2$. The sublevel set Λ_{d} for $\jmath_{\mathrm{F}} := f_{\mathrm{e}}(u_{\mathrm{L}}, u_{\mathrm{F}})$ with the desired equilibrium arbitrarily chosen to be $(u^{\mathrm{d}}_{\mathrm{L},1}, u^{\mathrm{d}}_{\mathrm{L},2}, u^{\mathrm{d}}_{\mathrm{F}}) = (-0.2, -0.27, 0.15)$. Since

$$\begin{aligned}
\nabla_{u_{\mathrm{L}}} \jmath_{\mathrm{F}}(u^{\mathrm{d}}_{\mathrm{L}}, u^{\mathrm{d}}_{\mathrm{F}}) &= \nabla_{u_{\mathrm{L}}} [(1-u^{\mathrm{d}}_{\mathrm{F}})^2 + (1-u^{\mathrm{d}}_{\mathrm{L},1})^2 + \ldots \\
&\qquad 100(u^{\mathrm{d}}_{\mathrm{L},1} - {u^{\mathrm{d}}_{\mathrm{F}}}^2)^2 + 100(u^{\mathrm{d}}_{\mathrm{L},2} - {u^{\mathrm{d}}_{\mathrm{L},1}}^2)^2] \\
&= \begin{bmatrix} \left(-2 + 2u_{\mathrm{L},1} + 200u_{\mathrm{L},1} - 200u_{\mathrm{F}}^2 - 400u_{\mathrm{L},2}u_{\mathrm{L},1} + 400u_{\mathrm{L},1}^3\right) \\ \left(200u_{\mathrm{L},2} - 200u_{\mathrm{L},1}^2\right) \end{bmatrix} \\
&= \begin{bmatrix} -56.21 & 15.42 \end{bmatrix}^{\mathrm{T}} \neq \begin{bmatrix} 0 & 0 \end{bmatrix}^{\mathrm{T}}, \\
\nabla_{u_{\mathrm{F}}} \jmath_{\mathrm{F}}(u^{\mathrm{d}}_{\mathrm{L}}, u^{\mathrm{d}}_{\mathrm{F}}) &= -2 + 2u^{\mathrm{d}}_{\mathrm{F}} - 400u^{\mathrm{d}}_{\mathrm{L},1}u^{\mathrm{d}}_{\mathrm{F}} + 400{u^{\mathrm{d}}_{\mathrm{F}}}^3 = -27.20,
\end{aligned}$$

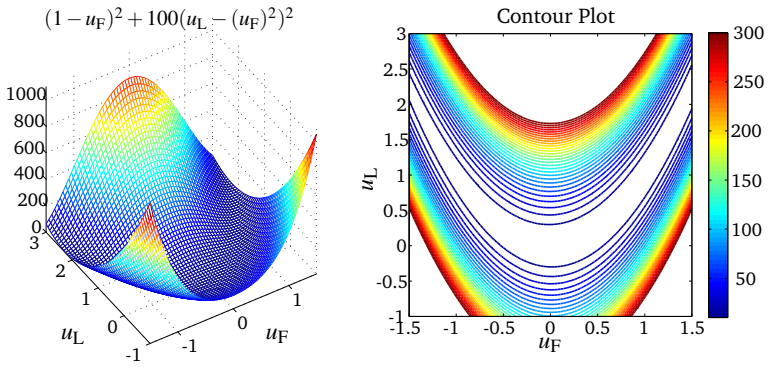


Figure 3.6: *The Rosenbrock function and several level curves.*

the unique strictly supporting hyperplane at (u_L^d, u_F^d) can be defined by:

$$\Pi_{\Lambda_d}(u_L^d, u_F^d) = \begin{bmatrix} -56.21 & 15.42 \end{bmatrix} \left(\begin{bmatrix} -0.27 \\ 0.15 \end{bmatrix} - \begin{bmatrix} u_{L,1} \\ u_{L,2} \end{bmatrix} \right) - 27.20(-0.2 - u_F) = 0.$$

Now, $\gamma_L(\cdot)$ characterized by (3.9) with $R_F = 1$ has associated matrices R_L according to (3.18), where $R_L^0 = \begin{bmatrix} (15.42/-27.20) & (-56.21/-27.20) \end{bmatrix}^T$ is a particular solution of $\nabla_{u_L}^T \jmath_F(u_L^d, u_F^d) R_L = \nabla_{u_F} \jmath_F(u_L^d, u_F^d)$, and with $\mathcal{B}_{\text{null}\left(\nabla_{u_L}^T \jmath_F(u_L^d, u_F^d)\right)} = \begin{bmatrix} 0.2645 & 0.9644 \end{bmatrix}^T$, leading to the set of optimal affine solutions characterized by

$$\mathcal{R}_L := \left\{ R_L \,\middle|\, R_L = \begin{bmatrix} (15.42/-27.20) \\ (-56.21/-27.20) \end{bmatrix} + \begin{bmatrix} 0.2645 \\ 0.9644 \end{bmatrix} \cdot t,\ t \in \mathbb{R} \right\}. \qquad \diamond$$

3.5 Constrained Decision Spaces

So far the situation without constraints has been considered and conditions have been provided under which an optimal affine leader function exists that leads to the desired reverse Stackelberg equilibrium point. These conditions form necessary but not sufficient conditions for the existence of an optimal affine leader function in the constrained game in which $\Omega_L \subsetneq \mathbb{R}^{n_L}$ or $\Omega_F \subsetneq \mathbb{R}^{n_F}$. In the constrained case, the complexity arises that additionally the locally defined supporting hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ — or the tangent hyperspace $\Pi_{\Lambda_d}^t(u_L^d, u_F^d)$ for the case with $n_L > 1$ and $(u_L^d, u_F^d) \in \text{int}(\text{conv}(\Lambda_d))$ — should be within the constrained decision space $\Omega_L \times \Omega_F$, with $\Omega_L \subsetneq \mathbb{R}^{n_L}$ or $\Omega_F \subsetneq \mathbb{R}^{n_F}$. This implies that the supporting or tangent hyperplane should contain an n_F-dimensional affine subspace γ_L satisfying (i) γ_L should cover Ω_F, i.e., $\text{dom}(\gamma_L) = \Omega_F$ while (ii) $\gamma_L(\Omega_F) \subseteq \Omega_L$. However, since the hyperplanes are derived locally, it thus still has to be verified whether an optimal leader function γ_L exists in the bounded decision space such that the global conditions (i) and (ii) hold.

Hence, given the set of feasible solutions Γ_L^* characterized in Section 3.4 above that is essentially developed for the unconstrained decision space, constraints can

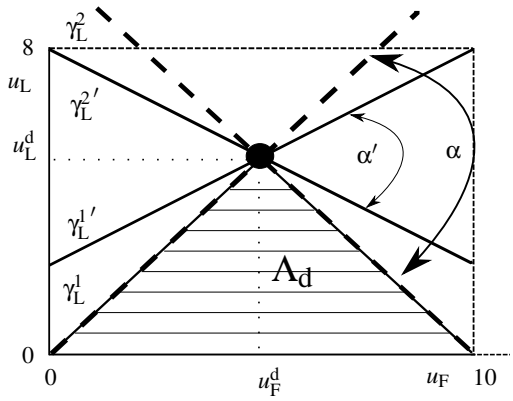


Figure 3.7: *Example of a set Γ_L^* that is reduced under the consideration of constraints on Ω_L, Ω_F. Here, α indicates the range of possible values of R_L. Note that the dashed functions γ_L^1, γ_L^2 have an infinitesimal gap with* $\mathrm{bd}(\Lambda_d)$ *for values of u_L below or above u_F^d, respectively.*

simply be incorporated to verify which elements of Γ_L^* are still valid under the constrained conditions. Here it should be noted that any constraints on the decision spaces can obviously affect the desired equilibrium point (u_L^d, u_F^d) as well as the set Λ_d, which elements are both assumed to be given in the conditions and the initial characterization of Sections 3.3 and 3.4. In order to use these results, applicable constraints on the decision spaces should therefore naturally be incorporated in the computation of a desired leader equilibrium (u_L^d, u_F^d) and in the derivation of the associated sublevel set Λ_d at the initial stage.

The following simple example illustrates this approach.

Example 3.24 Consider a reverse Stackelberg game with the desired equilibrium $(u_L^d, u_F^d) = (5,5)$ and a nonsmooth, convex sublevel set Λ_d as depicted in Fig. 3.7.

For $\Omega_L = \Omega_F = \mathbb{R}$, $\Gamma_L^* := \{\gamma_L : \Omega_F \to \Omega_L | \gamma_L \text{ satisfies (2.6)-(2.7), (3.9)}\}$ is characterized by the open interval $\mathcal{R}_L = (-1,1)$. Note that in this particular case, an optimal affine mapping $\Omega_F \to \Omega_L$ through (u_L^d, u_F^d) coincides with a line $\Pi_{\Lambda_d}(u_L^d, u_F^d)$. This interval of possible slopes of γ_L is indicated in Fig. 3.7 by α.

Now, consider the decision space imposed by the constraints $\Omega_L = [0,10], \Omega_F = [0,8]$. It can be seen that not all mappings $\gamma_L \in \Gamma_L^*$ return values for all $u_F \in \Omega_F$. Using the extrema $(0,10)$, $(8,0)$, and $(8,10)$ one can derive the intersection points of affine functions through (u_L^d, u_F^d) with the decision space boundaries $\mathrm{bd}(\Omega_L), \mathrm{bd}(\Omega_F)$, resulting in a new range α' for which $\mathcal{R}_L = (-3/5, 3/5)$.

The full set of possible optimal affine leader functions can thus be characterized as follows:

$$\Gamma_L^{*,\mathrm{con}} = \{u_L := \gamma_L(u_F) = 5 + B \cdot (u_F - 5), B \in (-3/5, 3/5)\}. \qquad \diamond$$

Remark 3.25 For clarification, we here elaborate on how the two examples of constrained decision spaces drawn in Fig. 3.4 and Fig. 3.7 compare to one another. Since

$n_L = 2$ in Fig. 3.4 whereas u_L and u_F are scalar in Fig. 3.7, in the former figure γ_L is a subset of $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ while in the latter figure this function coincides with the hyperplane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$. Moreover, $\Lambda_d(u_L^d, u_F^d)$ is smooth in Fig. 3.4, while it is nonsmooth in Fig. 3.7. Hence, in the first case the initial set Γ_L^* of optimal solutions for the unconstrained case consists of possible mappings through (u_L^d, u_F^d) lying on the unique plane $\Pi_{\Lambda_d}(u_L^d, u_F^d)$, while it consists of possible supporting hyperplanes $\Pi_{\Lambda_d}(u_L^d, u_F^d)$ in Fig. 3.7. In both constrained cases, elements $\gamma_L^*(\cdot) \in \Gamma_L^*$ are subsequently removed for which $\gamma_L^*(\Omega_F) \not\subseteq \Omega_L$. The following paragraph shows how such a derivation can be made. $\diamond$

Derivation of the set $\Gamma_L^{*,\mathrm{con}}$ The possible optimal matrices B of (3.11) computed in Section 3.4 for the cases with $\jmath_F(\cdot)$ differentiable and nondifferentiable respectively can be re-evaluated under the presence of constraints, where we assume the constrained decision spaces to be convex and closed and bounded, i.e., compact. In particular, one needs to verify whether $\gamma_L(\Omega_F) \subseteq \Omega_L$.

a) In case only Ω_F is restricted while $\Omega_L = \mathbb{R}^{n_L}$ we have that $\Gamma_L^{*,\mathrm{con}} = \Gamma_L^*$, which can be concluded from the fact that Γ_L^* is derived only locally based on Λ_d, while $\Omega_L = \mathbb{R}^{n_L}$. Hence, $\gamma_L(\Omega_F) \subseteq \Omega_L$ still holds for all $\gamma_L \in \Gamma_L^*$.

b) Further, the case with only Ω_L restricted while $\Omega_F = \mathbb{R}^{n_F}$ is only feasible if $\gamma_{L,i}(\Omega_F) = \{c_i\} \in \Omega_{L,i}$ for some $c_i \in \mathbb{R}$ for every index i such that $\Omega_{L,i} \subsetneq \mathbb{R}$, where $\Omega_{L,i}$ is the projection of Ω_L on the i-th coordinate and $\gamma_{L,i}(\cdot)$ denotes the i-th component of the vector-valued function $\gamma_L(\cdot)$. Indeed, the realization of the leader function for each such i-th component of the leader's decision space should be constant for any follower decision variable, as for any affine function $\gamma_{L,i}(u_F) \neq c_i \in \mathbb{R}$, $\lim_{u_F \to \pm\infty} \gamma_{L,i}(u_F)$ will not be finite and therefore not an element of $\Omega_{L,i} \subsetneq \mathbb{R}$.

c) In case both decision spaces are restricted by linear constraints, Ω_L, Ω_F represent polytopes. By applying the affine mapping $\gamma_L(\cdot)$, convexity is preserved, implying that its image $\gamma_L(\Omega_F)$ is a polytope as well [24]. As a result, it can be easily checked whether the image of $\gamma_L(\cdot)$ for its domain Ω_F satisfies the linear constraints imposed by Ω_L: it is sufficient to verify $\gamma_L(u_F^v) \in \Omega_L$ only for the vertices u_F^v of Ω_F.

   This result can also be used to obtain the reduced characterization of the set of optimal affine leader functions in the constrained case. If we denote the linear constraints on Ω_L by

$$A_L \cdot u_L \leq b_L, A_L \in \mathbb{R}^{n_c \times n_L}, b_L \in \mathbb{R}^{n_c}, n_c \in \mathbb{N},$$

   one should add the following set of linear inequality constraints to the characterizations in (3.17) or (3.18) respectively depending on differentiability of $\jmath_F(\cdot)$ at (u_L^d, u_F^d), where $\{u_F^v\}_{v=1}^{n_v}$ denotes the set of vertices associated with Ω_F:

$$\left\{ A_L \cdot \left(u_L^d + R_L(u_F^v - u_F^d) \right) \leq b_L \right\}_{v=1,\ldots,n_v}. \tag{3.24}$$

In the nondifferentiable case this results in the characterization (3.17) in which the nonnegative parameters $\alpha^s_{i,j}, \mu^s_{i,j}$ are now constrained by (3.24) as well as by $\sum_i \alpha^s_{i,j} = 1,\ j = 1,\ldots,n_{\mathrm{F}}$ and with $i = 1,\ldots,N^{\mathrm{f}}_{s,j}$ for $s \in \{+,-\}$. Similarly, in the differentiable case this will result in the characterization (3.18) in which T now belongs to the polyhedron $\mathbb{R}^{\dim(N)\times n_{\mathrm{F}}}$ subject to (3.24).

d) Finally, in the case of nonlinear constraints, a similar approach to c) can be adopted based on a piecewise affine approximation of the constraints. However, this may in general generate a large number of vertices. Further, in order to guarantee feasibility of the leader function, (conservative) inner approximations should be made with respect to the leader decision space and outer approximations should be made with respect to the follower decision space. Alternatively, a fully numerical evaluation of the set $\Gamma^*_{\mathrm{L}} := \{\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}} | \gamma_{\mathrm{L}}$ satisfies (2.6)-(2.7), (3.9)$\}$ for the unconstrained case may be made, e.g., based on a gridding of the decision spaces.

3.6 Secondary Objectives

Another motivation for determining the complete set $\Gamma^{*,\mathrm{con}}_{\mathrm{L}}$ of optimal leader functions is the consideration of a secondary objective within the reverse Stackelberg game. In other words, in case one would consider the risk of suboptimal solutions due to a suboptimal response of the follower to $\gamma_{\mathrm{L}} \in \Gamma^{*,\mathrm{con}}_{\mathrm{L}}$ due to either bounded rationality or uncertainty in the follower characteristics, one may look at the sensitivity of this follower response as a criterion for adopting a certain leader function. Such a secondary objective is relevant when one can expect deviations from the optimal decisions; in [29, 30] such a secondary objective is considered for the case the leader is uncertain regarding the parameter values in $J_{\mathrm{F}}(\cdot)$. In this dissertation, a deterministic setting with fully rational players is considered, where players adopt those decision variables that are computed to be optimal with respect to their objective functions. However, due to possible limits in the precision with which a follower player can derive decisions, e.g., in case these are discrete without the leader knowing so, deviations from the computed optimal decision variable are still a relevant issue.

A graphical example is given in Fig. 3.8, where the optimal leader function γ_{L} is subject to sensitivity to deviations of the follower from $\arg\min_{u_{\mathrm{F}}\in\Omega_{\mathrm{F}}} J_{\mathrm{F}}(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}})$. The solutions on γ_{L} for $u_{\mathrm{F}} > u^{\mathrm{d}}_{\mathrm{F}}, u_{\mathrm{L}} > u^{\mathrm{d}}_{\mathrm{L}}$ are close to $\mathrm{bd}(\Lambda_{\mathrm{d}})$ and therefore they return a follower objective function value that is close to $J_{\mathrm{F}}(u^{\mathrm{d}}_{\mathrm{L}}, u^{\mathrm{d}}_{\mathrm{F}})$. Hence, significantly sensitive solutions may be removed from the set Γ^*_{L} according to (3.16), where sensitivity of a solution γ_{L} can be defined through its vicinity to $\mathrm{bd}(\mathrm{conv}(\Lambda_{\mathrm{d}}))$, i.e., by the angle between γ_{L} and the mean normal vector

$$\bar{\nu} := \sum_{i=1}^{n} \frac{1}{n} \nu_i \text{ or } \bar{\nu} := \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \nu_i, \tag{3.25}$$

for the case with $n \in \mathbb{N}$ or with infinitely many generators of $\mathcal{V}\left(\mathrm{conv}\left(\Lambda_{\mathrm{d}}\right), (u^{\mathrm{d}}_{\mathrm{L}}, u^{\mathrm{d}}_{\mathrm{F}})\right)$, respectively. The alternative solution $\widetilde{\gamma_{\mathrm{L}}} \in \Gamma^*_{\mathrm{L}}$ such that $\widetilde{\gamma_{\mathrm{L}}} \perp \tilde{\nu}$ is the least sensitive to

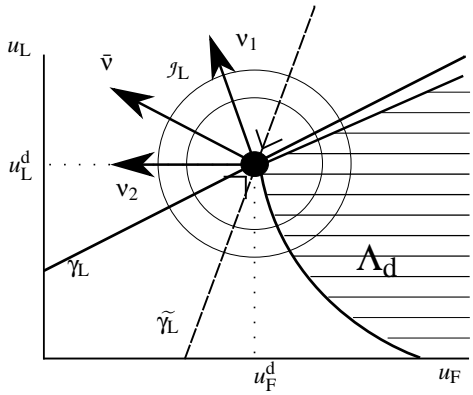


Figure 3.8: *Situation in which the optimal leader function* γ_{L} *is sensitive to deviations of the follower from* $\arg\min_{u_{\mathrm{F}}\in\Omega_{\mathrm{F}}} \vartheta_{\mathrm{F}}(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}})$.

such deviations from $\arg\min_{u_{\mathrm{F}}\in\Omega_{\mathrm{F}}} \vartheta_{\mathrm{F}}(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}})$ in Fig. 3.8.

3.7 Discussion

In this chapter, the single-leader single-follower static reverse Stackelberg game is considered, straightforwardly extendable to the open-loop dynamic game. Here, we adopt the indirect approach to the reverse Stackelberg game, according to which the leader player faces the problem of selecting a leader function – mapping her decision space into the follower's decision space – that will lead to a specific desired equilibrium. In order to be able to solve this generally complex game, it pays off to analyze particular leader function structures. In the literature, many examples and applications in which this type of game is considered adopt strictly convex follower objective functions and unconstrained decision spaces, in which case an affine leader function is sufficient to solve the game to optimality. In order to allow the reverse Stackelberg game to be more readily applicable as a decision-making structure for multilevel optimization-based control problems like in traffic tolling, there is a need to develop a more general solution approach.

In the current chapter, we have therefore first presented necessary and sufficient existence conditions and a characterization of the set of optimal affine leader functions that can be computed in a systematic manner. The solution approach when using an affine leader function structure can be summarized as follows. After an initial set Γ_{L}^{*} of optimal affine functions that are locally feasible is derived from the necessary and sufficient conditions of the existence of an optimal affine leader function for unconstrained decision spaces, this set can be further reduced to include only those elements that map the full follower's decision space into the leader's decision space, i.e., that satisfy $\gamma_{\mathrm{L}}(\Omega_{\mathrm{F}}) \subseteq \Omega_{\mathrm{L}}$, in case these spaces are constrained. Secondary optimization criteria can be incorporated similarly, e.g., selecting the leader function that is least sensitive to deviations from the optimum.

Clearly, an optimal affine leader function does not always exist, nor do the players behave fully rationally at all times as it is assumed in this thesis. Further steps therefore include the investigation of more diverse classes of nonlinear leader functions, e.g., piecewise affine and smooth (piecewise) polynomial structures, and an analysis of the robustness of a leader function in case of uncertain conditions and bounded rationality. The former is considered in the subsequent Chapter 4, while we leave the latter for future research. In addition, the development of solution approaches for the reverse Stackelberg game that are tractable remains a focal point of continued research.

Chapter 4

On Systematic Computation of Optimal Nonlinear Leader Functions

In the previous chapter, conditions on the existence of a leader function with an affine structure have been presented that lead to the leader's desired equilibrium point. Furthermore, the set of such optimal leader functions has been delineated and it has been shown how this set can be reduced to exclude infeasible elements in case of constrained decision spaces. Under the objective of developing a systematic solution approach for the reverse Stackelberg game, in the current chapter a broader class of nonlinear leader functions is considered. In particular, a continuous multi-level optimization approach as well as a gridding approach are proposed to compute an optimal leader function based on basis functions; also leader functions derived by interpolation are discussed and a comparison is made with evolutionary solution approaches proposed in the literature.

The research discussed in this chapter is based on [86], supported by the results presented in [85].

4.1 Introduction

As has been pointed out in Chapter 2, many reverse Stackelberg games considered in the literature involve particular problem instances with quadratic objective functions and linear state update equations [34, 56, 139]. In these games, affine leader functions prove sufficient to solve the game to optimality [83, 209], explaining the attention in the literature to leader functions of this type [125]. In case no optimal affine leader functions exist, more advanced nonlinear function structures are necessary. Moreover, they may embody more desirable properties related to, e.g., sensitivity to deviations from the optimal follower response.

To our best knowledge, only few papers consider nonlinear leader functions; they appear in specific, often numerical, examples [153, 154, 172, 208]. Nonetheless, there are few solution approaches for nonlinear leader functions proposed in the literature. While *indirect* approaches are suggested to be better able to solve the generally complex Stackelberg game in theory [16, 98], the suggested solution

approaches, based on evolutionary learning algorithms, are mostly devised for the *direct* game formulation [131, 179, 186, 187]. Recall that in the indirect game variant (Fig. 2.3(a)), a leader function is derived based on a particular desired equilibrium that the leader strives to achieve, while in the direct game (Fig. 2.3(b)) a leader function is computed that results in the best leader objective function value. Therefore, it can in fact be more efficient to adopt an indirect approach in which a leader function is determined that exactly yields an *a priori* decided equilibrium that is desired by or acceptable for the leader.

We here provide a structured solution approach for the class of nonlinear leader functions for the indirect variant of the single leader-single follower static, deterministic reverse Stackelberg game in which a specific nonlinear parametrization of the leader function is considered. Due to the inherent nonlinear and nonconvex nature of the optimization problems that result from considering nonlinear leader functions, instead of developing existence conditions for an optimal leader function, approaches to compute an optimal nonlinear leader function are developed.

The first proposed method is based on determining optimal parameters of a particular set of basis functions such that the resulting leader function (i) maps the full follower's decision space to the leader's decision space, and such that (ii) the associated optimal follower response coincides with the value of the decision variable associated with the leader's desired equilibrium. Constraints in this method can be based on an analytic expression of the follower objective function, as well as rely on a grid-based representation of the follower's sublevel set for a given equilibrium point. In a final alternative approach, the leader function is represented by an interpolating spline through a selection of data points in the decision space.

These indirect solution approaches are illustrated and applied to a worked example in which different levels of complexity are considered. The two evolutionary algorithms from the literature, i.e., a genetic algorithm and a neural network approach as considered in respectively [186] and [179] are also included in the comparison. Next to comparing their performance with respect to required computation times and the level of suboptimality or deviation from the desired equilibrium, the advantages and disadvantages of the different methods are analyzed.

This chapter is organized as follows. Several brief additions to the reverse Stackelberg game as defined in Chapter 2 are provided in Section 4.2. A summary of evolutionary algorithms from the literature that are applied to the direct game formulation is provided in Section 4.3. Section 4.4 includes a discussion of the indirect continuous multilevel and the grid-based basis function approaches proposed to derive an optimal nonlinear leader function. Subsequently, a heuristic interpolation method is considered in Section 4.5. A discussion on the complexity of the approaches and the choice of basis functions is provided in Section 4.6, after which the methods are illustrated and their results are compared and discussed in a worked example in Section 4.7. Finally, conclusions and recommendations are provided in Section 4.8.

Table 4.1: *Overview of methods to derive leader functions.*

Solution Methods		
Direct:	Genetic algorithm approach	[186]
	Neural network approach	[179]
Indirect:	Continuous multilevel approach	[86]
	Grid-based approach	[86]
	Interpolating spline approach	[86]

4.2 Preliminaries

Recall that the reverse Stackelberg game can be formulated according to a direct (2.2) and an indirect (2.5)–(2.7) approach. In this chapter, we develop methods focused at solving the latter game variant, and we compare them with methods from the literature that are devised to solve the game according to the direct formulation.

4.2.1 Nonlinear Incentive Controllability

Given the indirect formulation of the reverse Stackelberg game, the problem is reduced to finding a leader function that solves the game to optimality, i.e., that leads to the desired leader equilibrium[1]. When assuming a particular parametrized leader function structure, the problem further reduces to finding parameters for which the given leader function is optimal. In this chapter, nonlinear leader functions are determined that satisfy (2.6)–(2.7), where the constraint (2.6) implies that for all $u_\mathrm{F} \in \Omega_\mathrm{F} \setminus \{u_\mathrm{F}^\mathrm{d}\}$, $(\gamma_\mathrm{L}(u_\mathrm{F}), u_\mathrm{F})$ should remain outside of the sublevel set

$$\Lambda_\mathrm{d} := \left\{ (u_\mathrm{L}, u_\mathrm{F}) \in \Omega_\mathrm{L} \times \Omega_\mathrm{F} \,\middle|\, \jmath_\mathrm{F}(u_\mathrm{L}, u_\mathrm{F}) \leq \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d}) \right\}. \tag{4.1}$$

Now, optimality and feasibility of a leader function can be properly defined, given the following conditions:

[C.1] $\gamma_\mathrm{L}(\Omega_\mathrm{F}) \subseteq \Omega_\mathrm{L}$;

[C.2] $\jmath_\mathrm{F}(\gamma_\mathrm{L}(u_\mathrm{F}), u_\mathrm{F}) > \jmath_\mathrm{F}(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$, for all $u_\mathrm{F} \in \Omega_\mathrm{F} \setminus \{u_\mathrm{F}^\mathrm{d}\}$;

[C.3] $\gamma_\mathrm{L}(u_\mathrm{F}^\mathrm{d}) = u_\mathrm{L}^\mathrm{d}$.

4.2.2 Definitions

In this chapter, we adopt the following definitions in relation to conditions C.1–C.3:

Definition 4.1 Game feasibility, optimality
A leader function is called *game-feasible* for an instance of the reverse Stackelberg game if C.1 holds and *game-optimal* if C.1, C.2, and C.3 are satisfied. ◇

[1] Recall that the leader's desired solution point $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$ is referred to as the *(desired) equilibrium (point)*, even if a suboptimal leader function is applied that leads to a follower response $u_\mathrm{F} \neq u_\mathrm{F}^\mathrm{d}$.

Definition 4.2 Program feasibility, optimality
A leader function returned as a solution by any of the feasibility programs proposed in Section 4.4 and Section 4.5 is called *program-feasible* with respect to that problem. Similarly, a leader function returned by any of the optimization procedures proposed in Section 4.3 is called *program-optimal*. ◇

4.2.3 Assumptions

In this chapter, we consider the decision spaces Ω_L, Ω_F to be compact, hence bounded Euclidean spaces of a finite dimension as has also been assumed in Chapter 3. In the following, we also assume (u_L^d, u_F^d) to be a boundary point of sublevel set Λ_d; if this does not hold, no game-optimal leader function exists.

Necessary and sufficient existence conditions for a game-optimal affine leader function, i.e., a leader function of the form

$$\gamma_L(u_F) = u_L^d + B(u_F - u_F^d), B \in \mathbb{R}^{n_L \times n_F}, \tag{4.2}$$

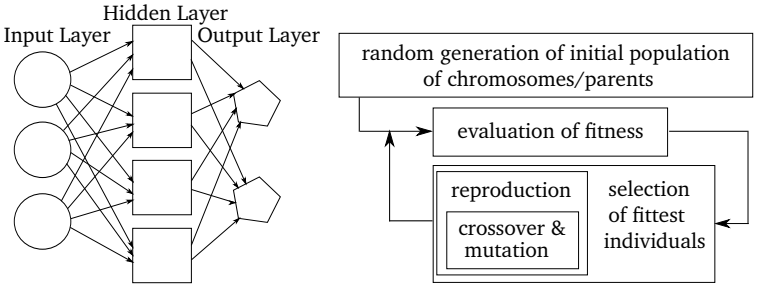
with $\Omega_p = \mathbb{R}^{n_p}, p \in \{L, F\}$ have been proposed and discussed in the previous chapter for the unconstrained case based on which a characterization of the full set of such optimal functions can be given. This set may be further reduced in the constrained case $\Omega_p \subsetneq \mathbb{R}^{n_p}, p \in \{L, F\}$. Clearly, in case a game-optimal affine leader function exists, nonlinear functions can also be considered. This may be preferred in case the nonlinear alternatives are more robust to deviations of the follower to his optimal decision.

In Section 4.3 below, the direct approaches proposed in the literature are summarized, after which our indirect methods are discussed in the subsequent Section 4.4.

4.3 Direct Evolutionary Algorithms

In the past decades, evolutionary algorithms have been adopted to solve different types of optimization problems that are computationally hard [48, 122, 197]. Amongst the various available algorithms are genetic algorithms and neural network-based algorithms. A basic schematic representation of both approaches is provided in Fig. 4.1. In the neural network approach, based on given values of the independent or decision variables (the 'input layer') and an associated value of the corresponding fitness function (the 'output layer'), a network of connections and weights is built (the 'hidden layer') that represents the relationship between inputs and outputs. In a genetic algorithm, a similar initial set of values of the decision variables (a population of 'chromosomes') is randomly generated. Based on the associated values of the corresponding fitness function, a new set of decision variable values is derived based on the evolutionary-based principles of crossover and mutation of the 'fittest' chromosomes.

Also in Stackelberg games in which the leader's strategy is a function of the state variable rather than a mapping $\gamma_L : \Omega_F \to \Omega_L$, heuristic search algorithms have been



(a) Artificial Neural Network

(b) Genetic Algorithm

Figure 4.1: *Schematic representation of an artificial neural network and a genetic algorithm.*

proposed [131, 187]. Specifically, in [187] a genetic algorithm is proposed as it is said to enable dealing with games in which the players do not know each other's cost functions, where it has the additional benefit that it can be implemented in a distributed fashion.

These approaches to solve the reverse Stackelberg game fall under the direct methods; instead of first computing the leader's global optimum and subsequently trying to derive a leader function that satisfies the intersection constraints as depicted in Fig. 2.3(a), possible leader functions are directly computed and the resulting equilibrium and associated objective function values are evaluated as depicted in Fig. 2.3(b). For reverse Stackelberg games, both genetic algorithms [186] and neural networks [179] have been proposed. In [186] several financial market applications were used to illustrate the approach, while in [179] a case study involving traffic tolling was performed.

However, the following drawback should be kept in mind: since neural network approaches as well as genetic algorithms are heuristic search methods, there are no performance guarantees or optimality bounds on the objective function value of the leader player, even if it the follower's information would be fully disclosed.

A summary of the solution approaches to the reverse Stackelberg game using either a neural network or a genetic algorithm are given in the following, where we refer to [122, 197] for a detailed description of the respective algorithms in a general context.

4.3.1 Neural Network Approach

A neural network is a structure or pattern that models the relation between inputs and outputs based on training and validation data. In this case, it is used to approximate the relation between leader function coefficients and the associated leader objective function value.

The main steps in the neural network algorithm of [179] are as follows:

1. Let a chosen leader function structure of $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$ be parametrized by coefficients $(a_i)_j \in \mathbb{R}, i = 1,\ldots,n_{\mathrm{a}}, j = 1,\ldots n_{\mathrm{L}}$ as in (4.4). A set $\mathcal{C}_{\mathrm{NN}}$ consisting

of $n_c \in \mathbb{N}_0^+$ realizations is selected for such coefficient values represented by tuples of vectors: $\mathcal{C}_{NN} := \left\{ \left(a_1^{(k)}, \ldots, a_{n_a}^{(k)} \right) | a_i \in \mathbb{R}^{n_L}, i = 1, \ldots, n_a, k = 1, \ldots, n_c \right\}$.

2. For each coefficient combination $c \in \mathcal{C}_N$, compute an optimal response of the follower: $u_F^c \in \arg\min_{u_F \in \Omega_F} \jmath_F(\gamma_L^c(u_F), u_F)$ and evaluate $\jmath_L(\gamma_L^c(u_F^c), u_F^c)$, where $\gamma_L^c(\cdot)$ is parametrized by the coefficients $c \in \mathcal{C}_{NN}$.

3. Divide the set of sample points, i.e., the pairs of coefficient values and associated leader function values, $\{(c, \jmath_L(\gamma_L^c(u_F^c), u_F^c)) | c \in \mathcal{C}_{NN}\}$ into a training and validation set and train the neural network to approximate the relation between coefficient values and the resulting leader objective function value.

4. Optimize the trained neural network with a solver for nonconvex optimization to find the coefficients minimizing the associated leader objective function values. Nonlinear constraints on $\gamma_L(\cdot)$, e.g., to satisfy C.1, can be incorporated in the optimization of the network.

Here, the choice of coefficients, of the number of coefficient pairs to evaluate, and of the ratio of the division of samples into training and validation pairs that lead to a good eventual performance, is made on the basis of experience and tuning.

4.3.2 Genetic Algorithm Approach

The main steps in the genetic algorithm are similar to those of the neural network approach, a main difference being in the choice of coefficients. While in the neural network approach a set of coefficients is determined by the user and used throughout the algorithm to approximate the neural network, in the genetic algorithm the coefficients are determined at random at the initial stage. The coefficients are adapted by probabilistic measures that are based on the fitness, i.e., the leader objective function value associated with the coefficients. The evolution of coefficients should lead to the best values and is terminated when a maximum number of iterations is reached:

1. Let a chosen leader function structure of $\gamma_L : \Omega_F \to \Omega_L$ be parametrized by coefficients $(a_i)_j \in \mathbb{R}, i = 1, \ldots, n_a, j = 1, \ldots n_L$ as in (4.4). Randomly select a population $\mathcal{C}_{GA}$ consisting of $n_c \in \mathbb{N}_0^+$ realizations: $\mathcal{C}_{GA} := \left\{ \left(a_1^{(k)}, \ldots, a_{n_a}^{(k)} \right) | a_i \in \mathbb{R}^{n_L}, i = 1, \ldots, n_a, k = 1, \ldots, n_c \right\}$, where $c \in \mathcal{C}_{GA}$ represents a chromosome of the algorithm.

2. For each of the n_c chromosomes or coefficient combinations $c \in \mathcal{C}_{GA}$, compute an optimal response of the follower: $u_F^c \in \arg\min_{u_F \in \Omega_F} \jmath_F(\gamma_L^c(u_F), u_F)$, where $\gamma_L^c(\cdot)$ is parametrized by the coefficients $c \in \mathcal{C}_{GA}$, and evaluate $\jmath_L(\gamma_L^c(u_F^c), u_F^c)$.

3. Based on the fitness $\jmath_L(\gamma_L^c(u_F^c), u_F^c)$ associated with the chromosomes, a new generation of chromosomes is derived by the mutation, crossover, and reproduction procedures of the genetic algorithm. Here, constraints on $\gamma_L(\cdot)$, e.g., $\gamma_L(\Omega_F) \subseteq \Omega_L$, can be incorporated to yield game-feasible chromosomes. For this selection, the algorithm continues with step (2).

Finally, it is argued that as an advantage of the evolutionary approaches, the leader does not need to possess information on $\jmath_{\mathrm{F}}(\cdot)$ [186]. However, the algorithms as just described *do* require knowledge of the reaction curve of the follower to each possible leader function for a correct computation of the fitness function values and for the training of the neural network.

In the worked example of Section 4.7 below our methods for computing a game-optimal leader function, as will be described in the following sections, will be compared also to the evolutionary algorithm implementations. There, the computation of the leader's global optimum $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ is taken into account as a subproblem of the indirect approaches in order to arrive at a fair comparison of the computational requirements.

4.4 A Basis Function Approach

We first consider $\gamma_{\mathrm{L}}(\cdot)$ to take the form of a linear combination of basis functions, and lateron of a piecewise polynomial (spline).

Leader functions can be represented by a weighted combination of basis functions that can be universal approximators, able to approximate any function with an arbitrary accuracy [27, 183]. Let the set of selected multidimensional basis functions $b_i : \mathbb{R}^{n_{\mathrm{F}}} \to \mathbb{R}^{n_{\mathrm{L}}}$, $i = 1, \ldots, n$ be written as:

$$\mathcal{B} = \{b_i(\cdot)\}_{i=1}^{n}. \tag{4.3}$$

We can now choose a leader function of the reverse Stackelberg game to be constructed as a linear combination of the chosen basis functions, parametrized by the coefficients $a_i \in \mathbb{R}^{n_{\mathrm{L}}}, i = 1, \ldots, n_{\mathrm{a}}$, i.e.:

$$\gamma_{\mathrm{L}}(u_{\mathrm{F}}) = \sum_{i=1}^{n} a_i \odot b_i(u_{\mathrm{F}}), \tag{4.4}$$

where $\odot$ denotes the element-by-element matrix (Schur) product.

4.4.1 A Continuous Approach Based on Basis Functions

In order to find a game-optimal leader function for a particular selection of basis functions $\{b_i\}_{i=1}^{n}$, the following feasibility program can be used:

$$\begin{aligned}
&\text{To find: } a_i \in \mathbb{R}^{n_{\mathrm{L}}}, && i = 1, \ldots, n, && (4.5)\\
&\text{such that } \jmath_{\mathrm{F}}(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}}) > \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}), && \forall u_{\mathrm{F}} \in \Omega_{\mathrm{F}} \setminus \{u_{\mathrm{F}}^{\mathrm{d}}\}, && (4.6)\\
&\qquad u_{\mathrm{L}}^{\mathrm{d}} = \gamma_{\mathrm{L}}(u_{\mathrm{F}}^{\mathrm{d}}), && && (4.7)\\
&\qquad \gamma_{\mathrm{L}}(u_{\mathrm{F}}) \in \Omega_{\mathrm{L}}, && \forall u_{\mathrm{F}} \in \Omega_{\mathrm{F}}, && (4.8)\\
&\qquad \gamma_{\mathrm{L}}(u_{\mathrm{F}}) = \sum_{i=1}^{n} a_i \odot b_i(u_{\mathrm{F}}). && && (4.9)
\end{aligned}$$

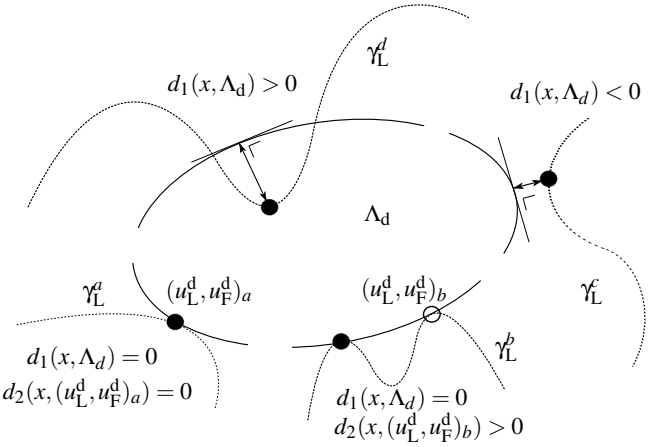


Figure 4.2: *Illustration of the adopted distance measures with respect to the sublevel set Λ_{d}. The symbol x is used to represent the points $(u_{\mathrm{L}},u_{\mathrm{F}})$ indicated by black dots on the different curves.*

Here it should be noted that the constraints (4.6) and (4.8) are complicating the solution process of the otherwise straightforward problem as they apply to specific subsets of the variable domains. In order to ease the solvability of the program (4.5)–(4.9), we therefore propose to substitute the complicating constraints by a lower-level optimization problem, the main elements of which will be introduced first.

We now introduce a function $d_1((u_{\mathrm{L}},u_{\mathrm{F}}),\Lambda_{\mathrm{d}})$ that expresses the signed measure of distance between a point $(u_{\mathrm{L}},u_{\mathrm{F}})$ and the boundary of the set Λ_{d}, i.e., the distance is assigned a positive sign only if it corresponds to a point belonging to the set Λ_{d}. Similarly, $\omega((u_{\mathrm{L}},u_{\mathrm{F}}),\Omega_{\mathrm{L}})$ refers to the Euclidean distance between the leader variable u_{L} of a point $(u_{\mathrm{L}},u_{\mathrm{F}})$ and the boundary of the decision space Ω_{L}, where the positive distances referring to points in Ω_{L} are substituted by zero to simplify the problem. The distance $d_2((u_{\mathrm{L}},u_{\mathrm{F}}),(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}))$ denotes the Euclidean distance between a point $(u_{\mathrm{L}},u_{\mathrm{F}})$ and $(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})$. More specifically, we define:

$$d_1((u_{\mathrm{L}},u_{\mathrm{F}}),\Lambda_{\mathrm{d}}) = \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}) - \jmath_{\mathrm{F}}(u_{\mathrm{L}},u_{\mathrm{F}}), \tag{4.10}$$

$$d_2((u_{\mathrm{L}},u_{\mathrm{F}}),(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})) = \|(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}}) - (u_{\mathrm{L}},u_{\mathrm{F}})\|_2, \tag{4.11}$$

$$\omega((u_{\mathrm{L}},u_{\mathrm{F}}),\Omega_{\mathrm{L}})) = \begin{cases} 0, & \text{for } u_{\mathrm{L}} \in \Omega_{\mathrm{L}}, \\ -\min_{u_{\mathrm{L}}^{\mathrm{b}} \in \mathrm{bd}(\Omega_{\mathrm{L}})} \|u_{\mathrm{L}}^{\mathrm{b}} - u_{\mathrm{L}}\|_2, & \text{for } u_{\mathrm{L}} \notin \Omega_{\mathrm{L}}. \end{cases} \tag{4.12}$$

Here, $\mathrm{bd}(\Lambda_{\mathrm{d}})$ can be characterized by the level set:

$$\left\{(u_{\mathrm{L}},u_{\mathrm{F}}) \in \Omega_{\mathrm{L}} \times \Omega_{\mathrm{F}} : \jmath_{\mathrm{F}}(u_{\mathrm{L}},u_{\mathrm{F}}) = \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}},u_{\mathrm{F}}^{\mathrm{d}})\right\}. \tag{4.13}$$

Given the definitions (4.10)–(4.12), the following auxiliary distance expressions

can be introduced:

$$d(\gamma_L, \Lambda_d) = \max_{u_F \in \Omega_F} \left(d_1((\gamma_L(u_F), u_F), \Lambda_d) + \alpha d_2 \left((\gamma_L(u_F), u_F), (u_L^d, u_F^d) \right) \right), \tag{4.14}$$

$$\omega(\gamma_L, \Omega_L) = \min_{u_F \in \Omega_F} \omega((\gamma_L(u_F), u_F), \Omega_L), \tag{4.15}$$

where $0 < \alpha \ll 1$. Examples of these distance expressions are depicted in Fig. 4.2. If $d(\gamma_L, \Lambda_d) \leq 0$, there is no intersection of the set $\{(\gamma(u_F), u_F) | u_F \in \Omega_F \setminus \{u_F^d\}\}$ with Λ_d, while inclusion of the point (u_L^d, u_F^d) is already guaranteed for any mapping $\gamma_L(\cdot)$ due to the constraint (4.7). In particular, γ_L does not intersect with $\mathrm{bd}(\Lambda_d)$ due to the addition of $d_2\left((u_L, u_F), (u_L^d, u_F^d)\right)$ – multiplied by the small constant α – to the signed distance $d_1((u_L, u_F), \Lambda_d)$.

In order to ensure that $\gamma_L(\Omega_F) \subseteq \Omega_L$, a similar approach is adopted: if the minimum signed shortest distance $\omega((u_L, u_F), \Omega_L)$ for some point $(\gamma_L(u_F), u_F)$ is nonnegative, all leader elements of the image of $\gamma_L(\cdot)$ are within the leader decision space, satisfying $\gamma_L(\Omega_F) \subseteq \Omega_L$.

Now, the original program (4.5)–(4.9) can be replaced by the following program (4.16)–(4.20):

$$\begin{aligned}
\text{To find:} \quad & a_i \in \mathbb{R}^{n_L}, i = 1, \ldots, n_a, && (4.16)\\
\text{such that} \quad & d(\gamma_L, \Lambda_d) \leq 0, && (4.17)\\
& \omega(\gamma_L, \Omega_L) \geq 0, && (4.18)\\
& u_L^d = \gamma_L(u_F^d), && (4.19)\\
& \gamma_L(u_F) = \sum_{i=1}^{n_a} a_i \odot b_i(u_F). && (4.20)
\end{aligned}$$

Here, the computation of the distances (4.10)–(4.15) required in (4.16)–(4.18) can be interpreted as lower-level optimization problems.

On program feasibility In order for the multilevel program (4.16)–(4.20) to lead to a game-optimal nonlinear leader function if such a function exists, α should be sufficiently small, i.e., in problem instances in which (u_L^d, u_F^d) is an interior point of the convex hull of Λ_d, choosing α too large can create program infeasibility due to (4.17).

The following assumptions are made for a derivation of an upper bound, as supported by the illustration in Fig. 4.3:

i) the (positive) difference in function values $J_F(u_L, u_F)$ from $J_F(u_L^d, u_F^d)$ increases with a strictly positive factor $M \in \mathbb{R}_0^+$ when the corresponding point (u_L, u_F) outside Λ_d moves away from (u_L^d, u_F^d), i.e.,

$$\begin{aligned}
J_F(u_L, u_F) - J_F(u_L^d, u_F^d) \geq M \cdot \|(u_L, u_F) - (u_L^d, u_F^d)\|_2 \\
\text{for all } (u_L, u_F) \in (\Omega_L \times \Omega_F) \setminus \Lambda_d;
\end{aligned} \tag{4.21}$$

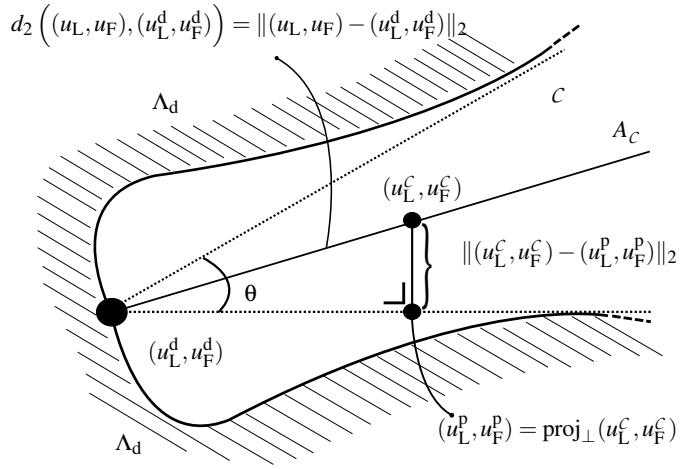


Figure 4.3: *Illustration of elements necessary for the derivation of an upper bound on* α.

ii) there exists a nondegenerate pointed cone[2] $\mathcal{C}$ with the apex (u_L^d,u_F^d) that does not intersect with $\Lambda_d \setminus \{(u_L^d,u_F^d)\}$. Let $\theta > 0$ be the maximum angle of this cone.

W.l.o.g. we can now assume that Λ_d is the complement of the cone $\mathcal{C}$.

In order for (4.17) to hold, since $d_2\left((u_L,u_F),(u_L^d,u_F^d)\right) \geq 0$ by definition, it follows that the following requirement should be satisfied for all $u_F \in \Omega_F$:

$$|d_1\left((\gamma_L(u_F),u_F),\Lambda_d\right)| \geq \alpha \cdot d_2\left((\gamma_L(u_F),u_F),(u_L^d,u_F^d)\right). \tag{4.22}$$

The limit case for this condition to still hold is when the set $\{(\gamma_L(u_F),u_F)\,|u_F \in \Omega_F\}$ coincides with the axis $A_{\mathcal{C}}$ of the cone $\mathcal{C}$. Consider a point $(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) \in A_{\mathcal{C}}$. Let (u_L^p,u_F^p) be the orthogonal projection of $(u_L^{\mathcal{C}},u_F^{\mathcal{C}})$ on $\mathrm{bd}(\mathcal{C})$, denoted in Fig. 4.3 by $\mathrm{proj}_\perp(u_L^{\mathcal{C}},u_F^{\mathcal{C}})$. Since the cone has a maximum angle, there exists such a point $(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) \in A_{\mathcal{C}}$ for which the orthogonal projection on the boundary of the cone approximates its orthogonal projection on the boundary of the sublevel set, as in the point (u_L^p,u_F^p), it holds that $\mathrm{bd}(\mathcal{C}) \approx \mathrm{bd}(\Lambda_d)$. Note that as a consequence we have that $\jmath_F(u_L^p,u_F^p) \approx \jmath_F(u_L^d,u_F^d)$ and thus

$$\left|d_1\left((u_L^{\mathcal{C}},u_F^{\mathcal{C}}),\Lambda_d\right)\right| = \jmath_F(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) - \jmath_F(u_L^p,u_F^p) \tag{4.23}$$

$$\geq M \cdot \|(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) - (u_L^p,u_F^p)\|_2. \tag{4.24}$$

Moreover,

$$\|(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) - (u_L^p,u_F^p)\|_2 = \sin\theta \cdot \|(u_L^{\mathcal{C}},u_F^{\mathcal{C}}) - (u_L^d,u_F^d)\|_2 \tag{4.25}$$

$$= \sin\theta \cdot d_2\left((u_L^{\mathcal{C}},u_F^{\mathcal{C}}),(u_L^d,u_F^d)\right). \tag{4.26}$$

[2]We call a cone nondegenerate if it has a strictly positive aperture, i.e., it does not solely contain a single half-line.

Hence,

$$\left| d_1\left((u_{\mathrm{L}}^{\mathcal{C}}, u_{\mathrm{F}}^{\mathcal{C}}), \Lambda_{\mathrm{d}}\right)\right| \geq M \sin\theta \cdot d_2\left((u_{\mathrm{L}}^{\mathcal{C}}, u_{\mathrm{F}}^{\mathcal{C}}), (u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\right). \tag{4.27}$$

In order to satisfy (4.22), it follows that α should be selected such that:

$$0 < \alpha \leq M \sin\theta. \tag{4.28}$$

Additionally, to satisfy $0 < \alpha \ll 1$, we can write:

$$0 < \alpha \leq \min\left(\xi, M \sin\theta\right), 0 < \xi \ll 1. \tag{4.29}$$

Here it should be noted that in case $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ is the apex of a pointed cone $\mathcal{C}$ as defined above that in addition is not sufficiently symmetric with respect to $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ when projected on Ω_{F}, no explicit leader function is feasible, as is illustrated in Fig. 4.4(a). In such case, an implicit function representation of a leader function may be adopted instead; this is considered as a topic for future research.

Also in case such a cone has an infinitesimal gap with the boundary of Λ_{d} in the neighborhood of $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$, no smooth explicit leader function is feasible. This situation is depicted in Fig. 4.4(b).

4.4.2 A Gridding Approach to Solve for a Basis Function

In general, multilevel optimization problems can consume a significant amount of computational resources, especially when the size of an instance of the optimization problem, e.g., influenced by the number of elements $n_{\mathrm{L}}, n_{\mathrm{F}}$ of the leader and follower decision variables in our case, increases. In order to prevent this issue, we here propose a simple, relaxed problem formulation based on the multilevel optimization approach defined in Section 4.4.1 above.

In particular, instead of solving (4.16)–(4.20) subject to the relations (4.10)–(4.15), we return to the original feasibility program (4.5)–(4.9) in which the complicating constraints (4.6) and (4.8) are transformed into regular constraints by performing a gridding approximation of the follower decision space. The left-hand side expressions in (4.6) and (4.8) are then evaluated at each relevant grid point $u_{\mathrm{F}} \in \Omega_{\mathrm{F}}^{\mathrm{g}}$, where $\Omega_{\mathrm{F}}^{\mathrm{g}} \subseteq \Omega_{\mathrm{F}}$ and with $u_{\mathrm{F}}^{\mathrm{d}} \in \Omega_{\mathrm{F}}^{\mathrm{g}}$, i.e., they are replaced by:

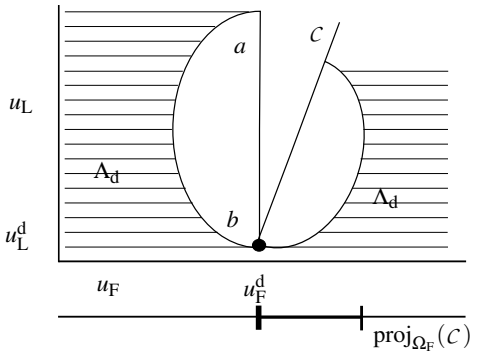
$$\jmath_{\mathrm{F}}(\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}}) > \jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}}), \qquad \text{for all } u_{\mathrm{F}} \in \Omega_{\mathrm{F}}^{\mathrm{g}} \setminus \{u_{\mathrm{F}}^{\mathrm{d}}\}, \tag{4.30}$$

$$\gamma_{\mathrm{L}}(u_{\mathrm{F}}) \in \Omega_{\mathrm{L}}, \qquad \text{for all } u_{\mathrm{F}} \in \Omega_{\mathrm{F}}^{\mathrm{g}}. \tag{4.31}$$

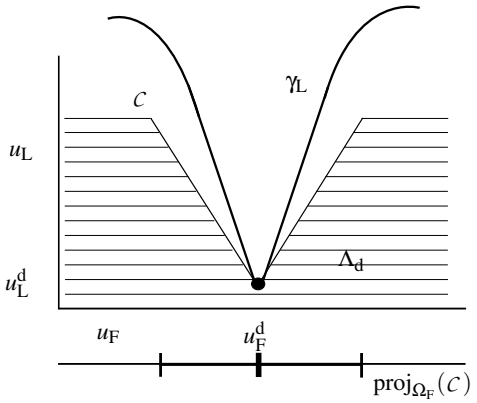
The implications of this approach with respect to finding a game-optimal leader function are analyzed next:

- Uniform grid.

  In case Ω_{F} is simply approximated by an equidistant grid with $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ as the basis grid point and with a grid size of precision δ, it can be easily verified that a point $u_{\mathrm{F}} \in \Omega_{\mathrm{F}}$ is separated at most $\varepsilon := \sqrt{n_{\mathrm{F}}} \cdot \frac{\delta}{2}$ in Euclidean distance from some grid point $u_{\mathrm{F}} \in \Omega_{\mathrm{F}}^{\mathrm{g}}$. If the follower's decision for a given leader function



(a) Case in which no feasible explicit leader function exists. The line *ab* is perpendicular to the follower axis.



(b) Case in which a nonsmooth optimal leader function γ_L exists.

Figure 4.4: *Sublevel set* Λ_d *and associated cone* $C : \Lambda_d \cap C = \{(u_L^d, u_F^d)\}$.

does not coincide with a grid point, there is no guarantee that the algorithm does not return a leader function that intersects with $\Lambda_d \setminus \{(u_L^d, u_F^d)\}$ and in fact the approach could then lead to a pair of decision variables that is far ($\gg \varepsilon$) from the desired pair (u_L^d, u_F^d).

- Non-uniform grid.

  Instead, one can consider a set of non-equidistant grid points that have a higher density in the space $\text{proj}_{\Omega_F}(\Lambda_d)$ then in Ω_F and that in particular includes the follower elements of the vertices of Λ_d in case it represents a polytope. A program-feasible leader function satisfying (4.30) and (4.31) then leads more likely to game-optimality.

Optimality Overall, if there exists a game-optimal nonlinear leader function within the applicable constraints, the gridding approach should be able to find it with a 'sufficiently' dense grid. We therefore consider an adaptive gridding approach in which the precision δ should be refined in case of the evaluation of a program-feasible function $\gamma_L(\cdot)$ from (4.5), (4.7), (4.9), (4.30), (4.31) it turns out that $\min_{u_F \in \Omega_F} J_F(\gamma_F(u_F), u_F) \neq u_F^d$.

Remark 4.3 Instead of evaluating the leader function at each grid point with respect to the sublevel set (i.e., with respect to $J_F(\cdot)$ according to (4.30)), this constraint could be tightened to exclude intersection with the convex hull of Λ_d. This is however only possible provided that $(u_L^d, u_F^d) \in \mathrm{bd}(\mathrm{conv}(\Lambda_d))$; otherwise the approach may not result in any game-optimal leader function. While it does not warrant game-optimality, considering the larger, convex area around Λ_d can reduce the likelihood of an undesired intersection of $\{(\gamma_L(u_F), u_F) \,|\, u_F \in \Omega_F\}$ with Λ_d. Further, this can also lead to the exclusion of any of the available feasible solutions with respect to the constraint $\gamma_L(\Omega_F) \subseteq \Omega_L$ in the available space $\mathbb{R}^{n_L} \setminus \mathrm{proj}_{\Omega_L}(\Lambda_d)$.

Finally, a hybrid approach may be adopted in which the continuous method is considered for $\mathrm{proj}_{\Omega_F}(\Lambda_d)$ while outside this space, a coarse gridding of Ω_F can be adopted to satisfy $\gamma_L(\Omega_F) \subseteq \Omega_L$. ◇

4.5 A Heuristic Approach: Interpolation

The last alternative indirect nonlinear solution approach we propose is that of multivariate interpolation also known as curve or (hyper)surface fitting, either by using piecewise polynomials (splines) or by fitting any given parametrized function [3, 47]. Although we are interested in any such curve or higher-dimensional manifold that does *not* intersect with $\Lambda_d \setminus \{(u_L^d, u_F^d)\}$ (according to C.2), the resulting leader function needs to *pass through* (u_L^d, u_F^d) (according to C.3) and *remain within* Ω_L for all $u_F \in \Omega_F$ (according to C.1). Therefore a set of data points can be selected that satisfy C.1–C.3; an interpolation algorithm fits a function through the elements of this set, which is especially facilitated in case of a function that is piecewise defined.

As it will be illustrated in Section 4.7, interpolation approaches can have significant advantages as regards computational efficiency. At the same time, a significant drawback is the difficulty of interpolation to take care of the requirements C.1 and C.2 as well as to determine an initial set of data points. Analogously to the gridding approach, interpolation is therefore an adaptive approach in which an evaluation of $\min_{u_F \in \Omega_F} J_F(\gamma_L(u_F), u_F)$ has to prove whether the leader function is indeed game-optimal, or whether an adapted set of data points is needed for the derivation of a new leader function. Another disadvantage of interpolation by fitting an involved function structure is that an ill-chosen set of data points may result in an infeasible fit of the specified function. This applies especially in functions involving many parameters, where the minimum number of data points required for interpolation is rather large.

A few elements of the interpolation approach are listed in the following:

Selection of data points One could systematically select data points that are a chosen measure in Euclidean distance away from $\mathrm{bd}(\Lambda_\mathrm{d})$, the boundary of Λ_d. Since the selection of proper interpolation points is very much dependent on the geometric shape of Λ_d and on the location of $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$, the approach is suggested for application to small problems in which $n_\mathrm{L} + n_\mathrm{F} \leq 3$, where graphical representation facilitates selection.

Smoothness In case smoothness does not exhibit a desired property, a game-optimal piecewise affine function is easily derived, i.e., by selecting interpolation points such that a linear interpolating function through these points does not intersect with $\Lambda_\mathrm{d} \setminus \{(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})\}$. Piecewise polynomial interpolation functions can however be subjected to any degree of continuity requirements.

On a high level, the piecewise polynomial (spline) approach can be summarized as follows[3]. For the sake of simplicity of the exposition we consider the case $n_\mathrm{L} = n_\mathrm{F} = 1$; the subscript j for $j = 1, \ldots, n_\mathrm{L}$ can thus be disregarded in the coefficient description $(a_i)_j$. Note, however, that the approach can also be extended to higher dimensions.

Here, S denotes the set of selected data points $\{(u_\mathrm{L}^{(s)}, u_\mathrm{F}^{(s)})\}_{s\in\{1,\ldots,S\}}$ including $(u_\mathrm{L}^\mathrm{d}, u_\mathrm{F}^\mathrm{d})$ and where $(u_\mathrm{L}^{(\hat{s})}, u_\mathrm{F}^{(s)}) \neq (u_\mathrm{L}^{(\bar{s})}, u_\mathrm{F}^{(s)})$ for $\hat{s} \neq \bar{s}$. Further, $(u_\mathrm{L}^{(s)}, u_\mathrm{F}^{(s)})$ and $(u_\mathrm{L}^{(s+1)}, u_\mathrm{F}^{(s+1)})$ are connected by the segment indexed $s \in \{1, \ldots, S-1\}$ and $n_\mathrm{a}^{(s)}$ denotes the degree of the polynomial associated with the segment $s \in \{1, \ldots, S-1\}$.

$$\text{To find} \quad \text{for } s = 1, \ldots, S-1: \tag{4.32}$$

$$a_i^{(s)} \in \mathbb{R}^{n_\mathrm{F}}, \qquad i = 1, \ldots, n_\mathrm{a}^{(s)}, \tag{4.33}$$

$$\text{such that} \quad \gamma_\mathrm{L}^{(s)}(u_\mathrm{F}^{(s)}) = u_\mathrm{L}^{(s)}, \tag{4.34}$$

$$\gamma_\mathrm{L}^{(s)}(u_\mathrm{F}^{(s+1)}) = u_\mathrm{L}^{(s+1)}, \tag{4.35}$$

$$\gamma_\mathrm{L}^{(s)}(u_\mathrm{F}) = \sum_{i=1}^{n_\mathrm{a}^{(s)}} \begin{bmatrix} \left(((a_i^{(s)}))_1\right)^i (u_\mathrm{F})^{i-1} \\ \vdots \\ \left(((a_i^{(s)}))_{n_\mathrm{L}}\right)^i (u_\mathrm{F})^{i-1} \end{bmatrix}. \tag{4.36}$$

In this program, only C^0 continuity is guaranteed. Here, no constraints are considered that can enforce C.1 nor C.2; a new data point should be selected and new segments should be created if the conditions are violated.

[3] Matlab's Curve Fitting toolbox provides several fitting options next to the spline option to solve (4.33)–(4.36). Depending on the specified function structure, in addition, C^1 and C^2 continuity can be incorporated.

4.6 Discussion on the Complexity and the Selection of Basis Functions

4.6.1 Complexity

The reverse Stackelberg game, defined in its general direct form (2.2) without the specification of a desired leader equilibrium, can be proven to be at least NP-hard. Recall from Chapter 2 that this follows from equivalence of the original Stackelberg game [196] with the strongly NP-hard linear bilevel programming problem [39], where the original Stackelberg game can be perceived as a special case of the reverse Stackelberg game for leader functions $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \{u_{\mathrm{L}}^{\mathrm{d}}\}$.

While the selection of a desired leader optimum reduces the number of solutions (if any exist), without the specification of a particular class Γ_{L} of leader functions, the search space for possible solutions is still very large. Moreover, the required computation time of the proposed solution methods depends strongly on the problem specifications, i.e., on the follower's objective function $f_{\mathrm{F}}(\cdot)$ or the shape of the sublevel set Λ_{d}, and on the choice of the type and the number of basis functions. The – in general – nonconvex nonlinear programming problems of the indirect basis function approaches are also NP-hard.

Nonetheless, for the continuous multilevel approach, the construction (4.10)–(4.12) can be simplified in case the decision space Ω_{L} is subject to simple box constraints with the upper and lower bounds for the i-th component of $u_{\mathrm{L}} \in \Omega_{\mathrm{L}}$ denoted by respectively $u_{\mathrm{L},i}^{\mathrm{low}}, u_{\mathrm{L},i}^{\mathrm{upp}}, i = 1, \ldots, n_{\mathrm{L}}$. In that case, (4.18) can be replaced by the following constraints in which only the following two optimizations are performed for $i = 1, \ldots n_{\mathrm{L}}$:

$$\begin{aligned} \min_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} \gamma_{\mathrm{L},i}(u_{\mathrm{F}}) &\geq u_{\mathrm{L},i}^{\mathrm{low}}, \\ \max_{u_{\mathrm{F}} \in \Omega_{\mathrm{F}}} \gamma_{\mathrm{L},i}(u_{\mathrm{F}}) &\leq u_{\mathrm{L},i}^{\mathrm{upp}}. \end{aligned} \tag{4.37}$$

In case Ω_{L} is a convex polytope, for $\gamma_{\mathrm{L}}(u_{\mathrm{F}}) \notin \Omega_{\mathrm{L}}$ it is easy to verify that determining the distance between $\gamma_{\mathrm{L}}(u_{\mathrm{F}})$ and the convex polytope Ω_{L} – replacing (4.18) – involves a quadratic programming problem. In [185] the distance between two convex polytopes is computed by solving a similar quadratic optimization problem with linear constraints, while in the specific case of three dimensions, an implementation that is approximately linear in the total number of vertices is adopted in [72]. Further, in case $f_{\mathrm{F}}(\cdot)$ is piecewise affine or in case it is approximated by a piecewise affine function, it can be shown using similar methods as in [21, 50], that the optimization of $d_1((\gamma_{\mathrm{L}}(u_{\mathrm{F}}), u_{\mathrm{F}}), \Lambda_{\mathrm{d}})$ in (4.14) reduces to a mixed-integer linear programming problem. Such an optimization program still belongs to the class of NP-hard problems [68], but has the advantage that a global optimum is found at once whereas nonconvex programming approaches require an *a priori* unknown number of iterations without the guarantee that a global optimum is reached.

4.6.2 Discussion on the Selection of Basis Functions

In order to reduce the computational complexity, for all solution approaches described in this chapter, a selection of a leader function structure needs to be made, e.g., a subset of basis functions should be considered, in which case the representation is no longer a universal function approximator. The choice of a set of basis functions depends on the requirements of the setting to which a reverse Stackelberg approach is applied.

First of all, the basis functions should be *sufficiently rich* to include a game-optimal leader function for the given problem instance, while it should not lead to a too large *computational burden* due to either a large number of coefficients or due to the existence of many local optima complicating the minimization of $\jmath_{\mathrm{F}}(\gamma_{\mathrm{L}}(\cdot),\cdot)$.

Next to the complexity of the leader function, a motivation for studying various leader function structures is to be able to influence the likelihood of follower players responding rationally, i.e., considering *robustness* of the leader function w.r.t. suboptimal behavior of the follower. Such deviations from the optimal follower response $u_{\mathrm{F}}^{\mathrm{d}}$ as has also been considered in Section 3.6 can occur due to, e.g.,

- the stochastic nature of the follower's objective function: some parameters of $\jmath_{\mathrm{F}}$ may be uncertain, e.g., in case $\jmath_{\mathrm{F}}(\cdot)$ depends on a state variable that is subject to measurement noise.

- incomplete information of the leader player concerning $\jmath_{\mathrm{F}}, \Omega_{\mathrm{F}}$, leading to the necessity to make an estimate, e.g., of the missing parameters. A simple example is the case in which there is a limit in the precision with which the follower can optimize, i.e., when the follower adopts a discretized set of decision variables that does not include $u_{\mathrm{F}}^{\mathrm{d}}$.

In such cases, deviations can occur if the (game-optimal) leader function passes close by the boundary of the follower's sublevel set Λ_{d}. This can be influenced by the choice of basis functions, depending on the geometric shape of Λ_{d} and on the tightness of bounds on Ω_{L} that are relevant for condition C.1 to hold. In other words, in case polynomial leader functions are considered, high-order polynomials may be required when Λ_{d} is highly nonconvex and if the distance between the boundary of $\mathrm{proj}_{\Omega_{\mathrm{L}}}(\Lambda_{\mathrm{d}})$ and the boundary of Ω_{L} is small in some coordinate directions, while lower-order polynomials are sufficient in case Λ_{d} is convex and C.1 is easily satisfied due to a large codomain Ω_{L} relative to a much smaller subset $\mathrm{proj}_{\Omega_{\mathrm{L}}}(\Lambda_{\mathrm{d}})$. Instead of adopting a feasibility objective, the basis function coefficients can be determined based on any desired optimization criterion, e.g., to take into account such robustness and sensitivity criteria.

Here, it could be interesting not only to take into account the sensitivity of the follower to deviations from $u_{\mathrm{F}}^{\mathrm{d}}$ when adopting different leader functions, but also to consider the associated deterioration of the leader's objective function value $\jmath_{\mathrm{L}}\left(\gamma_{\mathrm{L}}(u_{\mathrm{F}}^{\mathrm{dev}}), u_{\mathrm{F}}^{\mathrm{dev}}\right)$ due to such a deviation $u_{\mathrm{F}}^{\mathrm{dev}} \neq u_{\mathrm{F}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{dev}} \in \Omega_{\mathrm{F}}$. This will be considered as a next step of research.

Another criterion for the choice of a particular leader function structure is the *smoothness* of such a function. For instance, in multilevel control settings with

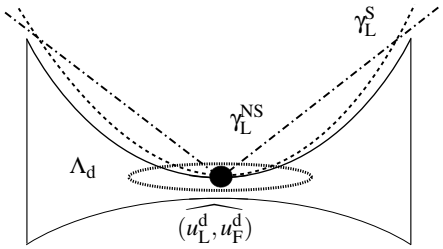


Figure 4.5: *Illustration of the effect of smoothness in case the 'true' desired solution* (u_L^d, u_F^d) *lies in an ellipse around the estimated solution* $(\widehat{u_L^d, u_F^d})$.

precise instruments and an accurate knowledge of the desired equilibrium (u_L^d, u_F^d), nonsmooth optimal leader functions can be applied, whereas in settings where the true equilibrium may be in a neighborhood of (u_L^d, u_F^d), the control actions may need to be steered with a smooth leader function in order to prevent from a large deterioration of the objective function value; see Fig. 4.5 for an illustration. There, $(\widehat{u_L^d, u_F^d})$ is the assumed desired leader equilibrium, while the true value is within an ellipsoidal set. However, such analyses strongly depend on the geometry of the sublevel set Λ_d.

For different types of orthonormal basis functions and their properties, please refer to [96]. In principle, also combinations of basis functions can be considered and the basis functions can be further parametrized; such extensions are left for more elaborate analysis of possible leader function structures.

4.7 Worked Example

In this section, the newly proposed multilevel and grid-based basis functions approaches and the interpolation method as well as the existing methods for computing a leader function with a nonlinear structure are applied to a static reverse Stackelberg game. The aim of this comparison is to evaluate the computation time differences as well as the gap from optimality in case one of the heuristic approaches is used.

4.7.1 Set-up

For both the leader as well as the follower objective functional, the Rosenbrock as well as the extended Rosenbrock function is adopted. Recall from Chapter 3 (Example 3.23) that the nonconvex Rosenbrock function [165] is often used to illustrate the performance of optimization algorithms. In terms of the decision variables of the reverse Stackelberg game it can be written as follows:

$$f(u_L, u_F) = (1-u_L)^2 + 100(u_F - u_L^2)^2, \tag{4.38}$$

with $u_\mathrm{F}, u_\mathrm{L}$ scalar, as depicted in Fig. 3.6 together with several level curves. In this case, we apply (4.38) for $\jmath_\mathrm{F}(\cdot)$ and with reversed signs for the leader:

$$\jmath_\mathrm{L} = (1+u_\mathrm{L})^2 + 100(u_\mathrm{F}+u_\mathrm{L}^2)^2. \tag{4.39}$$

In the three-dimensional case, both the cases of $\Omega_\mathrm{L} \subset \mathbb{R}^2, \Omega_\mathrm{F} \subset \mathbb{R}$ and of $\Omega_\mathrm{L} \subset \mathbb{R}, \Omega_\mathrm{F} \subset \mathbb{R}^2$ are considered in order to show possible leader functions as a one-dimensional curve or as a surface, respectively. For these higher dimensions we adopt variations of the extended Rosenbrock function (3.23) [49], i.e.:

$$\jmath_\mathrm{F}\left(u_{\mathrm{L},1}, u_{\mathrm{L},2}, u_\mathrm{F}\right) = (1-u_{\mathrm{L},1})^2 + 100(u_{\mathrm{L},2}-u_{\mathrm{L},1}^2)^2 + (1-u_{\mathrm{L},2})^2 + 100(u_\mathrm{F}-u_{\mathrm{L},2}^2)^2, \tag{4.40}$$

$$\jmath_\mathrm{F}\left(u_\mathrm{L}, u_{\mathrm{F},1}, u_{\mathrm{F},2}\right) = (1-u_\mathrm{L})^2 + 100(u_{\mathrm{F},1}-u_\mathrm{L}^2)^2 + (1-u_{\mathrm{F},1})^2 + 100(u_{\mathrm{F},2}-u_{\mathrm{F},1}^2)^2, \tag{4.41}$$

$$\jmath_\mathrm{L}\left(u_\mathrm{L}, u_{\mathrm{F},1}, u_{\mathrm{F},2}\right) = 100\left(0.5u_{\mathrm{F},1} + u_\mathrm{L}^2)^2 + (1+2u_\mathrm{L}\right)^2 + 100\left(0.5u_{\mathrm{F},2} + u_{\mathrm{F},1}^2)^2 + (1+2u_{\mathrm{F},1}\right)^2, \tag{4.42}$$

$$\jmath_\mathrm{L}\left(u_{\mathrm{L},1}, u_{\mathrm{L},2}, u_\mathrm{F}\right) = 100\left(0.5u_{\mathrm{L},2} + u_{\mathrm{L},1}^2)^2 + (1+2u_{\mathrm{L},1}\right)^2 + 100\left(0.5u_\mathrm{F} + u_{\mathrm{L},2}^2)^2 + (1+2u_{\mathrm{L},2}\right)^2. \tag{4.43}$$

We have chosen to adopt a (piecewise) cubic spline interpolator as well as a special fifth-order polynomial for the basis function cases and the evolutionary approaches, implying the following leader functions for the 2D (4.44) and 3D (4.45), (4.46) cases as realizations of (4.9):

$$\gamma_\mathrm{L}(u_\mathrm{F}) = \sum_{i=0}^{5} a_i \cdot (u_\mathrm{F})^i, \qquad u_\mathrm{L} \in \mathbb{R}, u_\mathrm{F} \in \mathbb{R}, \tag{4.44}$$

$$\gamma_\mathrm{L}(u_\mathrm{F}) = \sum_{i=0}^{5} \begin{bmatrix} (a_i)_1 \cdot (u_\mathrm{F})^i \\ (a_i)_2 \cdot (u_\mathrm{F})^i \end{bmatrix}, \qquad u_\mathrm{L} \in \mathbb{R}^2, u_\mathrm{F} \in \mathbb{R}, \tag{4.45}$$

$$\gamma_\mathrm{L}(u_\mathrm{F}) = \sum_{i=0}^{5} (a_i)_1 (u_{\mathrm{F},1})^i + \sum_{j=6}^{11} (a_j)_2 (u_{\mathrm{F},2})^{j-6}, \qquad u_\mathrm{L} \in \mathbb{R}, u_\mathrm{F} \in \mathbb{R}^2, \tag{4.46}$$

where the respective decision spaces in the first two cases are restricted to $u_\mathrm{F} \in [-2,2]$, $u_\mathrm{L}, u_{\mathrm{L},1}, u_{\mathrm{L},2} \in [-5,5]$ and the coefficient values are constrained to $(a_i)_j \in [-5,5]$, $i = 1,\ldots,n_\mathrm{a}, j = 1,\ldots,n_\mathrm{L}$, $n_\mathrm{a} \in \{6,12\}$. In order to ensure the existence of a game-optimal leader function, in the last case these bounds are relaxed to $u_{\mathrm{F},1}, u_{\mathrm{F},2} \in [-5,5]$, and coefficient values to $a_i \in [-10,10], i = 1,\ldots,n_\mathrm{c}$.

As discussed in Section 4.4.2, the simple, uniform gridding approach may not reach the leader's desired equilibrium for an arbitrary precision. In this example, the gridding approach is evaluated for the precision $\delta = 0.01$ in the cases $n_\mathrm{F} = 1$ and $\delta = 0.05$ in the case $n_\mathrm{F} = 2$. These precisions were found to be sufficient, i.e., they solved the problem to optimality. For the scenario $u_\mathrm{L} \in \mathbb{R}, u_\mathrm{F} \in \mathbb{R}$, the results for both a sufficient precision and an unnecessarily fine precision are included. An

evaluation of the computation time and optimality for different degrees of precision can be found in Table 4.5 and Fig. 4.8 below.

For the evolutionary approaches, the Matlab neural network fitting toolbox and the genetic algorithm tool of the global optimization toolbox are used. Standard settings have been adopted. The evolutionary algorithms are only used for the main optimization runs of these methods as described in Section 4.3. For the purpose of a comparison of computation times, constrained nonlinear programming of Tomlab's SNOPT optimization toolbox is used for all of the following nonconvex optimization problems[4]:

(i) the optimization of $\jmath_{\mathrm{F}}(\cdot)$ for each combination of coefficients of $\gamma_{\mathrm{L}}(\cdot)$ that is evaluated in the neural network approach;

(ii) the optimization of the neural network to obtain the best leader function coefficient values for the leader;

(iii) the computation of the follower's response for a population of leader function coefficient values, needed to obtain the genetic algorithm's fitness function value;

(iv) the computation of the follower's response to the functional $\gamma_{\mathrm{L},i}$ for each leader decision element $i = 1,\ldots,n_{\mathrm{L}}$, needed in the constraints (4.37) of the indirect approaches;

(v) the derivation of the auxiliary distance expression (4.14) within the indirect approaches;

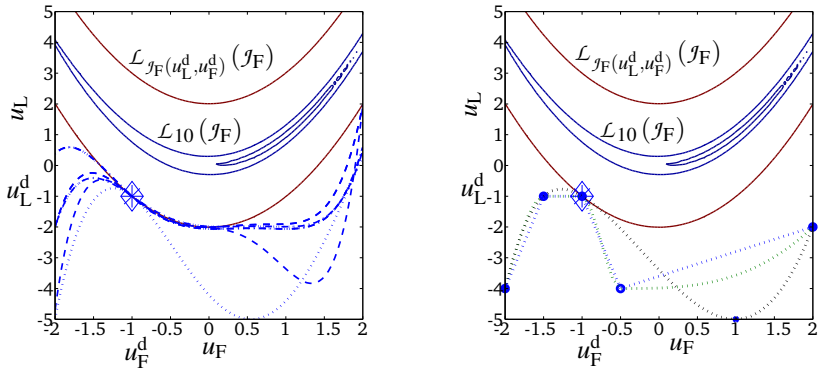
(vi) the computation of $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ in the indirect approaches.

To prevent local optima, in these cases, ten iterations with different initial points were applied that led to a (local) optimum.

4.7.2 Results and Discussion

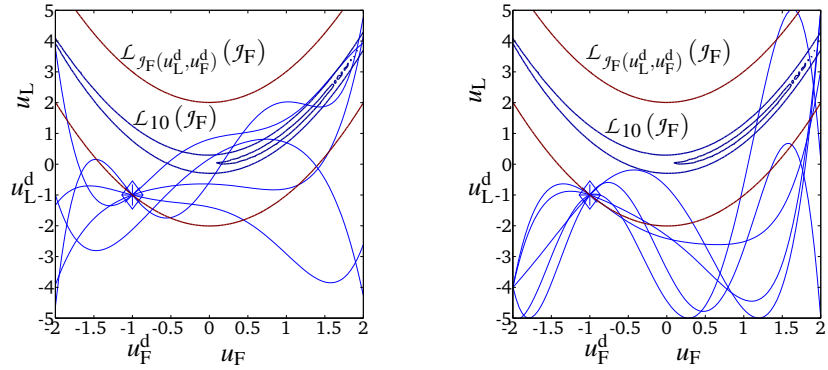
In Fig. 4.6 and Fig. 4.7 several leader functions computed by the discussed algorithms are plotted for the 2-dimensional case, while leader functions derived using the indirect methods in the 3-dimensional case are depicted in Fig. 4.9 (curve) and 4.10 (surface). It can be seen that for the indirect methods, the curves indeed remain within the constrained decision space Ω_{L} and outside $\Lambda_{\mathrm{d}} \setminus \{(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\}$, which is not the case for the evolutionary approaches. The extent of the deterioration in the leader objective function values can be found in Table 4.3, where it should be emphasized that this performance should be regarded in the context of a trade-off with the computation times. Further, enforcing the leader function to pass through $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ (the 'desired equilibrium constraint', refer to Table 4.4) did improve the average and minimum value of $\jmath_{\mathrm{L}}(\gamma_{\mathrm{L}}(\cdot),\cdot)$ for the neural network approach, but not for the genetic algorithm.

[4] It should be noted that alternative to using these regular constrained solvers such as SQP-based, interior-point or active-set methods, also here, evolutionary approaches could be used.



(a) Continuous (dashed) and gridding (dotted) (b) Interpolation

Figure 4.6: *Leader functions resulting from the indirect approaches: Case* $u_F \in \mathbb{R}, u_L \in \mathbb{R}$.



(a) Direct genetic algorithm approach (b) Direct neural network approach with equilibrium constraint

Figure 4.7: *Leader functions resulting from the direct approaches: Case* $u_F \in \mathbb{R}, u_L \in \mathbb{R}$.

In order to give an indication of the computational requirements, Tables 4.2 and 4.3 show the computation time[5] for deriving the leader functions. As the computation time differed for consecutive runs, the average, minimal, and maximal values of 10 runs are indicated as well as the standard deviation. The proposed methods for deriving optimal nonlinear leader functions can now be compared to genetic and neural networks approaches, where the desired optimum is not computed *a priori*.

As mentioned before, the results of the genetic algorithm are obtained under the standard settings indicated by the Matlab documentation. This holds also for

[5]These CPU times were obtained adopting the 64-bit Matlab 8 (R2012b) Tomlab (SNOPT) environment on a Linux PC with a 3GHz Intel Core Duo processor and 4Gb RAM.

[6]Under the standard genetic algorithm settings, the 3D surface case did not result in any feasible solution after 25 runs that each took on average 850 s.

Table 4.2: *Summary of computational results: Indirect methods.*

Scenario	Dimension	Indirect method	Mean CPU time [s]	min [s]	max [s]	st.dev. [s]
Case 1	2D	Continuous multilevel	49.2	46.9	57.7	3.19
	2D	Gridding $\delta = 0.01$	0.695	0.419	1.88	0.480
	2D	Gridding $\delta = 0.001$	6.92	6.47	7.33	0.208
	2D	Interpolating cubic spline (5 points)	$2.60{\cdot}10^{-3}$	$2.30{\cdot}10^{-4}$	$2.42{\cdot}10^{-2}$	$7.60{\cdot}10^{-3}$
	2D	Computation of $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$	0.206	0.192	0.216	$7.20{\cdot}10^{-3}$
Case 2	3D curve	Continuous multilevel	55.6	34.2	95.8	19.6
	3D curve	Gridding $\delta = 0.01$	1.416	1.06	2.03	0.410
	3D curve	Interpolating cubic spline (4 points)	$1.19{\cdot}10^{-2}$	$4.90{\cdot}10^{-3}$	$9.96{\cdot}10^{-2}$	$2.21{\cdot}10^{-2}$
	3D curve	Computation of $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$	0.293	0.269	0.313	$1.39{\cdot}10^{-2}$
Case 3	3D surface	Continuous multilevel	55.6	36.2	95.5	24.1
	3D surface	Gridding $\delta = 0.05$	37.0	30.8	45.4	7.09
	3D surface	Interpolating surface (12 points)	0.312	0.207	0.481	$9.62{\cdot}10^{-2}$
	3D surface	Interpolating piecewise cubic surface (8 points)	$4.89{\cdot}10^{-2}$	$2.30{\cdot}10^{-2}$	$8.69{\cdot}10^{-2}$	$2.47{\cdot}10^{-2}$
	3D surface	Computation of $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$	0.248	0.230	0.271	$1.16{\cdot}10^{-2}$

Table 4.3: *Summary of computational results: Direct methods.*

Scenario	Dimension	Direct method	Mean CPU time [s]	min [s]	max [s]	st.dev. [s]	Mean $\jmath_L$	min	max	st.dev.
Case 1	2D	Genetic Algorithm	752	545	$1.11{\cdot}10^3$	167	0.783	0.494	0.940	0.159
	2D	Neural Network (sample size 4^6)	389	357	489	39.1	$2.30{\cdot}10^3$	40.9	$6.12{\cdot}10^3$	$2.75{\cdot}10^3$
	2D	Neural Network (sample size 5^6)	$1.22{\cdot}10^3$	$1.10{\cdot}10^3$	$1.36{\cdot}10^3$	111	$2.08{\cdot}10^3$	1.12	$6.07{\cdot}10^3$	$2.69{\cdot}10^3$
Case 2	3D curve	Genetic Algorithm	997	824	$1.22{\cdot}10^3$	165	115	2.21	247	129
Case 3	3D surface[6]	Genetic Algorithm	–	–	–	–	–	–	–	–

Table 4.4: *Summary of computational results: Evolutionary algorithms with desired equilibrium constraint.*

Scenario	Dimension	Direct method	Mean CPU time [s]	min [s]	max [s]	st.dev. [s]	Mean $\jmath_L$	min	max	st.dev.
Case 1	2D	Genetic Algorithm	955	608	$1.88{\cdot}10^3$	406	10.6	1.00	38.0	14.8
	2D	Neural Network (sample size 3^6)	148	72.1	220	45.6	$1.27{\cdot}10^3$	0.983	$6.09{\cdot}10^3$	$2.53{\cdot}10^3$
	2D	Neural Network (sample size 4^6)	383	346	508	49.4	$1.17{\cdot}10^3$	0.441	$6.08{\cdot}10^3$	$2.43{\cdot}10^3$
	2D	Neural Network (sample size 5^6)	$1.23{\cdot}10^3$	$1.10{\cdot}10^3$	$1.46{\cdot}10^3$	128	658	0.935	$5.85{\cdot}10^3$	$1.83{\cdot}10^3$

Table 4.5: *CPU times for different gridding precisions.*

	Dimension	Method	CPU time
Case 1	2D	Gridding $\delta = 0.1$	1.3485 s
	2D	Gridding $\delta = 0.01$	1.7664 s
	2D	Gridding $\delta = 0.001$	6.7659 s
Case 2	3D curve	Gridding $\delta = 0.1$	1.1548 s
	3D curve	Gridding $\delta = 0.01$	1.6780 s
	3D curve	Gridding $\delta = 0.001$	9.5181 s

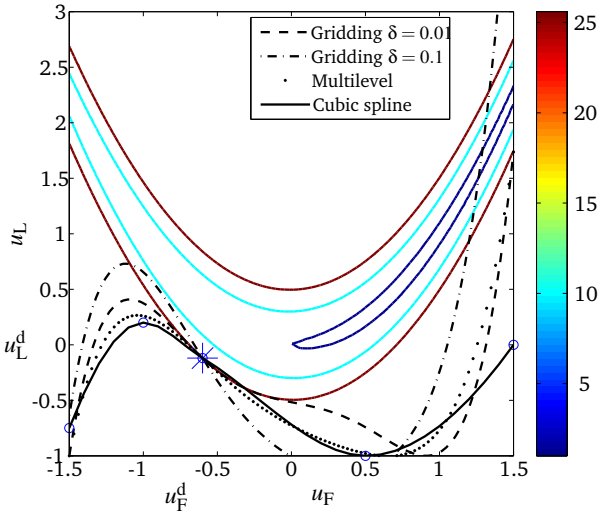
the training and validation in the neural network approach, where different sizes of sample sets (i.e., training plus validation samples) of the leader function coefficients were evaluated. For the neural network approach, it should be noted that the evaluation of different coefficient combinations takes a significant amount of time, which proved to be a bottleneck in our example. The inclusion of too few coefficient values in the samples may however result in a bad performance. In particular, in the 3D surface case, evaluating only 3 values for each coefficient does not give a sufficient representation of the possible range of coefficient values, while the follower optimization problem has to be solved already an intractable number (3^{12}) of times. The 3D cases are therefore not included in the comparison.

As it is confirmed in the results for the 2D case, the larger the number of coefficient value evaluations, the lower the achieved leader objective function value. For a satisfactory performance of the neural network approach, parallel evaluations are therefore necessary, which were not performed in any of the approaches of this example. Instead, one may revert to leader functions with fewer coefficients. However, those may not be sufficiently rich to lead to a game-optimal leader function. Further, it is noteworthy that while the minimum leader objective function values obtained by the evolutionary algorithms are close to optimal ($J_L(u_L^d, u_F^d) = 0$), the standard deviation of both computation time and objective function values is significant.

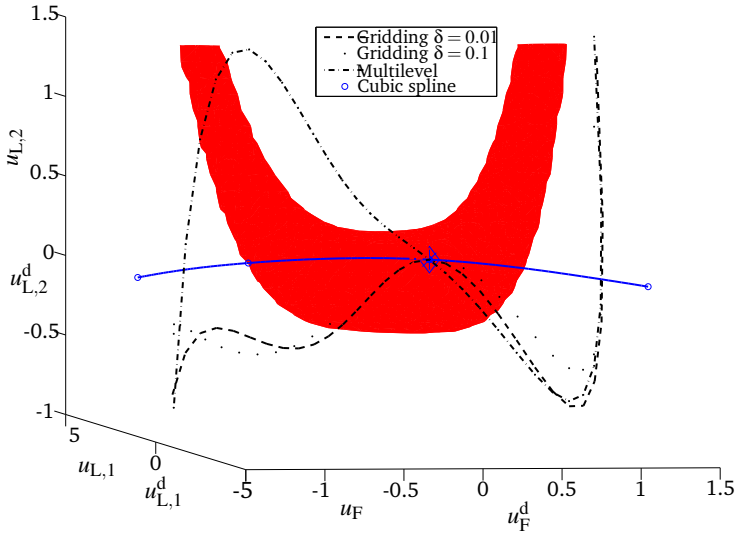
Furthermore, the interpolating surface methods required significantly less computation time than the other algorithms. In the 3D surface case, both piecewise cubic interpolation as well as a fitting based on the polynomial (4.46) adopted in the other methods is used. In order to find a surface satisfying (4.46), at least 12 data points had to be determined; in case of piecewise (cubic) interpolation fewer points can be specified.

Finally, Table 4.5 gives another indication of computation times for different gridding precisions for the worked example. Here, a gridding precision of $\delta = 0.1$ turned out to be insufficient, as is also plotted in Fig. 4.8. There it can be seen that the curves indeed remain within the constrained decision space Ω_L and outside $\Lambda_d \setminus \{(u_L^d, u_F^d)\}$, except in case gridding was applied with $\delta = 0.1$; for that case, the computed curves intersect with Λ_d. Further, it can be observed that increasing the gridding precision to $\delta = 0.001$ leads to a relatively large increase in time, while an optimal curve is already obtained for $\delta = 0.01$.

Overall, the main difference between the direct evolutionary approaches and our proposed, indirect methods is that in the latter approaches, the leader function



(a) $u_F \in \mathbb{R}, u_L \in \mathbb{R}$



(b) $u_L \in \mathbb{R}^2, u_F \in \mathbb{R}$

Figure 4.8: *Nonlinear leader functions for sufficient and insufficient gridding precisions.*

is constrained based on the sublevel set Λ_d for a certain equilibrium (u_L^d, u_F^d), which can be derived directly from knowledge of $J_F(\cdot)$. This excludes the need inherent in the direct methods to repetitively solve the nonconvex follower optimization problem for every set of leader function coefficients, where it should also be noted that in each of those problems, there is a risk of selecting a local optimum based on which further optimization is performed. In those direct approaches, the main aim

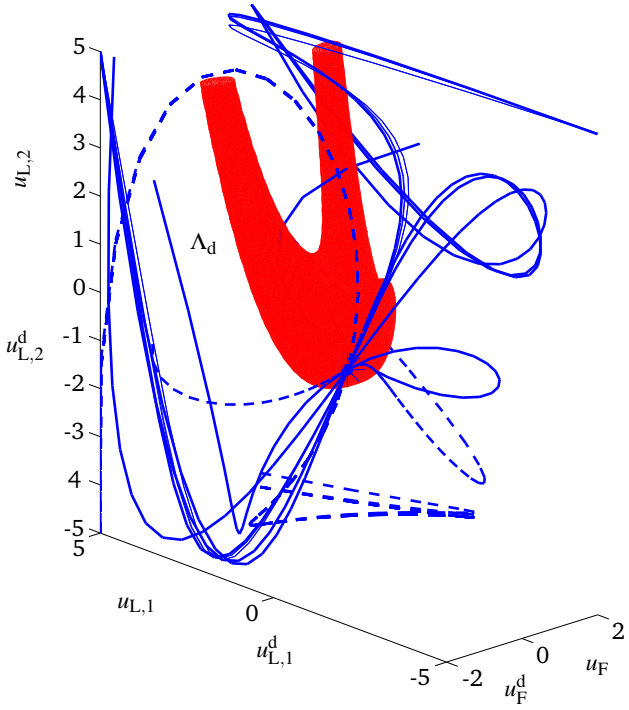


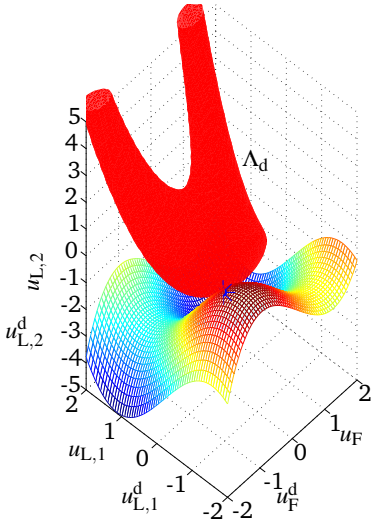
Figure 4.9: *Game-optimal nonlinear leader functions from the gridding (solid) and the continuous (dashed) approach: Case* $u_L \in \mathbb{R}^2, u_F \in \mathbb{R}$.

is to approximate the – in general complex – relationship between leader function coefficients and the resulting leader objective function value, with the follower's optimization problem as an intermediate step. In cases in which many coefficients are needed for the leader function to be game-optimal, this difference between indirect and evolutionary approaches shows clearest.

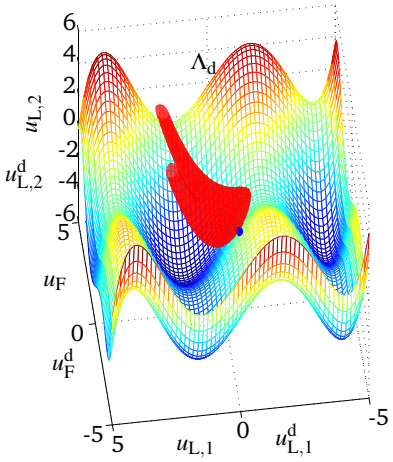
4.8 Discussion

In this chapter, approaches to systematically solve the reverse Stackelberg game have been proposed, their properties have been analyzed, and their performance has been compared to existing direct methods based on evolutionary algorithms. Available solution approaches that derive optimal affine leader functions have thus been extended to include more diverse, nonlinear function structures under which the leader's desired equilibrium can be reached.

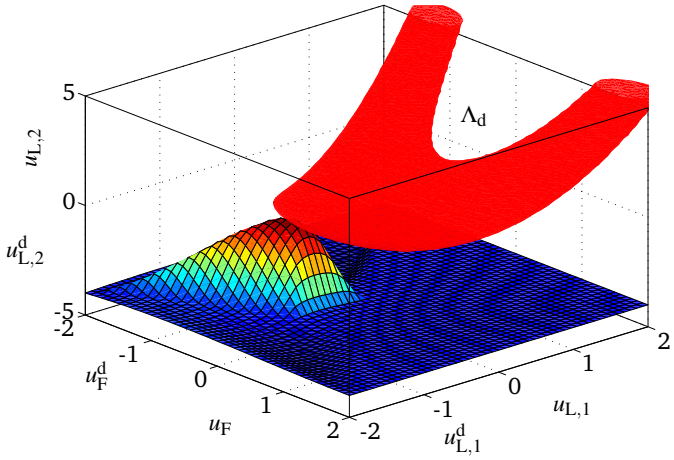
In particular, a continuous multilevel optimization problem leading to an optimal solution for the leader as well as a gridding approach have been developed next to an interpolating spline approach. In a case study, these indirect approaches in which



(a) Gridding approach



(b) Continuous approach



(c) Piecewise cubic interpolation

Figure 4.10: *Game-optimal nonlinear leader functions: Case* $u_L \in \mathbb{R}, u_F \in \mathbb{R}^2$.

the leader's desired equilibrium is computed *a priori* have been compared to direct methods based on the evolutionary neural network and genetic algorithms that have been proposed in earlier literature. In particular, the required CPU times as well as the level of optimality of the methods have been compared for a varying number of decision variables and leader function coefficients.

Examples mostly deal with two- and three-dimensional decision spaces for the sake of exposition. For these cases that are on the one hand relatively low-dimensional but on the other hand deal with relatively rich leader functions that

involve many coefficients, the indirect approaches turned out to consume much less computational effort, while the direct approaches resulted in significantly less desirable solutions for the leader under the standard parameters of the evolutionary algorithms used. This difference could be explained by the fact that in comparison to the indirect methods, the underlying aim of the latter, direct approaches to approximate the relationship between leader function coefficients and resulting leader objective function values is complex, i.e., it involves solving the nonconvex follower optimization problem repetitively for different sets of selected leader function coefficients. While tuning of the evolutionary algorithms used within the direct approaches is likely to reduce this gap to optimality, in the indirect methods such tuning is not indispensable in order to yield an optimal leader function, which can prove to be beneficial especially in situations with frequently changing problem settings.

With respect to the current chapter, in future research, the qualitative as well as quantitative performance of the different algorithms could be evaluated for various leader function parameterizations in more diverse, higher-dimensional problem settings with varying shapes of the follower's sublevel set. In particular, attention could be paid to the scalability of the algorithms, focusing on methods for reducing the computational requirements of the solution methods, e.g., by performing more tuning and by applying parallel computations as well as by distributed algorithms. Finally, the considered approaches can also be combined, e.g., by adopting a neural network to represent a leader function as a replacement of the basis functions representation. The indirect solution approaches could then be applied in order to find an optimal leader function in the form of a neural network.

Chapter 5

Reverse Stackelberg games in Dynamic Route Guidance

In the previous two chapters, systematic solution methods for the general reverse Stackelberg game have been considered by characterizing optimal affine leader functions (Chapter 3) and through the computation of optimal leader functions of a nonlinear structure (Chapter 4). In the current chapter, novel schemes for applying the reverse Stackelberg game are proposed in the context of traffic control. Here, the previously considered solution methods can be applied to obtain optimal leader functions. In particular, we present methods in which the reverse Stackelberg game is adopted to close the gap between user-optimal and globally system-optimal behavior. It is shown how a reverse Stackelberg game approach can enable traffic authorities to induce drivers to follow routes that are computed to reach a system-optimal distribution of traffic on the available routes of a freeway, e.g., to minimize the total time spent (TTS) of traffic in the network, as well as to reduce traffic emissions in urban traffic networks.

5.1 Introduction

Two significant and correlated problems in everyday traffic are congestion and environmental pollution by vehicular emissions. These problems can be said to be caused by the inability of individual travelers in current traffic systems to communicate with each other and to reach a consensus on how to behave in order to collectively improve the traffic situation. At the same time it may be impossible to improve the conditions like the travel time and the vehicular emissions of some individual travelers without creating a worse situation for other drivers, thus reaching an impasse.

Two well-known notions that are relevant in this traffic context and that can at the same time be placed in a game-theoretical context are stated next, viz., the *Wardrop equilibrium principles* and the *Braess paradox*. These notions involve the disparity between a user optimum and a (global) system optimum.

Definition 5.1 Wardrop Equilibrium Principles [198]
The following two excerpts summarize the equilibrium traffic state from the perspective of the individual drivers and a traffic authority, respectively.

(I) “The journey times on all the routes actually used are equal, and less than

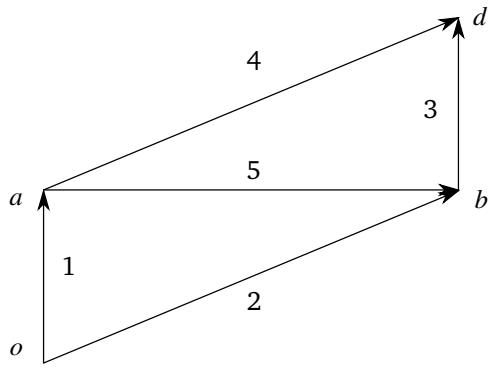


Figure 5.1: *Illustration of the Braess paradox.*

those which would be experienced by a single vehicle on any unused route." [198, p. 345]

In this case in which drivers act noncooperatively and adopt their individual best, i.e., shortest, route the so-called *user equilibrium* of traffic flow is achieved.

(II) "The average journey time is a minimum." [198, p. 345]

This so-called *system-optimal equilibrium* is achieved in case drivers act according to the objective of minimizing the total time that all vehicles together spend in a given traffic system (TTS [veh·h]). ◇

A related game-theoretical notion is described by Braess:

Example 5.2 The Braess Paradox [25]

" ... an extension of the road network may cause a redistribution of the traffic which results in longer individual running times." [25, p. 258]

The following example (depicted in Fig. 5.1) illustrates this paradox: suppose the average travel times on the links can be described by the following linear functions of the link flow q (in number of vehicles per time unit): $t_1(q) = t_3(q) = 10q$, $t_2(q) = t_4(q) = 50+q$, $t_5(q) = 10+q$. When a total flow demand (denoted by Q) of (i) $Q = 2$, (ii) $Q = 6$, and (iii) $Q = 20$ is led from o to d, these result in an optimal total travel time if these are (i) allocated fully to (o,a,b,d) and (ii) and (iii) divided equally between (o,a,d) and (o,b,d), respectively. It can be verified that in case (i) and (iii) the resulting travel times for these optimal route(s) from o to d are also the shortest amongst the times on alternative routes. However, given that each driver will adopt the shortest path at each node according to Wardrop's first principle of the user equilibrium, in case (ii) with $Q = 6$, traffic will deviate to the shortest route (o,a,b,d) and disturb the optimality of the route travel time (o,a,d) and (o,b,d). Since the travel time on these previously fastest routes (o,a,d), (o,b,d) has now been increased, no individual driver will deviate from the suboptimal route (o,a,b,d), resulting in a suboptimal equilibrium. ◇

In this chapter we address the disparity as just described between a system-optimal traffic distribution and the behavior of traffic participants following their individual, non-aligned interests, while being subject to insufficient information on the current as well as on the predicted traffic situation in a dynamic setting. We show how the road authority can in fact control the traffic behavior by collecting the (urgency) priorities of the drivers and by providing each individual with monetary incentives in return for a certain restriction in its traffic behavior, e.g., by following an imposed speed limit or route. In other words, we propose a hierarchical strategy based on the framework of the reverse Stackelberg game in which the objectives of travelers as well as of the road authority can be aligned. First we let the traffic authority (the leader) communicate her knowledge of the network to the drivers (the followers) via the proposition of a leader function that maps the follower's decision variables to positive or negative monetary incentives. In order to collect both the origin-destination information of the drivers and their individual values of travel time, on-board computers are assumed to be installed. Based on this information and on information regarding the current traffic state and the predicted future travel demand, a system-optimal assignment or distribution of traffic over the available routes can be computed.

In order to assign traffic over the available routes according to this optimal distribution, a mapping of monetary incentives should be proposed under which the rational travelers adopt those values of their decision variable – such as the predicted route travel time or route splitting rate – that is associated with a route that is desired by the leader. Evidently, the computation of such an optimal leader function is a crucial element of the reverse Stackelberg game. It can be shown that in the proposed applications the derivation of an optimal leader function is relatively straightforward, given the desired optimal distribution and the drivers' value of travel time.

In particular, three variants of the reverse Stackelberg game in a traffic context are proposed, where the follower decision variables are taken to be:

(1a) the route splitting rates of a homogeneous group of travelers in a freeway network;

(1b) the desired travel time for a driver to reach the desired destination in a freeway network;

(2) the desired travel time for a driver to reach the desired destination in a (single-corridor) urban traffic network.

Compared to dynamic routing practices based on, e.g., dynamic route information panels, our proposed method can bring about a system-optimal distribution of traffic over the alternative roads in a freeway network by directly assigning drivers to a route. Moreover, different from methods that assign constant or flow-dependent tolls to road stretches [4, 105, 180, 191], in the proposed reverse Stackelberg approaches the tolls are replaced by incentive functions that are directly dependent on the decision variable of the drivers. In this manner, it may be more practical to differentiate between types of drivers (e.g., in terms of their desired origin and destination or the value-of-time profile) as compared to when flow-dependent tolls are

adopted. With respect to the fairness of applying monetary incentives to heterogeneous drivers, it will later become clear that the precise amount associated with the optimal follower decision is not fixed nor is it important for the game, i.e., the leader may assign a zero incentive to the desired values of the decision variables and still achieve a system-optimal situation. The shape or geometry of the leader function is relevant for influencing the decision of the follower, but the road authority is free to associate any monetary incentives for the heterogeneous driver groups as she deems appropriate. As in other toll-based approaches, possible revenue yielded from the monetary incentives in return for a less efficient behavior of the drivers can enable the road authority to implement additional, e.g., environmental countermeasures.

The remainder of this chapter is organized as follows. After a concise summary of available literature in the area of road (congestion) pricing in Section 5.2, some relevant aspects of the general reverse Stackelberg game introduced in Section 2.1 are placed in a traffic context in Section 5.3. The two proposed reverse Stackelberg routing approaches and the third game variant mainly designed for urban networks are described in respectively Section 5.4 and 5.5. More specifically, the elements defining the games are described, after which the dynamic traffic assignment problems and the derivation of an optimal leader function is discussed. While the applications are not intended to be readily implementable in real-life, the performance of the travel time-based reverse Stackelberg approach in a freeway setting is illustrated in a case study in which overlapping routes apply in Section 5.6 and a discussion on implementability is provided. Finally, conclusions and recommendations are presented in Section 5.7.

5.2 A Brief Overview of Road Pricing Literature

There is an abundance of literature available on tolling approaches that aim to improve traffic conditions such as by reducing congestion [146, 169]. Often these toll design approaches [141, 160] are based on the original Stackelberg game [195, 196] as described in Section 2.1 in which the road authority as a leader acts first by pricing road stretches. The drivers as follower players are assumed to adapt their behavior, i.e., their route choice, to this toll. An overview of road or congestion pricing literature of the 70s and 80s on foundations as well as implementational aspects can be found in [146, 150].

Different types of toll have been considered since then, e.g., static versus dynamic tolling, constant, time-varying or traffic flow-based tolls, while either deterministic or stochastic models of the traffic behavior have been adopted. Another distinction is in whether all links in a traffic network are tolled (first-best tolling [201]) or only a subset of links (second-best tolling [191]). Further, also the choice of departure time has been considered as an additional decision variable of the drivers, leading to time-variant tolls [4, 105]. In [104], multicriteria road pricing is considered, focusing not only on minimizing the total time spent, but in addition on the reduction of vehicular emissions and fuel consumption. Case-studies are often based on conceptual parallel-link networks [51, 193], on the Chen or Beltway network [35, 105, 178], or on real-life ringroads around cities like San Diego [182]

and Singapore [73, 155]. A higher-level game considering the different (conflicting) objectives between multiple stakeholders or road authorities of different regions in road pricing games is considered in [207]. Also in [181, 200, 206], situations are considered in which multiple players adopt their own road pricing schemes, given the presence of private toll roads and different ownership regimes.

Finally, in the road pricing methods suggested in [180], instead of adopting the original Stackelberg game with constant tolls, constant as well as first- and second-degree polynomial toll functions of traffic flow are considered. As pointed out in the previous section, the approaches adopted in the current chapter are different because the incentives – that can be both positive and negative – are functions of the decision variables of the drivers. Not only does this enable to distinguish between heterogeneity of drivers as compared to when the flow-dependent and constant toll approaches are adopted; it can also be used to induce the drivers to adopt a desired behavior that is associated with their choice of decision variables.

In order to compute traffic flow-dependent leader functions for different cost functions, e.g., to minimize the TTS or to maximize the toll revenue of the road authority, previously, neural networks have been adopted [179, 180]. There, constant up to second-degree polynomial leader functions were computed using a direct approach as was argued to be necessary because of the complexity of computing an optimal solution for the road authority. In the routing application of this chapter, we show how such a solution can nonetheless be (efficiently) computed when adopting an indirect solution method.

5.3 General Framework for Reverse Stackelberg Games in Traffic Networks

An outline of the basics of the reverse Stackelberg game has been provided in Chapter 2 (Section 2.1). Just as in the previous chapters, we consider the indirect formulation of the game under the assumption that both leader and follower players behave fully rationally, i.e., they optimize their respective goal functions over the given decision spaces. However, for the application considered in this chapter, it does not suffice to consider a static, nor a single leader-single follower game. We briefly elaborate on the differences with respect to these relevant aspects for the implementation in a routing context in the following. A discussion on how to deal with deviations from the rational, optimal behavior can be found in Section 5.6.2.

On the Existence of an Optimal Solution

Different from the direct approach in which no *a priori* desired equilibrium is adopted, it is possible that for a particular desired leader equilibrium in the indirect approach, no leader function exists that can induce the follower player to act according to the desired decisions. This can be the case if due to strict constraints on the decision spaces Ω_L, Ω_F, no leader functions exist that map the full space Ω_F to Ω_L and that pass through (u_L^d, u_F^d) while staying outside the sublevel set Λ_d. Whereas this has not been an issue in the general setting considered before, in the

current application, no matter the traffic situation, a feasible leader function should be provided to the drivers. However, as will be shown in Section 5.4.2, an optimal leader function does in fact exist under the considered objective functions and the associated level curves of the drivers.

Multiplayer Game

In the current traffic routing setting, one road authority will be considered together with multiple classes of homogeneous drivers. In the literature on reverse Stackelberg games, some authors that consider settings with one leader and multiple followers assume the followers to play a noncooperative game amongst themselves in which they act simultaneously, leading to a Nash equilibrium amongst the follower players [99, 167]. The leader can then consider the actions of the followers together as the response of one follower. Alternatively, different leader functions $u_{\mathrm{L},i} = \gamma_{\mathrm{L},i}(u_{\mathrm{F},1}, \ldots, u_{\mathrm{F},N}), i = 1, \ldots, N$ are proposed in [98] with the objective to align the objective functions of the N different followers. Under assumptions on continuous differentiability and (strict) convexity of the follower objective functions as summarized in Chapter 2 (Section 2.4.6), even a single leader function $u_{\mathrm{L}} = \gamma_{\mathrm{L}}(u_{\mathrm{F},1}, \ldots, u_{\mathrm{F},N})$ can prove sufficient to reach the leader's desired equilibrium [103].

As it will become clear, in this chapter we can consider each game between the leader and one follower separately, without the need to consider the relation between the followers themselves. After the leader has computed her desired, system-optimal solution, an optimal leader function can be computed for each of the homogeneous driver groups separately. However, as it will also become clear in the following sections, in the splitting rate-based variant of the game also the followers have to play a game amongst themselves. In particular, they have to decide how individual drivers will be distributed according to the accumulated route splitting rates that hold for the entire group. Since these lower-level problems do not influence the main game that is of interest in the current chapter, they are left for future research.

Here, independence of the follower players is only guaranteed for the case in which players are fully rational, as assumed throughout the largest part of this dissertation. As it will become clear in the remainder, especially the travel time-based game variant may result in infeasibilities in case followers deviate significantly from their optimal decisions. In other words, if followers adopt suboptimal travel times and are therefore associated with a suboptimal route, the (optimal) desired travel time of other players on overlapping routes will be influenced. While travel times of drivers are also influenced in case of suboptimal behavior in the splitting rate-based case, each follower's (suboptimal) desired splitting rates can be realized in practice. Here it should be noted that also in case of unpredictable events like accidents, promised travel times may not be realizable. In such cases, which will not be considered in this chapter, a new incentive scheme could be proposed. Further, penalization schemes as discussed in Section 5.6.2 may mitigate the effect of suboptimal behavior on a longer term.

Dynamic Game

Recall that in a multi-stage (discrete-time) framework, the system to which the game is applied is identified by a state variable $x(t) \in X \subseteq \mathbb{R}^{n_x}, n_x \in \mathbb{N}$ with a corresponding update equation, e.g., $x(t+1) = f(x(t), u_L(t), u_F(t))$, $t = 1, 2, \ldots, T$. This variable is part of the players' objective function $J_p : \Omega_L \times \Omega_F \times X \to \mathbb{R}, p \in \{L, F\}$.

Next to the static game (2.5)–(2.7), also open-loop [209], state feedback [98, 184], and closed-loop [210] variants of the reverse Stackelberg game have been considered, in which the leader function can be a mapping of decision variables of the current time instant as well as of one or of a series of previous time instants and in addition of the state variable, to the leader's decision space.

While we do consider a dynamic routing problem in which the evolution of traffic states over time is taken into account, the leader functions of the game are computed and proposed once to the incoming traffic at each control time step. Hence, it is sufficient to adopt the open-loop dynamic variant in which a set of leader functions is determined at each control time step. For ease of notation, each new leader function proposed to a certain follower player at a certain control time step is simply denoted by $\gamma_L(\cdot)$. The methods available for solving the static reverse Stackelberg game can also be applied to the open-loop variant.

At the same time, for an accurate prediction of the traffic flow evolution that is necessary for the road authority to compute an accurate system-optimal traffic assignment, considering the traffic dynamics is important. This will be elaborated upon in the discussion on the computation of an optimal traffic distribution in Section 5.4.2(i).

Solution Methods

Recall from Chapter 2 that the general static reverse Stackelberg game (2.2) can be proven to be NP-hard by reduction to the original Stackelberg game, which is in its turn equivalent to a bilevel programming problem [39] that is (strongly) NP-hard [89] already for the case of linear higher- and lower-level optimization objectives. Nonetheless, different solution methods are available for the static reverse Stackelberg game as will be briefly recapitulated next. Please refer back to Chapter 4 for a more detailed description of the approaches.

For (quasi) convex follower objective functions or for cases in which the desired equilibrium (u_L^d, u_F^d) is a boundary point of the convex hull of the sublevel set Λ_d, an affine leader function has been proven to lead to the desired leader equilibrium (see Chapter 3, [83, 209]). For general problems, direct ([179, 186]) and indirect (Chapter 4, [86]) approaches have been proposed that consider respectively the game formulation (2.2) and (2.5)–(2.7):

- Direct Approaches
  - *Evolutionary algorithms* [179, 186]: the relationship between the follower decision variable $u_F \in \Omega_F$ and the leader objective function value $J_L(\gamma_L(u_F), u_F)$ for a leader function $\gamma_L(\cdot)$ parametrized by coefficients is approximated. Optimization of this relationship should lead to the best

leader function coefficients. However, no optimality guarantees can be given as regards the approximation.

- Indirect Approaches [Chapter 4]
  - *Continuous basis-function approach:* based on a leader function structure represented by a linear combination of basis functions, a multi-level optimization problem can be solved to yield optimal leader function coefficients.
  - *Grid-based basis-function approach:* this method is computationally more efficient than the continuous equivalent for relatively small problem instances, but computation times increase exponentially for an increasing number of dimensions $n_{\mathrm{L}}, n_{\mathrm{F}}$ and a more dense grid. In general, a trade-off between computational efficiency and suboptimality of the leader function can be made by the choice of the gridding precision.
  - *Interpolation and spline approach:* leader functions passing through a set of selected data points can be derived efficiently. This heuristic approach can in general not guarantee the selection of an optimal leader function, implying an adaptive approach with an *a priori* unknown number of iterations. In order to facilitate the selection of proper data points outside $\Lambda_{\mathrm{d}} \setminus \{(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\}$, this approach is best applied to problems where $n_{\mathrm{L}} + n_{\mathrm{F}} \leq 3$.

In the Section 5.4 below, the details of the game approaches proposed for the freeway context are described, followed by the urban game formulation in Section 5.5.

5.4 Reverse Stackelberg Approaches for Traffic Routing in Freeway Networks

5.4.1 Problem Statement

The game-based methods proposed in this section are developed for the purpose of reaching a system-optimal distribution of traffic over the available routes based on any desired criterion such as the total time spent of traffic in a network. In particular, networks with multiple available routes for any origin-destination (OD) pair are considered, where the reverse Stackelberg approach is adopted to be able to distribute the traffic by assigning drivers to the available routes.

Here, the traffic network is modeled as a directed graph with a set of origin and destination nodes $\mathcal{O}$ and $\mathcal{D}$ respectively. The total number of OD-pairs $(o,d) \in \mathcal{O} \times \mathcal{D}$ is denoted by $N_{\mathrm{OD}} = |\mathcal{O}| \cdot |\mathcal{D}|$, where $|X|$ represents the cardinality of X. The number of alternative routes, i.e., paths without cycles, for the i-th OD-pair, $i \in \{1, \ldots, N_{\mathrm{OD}}\}$, is denoted by $n_i \in \mathbb{N}$. Homogeneous freeway stretches are represented by links $m \in \mathcal{L}$, that connect origins, destinations, and internal nodes of the set $\mathcal{N}$ of nodes that do not constitute a source or sink of outgoing and incoming traffic, respectively. Hence,

we define the node sets such that $\mathcal{O} \cap \mathcal{N} = \varnothing$ and $\mathcal{D} \cap \mathcal{N} = \varnothing$. In addition, w.l.o.g. we assume that each origin $o \in \mathcal{O}$ has a single outgoing link $m_{\text{out}}(o)$. In case of multiple outgoing links of a given origin, a single virtual link can be created with zero length, hence with zero travel time, that connects the given origin to a virtual internal node from which the multiple outgoing links then depart. Similar applies to a destination node $d \in \mathcal{D}$ with multiple incoming links. When adopting a traffic prediction model, the links can be further divided into road segments of equal length, in order to achieve an accurate prediction of the traffic behavior. For clarity, only link indices are adopted throughout the greater part of the chapter.

Further, a receding horizon approach is adopted to take into account not only the present but also the future traffic conditions. In particular, $k_{\text{c}} \in \mathbb{N}$ indicates the time instant $t = k_{\text{c}} T_{\text{c}}$, with T_{c} [s] the sample or control time interval of the dynamic routing approach. Similarly, $k \in \mathbb{N}$ indicates the time instant $t = kT_{\text{s}}$, with T_{s} [s] the time interval for the simulation of the traffic flow behavior based on a prediction model that will be presented in Section 5.4.2. This yields a time horizon $[kT_{\text{s}}, (k+N_{\text{p}})T_{\text{s}}]$ with the prediction horizon N_{p} and where $T_{\text{c}} = MT_{\text{s}}, M \in \mathbb{N}_0^+$.

To complete the traffic link flow description, the inflow of origins can be written

$$q_{\text{in}}^{o,d}(k) = q_{m_{\text{out}(o)},d}(k), \tag{5.1}$$

with $q_{\text{in}}^{o,d}$ the effective demand at time step k of vehicles traveling from $o \in \mathcal{O}$ to $d \in \mathcal{D}$. Then, the total inflow of node $n \in \mathcal{N}$ at time step k that has a destination d is denoted by

$$Q_{n,d}(k) = \sum_{m \in I(n)} q_{m,d}(k), \tag{5.2}$$

with $I(n)$ the set of incoming links of node n. Similarly, $O(n)$ denotes the set of outgoing links for node n, with a traffic flow

$$q_{m,d}(k) = \beta_{n,m,d}(k) Q_{n,d}(k), \tag{5.3}$$

where $\beta_{n,m,d}(k) \in [0,1]$ represents the splitting rate for link m at node n with the destination d. We can now write the total flow $q_m(k)$ on link m by

$$q_m(k) = \sum_{d \in \mathcal{D}} q_{m,d}(k). \tag{5.4}$$

Aggregated link flows are obtained simply by summing up the flows for the possible destinations.

We now focus on achieving a system-optimal distribution of vehicle flows over a traffic network with respect to the TTS. In order to optimize the use of the available routes in a traffic network by the drivers, we will strive for congestion avoidance by keeping the flow on a route below the bottleneck capacity or capacity flow as long as possible. The general cost function in case the system-optimal traffic distribution is defined to minimize the TTS for a given prediction horizon N_{p} that is incorporated to take into account not only the present but also the future traffic, can be described by

$$\mathcal{J}(k) = T_{\text{s}} \sum_{j=1}^{N_{\text{p}} \cdot M} \left(\sum_{m \in \mathcal{L}} q_{\text{in},m}(k+j) t_m(k+j) + \sum_{o \in O_{\text{all}}} w_o(k+j) \right), \tag{5.5}$$

where $t_m(k)$ denotes the mean travel time associated with link $m \in \mathcal{L}$ at time step k. The value of $t_m(k)$ is determined using a traffic prediction model as will be elaborated upon in paragraph (i) of the dynamic game framework below. In addition, the queue length $w_o(k)$ at origin $o \in \mathcal{O}_{\text{all}}$ is considered to incorporate situations in which the demand $d_o(k)$ of the time interval $[kT_s,(k+1)T_s]$ (including the current queue length) exceeds the capacity minus the outflow of the given origin, where $\mathcal{O}_{\text{all}}$ denotes the set of indices of all origins. Further, $q_{\text{in},m}(k)$ denotes the effective traffic inflow of link $m \in \mathcal{L}$ in the time interval $[kT_s,(k+1)T_s]$, i.e., where $q_{\text{in},m_{\text{out}(o)}}(k) \leq d_o(k)$, with $m_{\text{out}(o)} \in \mathcal{L}$ the (virtual) outgoing link of origin $o \in \mathcal{O}$.

5.4.2 The Reverse Stackelberg Approaches

In order to introduce the reverse Stackelberg approach to the dynamic traffic assignment problem, we start with a definition of the basic elements of the reverse Stackelberg game, linked to the traffic control context, where the following game variants are proposed:

- *splitting rate-based:* The leader function represents a mapping of route splitting rates to monetary incentives.
- *travel time-based:* The leader function represents a mapping of desired travel time to monetary incentives.

The scheme of Fig. 5.2 illustrates the game-based framework that leads to a dynamic route assignment, using a leader-follower approach in which the road authority associates monetary incentives denoted by θ with a follower's choice of either the route splitting rate ζ or the travel time denoted by τ in which a driver or a group of drivers – represented by a follower player – desires to reach a destination. Based on the follower's choice of the pair (θ,ζ) or (θ,τ) according to the relation $\theta = \gamma_L(\cdot)$ proposed by the leader, the driver is assigned to a route. The leader's aim is therefore to compose a leader function $\gamma_L(\cdot)$ such that a system-optimal distribution of traffic can be achieved. The characteristics of the drivers as well as the incentive functions and travel time choice of the drivers are communicated via an on-board computer.

Before the overall approach is elaborated upon, first the basic elements of the reverse Stackelberg game are translated to the traffic domain.

The Players and Their Decision Variables

- The single **leader player** represents the road authority responsible for accomplishing an optimal use of a given traffic network.
- In the *splitting rate-based* reverse Stackelberg approach, a **follower player** represents a homogeneous group of vehicles, i.e., drivers characterized by the same OD-pair and monetary value-of-time class index $h \in \mathcal{H} := \{1,\ldots,H\}$, where the set of classes of drivers with a particular origin and destination index $i \in \{1,\ldots,N_{\text{OD}}\}$ is denoted by $\mathcal{H}_i$. The differentiation in monetary value-of-time could be based on, e.g., the urgency of the particular driver given the travel purpose that could be indicated via the on-board computer, or on the class or

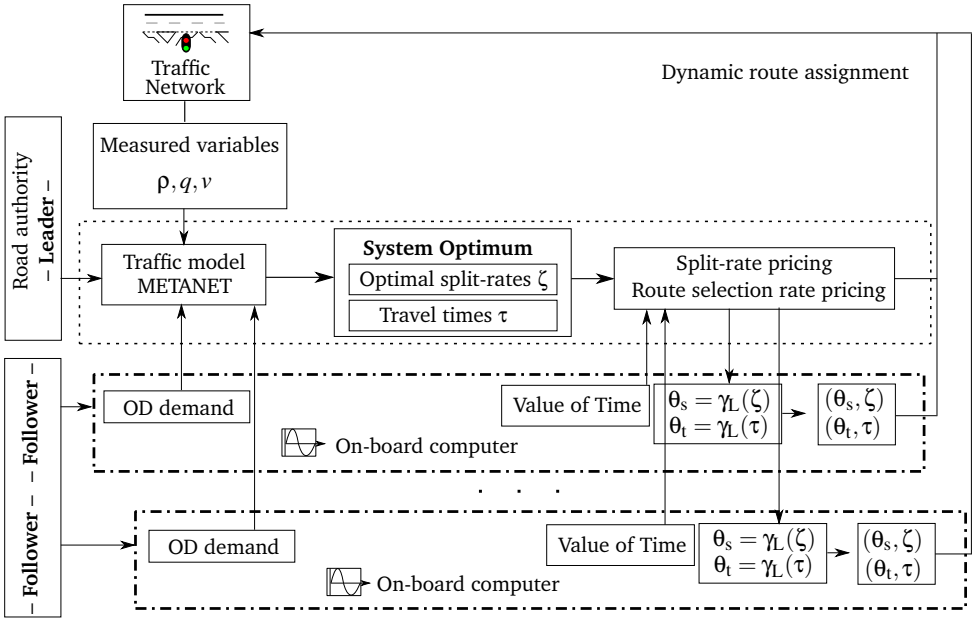


Figure 5.2: *Schematic framework for the splitting rate-based and the travel time-based reverse Stackelberg approaches to dynamic route assignment.*

type of vehicle. Alternatively, this parameter could be determined via an iterative learning process in which the monetary value of time is adapted over time based on the decision variables selected by the given driver.

The total number of follower players can be written $N_{\mathrm{F,s}} = \sum_{i=1}^{N_{\mathrm{OD}}} |\mathcal{H}_i|$ where we denote the set of followers by

$$\mathcal{F}_{\mathrm{s}} := \left\{ (h,i) | i \in \left\{ 1, \dots N_{\mathrm{OD}}, h \in \mathcal{H}_i \right\} \right\}. \tag{5.6}$$

In the *travel time-based* reverse Stackelberg approach, in order for the leader player to accomplish the desired traffic distribution, drivers in a homogeneous follower group will have to be distributed over the different available routes. Therefore, a follower player is assigned another index specifying the route $j \in \{1, \dots, n_i\}$ that the leader desires the drivers of the group to take. The leader can accomplish this by presenting a different leader function $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F,t}} \to \Omega_{\mathrm{L}}$ to n_i subgroups of the drivers within value-of-time-classes $h \in \{1, \dots, H\}$ and OD-pair $i \in \{1, \dots, N_{\mathrm{OD}}\}$. In the extreme case, each individual driver could be treated as a separate follower player that gets assigned an individual leader function.

Here, the total number of follower players is thus denoted by $N_{\mathrm{F,t}} = \sum_{i=1}^{N_{\mathrm{OD}}} \sum_{j=1}^{n_i} n_i \cdot |\mathcal{H}_i|$ where we can represent the set of followers by

$$\mathcal{F}_{\mathrm{t}} := \left\{ (h,i,j) | i \in \left\{ 1, \dots N_{\mathrm{OD}}, j \in \{1, \dots, n_i\}, h \in \mathcal{H}_i \right\} \right\}. \tag{5.7}$$

For the sake of implementation in a dynamic receding horizon framework, we consider the traffic that enters (or plans to enter) the system in the time period $Mk_c = [k_c T_c, (k_c+1)T_c]$ as one follower player at the control time step k_c.

The associated **decision variables** represent:

- A monetary incentive $\theta_s^{hi}(k_c) \in \Omega_L^{hi} \subseteq \mathbb{R}$ to be awarded or collected by the **leader player** regarding the follower player $(h,i) \in \mathcal{F}_s$, where $\Omega_L^{hi} := [\theta_{min}^{hi}, \theta_{max}^{hi}]$ denotes an acceptable range of monetary incentive values.

  The same applies to the incentive $\theta_t^{hij}(k_c) \in \Omega_L^{hij} \subseteq \mathbb{R}$ for a follower $(h,i,j) \in \mathcal{F}_t$ in the travel time-based game, with $\Omega_L^{hij} := [\theta_{min}^{hij}, \theta_{max}^{hij}]$.

- The follower decision variable is significantly different in the two proposed methods:

  – *Splitting rate-based game:*

  Each homogeneous group of drivers that constitutes the follower player $(h,i) \in \mathcal{F}_s$ decides upon the vector of route selection variables $\zeta^{hi}(k_c) \in \Omega_{F,s}^{hi} \in \mathbb{R}^{n_i}, i \in \{1,\dots,N_{OD}\}, h \in \mathcal{H}$ that specifies the fractions of this homogeneous group of drivers that take the different routes associated to OD-pair index $i \in \{1,\dots,N_{OD}\}$, where $0 \le \zeta_m^{hi}(k_c) \le 1, \sum_{m=1}^{N_{OD}} \zeta_m^{hi}(k_c) = 1$, and where

  $$\zeta^{hi}(k_c) := \left(\zeta_1^{hi}(k_c)\ ,\ \dots\ ,\ \zeta_{n_i}^{hi}(k_c)\right)^T. \tag{5.8}$$

  The effective inflow of vehicles associated with route $j \in \{1,\dots,n_i\}$ for the OD-pair index $i \in \{1,\dots,N_{OD}\}$ at control time step k_c can thus be written $q_{in}^{ij}(k_c) := \sum_{h\in\mathcal{H}} \left(q_{in}^{hi}(k_c)\zeta^{hi,d}(k_c)\right)_j$.

  – *Travel time-based game:*

  Each follower player $(h,i,j) \in \mathcal{F}_t$ decides upon the desired travel time (including the average time of waiting in the queue at time step k_c) $\tau^{hij}(k_c) \in \Omega_{F,t}^{hij} \subseteq \mathbb{R}^+$ to reach the associated destination, where $\Omega_{F,t}^{hij}(k_c) := [\tau_{min}^{hij}(k_c), \tau_{max}^{hij}(k_c)]$ denotes the range of possible, i.e., realizable travel times for a specific OD-pair that is provided by the leader at the current time step k_c and given the route length and the predicted velocities.

Leader and Follower Objective Functions

- It is important to note that the proposed reverse Stackelberg game can be applied to reach system-optimal traffic behavior for many different objectives. The only requirement for a suitable formulation of the leader's objective is that the drivers' decision variable should be an independent variable in the leader's objective function, as is described in a general game context in Remark 2.5. In both approaches we take the leader player's aim as the minimization of the sum of total travel time (including possible waiting time in queues) of traffic in the system, subject to consistency and capacity constraints.

– *Splitting rate-based game:*

$$\jmath_{\mathrm{L,s}}(k_\mathrm{c}) = T_\mathrm{c} \sum_{h\in\mathcal{H}_i} \sum_{i=1}^{N_\mathrm{OD}} \left(t^{hi}(k_\mathrm{c})\right)^\mathrm{T} \cdot \left(q_\mathrm{in}^{hi}(k_\mathrm{c})\zeta^{hi}(k_\mathrm{c})\right), \tag{5.9}$$

where $q_\mathrm{in}^{hi}(k_\mathrm{c})$ [veh/h] denotes the effective demand or inflow of drivers in the value-of-time class $h \in \mathcal{H}_i$ for the i-th OD-pair that enters the system during the time window $[k_\mathrm{c}T_\mathrm{c}, (k_\mathrm{c}+1)T_\mathrm{c}]$. Furthermore,

$$t^{hi}(k_\mathrm{c}) := \left(t_1^{hi}(k_\mathrm{c}) \ , \ \ldots \ , \ t_{n_i}^{hi}(k_\mathrm{c})\right)^\mathrm{T} \tag{5.10}$$

denotes the vector of associated predicted travel times for reaching the destination via each of the routes associated with the i-th OD-pair, where the previous waiting times of the vehicles corresponding to the effective inflow $q_\mathrm{in}^{hi}(k_\mathrm{c})$ are incorporated.

– *Travel time-based game:*

$$\jmath_{\mathrm{L,t}}(k_\mathrm{c}) = T_\mathrm{c} \sum_{h\in\mathcal{H}_i} \sum_{i=1}^{N_\mathrm{OD}} \sum_{j=1}^{n_i} \tau^{hij}(k_\mathrm{c}) q_\mathrm{in}^{\mathrm{ctrl},hij}(k_\mathrm{c}), \tag{5.11}$$

where $q_\mathrm{in}^{\mathrm{ctrl},hij}(k_\mathrm{c}) \in \mathbb{R}^+$ [veh/h] denotes the part of the effective demand $q_\mathrm{in}^{hi}(k_\mathrm{c})$ [veh/h] for the OD-pair $i \in \{1,\ldots,N_\mathrm{OD}\}$ that involves the vehicles from one of the $h \in \{1,..,H_i\}$ classes of drivers with a certain monetary value of time, that is *distributed* by the road authority over route $j \in \{1,\ldots,n_i\}$, i.e., $\sum_{j=1}^{n_i} q_\mathrm{in}^{\mathrm{ctrl},hij}(k_\mathrm{c}) = q_\mathrm{in}^{hi}(k_\mathrm{c})$ for $i = 1,\ldots,N_\mathrm{OD}$, $h = 1,\ldots,H_i$.

- The followers' aim is to minimize the travel cost that can be represented by a function of monetary incentives and the expected travel time. Note that while the leader player uses a receding-horizon approach, the follower is only interested in the traffic conditions predicted at the control time step k_c of entering the system and with a horizon until the destination is reached:

  – *Splitting rate-based game:*

$$\jmath_{\mathrm{F,s}}^{hi}(k_\mathrm{c}) = \alpha_\mathrm{F}^h \left(t^{hi}(k_\mathrm{c})\right)^\mathrm{T} \cdot \left(q_\mathrm{in}^{hi}(k_\mathrm{c})\zeta^{hi}(k_\mathrm{c})\right) + \theta_\mathrm{s}^{hi}(k_\mathrm{c}), \tag{5.12}$$

where the vector of expected route travel times $t^{hi}(k)$ including waiting times follows from the traffic flow distribution $\zeta^{hi}(k) \cdot q_\mathrm{in}^{hi}(k)$ in a manner explained in paragraph (i) of the dynamic game framework below. Here, $\alpha_\mathrm{F}^h \in \mathbb{R}^+$ is the possibly time-variant monetary value of travel time, also known as the value of travel time savings that indicates a driver's willingness to pay for a change of travel time. This value is often taken to be a constant parameter in the road-pricing literature, implying a linear mapping of travel time to monetary value [106]. Nonetheless, for the reverse Stackelberg methods, α_F^h could be replaced by a more involved, nonlinear

relation as considered in, e.g., [23, 53]. This choice could influence the function structure required for an optimal leader function γ_{L} to exist, as will be elaborated upon later in this section.

– *Travel time-based game:*

$$\jmath_{\mathrm{F,t}}^{hij}(k_{\mathrm{c}}) = \alpha_{\mathrm{F}}^{h}\tau^{hij}(k_{\mathrm{c}}) + \theta_{\mathrm{t}}^{hij}(k_{\mathrm{c}}), \tag{5.13}$$

where α_{F}^{h} is as defined above.

The Dynamic Game Framework

The dynamic reverse Stackelberg routing approach consists of the following main steps:

(i) Given the traffic state at the control time step k_{c} and the demand for the OD-pairs as indicated by the drivers via on-board computers, a **system-optimal distribution** of the new vehicles over the available routes with respect to criterion (5.9) or (5.11) is computed by the road authority, together with the corresponding predicted mean travel times. For more details on this computation, refer to the next subsection 5.4.2(i).

(ii) Given the *desired*, optimal distribution of vehicles $q_{\mathrm{in}}^{ij,\mathrm{d}}(k_{\mathrm{c}})$ for the route index j for the OD-pair index i and and the associated predicted mean travel times, optimal **leader functions** are computed by the leader. This computation is elaborated upon in Section 5.4.2(ii):

– *Splitting rate-based game*: An optimal leader function $\gamma_{\mathrm{L,s}}^{hi}$ for each of the followers $(h,i) \in \mathcal{F}_{\mathrm{s}}$ leads to the choice of a *desired* route splitting rate by the drivers of the i-th OD-pair and with a monetary value of time-class $h \in \mathcal{H}_i$, where $q_{\mathrm{in}}^{ij,\mathrm{d}}(k_{\mathrm{c}}) := \sum_{h=1}^{H_i} \left(q_{\mathrm{in}}^{hi}(k_{\mathrm{c}})\zeta^{hi,\mathrm{d}}(k_{\mathrm{c}})\right)_j$ for $i \in \{1,\ldots,N_{\mathrm{OD}}\}$, $j \in \{1,\ldots,n_i\}$.

– *Travel time-based game*: An optimal leader function $\gamma_{\mathrm{L,t}}^{hij}$ for each of the followers $(h,i,j) \in \mathcal{F}_{\mathrm{t}}$ leads to the choice of a *desired* predicted travel time $\tau^{hij,\mathrm{d}}(k_{\mathrm{c}})$ associated with one of the $j \in \{1,\ldots,n_i\}$ routes for a specific share $q_{\mathrm{in}}^{\mathrm{ctrl},hij}(k_{\mathrm{c}})$ of the drivers for the i-th OD-pair and with a monetary value of time-class $h \in \mathcal{H}_i$, where $\sum_{j=1}^{n_i} q_{\mathrm{in}}^{ij,\mathrm{d}}(k_{\mathrm{c}}) := \sum_{h=1}^{H_i} q_{\mathrm{in}}^{hi}(k_{\mathrm{c}})$, for $i = 1,\ldots,N_{\mathrm{OD}}$.

(iii) As a response to the optimal leader functions, a rational follower in the travel time-based game will choose a combination of monetary incentive and travel time, which the leader associates with a certain route that the follower should[1] follow. In the splitting rate-based game, the optimal splitting rates associated with an optimal leader function automatically lead to the desired, system-optimal distribution.

[1] Penalty schemes could be implemented to mitigate disobedience outside the game context; more suggestions in this context can be found in the discussion paragraph of Section 5.6.2.

(i) Computing an Optimal Distribution of Traffic Flow

Several methods exist to determine a desired system-optimal traffic distribution over the available routes. For an overview on approaches for dynamic traffic assignment, the reader is referred to [35, 142, 162]; to [37] for integration with departure time choice, and to [62] for the equilibrium traffic assignment problem. Distributed routing policies with a specific focus on resilience, viz., robustness to link failures and disruption in general types of networks are considered in [40, 41]. We here describe how such methods for computing an optimal assignment can be related to our game.

Using a prediction model as will be elaborated upon in the following paragraph, given a driver demand pattern that is communicated via the on-board computer for the control time window $[k_c T_c, (k_c+1)T_c]$, i.e., $q_{\text{in}}^{ij}(k_c)$ [veh/h], $i \in \{1,\dots,N_{\text{OD}}\}, j \in \{1,\dots,n_i\}$, a system-optimal – with respect to the TTS – distribution of traffic flow $q_{\text{in}}^{ij,\text{d}}(k_c)$ over the road network can be computed, as well as the corresponding predicted mean travel times $t^{ij}(k_c)$ for each of the routes $j \in \{1,\dots,n_i\}$. These travel times are accumulated from the predicted mean travel time on a link $m \in \mathcal{L}$ for the effective flow of vehicles $q_{\text{in},m}(k)$ entering link m in the period $[kT_s, (k+1)T_s]$ and are denoted by $t_m(k)$. This symbol t_m is adopted to differentiate between the predicted travel times t^{hi} and the desired route travel time variable τ^{hij} used in the travel time-based *game*, where the predicted mean travel times $t^{ij}(k_c)$ associated with the system-optimal traffic distribution for the routes $j \in \{1,\dots,n_i\}$ are denoted by $\tau^{ij,\text{d}}(k_c)$.

The traffic assignment can now be determined by solving a dynamic version of the minimum cost flow problem, with the objective to minimize (5.5).

Additionally, for $j = 1,\dots N_{\text{p}}$, the following constraints are needed to correctly represent the flows in the traffic network, where $o(i) \in \mathcal{O}, d(i) \in \mathcal{D}$ represent the origin and destination associated with the OD-pair i: $\{(o,d)\}_i$:

$$\sum_{h\in\mathcal{H}_i} q_{\text{in}}^{hi}(k+j) = q_{m_{\text{out}}(o(i)),d(i)}(k+j), \qquad i = 1,\dots,N_{\text{OD}}, \tag{5.14}$$

$$\sum_{d\in\mathcal{D}} \sum_{m\in I(n)} q_{m,d}(k+j) = \sum_{d\in\mathcal{D}} \sum_{m\in O(n)} q_{m,d}(k+j), \qquad \forall n \in \mathcal{N}, \tag{5.15}$$

$$\sum_{d\in\mathcal{D}} q_{m,d}(k+j) \le q_{\text{cap},m}, \qquad \forall m \in \mathcal{L}, \tag{5.16}$$

$$q_{m,d}(k+j) \ge 0, \qquad \forall m \in \mathcal{L}, \forall d \in \mathcal{D}, \tag{5.17}$$

The system-optimal, desired route selection rates $\zeta^{hi,\text{d}}$ that are required for the game of the homogeneous followers $(h,i) \in \mathcal{F}_s$ now follow straightforwardly from the desired node splitting rates $\beta_{n,m,d}^{\text{d}}$ that again follow from the optimal flows $q_{m,d}(k)$ for each link $m \in \mathcal{L}$ towards the destination $d \in \mathcal{D}$, i.e.: $\beta_{n,m,d}^{\text{d}}(k) := \frac{q_{m,d}(k)}{\sum_{m\in O(n)} q_{m,d}(k)}$. Hence, the distribution of traffic demand over the particular routes, leading to a flow q_{ij} on route $j \in \{1,\dots,n_i\}$ for OD-pair index $i \in \{1,\dots,N_{\text{OD}}\}$ can be written:

$$q_{ij}(k) := \sum_{h\in\mathcal{H}_i} q_{\text{in}}^{hi}(k) \cdot \zeta_j^{hi}(k), \text{ with } \zeta_j^{hi}(k) := \prod_{\substack{n\in\mathcal{P}_{ij}^{\mathcal{N}} \\ m\in O(n)}} \beta_{n,m,d(i)}(k), \tag{5.18}$$

where $\mathcal{P}_{ij}^{\mathcal{N}} \subseteq \mathcal{N}$ denotes the set of nodes on the path j for OD-pair i. Further, for the flow on a link $m \in \mathcal{L}$ with destination $d \in \mathcal{D}$ we have:

$$q_{m,d}(k) = \sum_{i \in I_d} \sum_{h \in \mathcal{H}_i} \sum_{j \in J_{i,m}} q_{\text{in}}^{hi}(k) \xi_j^{hi}(k), \tag{5.19}$$

with I_d the set of OD-pairs with destination node d and $J_{i,m}$ the set of routes for the i-th OD-pair that contains link m.

In order to derive the associated predicted travel time for a route as required in (5.5), several traffic models as well as prediction approaches of various degrees of precision can be adopted that predict the travel time $t_m(k)$ for vehicles entering a particular link $m \in \mathcal{L}$ at time kT_s. Here, we take the average travel time to be $t_m(k) = L_m/\tilde{v}_m(k)$, with $\tilde{v}_m(k)$ the average space-mean speed in link m at time step k.

It is important to note that this choice of accuracy determines the complexity of the above problem. If one takes $\tilde{v}_m(k)$ as a constant, e.g., equal to the currently measured (space-mean) speed or equal to the maximum or free-flow speed that applies during uncongested traffic conditions, a linear programming problem results [2]. Here it should be noted that in order to reduce the complexity of the optimization problem leading to a route assignment, one can also select a particular subset of shortest routes that are together sufficient in capacity for the given demand – but may not yield a system-optimal distribution as when the complete network were considered – for the traffic to be distributed amongst.

The following paragraph elaborates on different prediction models as well as on simulation approaches that can be adopted to obtain the predicted route travel time.

Traffic Flow Prediction In order to capture the behavior of traffic over time, a traffic prediction model is used to track the traffic states, i.e., to analyze the impact of a traffic routing assignment on these states and to determine an accurate prediction of the travel time of drivers. Based on the current and future state of the network and on an estimation of future demand, the desired traffic flow distribution and the associated decision variables that can lead to this state can be determined as will be explained next. In particular, the state of the traffic network can be described by the following macroscopic variables:

- average traffic density $\rho_{m,i}(k)$ [veh/km/lane] in segment i of link m at time $t = kT_s$;
- average space-mean speed $v_{m,i}(k)$ [km/h] of vehicles in segment i of link m at time $t = kT_s$;
- traffic flow $q_{m,i}(k)$ [veh/h] leaving segment i of link m in time interval $[kT_s, (k+1)T_s]$.

Instead of the space-mean speed, also the time-mean speed can be adopted, for which estimation a method is proposed in [189] on the basis of locally measured vehicle speeds.

Different traffic flow models can here be adopted in varying degrees of accuracy, which again influences the complexity of the minimum cost-flow problem. Especially in case of the integration of the route guidance problem with control measures like variable speed limits and ramp metering signals, an accurate prediction of traffic flow is needed. Taking into account the trade-off between accuracy and computational speed, we chose to adopt the macroscopic METANET traffic flow model [115, 144]; for completeness the basic METANET equations as well as the modifications for the route-dependent variant needed for the current setting can be found in Appendix A. Note that other traffic models can also be adopted, e.g., the cell transmission model (CTM) [46] and link transmission model (LTM) [202]. The reader is referred to the literature for details; a comparison of the accuracy and complexity of several of these models can be found in [101].

A less accurate but computationally more efficient alternative for deriving $\tilde{v}_m(k)$ is based on the fundamental diagram, e.g., a piecewise affine diagram, together with the expression $q_m(k) = \rho_m(k) v_m(k) \lambda_m$, with λ_m the number of lanes of link m, i.e.:

$$\tilde{v}_m(k) = \begin{cases} v_{\text{free},m}, & \text{if } \rho_m(k) \leq \rho_{\text{crit},m}, \\ \dfrac{q_{\text{cap},m} - w(\rho_m(k) - \rho_{\text{crit},m})}{\lambda_m \rho_m(k)}, & \text{if } \rho_{\text{crit},m} < \rho_m(k) \leq \rho_{\text{max},m}, \end{cases} \tag{5.20}$$

where $v_{\text{free},m}$ denotes the free-flow speed on link m, $q_{\text{cap},m}$ the capacity flow, and $\rho_{\text{max},m}$, $\rho_{\text{crit},m}$ [veh/km/lane] respectively the maximum and critical density of link $m \in \mathcal{L}$. Instead of the piecewise affine fundamental diagram, the expression

$$\tilde{v}_m(k) = v_{\text{free},m} \exp\left[-\frac{1}{a_m} \left(\frac{\rho_m(k)}{\rho_{\text{crit},m}} \right)^{a_m} \right] \tag{5.21}$$

could be adopted, where a_m represents a model parameter [140].

The following two paragraphs describe simulation approaches that can be adopted, next to the different types of prediction models, to obtain an accurate prediction of link and route travel times.

Backtracking of Vehicles In order to predict an accurate travel time for traversing a link and hence to complete a route to which a vehicle is to be assigned, so-called backtracking of the individual vehicles[2] should be performed. In this backtracking procedure, depicted in Fig. 5.3, at each simulation time step in the period $\left[k_c T_c, (k_c + N_p) T_c\right]$ it is determined whether a vehicle is still predicted to be in a certain segment of a link or whether it is predicted to have left the segment, updating the segment travel time according to the dynamic speed profile $v_{m,i}(k)$. Here, as shown in Fig. 5.3, at the indicated 'jump points', the current speed $v_{m,i}(k)$ is updated to either $v_{m,i}(k+1)$ or $v_{m,i+1}(k)$. The final link costs for the particular vehicle is thus based on the (predicted) time difference between entering and leaving a link, where the link travel times are determined based on the updated velocities on the segments while the vehicle travels through the links during the prediction window.

[2] In fact, it is sufficient to distinguish between the homogeneous driver groups as each individual driver within a homogeneous group is assumed to act identically according to the decisions of the associated follower player.

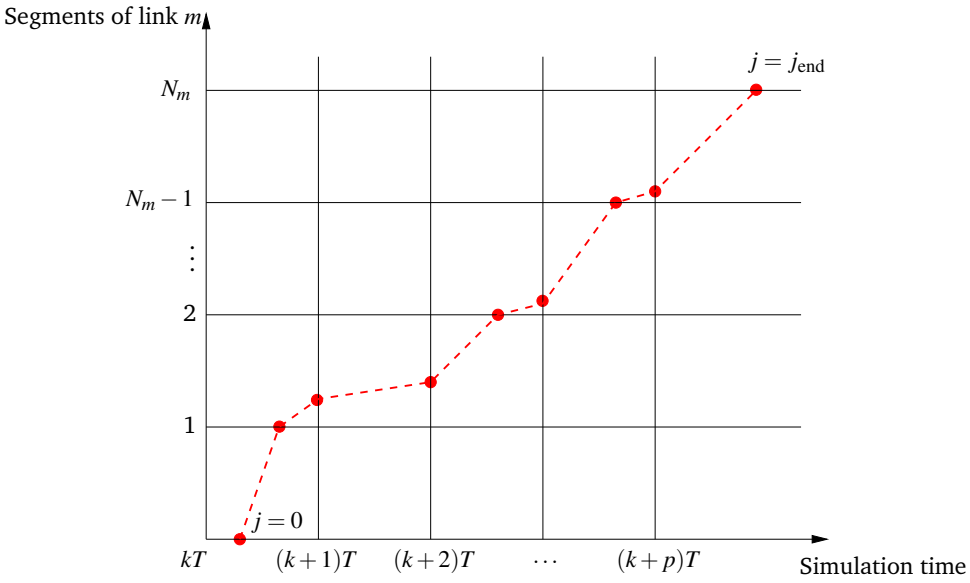


Figure 5.3: *Backtracking of the position of individual vehicles on a link [42].*

A Quasi-Dynamic Approach Instead of performing a computation time-intensive optimization to determine a system-optimal traffic distribution while adopting an accurate prediction model, the following quasi-dynamic approach can be adopted as suggested in [42]. Given a demand profile at a control time step k_c and the current velocities on the different links, these velocities are updated by iteratively computing a desired (optimal) distribution of traffic over the routes and by simulating the corresponding traffic flows for a chosen simulation horizon. By adopting a predicted demand profile for the simulation horizon, the situation is taken into account in which traffic with an overlapping path that entered the network later than a given vehicle takes over that vehicle and thus influences its course. Here, the desired (optimal) distribution is computed based on the link costs $c_m(k_c)$ that do not simply represent the expected travel time on each segment, but the summation of all predicted travel times for simulation steps $k = Mk_c, Mk_c + 1, \ldots, M\left(k_c + N_p\right) - 1$ in the prediction period $\left[k_c T_c, (k_c + N_p)T_c\right]$:

$$c_m(k_c) = \sum_{k=Mk_c}^{M(k_c+N_p)-1} \sum_{i=1}^{N_m} \frac{L_m/N_m}{v_{m,i}(k)}, \tag{5.22}$$

with N_m the number of segments of link $m \in \mathcal{L}$. The current velocities are then substituted by the predicted velocities, which update is performed for a fixed number of iterations or until the difference in, e.g., the link flows between two iterations is below a chosen threshold. In this manner, the predicted behavior of traffic is more accurately captured than in an approach in which travel times are simply predicted based on the evolution of the current traffic demand according to the prediction model.

(ii) Computing an Optimal Leader Function

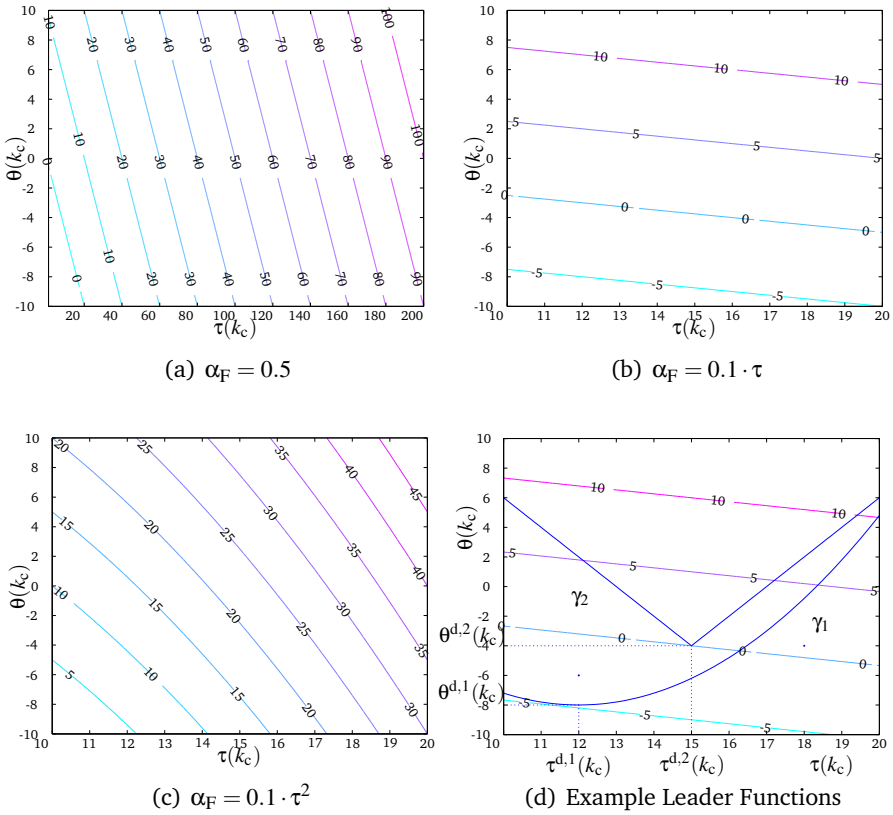
Given the desired traffic flow distribution $q_{\text{in}}^{hij,\text{d}}(k_\text{c})$ (travel time-based game) or $\zeta^{hi,\text{d}}(k_\text{c}) \cdot q_{\text{in}}^{hi}(k_\text{c})$ (splitting rate-based game) computed at the control time step k_c for all $h \in \mathcal{H}_i, i \in \{1,\ldots,N_\text{OD}\}, j \in \{1,\ldots,n_i\}$, the leader should associate with each feasible value of the follower variable a monetary incentive $\theta^{hij}(k_\text{c}) \in \Omega_{\text{F,t}}, \theta^{hi}(k_\text{c}) \in \Omega_{\text{F,s}}$, respectively. Optimal leader functions are those that cause the follower to adopt the associated desired travel time $\tau^{hij,\text{d}}(k_\text{c})$ or the desired splitting rates $\zeta^{hi,\text{d}}(k_\text{c})$.

Possible algorithms to compute optimal leader functions have been recapitulated in Section 5.3. Recall that in order for conditions C.1-C.3 (see Chapter 4, Section 4.2.1) to be satisfied, the leader function should remain outside the follower's sublevel set $\Lambda_\text{d} \setminus \{(u_\text{L}^\text{d}, u_\text{F}^\text{d})\}$, with $(u_\text{L}^\text{d}, u_\text{F}^\text{d})$ the decision variables desired by the road authority, i.e., $\big(\theta^{hij,\text{d}}(k_\text{c}), \tau^{hij,\text{d}}(k_\text{c})\big)$ or $\big(\theta^{hi,\text{d}}(k_\text{c}), \zeta^{hi,\text{d}}(k_\text{c})\big)$. If this does not hold, a rational follower would adopt a decision variable value other than the leader's desired, precomputed value.

For the current application, several level curves of the travel cost function $\jmath_{\text{F,t}}(\cdot)$ according to (5.13) are depicted in Fig. 5.4. Here, a constant value-of-time coefficient $\alpha_\text{F} \in \mathbb{R}^+$ as well as a linear and nonlinear relation $\alpha_\text{F} = 0.5 \cdot \tau, \alpha_\text{F} = 0.5 \cdot (\tau)^2$ are considered. For increasing values $\alpha_\text{F} \geq 0$, the slope of the level curves changes from horizontal lines to steep curves (Fig. 5.4), since the monetary incentive becomes the more significant decision criterion for the follower player.

It is important to note that the sublevel sets Λ_d associated with the (here, concave) level curves are convex, i.e., it is relatively easy to find optimal leader functions for which C1-C3 hold (see Chapter 3). While we refer to Chapter 4 for details on the available algorithms for computing leader functions, two possible optimal leader functions are indicated in Fig. 5.4(d), i.e., a quadratic (γ_1) and a piecewise affine (γ_2) mapping of desired travel time to a monetary value. In this example, the follower's optimal decision variables are $\tau^{\text{d},1}, \tau^{\text{d},2}$, associated with a monetary value $\theta^{\text{d},1}, \theta^{\text{d},2}$.

Remark 5.3 Note that any range of monetary incentives could be adopted as an image of the leader function given a follower's decision space. Since $\theta^{hij}(k_\text{c})$ and $\theta^{hi}(k_\text{c})$ are not included in the leader objective functions (5.11) and (5.9), the selection of the variable values $\tau^{hij,\text{d}}(k_\text{c})$ or $\zeta^{hi,\text{d}}(k_\text{c})$ by the follower is only dependent on the shape or geometrical specifications of the leader function. In other words, it is important that the leader function is designed such that $\tau^{hij,\text{d}}(k_\text{c})$ or $\zeta^{hi,\text{d}}(k_\text{c})$ is attained as a global minimum for the follower. Here, any value $\theta^{hij,\text{d}}(k_\text{c}) \in \Omega_\text{L}$ respectively $\theta^{hi,\text{d}}(k_\text{c}) \in \Omega_\text{L}$ can therefore be associated with the desired follower response. This implies that monetary values could be set, e.g., such that all homogeneous drivers with a given OD-pair and value-of-time, but assigned to different routes, yield the same objective function value. Differently, all drivers could be assigned a zero monetary value at the desired value of their decision variable. ◇



(a) $\alpha_F = 0.5$ (b) $\alpha_F = 0.1 \cdot \tau$

(c) $\alpha_F = 0.1 \cdot \tau^2$ (d) Example Leader Functions

Figure 5.4: *Level curves of the follower player in the travel time-based game for varying value-of-time parameters* α_F.

5.5 A Reverse Stackelberg Approach for Emission Reduction in Urban Corridors

A third and final application variant of the reverse Stackelberg game in traffic control is presented next.

5.5.1 Problem Statement

Different from the approaches presented in Section 5.4, the objective of the last game approach presented in this section is not to assign vehicles over the possible routes in a traffic network. Instead, the main factor of interest is traffic emissions and fuel consumption. In urban traffic networks, there is a significant exhaust of vehicular emissions due to the high frequency of stopping and taking off at intersections and traffic signals, where it is known that especially in the lower gears when starting up the motor, larger emission volumes are obtained and more fuel is con-

sumed [1]. Both in an uncongested state as well as during rush hours, vehicles have to stop as a simple result of controlling the (crossing) flows of motorized vehicles, pedestrians, and cyclists. At the same time, both vehicles in the mainstream traffic flow as well as vehicles on the cross roads like to minimize their total travel time, of which waiting time constitutes an important factor. This setting is depicted in Fig. 5.5.

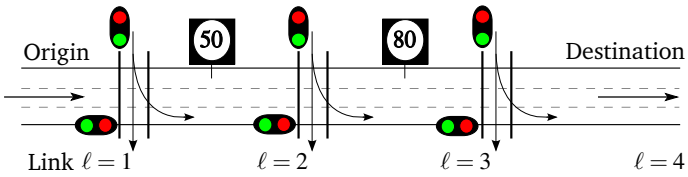


Figure 5.5: *Illustration of the urban corridor setting with variable speed limits and traffic signals for mainstream as well as traffic crossing and joining the mainstream.*

Results on integrated control strategies for reducing emissions and travel delays in a freeway context can be found in, e.g., [204], while an urban context is considered in [128]. Different from the latter results, we will propose again a hierarchical game-based approach in which the reverse Stackelberg game framework is adopted to allow the follower to make a trade-off between travel time and monetary incentives, where the leader can induce the follower to behave according to the desired or system-optimal speed and stopping profile as achieved by the on-ramp meters. As control handles to achieve the desired emission levels and total travel times, traffic signal timings for both mainstream and crossing vehicle flows are set, where in addition, variable speed limits can be considered to influence the travel time.

While the road authority is also interested in achieving a minimum TTS of vehicles in the urban traffic network, it is known that the usual speed allowed in urban areas under optimal, i.e., uncongested traffic conditions also causes the least emissions, contrary to situations in which the vehicles have to decelerate and accelerate. Emissions in urban areas are mostly influenced by these two aspects of acceleration and the average speed. The latter relation between speed and emissions is depicted in Fig. 5.6. Nonetheless, the TTS can also be incorporated as an optimization criterion of the road authority via, e.g., a soft constraint. Similarly, the emissions of crossing traffic can also be incorporated in the cost function of the road authority. Finally, also on-ramp waiting times of the crossing traffic can be considered, either as a hard constraint or as a term in the objective function of the road authority.

5.5.2 The Reverse Stackelberg Approach

The Players and Their Objectives and Decision Variables

The trade-off between the objectives of the road authority and the drivers is again modeled as a reverse Stackelberg game in which the road authority is represented by a **leader player** and a homogeneous class of drivers from the origin to the desti-

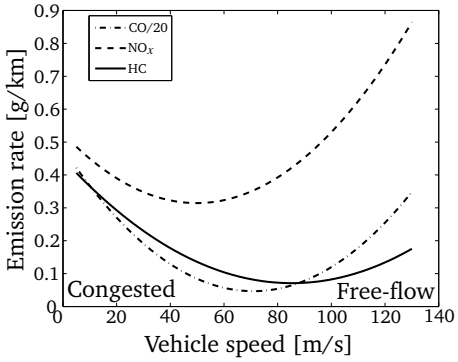


Figure 5.6: *General evolution of emission rates as a function of vehicle speed.*

nation by a **follower player**.

Since we consider a single link or origin-destination pair as depicted in Fig. 5.5 on which several traffic signals are necessary to let traffic of the side streets pass or join the mainstream road, we assume all mainstream drivers to behave alike and we thus consider all drivers entering the network between two consecutive control time steps as one follower player.

Assuming drivers to behave rationally, heterogeneity of drivers w.r.t., e.g., the value-of-time could be incorporated, thus proposing different leader functions that however result in the same optimal follower response. However, if drivers adopt suboptimal decisions, it may be infeasible to realize different speed and waiting time profiles for different classes of drivers. For the sake of conciseness, we omit heterogeneity indices in the following.

As in the travel time-based routing game, the associated **decision variables** represent:

- The monetary incentive $\theta(k_c) \in \Omega_L \subseteq \mathbb{R}$ to be awarded or collected by the **leader player** at control time step k_c, where $\Omega_L := [\theta_{min}, \theta_{max}]$, $\theta_{min} \in \mathbb{R}, \theta_{max} \in \mathbb{R}$ indicates an acceptable range of monetary incentive values.

- The predicted total time $\tau_u(k_c) \in \Omega_F \subseteq \mathbb{R}^+$ desired by the **follower player** for spending in the (urban) network including possible waiting times in a queue before entering the system. Again, $\Omega_F(k_c) := [\tau_{u,min}(k_c), \tau_{u,max}(k_c)]$ denotes the range of possible, i.e., realizable travel times for the incoming traffic, given the current state of the network.

Leader and Follower Objective Function

- Taking the road authority to be the leader in the game-based framework, we can write her cost function as follows:

$$J_L(k_c) = \tau_u(k_c) + \alpha_{L,\gamma} \sum_{g \in \Gamma} \gamma_g(k_c), \tag{5.23}$$

where $\gamma_g(k_c), g \in \Gamma$ denotes the emissions or fuel consumption of type g caused by the mainstream traffic entering the system in the time window $[k_c T_c, (k_c + 1)T_c]$, with $\Gamma := \{\text{CO emission}, \text{CO}_2 \text{ emission}, \text{HC emission}, \text{NO}_x\text{emission}, \text{fuel consumption}\}$. These emissions are a function of the number of vehicles, acceleration, and velocity, the values of which are influenced by the control handles of the road authority as will be explained below. Further, $\alpha_{L,\gamma} \in \mathbb{R}^+$ represents a weighing coefficient in which also the conversion of units is incorporated.

Note that no effective traffic demand (in veh/h) needs to be incorporated at this stage, as the drivers entering the system are considered as one player. For the computation of a system-optimal profile of traffic signals and speed limits as elaborated upon in the following paragraph, this explicit value of traffic demand as well as the values of predicted traffic demand will be incorporated.

- The objective of the drivers is equivalent to (5.13) as adopted in the travel time-based reverse Stackelberg game in freeway traffic networks, i.e.,

$$\jmath_F(k_c) = \alpha_F \cdot \tau_u(k_c) + \theta(k_c), \tag{5.24}$$

with $\theta(k_c) \in \Omega_L \subseteq \mathbb{R}$ the positive, zero, or negatively-valued monetary incentives that are respectively provided by, neutral, or paid to the leader at control time step k_c. Further, $\tau_u(k_c) \in \Omega_F(k_c) \subseteq \mathbb{R}^+$ denotes the travel time in which the follower wants to reach his destination, where the follower's decision space Ω_F is time-variant as it is subject to feasibility bounds that depend on the state of the traffic network. The factor $\alpha_F \in \mathbb{R}^+$ again represents the drivers' value-of-time.

The Dynamic Game Framework

The reverse Stackelberg approach for urban corridors consists of the following main steps:

(i) Given the traffic state at the control time step k_c and the incoming mainstream traffic demand during the time window $[k_c T_c, (k_c+1)T_c]$ as well as a prediction of the demand of mainstream as well as of side-street traffic for a horizon $[k_c T_c, (k_c + N_p)T_c]$, a system-optimal profile of speed limits and green phases at the traffic signals for the traffic entering the mainstream corridor is computed by the road authority, together with the corresponding desired predicted travel times $\tau_u^d(k_c)$.

In particular, adopting a prediction horizon as in the previous approaches of Section 5.4, the overall objective based on (5.23) with the effective demand and queue length before entering the system explicitly incorporated, can be written as:

$$\begin{aligned} \min_{\substack{\tau_{\text{wait},\ell}(k_c),\ell\in\mathcal{U} \\ v_{\text{ctrl},\ell}(k_c),\ell\in\mathcal{U}}} \jmath(k_c) \quad = T_s \sum_{j=1}^{N_p \cdot M} \Big(\sum_{\ell\in\mathcal{U}} \Big(& \alpha_{L,\tau}\tau_{u,\ell}(k_c+j) q_{\text{in},\ell}(k_c+j) + \\ & \alpha_{L,\gamma} \textstyle\sum_{g\in\Gamma} \gamma_{g,\ell}(k_c) \Big) + w(k_c+j) \Big), \end{aligned} \tag{5.25}$$

where the urban corridor is divided into links $\ell \in \mathcal{U}$ that end with a traffic signal with the corresponding index ℓ as depicted in Fig. 5.5 (the last link is allocated a virtual traffic signal for consistency).

Here, the total emissions and travel time are straightforwardly composed according to $\gamma_g(k_c) = \sum_{\ell\in\mathcal{U}} \gamma_{g,\ell}(k_c)$ and $\tau_u(k_c) = \sum_{\ell\in\mathcal{U}} \tau_{u,\ell}(k_c) + w(k_c)$. The travel time on link ℓ for traffic entering link ℓ during the time interval $[k_c T_c, (k_c+1)T_c]$ can be written:

$$\tau_{u,\ell}(k_c) = \tau_{nom,\ell}(k_c) + \tau_{wait,\ell}(k_c), \tag{5.26}$$

where $\tau_{wait,\ell}(k_c) \in \mathbb{R}^+$ denotes the waiting time for mainstream traffic signal and where $\tau_{nom,\ell}(k_c) \in \mathbb{R}^+$ denotes the nominal or average travel time for a mainstream driver through link ℓ that entered during $[k_c T_c, (k_c+1)T_c]$, which can be derived according to[3] $\tau_{nom,\ell}(k_c) = L_\ell / v_{ctrl,\ell}(k_c)$. The variable speed limit $v_{ctrl,\ell}(k_c)$ for link ℓ at control time step k_c is adopted as a control handle of the road authority to influence both emissions and the total travel time. Further, $\alpha_{L,\tau} \in \mathbb{R}^+$ represents another weighing coefficient.

The waiting time $\tau_{wait,\ell}(k_c)$ for mainstream traffic during the time interval $[k_c T_c, (k_c+1)T_c]$ implies an equivalent time window of green time for the traffic queuing at the side-street traffic signal with index ℓ. While we do not elaborate on the modeling of this queue, the road authority may consider a soft or hard constraint to limit the waiting time of traffic at the cross roads. Further note that after each green phase, a fraction of side-street traffic will join the mainstream. A prediction of this division, e.g., based on historical data, should be incorporated in the traffic flow model.

Further, the effective traffic demand entering link ℓ during the time interval $[k_c T_c, (k_c+1)T_c]$ is denoted by $q_{in,\ell}(k_c)$. In addition, $w(k_c)$ denotes the waiting time for (a part of) the total demand in case the first link of the corridor is saturated.

Finally, the total emissions and fuel consumption can be derived from the acceleration and the velocity of vehicles, as briefly elaborated upon in the paragraph on prediction models below.

(ii) Given the *desired*, optimal profile of speed and waiting times and the resulting predicted mean travel time, an optimal **leader function** $\gamma_L(k_c) : \Omega_F(k_c) \to \Omega_L$ can be computed by the leader.

Since the follower's objective is equivalent to the cost function (5.13) described in the travel time-based game in a freeway context, for the derivation of an optimal leader function, the analysis of Section 5.4.2(ii) also applies here.

(iii) As a response to an optimal leader function, a rational follower will adopt a pair of monetary incentive and travel time, which the leader associates with the optimal profile of variable speed limits and traffic signals.

[3] A more accurate description of the speed as a function of density and variable speed limit is given in Appendix B, Equation (B.4).

Prediction Models

In order to be able to make accurate estimations of travel time and emissions that are necessary to compute a desired, system-optimal traffic behavior according to (5.23), good prediction models are necessary that at the same time are fast enough to deal with in dynamic optimization. For urban traffic networks, the BLX model [128] can be adopted as a possible prediction model. The reader is referred to the literature for an elaborate description of different traffic models [101, 107].

In the VT-micro model [1], values of the vehicular emissions CO, NO_x, HC [g] and of fuel consumption [l] are determined based on the mean velocity and acceleration of an individual vehicle in the set I of vehicles at every time step, i.e.,

$$\gamma_\ell(k) = \sum_{i \in I} \exp\left(\check{v}_{\ell,i}^{\mathrm{T}}(k) P_\gamma \check{a}_{\ell,i}(k)\right), \tag{5.27}$$

with the speed and acceleration vectors $\check{v}_{\ell,i}^{\mathrm{T}}(k) = [1 \;\; v_{\ell,i}(k) \;\; v_{\ell,i}^2(k) \;\; v_{\ell,i}^3(k)]$ and $\check{a}_{\ell,i}^{\mathrm{T}}(k) = [1 \;\; a_{\ell,i}(k) \;\; a_{\ell,i}^2(k) \;\; a_{\ell,i}^3(k)]$, and where P_γ denotes the matrix of model parameters for $\gamma \in \Gamma = \{\text{CO emission, HC emission, } NO_x \text{ emission, fuel consumption}\}$, the values of which can be found in [1].

In addition to adopting an accurate prediction model for the velocities and accelerations needed to determine the emission values and travel times, backtracking of vehicles may be adopted as has been explained in Section 5.4.2(i) to accurately predict when vehicles enter and leave a link.

Discussion

It is important to note that the problem is formulated as a single leader-single follower game in which the driver class for a particular OD-pair is taken to be homogeneous. Only in case it would be feasible to consider different traffic signals for different classes of vehicles, e.g., in case of strictly separated lanes on a homogeneous segment of the corridor, a distinction between classes can be made.

Further, the travel time of side street traffic is directly considered in the objective function of the road authority. Alternatively, the waiting time may be posed as a constraint, focusing solely on the minimization of vehicular emissions and fuel consumption.

Finally, different from the approaches discussed in Section 5.4 in which no traffic signals can bring individual travelers to behave according to the system-optimal behavior (i.e., adopt a certain route), in the urban game setting described in Section 5.5, traffic signals and variable speed limits can in fact enforce[4] the desired speed and acceleration scheme. In particular, the approach can be adopted in combination with the travel time-based routing approach in a freeway context in order to simplify reaching the desired travel times promised by the road authority.

[4] Under the assumption of fully rational players; for penalty schemes implemented to mitigate disobedience outside the game context, we refer once more to the discussion paragraph in Section 5.6.2.

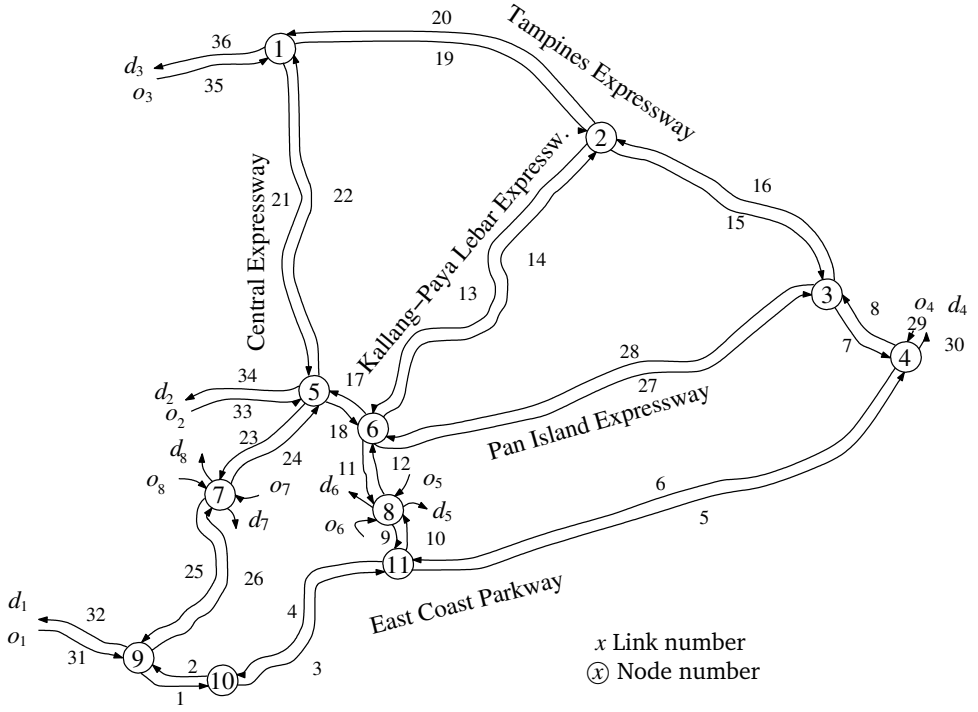


Figure 5.7: *Ring road network of the city of Singapore.*

5.6 Case Study

In this section a simulation study of a real-life multi-route freeway traffic network is considered in order to give an indication of the performance of the proposed approach. In order to compare the TTS obtained when using the travel time-based reverse Stackelberg game, we also simulate the traffic behavior when adopting dynamic route information panels (DRIPs).

5.6.1 Set-up

A real-life traffic network based on the Singapore ring road network is considered in which several road segments are shared by different routes associated with the same as well as with different origin and destination pairs, as depicted in Fig. 5.7.

It is assumed that leader functions are computed and proposed to new drivers each control time step corresponding to the control time interval $T_c = 1$ min with a simulation time step or sampling time step of $T_s = 10$ s and a simulation time of two hours. For online implementation, this is feasible if drivers enter their travel plans at least a time period consisting of this control time window plus the maximum time required for the computation of an optimal leader function, before departure. Further, the prediction horizon adopted to take into account future traffic conditions is

Table 5.1: *Comparison of TTS for different routing methods.*

Scenario	TTS (veh·h)
Shortest route	$1.089 \cdot 10^3$
Fastest route	$1.109 \cdot 10^3$
DRIP	$1.109 \cdot 10^3$
RSG	955.6

set to $N_\mathrm{p} = 30$, implying a prediction time duration of 5 min. The METANET traffic model is adopted to simulate the traffic behavior, where we adopt a critical density $\rho_{\mathrm{crit},m} = 27$ veh/km/lane, a free-flow speed of $v_{\mathrm{free},m} = 90$ km/h, and model parameters $a_m = 2.34, \eta = 30$ km^2/h,$\tau = 10$ s,$\kappa = 20$ veh/km. Further, we consider a traffic demand profile at origin o_4 to destinations $d_5 - d_8$, according to which the inflow of the network increases from 0 veh/h to 8000 veh/h during the first half hour, stays at 8000 veh/h for one hour, and then decreases back to 0 veh/h during the last half hour.

The TTS achievable by application of the travel time-based reverse Stackelberg approach ('RSG') is compared to three scenarios. The 'shortest route' scenario represents the TTS value obtained when travelers entering the system take the route that they are informed to be the route of shortest distance to their destination. The TTS for the 'fastest route' scenario represents the TTS value obtained when travelers entering the system take the route that they are informed to be associated with the shortest travel time to their destination, based on current traffic conditions. Finally, the TTS is considered that is obtained when DRIPs are adopted. Here, we assume that the travelers split according to the following simple logit model [43]:

$$\beta_i^{n_i}(k) = \frac{\exp\left(-\sigma t^{ij}(k)\right)}{\sum_{j=1}^{n_i} \exp(-\sigma t^{ij}(k)}, \qquad i \in \{1,\dots,N_{\mathrm{OD}}\}, j \in \{1,\dots,n_i\}, \tag{5.28}$$

where $\sigma \in \mathbb{R}$ indicates the sensitivity of a driver to changes in travel time, similar to the value-of-time, and with $t^{ij} \in \mathbb{R}^+$ the predicted travel time for the route $j \in \{1,\dots,n_i\}$ available for the OD-pair $i \in \{1,\dots,N_{\mathrm{OD}}\}$, as derived from the travel time prediction performed in the fastest route scenario.

The TTS values that resulted from the simulations when adopting the different approaches can be found in Table 5.1. The equivalent results of the DRIP and the fastest route-based approach can be explained by the fact that only a few alternative routes from origins to destinations are available in the given network; the travel times need to differ significantly in order to use alternative routes. Further, the worse performance of the fastest-route scenario as compared to the shortest-route may be explained by the model mismatch of the prediction and the actual simulation of traffic behavior.

5.6.2 Discussion

The main implications of using different follower decision variables in the travel time-based and route splitting rate-based approaches are in (i) the realization of

suboptimal follower decisions and in (ii) the dimension of the leader function domain.

While the benefit of the splitting rate-based approach as compared to the travel time-based approach is in the fact that any (suboptimal) route splitting rates can be achieved in contrast to suboptimal desired travel times in the travel time-based approach, there are also computational considerations. Both aspects will be briefly considered next.

Robustness Considerations So far, we have assumed a perfect situation in which both the road authority and the drivers behave as fully rational players, optimizing their respective goal functions for the given decision spaces.

In order to account for deviations from the expected (and *promised*) travel time due to unexpected traffic conditions, the following measures could be taken:

- Provide updates of the traffic conditions through an updated leader function for traffic that is already in the system. Here, drivers may get assigned a new route or speed profile while the monetary incentives can be re-computed. In particular, at each node the updated optimal splitting rates could be considered rather than route-splitting rates. However, the increased computational complexity as well as the increase in online and on-board communication would have to be evaluated.
- Work with desired travel time windows as the followers' decision variable instead of a precise value in minutes in the travel time-based approaches.
- In cases where lanes of a homogeneous freeway stretch are strictly separated, one can allow for different speed limits on these different lanes.

The third, urban game variant can also be integrated with the travel time-based or splitting rate-based freeway approaches to influence the travel time by variable speed limits and ramp metering in a freeway context, which we leave for future research.

Computational Considerations As regards the computational complexity of the different game variants, it should be noted that the decision space of the follower is inherently multidimensional in the splitting rate-based game. The leader function should allocate a monetary incentive to each possible combination of route splitting rates for a given OD-pair. The derivation of such an $n_{\mathrm{F,s}}$-dimensional optimal leader function (a hyperplane) in a space of dimension $n_{\mathrm{F}}+1$ as $n_{\mathrm{L}}=1$ is in general a more complex problem than the derivation of a curve for $n_{\mathrm{F},t}=1$.

At the same time, $N_{\mathrm{F,s}} \leq N_{\mathrm{F,t}}$, implying the need for at least as many one-dimensional leader functions in the travel time-based game.

5.7 Discussion

In this chapter we have described how the reverse Stackelberg game can be used for application in traffic control, i.e., as a means for achieving a system-optimal

evolution of traffic states, with as one of the possible criteria the minimization of the total time spent by vehicles in a network by influencing the traffic distribution over the available roads in a network.

In particular, the leader player is taken to represent the road authority, while homogeneous drivers with the same value-of-time and origin and destination pair are represented by a follower player. Three reverse Stackelberg game variants have been proposed, according to which the decision variables to which the leader assigns a positive or negative monetary incentive is taken to be either (1a) the desired travel time or (1b) the route splitting rates of a homogeneous group of vehicles in a freeway context, or (2) the desired travel time in an urban setting. In the splitting rate-based case, the distribution of individual drivers within the group is taken to be a lower-level problem that is not explicitly addressed in this dissertation. Further, while the system-optimal distribution of traffic amongst possible routes is considered in a freeway network with overlapping routes, in a single-corridor urban traffic context, the game is adopted to minimize vehicular emissions.

The proposed game approaches are conceptually different from road tolling approaches in which tolls are determined for road stretches, as it is easier in the former case to differentiate between heterogeneous drivers. Moreover, one can rely on the theory developed on, e.g., the complexity and solution methods for the reverse Stackelberg game. A simulation study of a real-life ringroad network showed the possible improvements when adopting the game approach in comparison with cases in which drivers take the shortest respectively fastest route, or when dynamic route information panels are considered.

In principle, a system-optimal state with respect to any objective can be achieved, provided that the drivers' decision spaces, i.e., the ranges of feasible travel times in which they can arrive at their destination, are considered in the determination of the system optimum. In the current setting, we assume that the road authority informs the drivers of their decision space of feasible travel times.

In future work, real-life implementability concerning unexpected deviations in an uncertain traffic setting should be considered. Also, an analysis of the possible leader function structures – ranging from smooth curves to piecewise affine functions – should be made as regards the acceptance and ease of employment by the drivers in a real-life scenario. Further, the freeway and urban applications can also be integrated, i.e., by considering route guidance in urban traffic networks or by adopting the control measures as described in the single-corridor urban setting also in a freeway network. In such a situation, integrated variable speed limit and ramp metering control can be adopted, which is discussed as a separate topic in Appendix B.

Chapter 6

Conclusions and Recommendations

As stated in the introduction to this dissertation, while a broad body of results is available regarding the original Stackelberg game, this does not apply to the – more general – reverse Stackelberg game. In this thesis, we have therefore presented extensions to the available theory, in particular aiming at the development of methods to systematically obtain solutions to this game. Several applications in the context of traffic control were further discussed as one of many areas in which the reverse Stackelberg game can be applied to structure decision making. In this final chapter, first the main findings from the previous chapters will be summarized, after which several suggestions and recommendations are discussed on topics that can prove to be interesting for further research in the area of reverse Stackelberg games.

"*A conclusion is the place where you got tired of thinking.*" ...
- Martin H. Fischer -

6.1 Conclusions

The reverse Stackelberg game is considered as a framework for hierarchical or sequential decision making that applies, e.g., in multi-level decision or optimization-based control problems. This game evolves around the so-called leader function or mapping of a follower's decision space into the leader's decision space, which the leader proposes to a follower player. The well-known Stackelberg game can be perceived as a special case of the reverse Stackelberg game in which the image of the leader function is a singleton. In case a follower's response to this leader decision variable value is nonunique, the leader can no longer control the follower's decision, unless she proposes a more involved leader function, i.e., applying a *reverse Stackelberg game*. In spite of these merits, the reverse Stackelberg game has received much less attention in comparison with the original, purely hierarchical Stackelberg game. We therefore see much potential in considering the reverse Stackelberg game at least in those problems to which the original Stackelberg game has been applied.

Having considered the research conducted in the area of reverse Stackelberg games since the 1970s, a list of open issues resulted, both summarized in Chapter 2. We identified the lack of systematic solution approaches for the indirect formu-

lation of the reverse Stackelberg game – in which the leader function is derived for a given equilibrium point desired by the leader – as the main challenge to be addressed first, as it prohibits the game from being applicable in general settings without restrictive assumptions regarding, e.g., smoothness and (strict) convexity of leader and follower objective functions. In this analysis, affine and nonlinear leader function structures are distinguished between in Chapter 3 and 4 respectively, where the leader function in the latter case is represented by a linear combination of basis functions. Finally, in order to frame the applicability of the reverse Stackelberg game in a new setting, suggestions for the application of the game in a traffic control context have been presented in Chapter 5.

The contributions of the conducted research are elaborated upon below, followed by suggestions for improvements as well as extensions.

Contributions The main contributions of the work presented in this dissertation can be enumerated as follows and will be summarized in the remainder of this paragraph.

1. We have developed necessary and sufficient conditions for the existence of an optimal affine leader function assuming an unconstrained decision space.
2. We have derived a characterization of the set of optimal affine leader functions for unconstrained decision spaces, and we have explained how this set can be reduced to obtain the set of optimal affine solutions in case of a constrained decision space.
3. We have developed three numerical methods to compute optimal nonlinear leader functions for the indirect formulation of the reverse Stackelberg game.
4. We have proposed three realizations of the reverse Stackelberg game in urban and freeway traffic networks and showed how implementing such games can result in system-optimal traffic behavior.
5. We have considered the computational intractability of nonconvex model predictive traffic and emission control and we have adopted piecewise affine approximations of the nonlinear model equations, resulting in a mixed-integer linear programming problem.

Affine solutions:
Before the reverse Stackelberg game is ready to be considered as a viable decision framework in various settings, first, a solution method to tackle the general reverse Stackelberg game should be available, which is not yet the case.

As a first step towards obtaining a systematic solution approach, leader functions of the affine structure have been considered, in particular by providing necessary and sufficient conditions for the existence of an optimal affine leader function in the general static, deterministic single leader-single follower game. With these results, it can thus be first verified whether an – easy to derive – optimal affine leader function exists, instead of adopting more complex, nonlinear function structures. Here, restrictive assumptions on (locally strict) convexity and differentiability of the follower objective function that are often required in the literature have been relaxed.

Based on these conditions, a characterization of the full set of optimal affine leader functions in an unconstrained decision space has been derived. Moreover, it has been shown how a reduced set of optimal affine leader functions can be obtained in case of constrained decision spaces. This set can subsequently be used for further optimization, e.g., as a result of a sensitivity analysis on the deviation of a follower from his optimal response.

Nonlinear solutions:
As only a subset of possible problems can be covered for which an affine leader function results in the leader's desired equilibrium and moreover, as leader functions of a nonlinear structure can have beneficial characteristics, e.g., regarding sensitivity, next, nonlinear leader functions have been considered. For this case, instead of focusing on existence conditions, computational methods have been provided for systematically deriving an optimal leader function written as a linear combination of basis functions, i.e.:

- a continuous multi-level optimization program;
- a grid-based variant thereof;
- a heuristic interpolating spline approach.

These methods – devised for the indirect formulation of the reverse Stackelberg game in which the leader's desired equilibrium point is computed before the derivation of a leader function is made – have been compared with heuristic evolutionary approaches that have been proposed in the literature for the direct game variant, i.e., based on either a genetic algorithm or an artificial neural network approach. Regarding the computational complexity of the solution methods, while these are in general nonconvex optimization problems, in comparison with the evolutionary approaches, our methods for the indirect game variant proved significantly more time-efficient in the case studies considered. At the same time, these indirect methods proved to be able to compute an optimal (polynomial) leader function – that resulted in the desired leader equilibrium, while the evolutionary approaches resulted in a suboptimal solution. Whereas the third direct interpolation approach is most efficient, it is also mostly restricted to two- and three-dimensional decision spaces due to the lack of feasibility conditions.

Examples mostly deal with two- and three-dimensional decision spaces for the sake of exposition. Nonetheless, the characterization as well as the computational methods involve an arbitrary (finite) number of decision elements of both leader and follower.

Applications:
Further, in order to contribute to the available applications of the reverse Stackelberg game, new applications of the game have been introduced in the area of traffic control and route guidance. There, the leader and follower players represent the road authority and a group of homogeneous drivers with respect to their origin and destination and value of travel time, respectively. While the applications are

not intended to be readily implementable in real-life, it has been shown how the game approaches can ideally result in a system-optimal traffic behavior, e.g., with respect to the total time spent by vehicles in a network. Suggestions for real-life implementation are provided. In particular, the following three games have been proposed, where the domain of the leader functions and the type of traffic network are distinguished between:

- a game with a leader function mapping the desired travel time to monetary incentives, applied to a dynamic routing approach in freeway networks;
- a game with a leader function mapping the route splitting rates to monetary incentives, applied to a dynamic routing approach in freeway networks;
- a game with a leader function mapping the desired travel time to monetary incentives, applied to a single-corridor urban network aiming to reduce vehicular emissions.

Model predictive traffic control (Appendix B):
In the context of the applications just discussed, it should be mentioned that in Appendix B integrated model predictive traffic and emission control has been studied as a means to reduce the total time spent of vehicles as well as the vehicular emissions in a freeway network. In particular, the computational intractability was addressed that applies when the nonlinear METANET and VT-macro traffic flow and emission prediction models are applied in a model predictive control (MPC) framework.

To this end, we have approximated the nonlinear model equations of the METANET and the VT-Macro models with piecewise affine functions. Subsequently, MPC was applied to the resulting mixed-logical dynamic prediction model, leading to a mixed-integer linear programming problem instead of an MPC controller that is based on nonconvex optimization. The control procedure including possible approximation procedures has been described and the approach has been applied to a freeway traffic network where the aim was to optimize the total time spent of traffic in the system as well as the vehicular emissions by means of both variable speed limits and on-ramp metering.

While the method showed an improved computational efficiency for the given case study, this was achieved at the cost of some deterioration of the control performance. Moreover, since (mixed) integer linear programming still falls within the complexity class of NP-hard problems, for application to larger networks, methods to speed up the optimization as well as to improve efficiency regarding the model predictive control framework should be investigated.

... "Life can only be understood backwards; but it must be lived forwards."
- Sören Kierkegaard -

6.2 Directions for Future Research

There is still a long road ahead before it can be stated that a systematic solution approach exists for the general reverse Stackelberg game.

In order to further expand the current theory, several of the open issues that have been identified from the literature and that have been listed in Chapter 2 (Section 2.6) should be considered. As regards the contributions presented in this thesis, several extensions of the derivations, solution methods, and applications can be found in the intermediate discussion sections. Here, a selection of the perhaps most promising topics is made.

General extensions In the largest part of the current dissertation, the single leader-single follower, static deterministic reverse Stackelberg game has been considered. The most obvious extensions are therefore in:

- extension to dynamic problems;

  While dynamic discrete-time as well as continuous-time differential Stackelberg games have been considered, no methods for deriving more general leader functions in such dynamic settings have been developed yet.

- extension to multi-player and multi-level problems;

  Possible schemes of interaction and cooperative as well as noncooperative games played amongst multiple players on a level should be investigated.

- extension to stochastic problems;

  Both cases in which the follower does not behave fully rationally and in which the leader lacks information on the follower that is crucial to derive an optimal strategy should be considered.

More specific suggestions for further research are considered in the following categories, the first three following the contents of Chapter 3 – Chapter 5 in corresponding order.

Theoretical considerations

- General existence conditions.

  No existence conditions have yet been derived for optimal leader functions of a general (nonlinear) structure. With the current solution methods nonlinear leader functions are computed without knowing whether such an optimal function in fact exists. While the derivation of such general existence conditions is a challenging task, instead, only necessary conditions for the existence of a continuous leader function can be derived that could apply, e.g., to cases where the follower sublevel set for the desired leader equilibrium intersects with the boundaries of the leader's decision space.

- Implicit leader functions.

  In case no optimal or even feasible explicit, continuous leader function exists (see e.g., Chapter 4, Fig. 4.4(a)), an implicit formulation may be considered to obtain the leader's desired equilibrium, allowing a follower decision to map to a nonunique element of the leader's decision space (i.e., an inverse mapping γ_L^{-1} that is surjective). Likewise, discontinuous, discrete- and set-valued leader functions can be considered.

- Set of desired leader equilibria.

  Instead of considering a particular selection of the leader's desired equilibrium point, a set of desired equilibria could be considered at once, e.g., deriving leader functions that pass through several desired equilibria that are equivalent in value. The equilibria may be weighted, especially in case also suboptimal equilibria are considered.

- Inclusion of the follower perspective.

  Instead of solely focusing on the leader's desired (global) optimum, the associated follower objective function value should also be considered in case the leader can select one of several desired equilibria. Moreover, a cooperative approach may be adopted in which the leader could incorporate the follower perspective directly when determining a desired equilibrium that is not solely based on the leader objective function.

- Sensitivity analysis.

  An explicit connection between a given leader function and the likelihood of deviation from the optimal follower response should be made, e.g., based on the closeness of a pair of the leader function argument and the image thereof to the boundary of the applicable follower sublevel set. When considering such a suboptimal response of the follower, the extent of deterioration that this would imply for the resulting suboptimal leader objective function value should be considered too.

- Suboptimal leader functions.

  Similarly, in case no optimal leader function exists, the implications of the selection of a suboptimal leader function for both leader and follower should be characterized.

Computational methods

- Computational analysis.

  Further assessments should be made of the computation time required to obtain optimal parameters for various basis functions. There, the convergence of the (evolutionary) approaches should also be taken into account. In addition, the scalability of the computational approaches to derive nonlinear leader functions should be investigated for high-dimensional decision spaces.

- Conditions on gridding precision.

  A formal optimality result should be obtained regarding the required gridding precision. Such conditions could be related to the geometry of the boundary of the follower sublevel set.

Real-life applicability

- Traffic applications.

  For the real-life applicability of the reverse Stackelberg game in a traffic context, in particular mechanisms to deal with unexpected changes in the predicted, *promised* behavior should be investigated, as well as restrictions in bandwidth for transferring the information load and uncertainties related to packet losses and speed for the communication amongst vehicles and between vehicles and the road authority.

- Leader function structure.

  Another crucial aspect to take into account in real-life applications is the type of leader functions that follower players are able to process. Unless a leader function is presented as a black box, leader functions for human players may need to be restricted to simple curves. In case a follower is an artificial agent, e.g., a controller purely dealing with the optimization problem, or an agent that has access to a solver for determining his optimal response to a leader function, only the (machine) precision is relevant. Moreover, in all settings in which the players are human, a complex leader function covering several decision spaces may be unreasonable. Here, a lower-level game could be defined in which the problem can be further decomposed.

Connection with other disciplines

While links between reverse Stackelberg games and the theory of incentives or contract theory as well as with mechanism design have been mentioned in Chapter 2, the current dissertation does not include an analysis of results within those fields. In particular, these disciplines could be consulted for insight in cases with incomplete information for the leader, when aiming to obtain a solution that is as good as possible for the leader while at the same time trying to induce the followers to convey their hidden information truthfully.

Final Outlook

Perhaps the main reason for considering the reverse Stackelberg game in multi-level decision problems rather than adopting the purely hierarchical Stackelberg game, is the fact that instead of a direct leader decision a leader *function* is adopted in order to account for a nonunique follower response. Following this reasoning, in this dissertation we have pursued the leader's aim to derive leader functions such that her desired optimum is achieved, where the leader function can be a very 'artificial' function, such as a highly piecewise nonlinear function with many local optima.

As suggested for future consideration regarding real-life applicability, this seems reasonable to the extent that an optimization program can *in principle* deal with such complex leader functions. Especially if the players embody human participants rather than artificial entities like controller units, however, such a complex leader function may be unreasonable. Moreover, the leader's problem of determining her global optimum for which the leader function is derived in this thesis, can be a complicated task.

A quite different perspective is to adopt a reverse Stackelberg game as a kind of feedback mechanism in a multi-stage context. Here, instead of perceiving the leader function as an incentive – perhaps better called manipulation – device, it enables the follower to at once decide upon his own decision variable as well as the associated leader decision. Providing the follower with decision making *freedom*, the leader can in turn extract useful hidden information from the follower's response.

Hence, while in the current thesis our focus has been foremost on achieving the pair of leader and follower decision variables that is desired by the leader, in future research, one may focus in addition on adopting a leader function as a mechanism to obtain feedback from a follower player, analogously to in the area of observers and estimation.

Appendix A

The METANET Model

For the sake of completeness, in this appendix chapter the basic METANET model for traffic flow used in Appendix B is provided together with the modifications necessary to obtain a route-dependent METANET model as is required for implementation of the reverse Stackelberg routing approaches of Chapter 5.

We first present the equations for the destination-independent mode in the following Section A.1, which is gratefully adopted from [92] with slight modifications. Subsequently in Section A.2 we give an extension to a route-dependent model as is used in Chapter 5. For the destination-dependent variant and for extensions with respect to, e.g., the modeling of dynamic speed limits and mainstream metering, we refer to [92].

A.1 The Basic METANET Model

The METANET traffic flow model, originally developed by Papageorgiou and Messmer [112, 144] is adopted in the case studies reported in Chapter 5 and Appendix B. This model was chosen because it provides a good trade-off between simulation speed and accuracy [114, 151]. The fact that this model is deterministic, discrete-time, discrete-space, and macroscopic makes it very suitable for model-based traffic control. Since the simulation time step and the segment length of the discretized freeway are relatively large, simulations of this model can be executed very fast.

Regarding the validation of the model we refer to [151, 157]. The reported validation results are in general satisfactory. Furthermore, the small number of parameters makes it easy to calibrate. An important property for model-based traffic control is that this model (including extensions) can reproduce the capacity drop at on-ramps and in shock waves. In the subsequent sections we describe the METANET model and a few extensions to this model that are relevant for the current thesis.

For the full description of METANET we refer to the literature [112, 115, 158].

A.1.1 Link Equations

The METANET model represents a network as a directed graph with the links (indicated by the index m) corresponding to freeway stretches. Each freeway link has uniform characteristics, i.e., no on-ramps or off-ramps and no major changes in geometry. Where major changes occur in the characteristics of the link or in the road geometry (e.g., at an on-ramp or an off-ramp), a node is placed. Each link

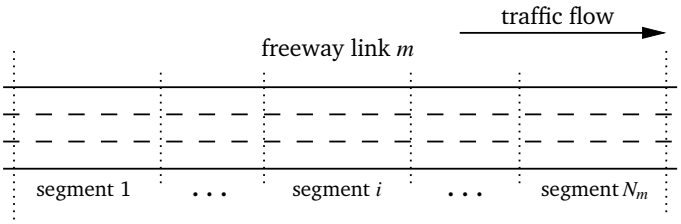


Figure A.1: *In the METANET model a freeway link is divided into segments.*

m is divided into N_m segments (indicated by the index i) of length L_m (typically 500-1000 m); see also Fig. A.1). Each segment i of link m is characterized by three quantities:

- *traffic density* $\rho_{m,i}(k)$ (veh/km/lane),
- *mean speed* $v_{m,i}(k)$ (km/h),
- *traffic volume* or *outflow* $q_{m,i}(k)$ (veh/h),

where k indicates the time instant $t = kT_s$, and T_s is the time interval used for the simulation of the traffic flow (typically $T_s = 10\,\text{s}$). For stability, the segment length and the simulation time interval should satisfy:

$$L_m > v_{\text{free},m} T_s \tag{A.1}$$

for every link m, where $v_{\text{free},m}$ is the average speed that drivers assume if traffic is freely flowing.

The outflow of each segment is equal to the density multiplied by the mean speed and the number of lanes on that segment (denoted by λ_m):

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) v_{m,i}(k). \tag{A.2}$$

The density of a segment equals the previous density plus the inflow from the upstream segment, minus the outflow of the segment itself (conservation of vehicles):

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T_s}{L_m \lambda_m}\big(q_{m,i-1}(k) - q_{m,i}(k)\big). \tag{A.3}$$

While equations (A.2) and (A.3) are based on physical principles and are exact, the equations that describe the speed dynamics and the relation between density and the desired speed are heuristic. In the METANET model the mean speed at the simulation step $k+1$ is taken to be the mean speed at time step k plus a *relaxation term* that expresses that the drivers try to achieve a desired speed $V(\rho)$, a *convection term* that expresses the speed increase (or decrease) caused by the inflow of vehicles, and an *anticipation term* that expresses the speed decrease (increase) as drivers

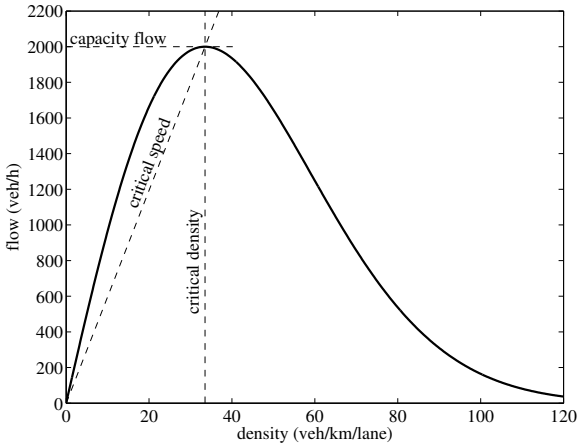


Figure A.2: *An example of the speed-flow relationship (A.5), with* $a_m = 1.867$*,* $v_{\text{free},m} = 102\,km/h$*,* $\rho_{\text{crit},m} = 33.5\,veh/km/lane$*.*

experience a density increase (decrease) downstream:

$$\begin{aligned} v_{m,i}(k+1) = v_{m,i}(k) &+ \frac{T_s}{\tau}\Big(V\big(\rho_{m,i}(k)\big) - v_{m,i}(k)\Big) \\ &+ \frac{T_s}{L_m} v_{m,i}(k)\big(v_{m,i-1}(k) - v_{m,i}(k)\big) \\ &- \frac{\eta T_s}{\tau L_m}\frac{\rho_{m,i+1}(k) - \rho_{m,i}(k)}{\rho_{m,i}(k) + \kappa}, \end{aligned} \tag{A.4}$$

where τ, η, and κ are model parameters, and with

$$V\big(\rho_{m,i}(k)\big) = v_{\text{free},m} \exp\left[-\frac{1}{a_m}\left(\frac{\rho_{m,i}(k)}{\rho_{\text{crit},m}}\right)^{a_m}\right], \tag{A.5}$$

with a_m a model parameter, and where the free-flow speed $v_{\text{free},m}$ is the average speed that drivers assume if traffic is freely flowing, and the critical density $\rho_{\text{crit},m}$ is the density at which the traffic flow is maximal on a homogeneous freeway. In Fig. A.2 the speed-density relationship $V(\rho)$, also called the fundamental diagram, is presented.

Mainstream origins and on-ramps are modeled with a simple queue model. The length $w_o(k)$ of the queue at origin o and at the time step k equals the previous queue length plus the demand[1] $d_o(k)$ (veh/h), minus the outflow $q_o(k)$ (veh/h) of origin o:

$$w_o(k+1) = w_o(k) + T_s\big(d_o(k) - q_o(k)\big). \tag{A.6}$$

The outflow of origin o depends on the traffic conditions on the mainstream and, for a metered on-ramp, on the ramp metering rate[2] $r_o(k)$, where $r_o(k) \in [0,1]$. The ramp flow $q_o(k)$ is the minimum of three quantities:

[1] Just as in [112, 113, 157], we assume that the demand is independent of any control actions taken in the network.

[2] For an unmetered on-ramp we also can use (A.7) by setting $r_o(k) \equiv 1$.

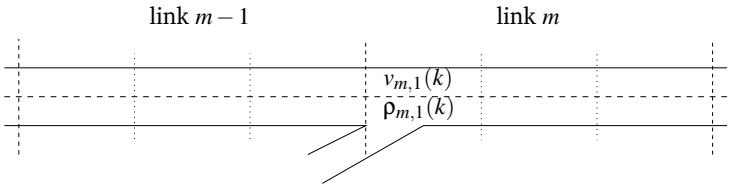


Figure A.3: *When there is an on-ramp connected to the freeway the speed $v_{m,1}(k)$ in the first segment of link m is reduced by merging phenomena according to (A.8).*

- the available traffic at simulation step k (queue plus demand),
- the maximal flow allowed by the metering rate,
- and the maximal flow that could enter the freeway because of the mainstream conditions.

So,

$$q_o(k) = \min\left[d_o(k) + \frac{w_o(k)}{T_s},\, C_o r_o(k),\, C_o\left(\frac{\rho_{\max,m} - \rho_{m,1}(k)}{\rho_{\max,m} - \rho_{\text{crit},m}}\right)\right], \tag{A.7}$$

where C_o (veh/h) is the capacity of origin o under free-flow conditions, the parameter $\rho_{\max,m}$ (veh/km/lane) represents the maximum density of a segment (also called jam density), which we assume as a global parameter independent of m in the following, and m is the index of the link to which the on-ramp is connected.

Note that in addition to the above formulation of ramp metering (A.7) that can be found in [111, 112], an alternative variant is discussed in the literature too [113].

In order to account for the speed drop caused by merging phenomena for the first segment of an outgoing link of an on-ramp or origin, the term

$$-\frac{\delta T_s q_o(k) v_{m,1}(k)}{L_m \lambda_m (\rho_{m,1}(k) + \kappa)} \tag{A.8}$$

is added to speed equation (A.4), where $v_{m,1}(k)$ and $\rho_{m,1}(k)$ are the speed and density of the segment that the on-ramp is connected to, as shown in Fig. A.3, and where δ is a model parameter.

When there is a lane drop as shown in Fig. A.4, the speed reduction due to weaving phenomena,

$$-\frac{\phi T_s \Delta\lambda_m \rho_{m,N_m}(k) v_{m,N_m}^2(k)}{L_m \lambda_m \rho_{\text{crit},m}} \tag{A.9}$$

is added to (A.4), where $\Delta\lambda_m = \lambda_m - \lambda_{m+1}$ is the number of lanes being dropped, and ϕ is a model parameter.

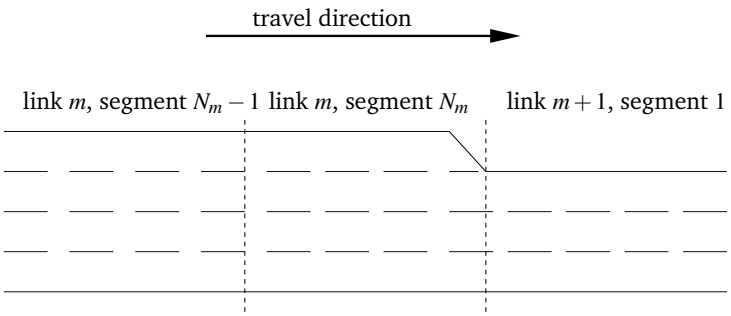


Figure A.4: *When there is a lane drop the speed $v_{m,N_m}(k)$ in the last segment of link m is reduced by merging phenomena according to (A.9).*

A.1.2 Node Equations

As a final part of the model, coupling equations should be considered that take into account the existence of multiple ingoing and outgoing links of the nodes and accordingly the distribution of flow and density. Recall that every time there is a major change in the link parameters or there is a junction or a bifurcation, a node is placed between the links. This node provides the incoming links with a downstream density (or a virtual downstream density when there are two or more leaving links), and the leaving links with an inflow and an upstream speed (or a virtual upstream speed when there are two or more entering links). The total flow $Q_n(k)$ (veh/h) that enters node n at simulation step k is distributed according to:

$$Q_n(k) = \sum_{\mu \in I_n} q_{\mu,N_\mu}(k), \tag{A.10}$$

$$q_{m,0}(k) = \beta_{n,m}(k) Q_n(k), \tag{A.11}$$

where I_n denotes the set of links entering node n, $\beta_{n,m}(k)$ is the turning rate (the fraction of the total flow through node n that leaves via link m), and $q_{m,0}(k)$ is the flow that leaves node n via link m, where link m is one of the links leaving node n. Recall that N_m is the index of the last segment of a link m that enters node n.

For a node n with multiple outgoing links as shown in Fig. A.5, the virtual downstream density $\rho_{m,N_m+1}(k)$ ('virtual' due to the nonexistence of a segment with the index N_m+1) of link m entering an origin is modeled as:

$$\rho_{m,N_m+1}(k) = \frac{\sum_{\mu \in O_n} \rho_{\mu,1}^2(k)}{\sum_{\mu \in O_n} \rho_{\mu,1}(k)}, \tag{A.12}$$

where O_n is the set of links leaving node n.

Similarly, for a node n with multiple ingoing links as shown in Fig. A.6, the virtual upstream speed $v_{m,0}(k)$ of outgoing link m is modeled as:

$$v_{m,0}(k) = \frac{\sum_{\mu \in I_n} v_{\mu,N_\mu}(k) q_{\mu,N_\mu}(k)}{\sum_{\mu \in I_n} q_{\mu,N_\mu}(k)}. \tag{A.13}$$

Note that the virtual downstream density and the virtual upstream speed are required in the speed update expression (A.4).

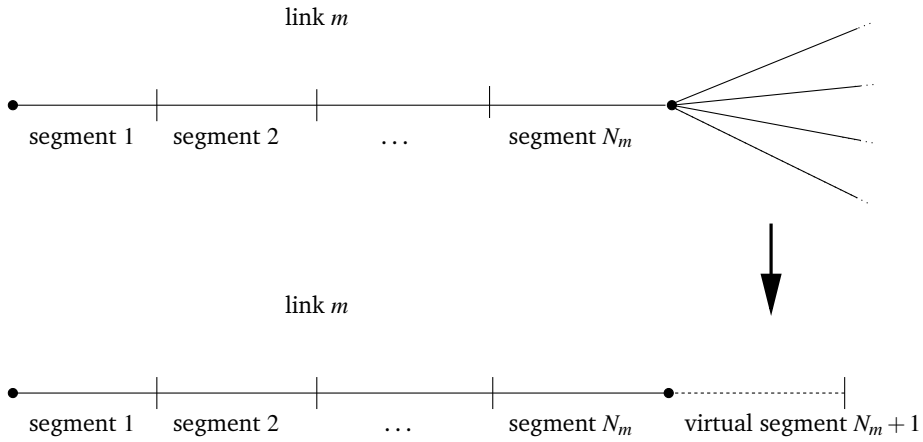


Figure A.5: *A node with one entering link m and several leaving links. The densities in the first segments of the leaving links are aggregated in the virtual downstream density $\rho_{m,N_m+1}(k)$ according to (A.12).*

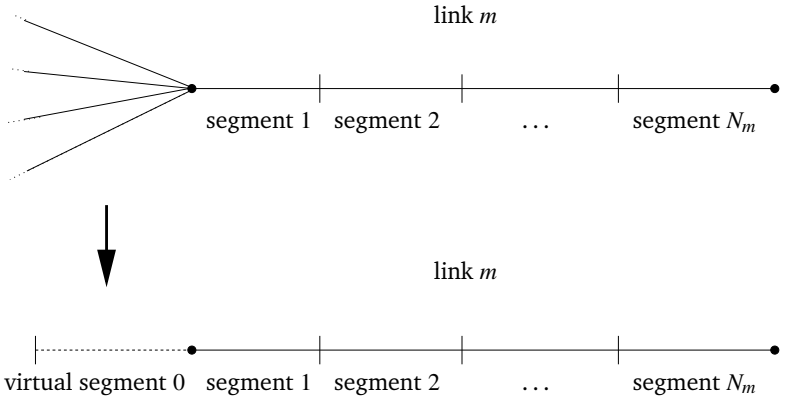


Figure A.6: *A node with one leaving link m and several entering links. The speeds in the last segments of the entering links are aggregated in the virtual upstream speed $v_{m,0}(k)$ according to (A.13) .*

A.1.3 Boundary Conditions

Boundary conditions need to be defined for the entry and exit points of the traffic network. As in METANET the state of a segment also depends on the upstream speed, the outflow of the upstream node, and the downstream density, we need to define the upstream speed and the inflow for the entry points of the network, and the downstream density for the exit points of the network. These boundary conditions can be user-specified or a default value can be assumed. We have already presented the boundary conditions for the traffic demand at on-ramps; now we present the boundary conditions for the upstream speed and the downstream density.

Upstream speed

When there is a mainstream origin entering node n, the virtual speed $v_\mu(k)$ of the origin can be user-specified, where μ is the index of the origin. If $v_\mu(k)$ is not specified then it could be taken to equal the speed of the first segment of the leaving link

$$v_\mu(k) = v_{m,1}(k). \tag{A.14}$$

If there is a second incoming link besides the origin, then the speed of the last segment of the second incoming link could be taken for $v_{m,0}(k)$.

Downstream density

Similarly, the virtual downstream $\rho_{m,N_m+1}(k)$ density for the entering link at a node that is connected to a destination, is calculated as follows. First, the user can specify a destination density scenario $\rho_\mu(k)$ where μ is the index of the destination link. Alternatively, a flow limitation $q_{\text{bound},\mu}(k)$ can be defined, and the virtual downstream density could be calculated according to

$$\rho_\mu(k+1) = \begin{cases} \rho_{\text{upstream},n}(k) & \text{if } q_\mu < q_{\text{bound},\mu}(k) \text{ and } \rho_{\text{upstream},n}(k) < \rho_{\text{crit},\mu}, \\ \rho_\mu(k) + C_\mu\big(q_\mu(k) - q_{\text{bound},\mu}(k)\big) & \text{otherwise.} \end{cases} \tag{A.15}$$

where $\rho_{\text{upstream},n}(k)$ is the density of the upstream link, and C_μ is a parameter. If $\rho_\mu(k)$ nor $q_{\text{bound},\mu}(k)$ is predefined then we could take

$$\rho_\mu(k) = \rho_{m,N_m}(k), \tag{A.16}$$

or if there is a second leaving link at node n, then the density of the first segment of that link could be taken for ρ_{m,N_m+1}.

A.2 A Route-Dependent Model

In this section we present a route-dependent METANET model, which can be obtained through straightforward modifications of the basic model and the destination-dependent METANET model. For the route-dependent mode the demand at each origin is further decomposed into demands for the available – possibly overlapping – routes for a particular OD-pair. The fraction of the demand of traffic entering route $r \in \mathcal{R}_j, j \in \{1,\ldots,N_{\text{OD}}\}$ – with N_{OD} the number of OD-pairs, $\mathcal{R}_j$ the set of routes available for OD-pair $j \in \{1,\ldots,N_{\text{OD}}\}$, and with $\mathcal{R} := \cup_{j\in\{1,\ldots,N_{\text{OD}}\}}\mathcal{R}_j$ – can be expressed by $\gamma_{j,r}(k)$, with $\sum_{r\in\mathcal{R}_j}\gamma_{j,r}(k) = 1$ for all $j \in \{1,\ldots,N_{\text{OD}}\}$, such that the demand at an origin o, $d_o(k)$, is now decomposed according to:

$$d_{o,r}(k) = \sum_{j\in\{1,\ldots,N_{\text{OD}}\}:o(j)=o} \gamma_{j,r}(k)\cdot d^j(k), \tag{A.17}$$

$$d_o(k) = \sum_{r\in\mathcal{R}} d_{o,r}(k), \tag{A.18}$$

where we assume that the traffic demand is specified for each OD-pair instead of for each origin as was the case in the basic model. Further, $o(j) \in \mathcal{O}$ denotes the origin associated with the OD-pair with the index $j \in \{1,\dots,N_{\text{OD}}\}$.

Origins are now modeled with a route-dependent queue model. The evolution of the partial queue length $w_{o,r}(k)$ at origin o associated with route $r \in \mathcal{R}$ is described by:

$$w_{o,r}(k+1) = w_{o,r}(k) + T_{\text{s}}\big(d_{o,r}(k) - q_{o,r}(k)\big), \tag{A.19}$$

$$w_o(k) = \sum_{r\in\mathcal{R}} w_{o,r}(k), \tag{A.20}$$

where $q_{o,r}(k)$ is the outflow of the origin associated with route $r \in \mathcal{R}$ according to:

$$q_{o,r}(k) = \min\left[q^{\text{des}}_{o,r}(k),\, C_o r_o(k)\frac{q^{\text{des}}_{o,r}(k)}{\sum_{r\in\mathcal{R}} q^{\text{des}}_{o,r}(k)},\, C_o\left(\frac{\rho_{\max}-\rho_{m,1}(k)}{\rho_{\max}-\rho_{\text{crit},m}}\right)\frac{q^{\text{des}}_{o,r}(k)}{\sum_{r\in\mathcal{R}} q^{\text{des}}_{o,r}(k)}\right], \tag{A.21}$$

where $q^{\text{des}}_{o,r}(k) = d_{o,r}(k) + \dfrac{w_{o,r}(k)}{T_{\text{s}}}$.

Here, the flow conservation equations modeling the distribution of flow that enters a node among the leaving links are simply route-dependent too, i.e., the previously applicable equations (A.10) and (A.11) are omitted in the new model, where the incoming and outgoing flows for a node n are linked according to the consecutive links l,m on the particular route $r \in \mathcal{R}$:

$$q_{m,0,r}(k) = q_{l,N_l+1,r}(k), \text{for all } l \in I_n, m \in O_n, \tag{A.22}$$

where, as explained in Section A.1.2, N_m+1 represents the index of the virtual last segment of link m in case of multiple outgoing links, and the virtual segment 0 is taken to represent the virtual first segment of a link in case of multiple incoming links.

Further, the total density $\rho_{m,i}(k)$ on segment i of link m is now decomposed into partial densities $\rho_{m,i,r}(k)$ for each route $r \in \mathcal{R}$, and similarly for the flow on a segment i of a link m:

$$\rho_{m,i}(k) = \sum_{r\in\mathcal{R}} \rho_{m,i,r}(k), \qquad q_{m,i}(k) = \sum_{r\in\mathcal{R}} q_{m,i,r}(k), \tag{A.23}$$

with a route-dependent conservation equation:

$$\rho_{m,i,r}(k+1) = \rho_{m,i,r}(k) + \frac{T_{\text{s}}}{L_m\lambda_m}\big(q_{m,i-1,r}(k) - q_{m,i,r}(k)\big), \tag{A.24}$$

and a route-dependent flow equation:

$$q_{m,i,r}(k) = \lambda_m \rho_{m,i,r}(k) v_{m,i}(k), \tag{A.25}$$

while the update equation for $v_{m,i}(k)$ is still according to Equation (A.4).

Appendix B

Integrated Model Predictive Traffic and Emission Control Using a Piecewise Affine Approach

The research presented in this appendix is the result of a separate line of research conducted in the beginning of the Ph.D. project. The chapter is based on [75], supported by the results presented in [76, 84].

This chapter addresses the computational intractability of traffic control when applying the integrated METANET freeway traffic model and the VT-macro emission model in a model-based predictive control (MPC) framework. In order to facilitate real-time implementation, a piecewise affine (PWA) approximation of the nonlinear METANET model is proposed. While a direct MPC approach based on the full PWA model is intractable for online applications, a conversion to a mixed-logical dynamic (MLD) model description is made instead. The resulting MLD-MPC problem, written as a mixed-integer linear program, can be solved much more efficiently as it does not explicitly state all model equations for each particular region. As a benchmark, the computational efficiency and accuracy of the MLD-MPC approach is tested on a case study including variable speed limits and a metered on-ramp while optimizing the total time spent, as well as while taking into account emissions and fuel consumption of the vehicles. The performance is evaluated against the original nonlinear and nonconvex MPC problem and shows an improved computational speed at the cost of some deviation in the objective function values.

B.1 Introduction

In online model-based control of large-scale traffic networks it is important to adopt a modeling framework that is both accurate and that yields a fast computation of solutions when incorporated within the optimization framework. An often used model for freeways is the second-order macroscopic METANET model [112, 114, 144]. Such a model is commonly used as it has shown to provide good accuracy while it does not require as much computation time as microscopic traffic models

that take individual vehicles into account [101]. The main variables considered in METANET are the average density, flow, and velocity of traffic. This model can be complemented by the VT-macro model for vehicular emissions and fuel consumption [204].

As control framework, a well-known method is model-based predictive control (MPC) [136, 163]. In the application of MPC to traffic systems, based on measurements of the system, optimal control inputs are computed, e.g., on-ramp metering rates and variable speed limits that yield – as one possible performance criterion – an optimal traffic throughput based on both the current state and predicted, future states. After the implementation of the first set of these optimal control inputs, the process is repeated, which is referred to as the moving horizon approach of MPC. MPC has often been adopted in various industries [136, 145] as it easily incorporates various constraints and adapts well to uncertain systems and structural changes in the system due to the moving horizon strategy. Additionally, the prediction model can be adapted during the control process.

When using the METANET traffic flow model complemented by the VT-macro emission and fuel consumption model in combination with MPC in order to minimize the total time spent (TTS) by traffic in the network as well as to reduce the vehicular emissions, a nonlinear and nonconvex optimization problem results. Such a problem can be solved with global or multi-start local optimization [20, 94, 115]. However, this approach is subject to computational issues that prevent the real-time implementation on realistically-sized traffic networks, and a global optimum cannot be guaranteed. Hence, the main objective we like to address is to develop an accurate yet computationally efficient approach for applying MPC based on the integrated METANET and VT-macro models. One way to address this computational issue is to approximate the underlying model [205]. Other ways are to address the optimization approach itself and to reformulate, e.g., to decompose or distribute the problem solving [70, 129].

In the current appendix, we focus on the underlying model and propose to adopt a piecewise affine (PWA) approximation of the nonlinear elements within METANET and VT-macro. Nonlinear functions can be approximated by PWA functions with arbitrary accuracy, partitioning the function domain in a finite number of polyhedra, each associated with an affine function. Based on the PWA model, an iteration of the MPC problem can then be more easily solved to optimality when formulated as a mixed-integer linear program (MILP). However, since the class of MILPs has been proven to be NP-hard [68], attention should be paid to keeping the number of binary variables in the PWA model formulation small, as they increase the MILP's complexity. At the same time, the fewer such variables are allowed in the approximation and thus the fewer the number of affine pieces, the larger the discrepancy with the original function. In other words, it is important to find a good trade-off between accuracy of the approximation and computational complexity. In this appendix, it will be explained how MPC based on a full PWA model turns out intractable, as it results in a large number of regions that need to be considered in the integrated PWA model for the entire traffic network. Instead, applying MPC based on a mixed-logical dynamic (MLD) description of the PWA system shows encouraging results towards the real-time application of MPC in traffic control.

The remainder of this chapter is organized as follows. In Section B.2 a brief recapitulation of the original METANET traffic flow model is presented together with the VT-macro model for vehicular emissions and fuel consumption. Section B.3 includes a description of the MPC approach and the optimization objectives we propose for traffic control. In Section B.4 we describe how the nonlinear model equations can be approximated in a PWA manner according to one of the selected methods. Subsequently, it is shown how the resulting PWA model can be recast as an MLD model in order to arrive at a feasible MILP (Section B.5). The proposed MLD-MPC approach is applied to a case study in Section B.6, where its performance with respect to accuracy and computational speed is illustrated in the context of integrated speed limit and ramp metering control, optimizing also the vehicular emissions, and compared to the original nonlinear MPC. Conclusions and recommendations are presented in Section B.7.

B.2 The METANET and VT-Macro Models

B.2.1 METANET

Although the METANET model has been described in the previous Appendix A, we here repeat the main equations that will be referred to in the remainder of this chapter. Recall that the traffic network is described by a graph with links representing homogeneous parts of a freeway, separated by nodes representing changes like on-ramps and the increase or decrease of the number of lanes. Links are further divided into segments of equal distance. As regards the discretization in time, typically a simulation time step T_s of about 10 s is used, where $t = kT_\mathrm{s}$ for time instant t and the corresponding time step counter k. Similarly, a sample or control time interval T_c [s] will be adopted, where $k_\mathrm{c} \in \mathbb{N}$ indicates the control time step counter for time instant $t = k_\mathrm{c}T_\mathrm{c}$, where $T_\mathrm{c} = MT_\mathrm{s}, M \in \mathbb{N}$. Commonly used values of the model parameters defined in Appendix A are provided in Table B.1.

The evolution of traffic flow $q_{m,i}$ (veh/h), density $\rho_{m,i}$ (veh/km/lane), and space-mean speed $v_{m,i}$ (km/h) for segment i of link m for time step k is described by:

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) v_{m,i}(k), \tag{B.1}$$

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T_\mathrm{s}}{L_m \lambda_m}\left[q_{m,i-1}(k) - q_{m,i}(k)\right], \tag{B.2}$$

$$\begin{aligned} v_{m,i}(k+1) = {} & v_{m,i}(k) + \frac{T_\mathrm{s}}{\tau}\left[V[\rho_{m,i}(k)] - v_{m,i}(k)\right] \\ & + \frac{T_\mathrm{s} v_{m,i}(k)\left[v_{m,i-1}(k) - v_{m,i}(k)\right]}{L_m} \\ & - \frac{T_\mathrm{s}\eta\left[\rho_{m,i+1}(k) - \rho_{m,i}(k)\right]}{\tau L_m\left(\rho_{m,i}(k) + \kappa\right)}. \end{aligned} \tag{B.3}$$

The desired speed $V[\rho_{m,i}(k)]$ (km/h) is represented by:

$$V[\rho_{m,i}(k)] = \min\left[v_{\mathrm{free},m}\exp\left[-\tfrac{1}{a_m}\left(\tfrac{\rho_{m,i}(k)}{\rho_{\mathrm{crit},m}}\right)^{a_m}\right], (1+\alpha)v_{\mathrm{ctrl},m,i}(k)\right], \tag{B.4}$$

Table B.1: *METANET parameter settings [93, 112]*

$T_s = 10\,\text{s}$	$\kappa = 40\,\text{veh/km/lane}$	$\delta = 0.0122$	$\eta = 60\,\text{km}^2\text{/h}$
$\tau = 18\,\text{s}$	$\rho_{\max} = 180\,\text{veh/km/lane}$	$a_m = 1.867$	$v_{\text{free}} = 102\,\text{km/h}$
$L_m = 1\,\text{km}$	$\rho_{\text{crit}} = 33.5\,\text{veh/km/lane}$	$\alpha = 0.1$	

where the first term appeared in (A.5) and the second term applies in case of variable-speed control on segment i of link m, where the speed limit is denoted by the speed control variable $v^{\text{c}}_{\text{ctrl},m,i}(k_{\text{c}})$ (km/h) at control time step k_{c}, where $v_{\text{ctrl},m,i}(k) = v^{\text{c}}_{\text{ctrl},m,i}(k_{\text{c}})$ for $k = k_{\text{c}}M, \ldots, (k_{\text{c}}+1)M-1$ [92, 93]. Here, α denotes a factor to model the non-compliance of traffic participants to the speed limit based on whether the limit is obligatory or recommended, resulting in a lower respectively higher actual speed. In practice, $\alpha \in [-0.25, 0.25]$.

As also mentioned in Appendix A, the METANET model can be further complemented to take into account e.g., merges and drops of lanes and the resulting speed drops, main-stream metering, or it can be adapted to different models for dynamic speed limits [32, 33, 93, 94, 112].

B.2.2 VT-Macro Emission Model

In order to take into account emissions and fuel consumption of the vehicles, the METANET model can be extended with the equations of the VT-macro model. For more detailed information on this model, the reader is referred to [204]. The VT-macro model estimates traffic emissions and fuel consumption using either the temporal or spatio-temporal accelerations of vehicles. For instance, the spatio-temporal acceleration and number of vehicles subject to it while moving from segment i to the next segment $i+1$ of a link m are given by:

$$a_{m,i,i+1}(k) = \frac{v_{m,i+1}(k) - v_{m,i}(k-1)}{T_s}, \tag{B.5}$$

$$n_{m,i,i+1}(k) = T_s q_{m,i}(k). \tag{B.6}$$

Similar expressions apply to e.g., on-ramps, off-ramps, junctions, etc. The temporal accelerations and number of vehicles subject to it refer to the values of those variables within a single segment i of a given link m:

$$a_{m,i}(k) = \frac{v_{m,i}(k) - v_{m,i}(k-1)}{T_s}, \tag{B.7}$$

$$n_{m,i}(k) = L_m \lambda_m \rho_{m,i}(k) - T_s q_{m,i}(k). \tag{B.8}$$

The (total) vehicular emissions [g] and fuel consumption [l] appear in the cost function of the traffic control problem where they are minimized over the time period $[kT_s, (k+1)T_s]$, i.e.:

$$J_{\gamma,\text{TEFC}}(k) = T_s \sum_{\ell \in L_{\text{all}}} n_\ell(k) \exp\left(\check{v}_\ell^{\text{T}}(k) P_\gamma \check{a}_\ell(k)\right), \tag{B.9}$$

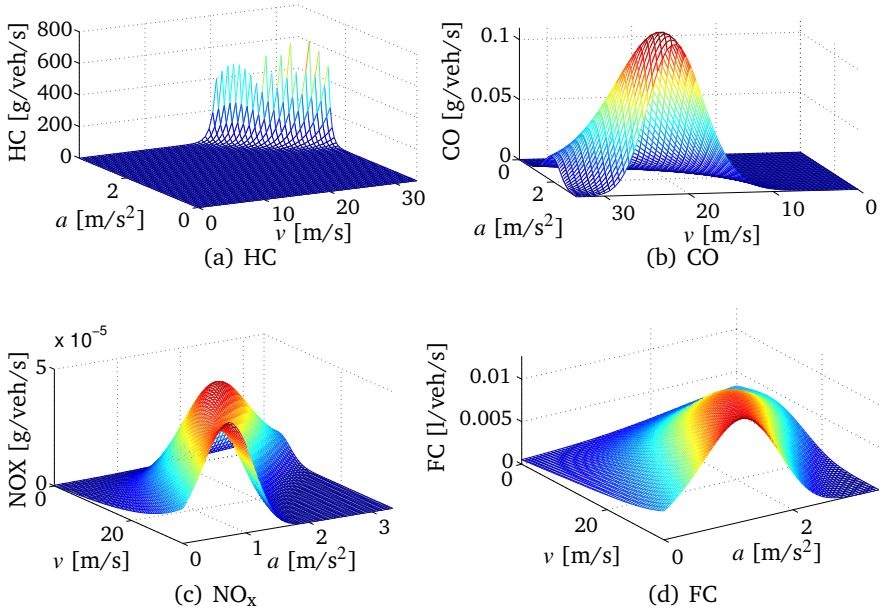


Figure B.1: *Plots of the vehicular emissions and fuel consumption term* $\exp\left(\check{v}_\ell^{\mathrm{T}} P_\gamma \check{a}_\ell\right)$ *in (B.9)*

with the speed and acceleration vectors $\check{v}_\ell^{\mathrm{T}}(k) = [1 \;\; v_\ell(k) \;\; v_\ell^2(k) \;\; v_\ell^3(k)]$ and $\check{a}_\ell^{\mathrm{T}}(k) = [1 \;\; a_\ell(k) \;\; a_\ell^2(k) \;\; a_\ell^3(k)]$, and with L_{all} the set of indices of all triples (a_ℓ, n_ℓ, v_ℓ) of spatio-temporal or temporal accelerations and the corresponding numbers of vehicles and speeds. Moreover, P_γ denotes the model parameter for $\gamma \in \Gamma = \{$CO emission, HC emission, $\mathrm{NO_x}$ emission, fuel consumption$\}$. The values of the parameter matrices P_γ can be found in [1]. Plots of the different emission components can be found in Fig. B.1.

B.3 MPC for Traffic Control

Using MPC [28, 67, 136, 163], based on measurements of the current state variables at the control step k_{c}, future states are predicted for a prediction horizon of N_{p} control steps, using the prediction model presented in the previous section. By optimization of the objective function over this horizon, a sequence of optimal decision variables is determined. In order to reduce the number of decision variables and therefore the computational burden, often a control horizon $N_{\mathrm{c}} < N_{\mathrm{p}}$ is defined, and the control signals are kept constant at the control steps $k_{\mathrm{c}}+N_{\mathrm{c}}, k_{\mathrm{c}}+N_{\mathrm{c}}+1, \ldots, k_{\mathrm{c}}+N_{\mathrm{p}}-1$. Here it should be noted that the prediction models should at the same time provide accurate predictions of the evolution of the states of the model, but not lead to a computation time that – when incorporated within the optimization algorithm – exceeds the specified time in which the control inputs are required. Implementing only

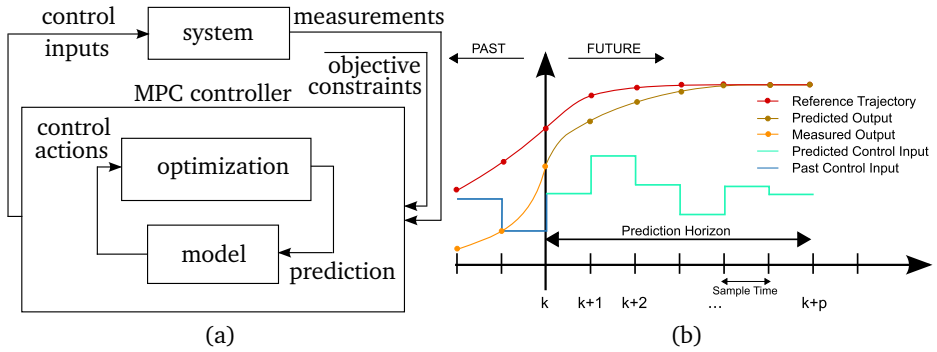


Figure B.2: *A schematic representation of model predictive control*

the first sample of these optimal control inputs, the procedure is repeated for the next control time step in a moving or rolling horizon fashion for a simulation time horizon with a simulation time interval T_s and control time interval T_c. A schematic representation of this control approach is shown in Fig. B.2.

Advantages of using MPC as a control scheme are that it facilitates the evaluation of combinations of performance criteria, using combinations of control measures, and it is easyto incorporate (feasibility and safety) constraints on the control signals and the system states. Moreover, by adopting a prediction horizon, the evolution of the behavior of the system during a longer time period can be incorporated when determining the control inputs. Finally, due to the moving horizon approach, it is easy to update system parameters as well as the model at different time instants; similarly, unexpected behavior influencing the system is accounted for by updating the system state by taking new measurements at each iteration.

Amongst the possible optimization goals for traffic networks are the maximization of traffic flow, spreading traffic density, and minimizing the variation in control variables [93]. We chose as our objective function the following linear combination of terms:

$$J(k_c) = c_1 \frac{J_{\text{TTS}}^{\text{MPC}}(k_c)}{\text{TTS}_{\text{nom}}} + \sum_{\gamma \in \Gamma} c_{2,\gamma} \frac{J_{\gamma,\text{TEFC}}^{\text{MPC}}(k_c)}{\text{TEFC}_{\gamma,\text{nom}}} + c_3 \frac{J_{\text{pen}}^{\text{MPC}}(k_c)}{\text{pen}_{\text{nom}}}, \tag{B.10}$$

namely the minimization of the total time vehicles spend in the system (TTS), i.e., the time vehicles wait at an on-ramp or mainstream origin before joining the freeway plus the time spent on the freeway, the traffic emissions and fuel consumption, and a penalty term on the variations of the decision variables, respectively, weighted by nonnegative constants $c_1, c_{2,\gamma}$, and c_3. The three respective terms are normalized with the nominal values TTS_{nom}, $\text{TEFC}_{\gamma,\text{nom}}$, and pen_{nom}, which are obtained by considering the uncontrolled system.

To elaborate, the first objective term of the MPC controller is to reduce the TTS

over the prediction horizon N_p, i.e.,

$$J_\mathrm{TTS}^\mathrm{MPC}(k_\mathrm{c}) = T_\mathrm{s} \sum_{k\in\mathcal{K}(k_\mathrm{c},k_\mathrm{c}+N_\mathrm{p})} \left(\sum_{(m,i)\in I_\mathrm{all}} L_m \lambda_m \rho_{m,i}(k) + \sum_{o\in O_\mathrm{all}} w_o(k) \right). \tag{B.11}$$

Here, I_all denotes the set of index pairs (m,i) of all links and segments in the network, and O_all denotes the set of indices of all origins. Further, $\mathcal{K}(k_\mathrm{c},k_\mathrm{c}+N_\mathrm{p}) = \{Mk_\mathrm{c}, Mk_\mathrm{c}+1,\ldots,M(k_\mathrm{c}+N_\mathrm{p})-1\}$ where M is such that $T_\mathrm{c} = MT$. Note that the TTS cost function term is linear in the state variables $\rho_{m,i}(k)$ and $w_o(k)$.

Further, the total vehicular emissions and fuel consumption introduced in (B.9) are captured in the expression

$$J_{\gamma,\mathrm{TEFC}}^\mathrm{MPC}(k_\mathrm{c}) = \sum_{k\in\mathcal{K}(k_\mathrm{c},k_\mathrm{c}+N_\mathrm{p})} J_{\gamma,\mathrm{TEFC}}(k). \tag{B.12}$$

Hence, if a linearized expression for (B.9) is found, the emission and fuel consumption factors enter the objective function linearly.

Finally, the penalty term on deviations of the decision variables is defined as:

$$\begin{aligned} J_\mathrm{pen}^\mathrm{MPC}(k_\mathrm{c}) = \sum_{j=1}^{N_\mathrm{c}-1} \Bigg\{ & \sum_{o\in O_\mathrm{all}} |r_o(k_\mathrm{c}+j) - r_o(k_\mathrm{c}+j-1)| \\ & + a_\mathrm{speed} \sum_{(m,i)\in C_\mathrm{all}} |v_{\mathrm{ctrl},m,i}(k_\mathrm{c}+j) - v_{\mathrm{ctrl},m,i}(k_\mathrm{c}+j-1)| \Bigg\}, \end{aligned} \tag{B.13}$$

where a_speed is a nonnegative weighting coefficient and where C_all is the set of all pairs of indices (m,i) of links and segments in which a variable speed limit is applied. The penalty term (B.13) can be transformed into a linear form, such that the overall objective function is linear and convex. This transformation, for which some additional real-valued auxiliary variables need to be introduced, will be explained in Section B.5. Note that instead of a 1-norm, alternatively a quadratic penalty term could be adopted.

All in all, given a linear expression for (B.9), the above MPC objective (B.10) is linear and convex in the control variables, yet the underlying METANET prediction model is nonlinear and nonconvex. As a result, the optimization problem based on the original models is intractable for real-life implementation. Hence, in order to reduce the computation time, in the following, a PWA approximation of the METANET model is proposed.

B.4 PWA Approximation

A function $f : \Omega \to \mathbb{R}^m$ is PWA if there exists a polyhedral partition $\{\Omega_i\}_{i\in I}$ of $\Omega \subseteq \mathbb{R}^n$ such that f is affine on each polyhedron Ω_i, i.e.,

$$f(x) = A_i x + b_i, \quad \text{for all } x \in \Omega_i, i \in I, \tag{B.14}$$

with A_i, b_i constants. Here, a polyhedral partition of Ω represents a finite number of nonempty polyhedra $\{\Omega_i\}_{i\in I}$ such that $\bigcup_{i\in I} \Omega_i = \Omega$ and $\Omega_i \cap \Omega_j = \varnothing$ for all $i \neq j$.

For a continuous PWA function we only require that $\text{int}(\Omega_i) \cap \text{int}(\Omega_j) = \varnothing$ for all $i \neq j$. For more information on general PWA theory, see [177].

One can approximate a nonlinear function in a PWA manner with arbitrary accuracy, i.e., by considering a sufficiently large number of regions. However, as will be pointed out in Section B.5, this comes at the cost of a larger computational burden if the approximated model equation is to be applied in the eventual optimization approach. Therefore, as a main consideration in the PWA approximation, the number of affine pieces should be kept small, while safeguarding a close match to the original traffic model. In return for a less accurate model, one then arrives at a PWA model description that is faster to deal with in optimizations and the solution of which could alternatively be used as an initial starting point for an MPC optimization when using the original nonlinear model.

In the remainder of this section a selection is provided of possible methods to arrive at a PWA approximation. These methods will subsequently be used for the approximation of the nonlinear METANET equations in Section B.4.2.

B.4.1 PWA Approximation Methods

Four frequently adopted approaches for PWA approximation of nonlinear functions are least-squares optimization, PWA identification, (partially) piecewise constant approximation, and PWA approximation of a multivariate function by reduction to separable quadratic terms. Here it should be noted that there is a large difference in complexity between the approximation of single and multi-variable functions, both of which occur in the METANET setting we consider in this chapter. More information on the various available methods for PWA function approximations can be found in [11] and [61].

Least-squares optimization

A well-known optimization-based approximation approach comprises the minimization of the squared error or difference between the original function and the approximation curve. For single-variate nonlinear functions this method is easy to apply and accurate. After one specifies the desired number of regions or intervals of the PWA function, both the optimal intervals and parameters of the affine functions are determined using least-squares optimization. Additionally, one may add positive weights to parts of the function that in particular require a high accuracy. E.g., the following PWA problem may be solved in a least-squares manner – here given for an approximation of a function f defined on an interval $[x_{\min}, x_{\max}]$ by a continuous PWA function $f_{\text{PWA}}(x)$ with three intervals:

$$\min_{\alpha,\beta,\gamma,\delta,\varepsilon,\zeta} \int_{x_{\min}}^{x_{\max}} w(x)\,(f_{\text{PWA}}(x) - f(x))^2\,dx, \tag{B.15}$$

such that

$$f_{\text{PWA}}(x) = \begin{cases} \gamma + \dfrac{x - x_{\min}}{\alpha - x_{\min}}(\delta - \gamma), & \text{for } x_{\min} \leq x < \alpha, \\ \delta + \dfrac{x - \alpha}{\beta - \alpha}(\varepsilon - \delta), & \text{for } \alpha \leq x < \beta, \\ \varepsilon + \frac{x-\beta}{x_{\max}-\beta}(\zeta - \varepsilon), & \text{for } \beta \leq x \leq x_{\max}, \end{cases} \tag{B.16}$$

where w denotes a weighting function. Least-squares optimization can be solved using e.g., a multi-start Gauss-Newton or Levenberg-Marquardt approach [63].

Piecewise affine identification

An alternative approach is hybrid or piecewise affine identification. Differently from the previous method, PWA identification is a clustering algorithm that returns a PWA approximation based on a set of data points. Therefore, this approach is especially useful for complex, multi-variate functions. In general, the three available methods described next create local data sets after which the clustering algorithm creates local affine models of the functions by classifying the points. Similar models are again grouped into clusters, depending on the number of regions required [61].

Amongst the available methods for identification, the most precise for bivariate identification is the algorithm Multicategory Robust Linear Programming (MRLP) [26]. However, this method is computationally expensive and generally works for the identification of up to three polytopes based on up to 200 data points. This is due to the fact that one linear program is solved to find boundaries of all regions simultaneously. Alternatively, the clustering algorithms Support Vector Classification (SVC) and Proximal Support Vector Classification (PSVC) can be used, yet for multivariate estimation the original domain of the variables may then not be completely covered by the union of computed subregions. In contrast to MRLP the SVC approach [190] solves several quadratic programs in order to sequentially find boundaries between two regions or half-spaces at a time. PSVC [66] is the most time-efficient algorithm of the three and only requires a single system of linear equations. Compared to the non-proximal version, it assigns data points to the closest of two parallel half-planes that are maximally separated, leading to a strongly convex objective.

These algorithms are implemented in the Hybrid Identification Toolbox (HIT) [60], a platform embedded within the Multi-Parametric Toolbox for Matlab [117].

Partially piecewise constant approximation

An approximation approach for bivariate functions that uses relatively few auxiliary variables consists in segmentation of the domain of one of the variables, where in each region or subdomain that variable is assigned a constant value. In general, a bivariate function $f(x,y)$ can be approximated as follows. Assume that based on the relative ranges $\frac{x_{\max}-x_{\min}}{x_{\max}}$ and $\frac{y_{\max}-y_{\min}}{y_{\max}}$ (in case $x_{\max}=0$ or $y_{\max}=0$, only the numerator applies) or the magnitude of the partial derivatives, the variable x is selected to be taken constant in each region. For a selection of N consecutive intervals $[x_i, x_{i+1}]$ for

$i = 1,\ldots,N-1$ and with $x_1 = x_{\min}$, $x_N = x_{\max}$, we can set e.g.,

$$f(x,y) \approx f\left(\frac{x_i + x_{i+1}}{2}, y\right), \text{ for } x \in [x_i, x_{i+1}]. \tag{B.17}$$

Now, if f is linear in y, as will be the case for several functions of the METANET model, this approach results in a PWA approximation of f. Alternatively, the least-squares optimization approach discussed above can be applied for each function $f(\frac{x_i+x_{i+1}}{2}, y)$.

Substitution of one of the variables can be seen as a specific case of PWA approximation where instead of an affine piece, a constant value is associated with each region of the domain. Nonetheless, this piecewise constant approach can deliver adequate approximation results for some functions. For single-variate functions this partially piecewise constant approach may also be applied, but in general it is not very difficult to obtain a more accurate PWA formulation for this class of functions.

PWA approximation by reduction

Finally, as proposed in [118, 199], a bivariate term of the form $x \cdot y$, with $x, y \in \mathbb{R}$ can be recast in the equivalent form $\frac{1}{4}[z_+^2 - z_-^2]$, where two new real variables are introduced, i.e., $z_+ := x + y$, $z_- := x - y$. The PWA approximation now reduces to replacing both quadratic terms z_+^2, z_-^2 in a PWA manner, for which any of the previous methods can be adopted. Since these terms can be approximated independently, this approach results in a relatively small number of binary variables, also if it would be extended to factors of higher order.

In the following section each nonlinear METANET equation is considered separately and approximated using one of the methods discussed above.

B.4.2 PWA Approximation of METANET and VT-Macro

In this section the nonlinear elements of the METANET model are dealt with. In general, application-specific knowledge could be applied in the approximation of model equations, e.g., by weighting areas that require a close match. In addition, information from other model equations can facilitate the approximation. When applied to this specific case, it indeed pays off to make use of physical information, e.g., the fundamental diagram of traffic flow depicted in Fig. B.3 will be used also in the approximation of other functions of the METANET model.

First, note that (B.2), (A.6), and (A.7) do not need to be approximated as the first two functions are already linear and the latter equation is PWA. The remaining nonlinear terms can be divided into the following main groups:

Fundamental diagram

The first nonlinear term of (B.4) corresponds to the right fundamental diagram of traffic flow depicted in Fig. B.3, which represent the equilibrium relations between speed, flow, and density in a homogeneous part of a freeway [140]. For this single-variate nonlinear term, a least-squares approach is adopted as shown in Fig. B.4. An

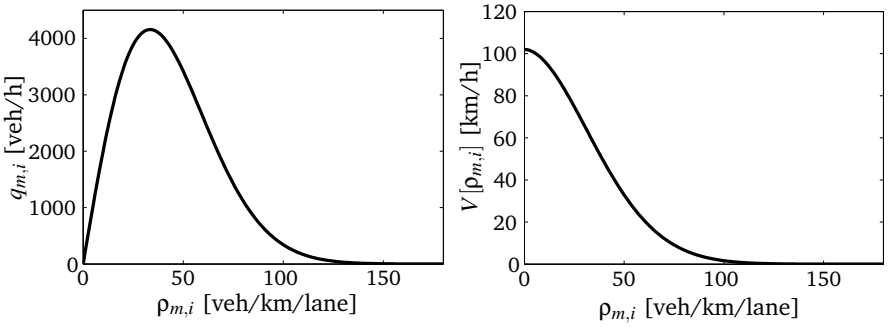


Figure B.3: *Fundamental diagrams of traffic flow*

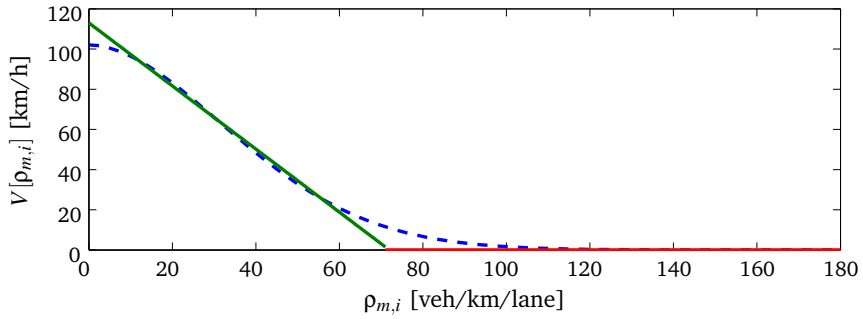


Figure B.4: *PWA approximation of the fundamental diagram in two pieces*

approximation in two affine pieces was applied, as this yielded a good accuracy at a better efficiency as compared to a more accurate division in three pieces. Since the second term in (B.4) is a linear expression, a PWA fundamental diagram combined with a variable speed limit leads to a PWA expression of the desired speed equation.

Speed equation (B.3)

We propose to keep several variables that give rise to the nonlinear terms in (B.3) constant at a value determined by historical data or equal to the currently measured value for predictions in receding horizon. The accuracy can be improved by adopting a sequence of different, predicted values that varies for each simulation step, i.e., by using the values of the state variables that result from a simulation of the traffic prediction model, in which the optimal control sequence consisting of N_c control inputs is considered. Alternatively, a more exact approximation could be obtained using PWA identification or a piecewise constant approach. However, the improvement in approximation accuracy may not justify the increase in computational complexity that this causes.

- $v_{m,i}(k)[v_{m,i-1}(k) - v_{m,i}(k)]$. Here, the first velocity variable $v_{m,i-1}(k)$ is substi-

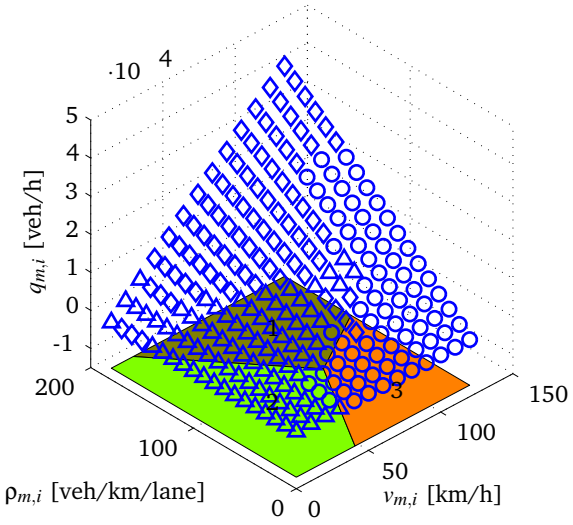


Figure B.5: *PWA approximation of flow equation (B.1) by hybrid identification using the Hybrid Identification Toolbox [60]*

tuted by a constant value. Note that the absolute approximation error caused by this method is mitigated in the context of the full model due to the multiplication of the replaced velocity by the relatively small term T_s/L_m ($2.78 \cdot 10^{-3}$ h/km: refer to Table B.1 for typical values of the parameters used).

- $\frac{\rho_{m,i+1}(k)-\rho_{m,i}(k)}{\rho_{m,i}(k)+\kappa}$. As in the previous item, the density term in the denominator is kept constant at a historically-based value, according to measurements, or it is determined based on a sequence of predictions using the computed control variables. Note that the multiplication factor $\eta T_s/\tau L_m$ for this factor in the numerator of (B.3) is rather large (33.33 km/h), as is the addition of $\kappa = 40$ veh/km/lane to the approximated variable in the denominator. Both aspects again cause a reduction in the effect of the absolute error of the approximated denominator with respect to the speed variable $v_{m,i}(k)$.

- Subtraction of the term (A.8). Final adaptations are made to this speed-drop term by substituting the density variable in the denominator by a constant value, combined with the substitution of $q_0(k) \cdot v_{m,1}(k)$ as in the PWA approximation of the flow equation (B.1) (see below).

Flow equation (B.1)

The bivariate equation modeling traffic outflow can be approximated by PWA identification, reduction to separable quadratic terms, and by a piecewise constant approach for one of the variables. When adopting the latter method we choose to

substitute the velocity variable $v_{m,i}(k)$, having the smallest domain in comparison with the flow variable, by the mean value of each subdomain, transforming (B.1) into:

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) \frac{v_j + v_{j+1}}{2}, \text{ for } v_{m,i}(k) \in [v_j, v_{j+1}]. \tag{B.18}$$

Here, the intervals $[v_j, v_{j+1}]$ can be chosen individually by taking into consideration the shape of the approximated function or determined in a more sophisticated way by using optimization.

In the approximation of (B.1) it is further important to take into account the shape of the fundamental diagram shown in Fig. B.3(a) and (b). To be more precise, in order to increase the accuracy of the approximation while keeping the set of auxiliary variables small one can put additional weight on data points where a small error is important. Looking at the shape of the fundamental diagram, it can be inferred that a situation of close-to maximum density and speed simultaneously is not likely to occur in real life. Therefore, the focus should be on a good match in the area around the function values as determined by the fundamental diagram.

The final PWA approximation by PWA identification can be seen in Fig. B.5. Here it should be noted that in both approximation methods, the relative approximation error of the flow variable $q_{m,i}(k)$ is of the same order as the approximation error in $\rho_{m,i}(k)v_{m,i}(k)$ (PWA identification) or in $v_{m,i}(k)$ (piecewise constant approach). To further put the error in perspective, it should be noted that the flow variable occurs in the density function (B.2), where the difference $q_{m,i-1}(k) - q_{m,i}(k)$ between consecutive segments is multiplied by the relatively small constant $T_s/\lambda_m L_m$ ($1.39 \cdot 10^{-3}$ h/km for $\lambda_m = 2$). Hence, approximation errors of (B.1) are relatively seen reduced when considering the complete METANET model. On the other hand, it should not be forgotten that approximations are re-used in different segments of the model and that by the iterative nature of MPC, approximation errors re-appear also in variables for later time steps of the prediction horizon.

Finally, note that the on-ramp flow equation (A.7) is already PWA, which means that together with the originally linear equations (B.2), (A.6), (B.5), and (B.6) we now have a system of only linear and PWA model equations. However, since the term $J_{\gamma,\text{TEFC}}(k)$ from (B.9) causes the optimization objective to become a nonlinear nonconvex function, a final PWA approximation should be made.

Total emission and fuel consumption (B.9)

In order to arrive at a PWA expression for (B.9), the exponential term $\exp\left(\check{v}_\ell^{\mathrm{T}}(k) P_\gamma \check{a}_\ell(k)\right)$ is replaced by a PWA approximation using PWA identification for each of the emission and fuel consumption elements. In Fig. B.6(a) a filled contour plot of this term for NO_x as it was plotted in Fig. B.1(c) is provided, together with a division of the domain into regions, each of which is associated with an affine expression that we obtained by hybrid identification as explained in Section B.4.1. Further, to reduce the number of additional variables and therefore the computational complexity, in the original VT-macro model it is suggested to substitute the variable $n_\ell(k)$ by a constant value [205]. Hence, $n_\ell(k)$ can again be taken as a constant equal to

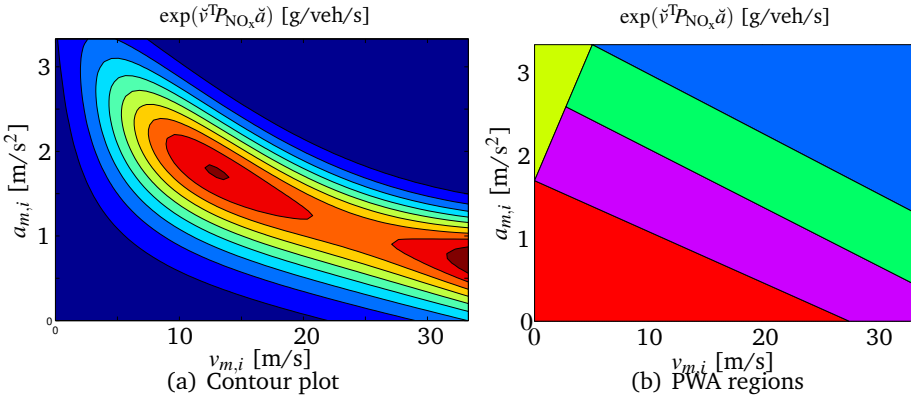


Figure B.6: *Domain and proposed regions of the PWA approximation of NO_x*

the currently measured or predicted values, as explained for the approximation of (B.3).

Now, in order to obtain a directly implementable optimization problem, some further adaptations using auxiliary binary variables are needed as will be explained next.

B.5 The PWA-MPC Problem

We now have gathered all ingredients for an implementation of model predictive traffic control using the integrated PWA METANET and VT-macro prediction models. As will be elaborated upon in this section, a full PWA model of the traffic system still leads to an intractable problem formulation when written as a mixed-integer linear program (MILP). Therefore we propose a mixed-logical dynamic formulation instead, which can as well be solved as an MILP when incorporated with MPC, yet in a more efficient manner.

B.5.1 Using a Full PWA Model

In order to be able to apply MPC to the PWA model, a logical next step would be to combine the individual linear and PWA model equations of METANET and VT-macro and to rewrite them into one coherent PWA description of the entire traffic network. A discrete-time PWA dynamical system in state space notation can be described as follows:

$$x(k+1) = A_i x(k) + B_i u(k) + f_i, \tag{B.19}$$

$$y(k) = C_i x(k) + D_i u(k) + g_i, \tag{B.20}$$

$$\left[x^{\mathrm{T}}(k)\ u^{\mathrm{T}}(k)\right]^{\mathrm{T}} \in \Omega_i,\ i \in I, \tag{B.21}$$

where $x(k) \in \mathbb{R}^{n_x}, u(k) \in \mathbb{R}^{n_u}$, and $y \in \mathbb{R}^{n_y}$ denote respectively the state, input, and output vector, and where Ω_i is a convex polyhedron with $\cup_{i \in I} \Omega_i = \mathbb{R}^{n_x+n_u}$ and $\mathrm{int}(\Omega_i) \cap \mathrm{int}(\Omega_j) = \varnothing, \forall i \neq j \in I$. For each region Ω_i, $A_i \in \mathbb{R}^{n_x \times n_x}, B_i \in \mathbb{R}^{n_x \times n_u}, C_i \in \mathbb{R}^{n_y \times n_x}, D_i \in \mathbb{R}^{n_y \times n_u}$, and $f_i \in \mathbb{R}^{n_x}, g_i \in \mathbb{R}^{n_y}$ represent constant system matrices and vectors.

Furthermore, recall that the state variables refer to the mean velocities $v_{m,i}(k)$, densities $\rho_{m,i}(k)$, and flows $q_{m,i}(k)$ of vehicles, together with the flows $q_{\mathrm{o}}(k)$ and queue lengths $w_{\mathrm{o}}(k)$ at the origins, and the (spatial)-temporal components of the space-mean accelerations $a_{\ell}(k)$ and numbers of vehicles $n_{\ell}(k)$, in addition to the auxiliary variables and constraints due to the PWA approximations. The input variables refer to the variable speed limits $v_{\mathrm{ctrl},m,i}(k)$ and ramp metering rates $r_{\mathrm{o}}(k_{\mathrm{c}})$.

To obtain the above PWA system description, the individual PWA model equations should be combined for each link, segment, node, and origin of the given traffic network, yielding a cross-product of the PWA regions and therefore an exponential growth of the model. Due to the large total number of regions this results in, the composition of the full PWA traffic model is already inefficient. Moreover, when using MPC as explained in Section B.3, this PWA model has to be evaluated over several future time steps, which causes this PWA-MPC approach for the METANET model (where an MILP is used for optimization [21]) to be computationally intractable already for a small network of only a few segments.

B.5.2 A Tractable Approach Using an MLD Model

In order to do be able to efficiently solve the MPC problem based on a PWA system description with a large number of regions, we do not compose the fully integrated PWA model, yet we propose to make a conversion of the individual model equations to the following equivalent MLD description:

$$x(k+1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) + f, \tag{B.22}$$

$$y(k) = Cx(k) + D_1 u(k) + D_2 \delta(k) + D_3 z(k) + g, \tag{B.23}$$

$$E_1 x(k) + E_2 u(k) + E_3 \delta(k) + E_4 z(k) \leq h, \tag{B.24}$$

where $\delta(k) \in \{0,1\}^{\mathrm{n}_b}$ denotes a vector of binary variables and $z(k) \in \mathbb{R}^{\mathrm{n}_z}$ represents the auxiliary variables resulting from the procedure discussed next (see also [21]). Similarly, the constraints defined through the system matrices E_1, E_2, E_3, and E_4 and the constant vector h arise along with the composition of the MLD model. As in the PWA system (B.19)–(B.21) $x(k) \in \mathbb{R}^{\mathrm{n}_x}, u(k) \in \mathbb{R}^{\mathrm{n}_u}$, and $y(k) \in \mathbb{R}^{\mathrm{n}_y}$ denote respectively the state, input, and output vector.

In the MLD representation one model applies in which the binary and auxiliary variables that are needed to define the regions are directly included in the model through additional constraints. As compared to the full PWA system description, in this MLD representation one large but tractable model applies, composed simply by stacking the individual linear and PWA model equations plus the auxiliary equations that define the PWA regions for the individual equations, resulting in a model size that grows linearly.

In order to arrive at a directly solvable optimization problem, an iteration of the MPC method based on the MLD model can be written as an MILP where some of the decision variables belong to an integer domain (in this case solely binary) and some to a real domain. The following equivalence statements summarize the conversion

(adapted from [21, 199]):

$$[f(x) \leq c \Leftrightarrow \delta = 1] \quad \Leftrightarrow \quad \begin{cases} f(x) \leq c + (M-c)(1-\delta), \\ f(x) \geq c(1-\delta) + \varepsilon + (m-\varepsilon)\delta, \end{cases} \tag{B.25}$$

$$\delta = \delta_1 \delta_2 \quad \Leftrightarrow \quad \begin{cases} -\delta_1 + \delta \leq 0, \\ -\delta_2 + \delta \leq 0, \\ \delta_1 + \delta_2 - \delta \leq 1, \end{cases} \tag{B.26}$$

$$z = \delta f(x) \quad \Leftrightarrow \quad \begin{cases} z \leq M\delta, \\ z \geq m\delta, \\ z \leq f(x) - m(1-\delta), \\ z \geq f(x) - M(1-\delta). \end{cases} \tag{B.27}$$

Here, binary dummy variables (denoted by $\delta \in \{0,1\}$) are introduced to indicate whether a certain region applies that is associated with one of the affine pieces of the PWA function. The function $f(\cdot)$ is affine and defined over a set of bounded variables, where we can assume that $x(k), u(k)$, and $y(k)$ are bounded; hence, the constants m, M that denote respectively a lower and upper bound of $f(\cdot)$ can be selected such that they are finite. Finally, c denotes an arbitrary constant and the constant ε denotes the machine precision (used to turn a strict inequality into a non-strict inequality that fits the MILP framework).

To briefly illustrate the transformation of a PWA model equation into an MLD model equation using the above statements, we take an expression of the form (B.4) or (A.7), i.e.: $f = \min(f_1, f_2)$ that can be replaced by $f = f_1\delta + f_2(1-\delta)$ where $\delta = 1$ iff $f_1 \leq f_2$ and $\delta = 0$ otherwise, according to the constraints (B.25). The latter expression of f can again be written $f = z_1 + f_2 - z_2$ with auxiliary variables $z_i = f_i\delta$, $i = 1,2$, according to the constraints (B.27). If in addition a third term f_3 arises in the PWA equation $f = \min(f_1, f_2, f_3)$, such that $f = f_1\delta_1 + f_2\delta_2 + f_3(1-\delta_1)(1-\delta_2)$, where

$$\delta_1 = 1 \Leftrightarrow (f_1 \leq f_2 \text{ and } f_1 \leq f_3), \tag{B.28}$$

$$\delta_2 = 1 \Leftrightarrow (f_2 \leq f_1 \text{ and } f_2 \leq f_3), \tag{B.29}$$

this gives rise to a multiplication of binary variables, $\delta_1\delta_2$, for which the constraints in (B.26) are needed.

Finally, as mentioned in Section B.3, it is needed to transform the norm $\|s\|_1, s \in \mathbb{R}^n$ in a similar manner. Here, an optimization objective $\min_{s\in\mathbb{R}^n} \|s\|_1 = \sum_{i=1}^n |s_i|$ can be substituted by the following linear expressions, as can be easily verified:

$$\min_{s,z\in\mathbb{R}^n} \sum_{i=1}^n z_i \text{ subject to } -z \leq s \leq z. \tag{B.30}$$

Likewise, an ∞-norm can be transformed into a linear problem, and alternatively a 2-norm in the penalty term (B.13) can be substituted by a quadratic objective.

All in all, we now end up with an MILP or MIQP formulation, which still belongs to the class of NP-hard problems, i.e., it is generally accepted that no polynomial time

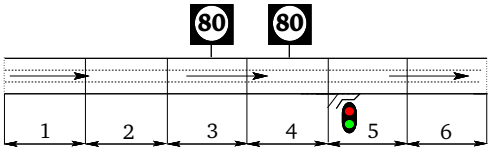


Figure B.7: *Set-up of the case study*

algorithm exists that solves the problem to optimality [68]. However, for relatively small problem instances efficient solvers are available that are based on e.g., column generation techniques, branch and bound, or cutting plane methods [8, 130]. Concerning the implications of this MILP conversion for the ease of computation of the final MPC problem, it thus remains of interest to keep the number of regions and therefore the number of additional binary variables small. However, a division using fewer regions may further increase the approximation error. Therefore, an appropriate balance between computational speed and approximation accuracy has to be found, which has already been considered in the approximation of the individual model equations in Section B.4.2 and which will be evaluated next.

B.6 Case Study

In this section, a benchmark from the literature [93] is adopted in order to facilitate the comparison of results. Here, simulation results from the original nonlinear METANET model with emissions are compared with the MLD-MPC approach based on the PWA models. Both the computation time and the performance with respect to the optimization objectives are analyzed. In addition to the benchmark just mentioned, a second, more diverse demand profile is considered in order to evaluate the performance of the MLD-MPC approach when experiencing larger variations in the traffic states.

B.6.1 Set-up

The freeway setting used in the simulations is depicted in Fig. B.7: six segments are considered of which segments 3 and 4 are subject to variable speed limits and a metered on-ramp is placed between segments 4 and 5. Additionally, an upper bound is added constraining the queue length $w_{o_2}(k)$ to 100 veh. The demand profiles are depicted in Fig. B.8, where the mainstream origin and on-ramp have a capacity of 4000 and 2000 veh/h respectively. Parameter values can be found in Table B.1. Further, the temporal aspect of the vehicular emissions and fuel consumption is considered (see Section B.2.2). Finally, conform to the analysis in [93], a prediction and control horizon of respectively $N_p = 7, N_c = 5$ is found to lead to the best results for demand profile 1. We simulate the freeway dynamics for a simulation horizon corresponding to 2.5 h with the controller sampling time $T_c = 1$ min.

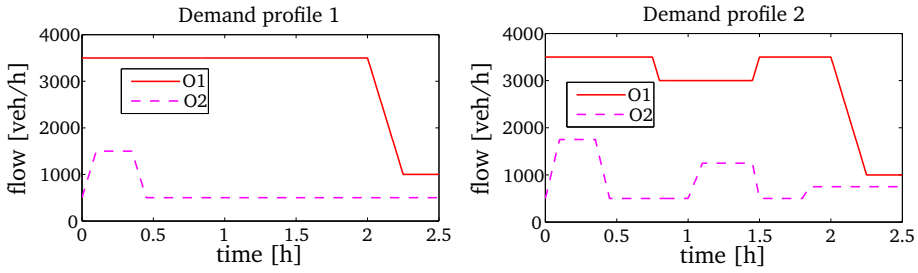


Figure B.8: *On-ramp and mainstream demand profiles*

B.6.2 Results

Table B.2 shows the TTS, total emission values, and mean CPU times obtained for three scenarios, i.e., minimizing

(S1) TTS only ($c_1 = 1, c_2 = c_3 = 0$)

(S2) TTS and HC emissions ($c_1 = 1, c_2 = 0.25, c_3 = 0$)

(S3) TTS and NO_x emissions as well as fuel consumption (FC) ($c_1 = 1, c_2 = 0.4, c_3 = 0$).

These scenarios were chosen to evaluate our approach for different problem sizes. Additionally, the standard deviation (σ) and the minimum and maximum CPU times are provided. The values obtained in the uncontrolled case are used to normalize the performance criteria. Further, the percentage difference is given between the objective function values obtained under nonlinear control or while using MLD-MPC.

As can be seen from Table B.2, for scenario S1 one run of the MLD-MPC optimization took 6 s on average concerning the in total 150 runs or simulation time steps, as compared to approximately 47 s, which is the time required for 10 iterations of the nonlinear solver[1]. Further, it should be noted that whereas the MILP is solved to optimality at once, nonlinear MPC requires an *a priori* unknown number of iterations before convergence is reached. Therefore, 10 iterations should be seen as a lower bound: in scenario S2 and S3 at least 15 runs were required to obtain good results, where the number of required runs is based on the variance in the solution returned by consecutive optimization runs for different random, initial starting points [93]. A certain number of runs may also be required in order to obtain a feasible solution in case of constraints. Finally, it can be seen that the computational advantage of applying MLD-MPC is of similar order for all three cases.

As for the deviations of the objective function values when comparing MLD-MPC to nonlinear MPC, the absolute emission values can be said to differ relatively little. On the other hand, the values of the TTS have a deviation of roughly 10% when

[1]The CPU times were obtained adopting the Tomlab CPLEX and fmincon environment within 32-bit Matlab 7.9.0 (R2009b) on a Linux PC with a 3GHz Intel Core Duo processor and 3.7Gb RAM.

(S1) TTS	TTS (veh·h)	TNO_x (kg)	TCO (kg)	THC (kg)	TFC (l)	CPU (s) [σ, min, max]
Uncontrolled	$1.463 \cdot 10^3$	6.683	59.49	4.103	$4.654 \cdot 10^3$	–
Nonlinear MPC	$1.268 \cdot 10^3$	6.988	60.45	3.916	$4.348 \cdot 10^3$	46.71 (10 runs) [25.0, 9.56,131]
MLD-MPC (%diff.)	$1.392 \cdot 10^3$ (9.8%)	6.814 (-2.5%)	59.92 (-0.9%)	4.122 (5.3%)	$4.526 \cdot 10^3$ (4.1%)	5.829 (-87.5%) [16.2, 0.2137, 186]

(S2) TTS + HC	TTS (veh·h)	THC (kg)	CPU (s) [σ, min, max]
Uncontrolled	$1.463 \cdot 10^3$	4.103	–
Nonlinear MPC	$1.287 \cdot 10^3$	3.857	62.05 (15 runs) [29.7, 30.3, 302]
MLD-MPC (%diff.)	$1.403 \cdot 10^3$ (9.1%)	4.045 (4.9%)	14.5 (-76.7%) [29,1.7,231]

(S3) TTS + TNO_x + TFC	TTS (veh·h)	TNO_x (kg)	TFC (l)	CPU (s) [σ, min, max]
Uncontrolled	$1.463 \cdot 10^3$	6.683	$4.539 \cdot 10^3$	–
Nonlinear MPC	$1.380 \cdot 10^3$	6.959	$4.353 \cdot 10^3$	80.44 (15 runs) [56.5, 24.2, 248]
MLD-MPC (%diff.)	$1.441 \cdot 10^3$ (4.4%)	6.805 (-2.2%)	$4.498 \cdot 10^3$ (3.3%)	29.34 (-63%) [29.7,0.221,183]

Table B.2: *Comparison of the different scenarios for demand profile 1.*

compared to nonlinear MPC. This reduction in accuracy results from the trade-off with the improvements in computational efficiency. Further, as expected, the TTS values increase in both the nonlinear and MLD-MPC approaches when emission control is applied, where the largest increase occurs in the most comprehensive scenario S3. For this scenario it should be noted that the total NO_x emissions were not reduced with respect to the uncontrolled case for both the MLD and the nonlinear MPC approach. However, this value as well as the value for TFC did get reduced in comparison to the values that were obtained under the TTS minimization objective of S1. In order to further decrease the NO_x emission, the weights for the NO_x factor in the goal function could be adjusted. Here it should be kept into account that due to the multi-objective nature of the optimization problem at hand, the minimization of some terms may have opposing effects on the minimization of other elements in the objective function. In particular, it should be noted that in scenario S2 and S3, the approximated emissions are directly incorporated in the objective functions in case of MLD-MPC.

Further, also under the more diverse, second demand profile depicted in Fig. B.8, a reduction in computational speed can be observed, while the improvement in TTS is very close to the reduction in TTS while adopting nonlinear MPC. Here, we only applied ramp metering for both the MLD-MPC and the nonlinear MPC approach. When taking into account the behavior of the traffic network while using the PWA approximated model, as depicted in Fig. B.10, it can be said that the behavior of the traffic flow, density and speed over the simulation horizon is similar with the nonlinear case in Fig. B.9. The uncontrolled behavior is plotted in Fig. B.11, where two large peaks in the queue length at origin 1 can be observed. In the controlled case, these peaks are reduced to a smaller peak of around 100 vehicles around 45 minutes time, and a larger peak around 2 hours. The queue at origin 2 is at its upper bound in both cases, while in the MLD-MPC case, more fluctuations in the queue lengths can be observed. Overall, there is more fluctuation in the different variables for MLD-MPC, which is smoothened in the nonlinear MPC case. This difference could be explained by the use of smooth versus nonsmooth, PWA model equations.

Recall that in this chapter a proof-of-concept is presented. To show the performance of the presented method in real-life traffic scenarios, it should be applied to diverse case studies based on real-life situations in which varying demand profiles could be investigated. There, in order to yield the best overall accuracy, one should consider fine-tuning the individual function approximations as well as calibrating the model parameters in the MLD-MPC approach for these real-life scenarios. These are topics for future research. Finally, it is important to note that instead of taking the optimal decision variables resulting from MLD-MPC as a final control input, these results could also be used as a starting point for one single nonlinear optimization run, which would still yield a faster solution compared to the conventional approach given the gains in computation times while using MLD-MPC.

B.6.3 Computational Efficiency

Concerning the computational efficiency of the MLD method it should be noted that the specific problem structure could be exploited to speed-up the optimization, i.e.,

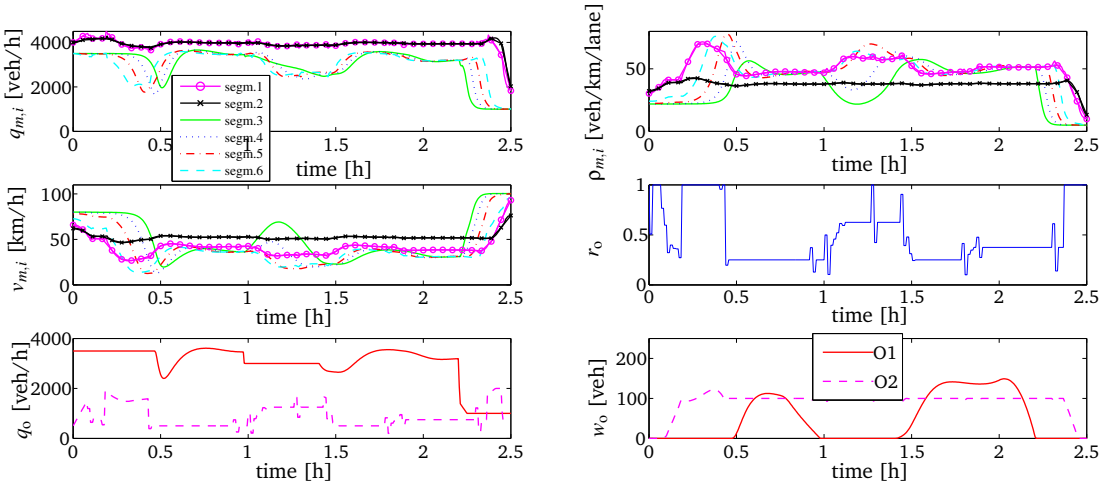


Figure B.9: *Simulation results for demand profile 2 - nonlinear MPC*

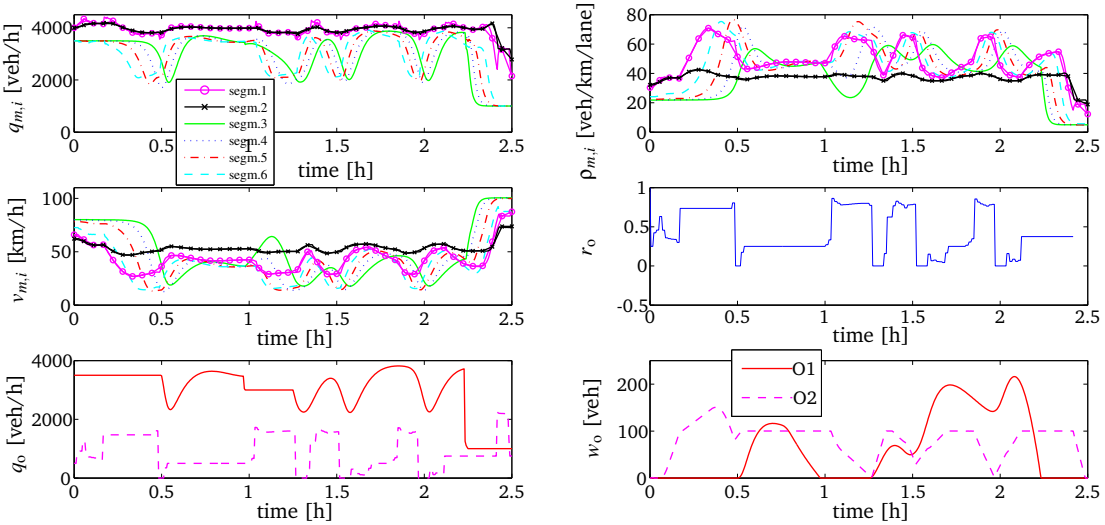


Figure B.10: *Simulation results for demand profile 2 - MLD-MPC*

Table B.3: *Comparison of TTS and CPU time and their relative differences for demand profile 2.*

(1) TTS	TTS (veh·h)	CPU (s) [σ, min, max]
Uncontrolled	$1.716 \cdot 10^3$	–
Nonlinear MPC	$1.641 \cdot 10^3$	8.98 (25 runs) [6.1, 3.4, 43]
MLD-MPC (% diff.)	$1.657 \cdot 10^3$ (1.0%)	1.92 (-78.6%) [2.8, 0.15, 14]

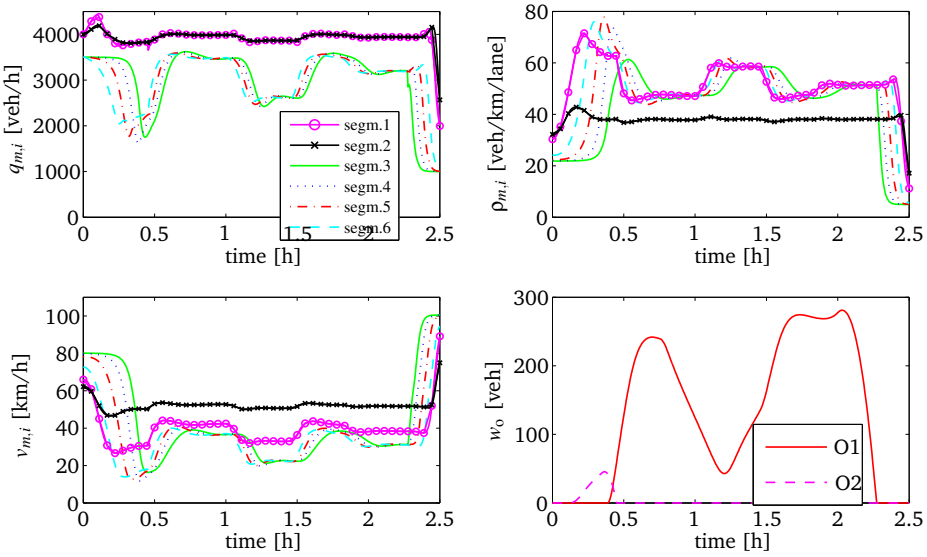


Figure B.11: *Uncontrolled behavior under demand profile 2*

computational advantages could result from tuning the solver to the particular problem like by changing the structure of the constraint matrices [166]. An overview of computational efficiency in the solving of MILPs when using different solvers can be found in [22]. Also, as an alternative to nonlinear control, the feasible-direction method proposed in [114] could be used to compare the computational requirements. In particular, results of the designated Advanced Motorway Optimal Control (AMOC) toolbox for the feasible-direction implementation of ramp metering can be found in [109, 110, 116], while integrated ramp metering and speed control is considered in [32], where the method is found to be fast enough for real-life implementation. Here, it should be noted that the constraints on the state variables as considered in our case study for the maximum queue length can result in an increase in the required computational time when implemented as a hard constraint; in [32], this constraint is incorporated in the penalty function as a soft constraint. The latter approach however does not in general yield the same traffic behavior; in instances where a rather large penalty is required to keep the queue length below the limit, the effect on the control variables will be substantial. Hence, it would be fruitful to

perform an analysis on the impact of optimal control actions on the traffic behavior as well as on computational efficiency of both methods under the same conditions.

B.7 Discussion

A piecewise affine (PWA) approximation of the nonlinear traffic flow model METANET integrated with the VT-macro model for vehicular emissions and fuel consumption has been proposed in order to deal with the computational complexity of the original nonlinear nonconvex model-based traffic control approach, which is currently hindered from application in real-life traffic networks due to the required computation time.

Here, a direct MPC implementation of the fully PWA system is still computationally intractable due to the large number of regions that should be considered. Instead, a mixed-logical dynamic (MLD) representation of the approximated model has been adopted, after which MPC can be applied. Compared to the exponential growth of the full PWA model size, the MLD model grows only linearly by stacking the individual model equations.

Several methods to approximate the nonlinear functions appearing in the models have been discussed and the transformation to a ready-to-implement mixed-integer linear optimization programming (MILP) problem has been provided. This method has been tested in a case study with hard state constraints comparing the performance of the MLD-MPC method with the original nonlinear-programming MPC approach with respect to the trade-off between computational speed and accuracy, where the latter is measured by the deviation of objective function values and traffic dynamics as compared to those based on the original nonlinear-programming MPC approach. The simulation results when considering two hypothetical demand profiles showed that MLD-MPC can indeed be applied at an improved computational efficiency at the cost of some deterioration of the control performance.

As further steps to bring the MLD-MPC approach to be applicable in real life, more elaborate case studies could be performed in order to investigate in detail the trade-off between approximation accuracy (i.e., the number of affine pieces or regions) and computational efficiency, in a setting that more closely resembles real-life traffic conditions. Moreover, an extensive analysis of the effect of the MLD-MPC method on the traffic behavior could be studied under different traffic scenarios, and compared to the performance to traffic control using e.g., the feasible-direction approach [32, 114]. Here, an extensive sensitivity analysis of various approximation methods could be fruitful in order to fine-tune the individual approximations. Finally, the MLD-MPC approach could also be applied to different traffic models like higher-order macroscopic models [95], the discretized Payne model [161] and the cell and link transmission models [46, 202].

Bibliography

[1] K. Ahn, A. Trani, H. Rakha, and M. Van Aerde. Microscopic fuel consumption and energy emission models. In *Proceedings of the 78th Annual Meeting of the Transportation Research Board*, Washington, DC, USA, January 1999.

[2] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Upper Saddle River, NJ, USA, 1993.

[3] R. Arcangéli, M.C. López de Silanes, and J.J. Torrens. *Multidimensional Minimizing Splines: Theory and Applications*. Kluwer Academic Publishers, Boston, MA, USA, 2004.

[4] R. Arnott, A. de Palma, and R. Lindsey. Departure time and route choice for the morning commute. *Transportation Research Part B: Methodological*, 24(3):209–228, 1990.

[5] K.J. Arrow. *Social Choice and Individual Values*. John Wiley & Sons, New York, NY, USA, 1st edition, 1951.

[6] G. Arslan, J.R. Marden, and J.S. Shamma. Autonomous vehicle-target assignment: A game-theoretical formulation. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):584–596, 2007.

[7] K.J. Åström and B. Wittenmark. *Computer-Controlled Systems – Theory and Applications*. Prentice Hall, Upper Saddle River, NJ, USA, 3rd edition, 1997.

[8] A. Atamtürk and M.W.P. Savelsbergh. Integer-programming software systems. *Annals of Operations Research*, 140(1):67–124, 2005.

[9] R.J. Aumann and M. Maschler. Game theoretic analysis of a bankruptcy problem from the talmud. *Journal of Economic Theory*, 36(2):195–213, 1985.

[10] A. Auslender and M. Teboulle. *Asymptotic Cones and Functions in Optimization and Variational Inequalities*. Springer Monographs in Mathematics. Springer, New York, NY, USA, 2003.

[11] S. Azuma, J. Imura, and T. Sugie. Lebesgue piecewise affine approximation of nonlinear systems. *Nonlinear Analysis: Hybrid Systems*, 4(1):92–102, 2010.

[12] T. Başar. On the relative leadershop property of Stackelberg strategies. *Journal of Optimization Theory and Applications*, 11(6):655–661, 1973.

[13] T. Başar. General theory for Stackelberg games with partial state information. *Large Scale Systems*, 3(1):47–56, 1982.

[14] T. Başar. Stochastic incentive problems with partial dynamic information and multiple levels of hierarchy. *European Journal of Political Economy*, 5(2–3):203–217, 1989.

[15] T. Başar and G.J. Olsder. *Dynamic Noncooperative Game Theory*. Classics in Applied Mathematics. SIAM, Philadelphia, PA, USA, 2nd edition, 1999.

[16] T. Başar and H. Selbuz. Closed-loop Stackelberg strategies with applications in the optimal control of multilevel systems. *IEEE Transactions on Automatic Control*, 24(2): 166–179, 1979.

[17] T. Başar, A. Bensoussan, and S.P. Sethi. Differential games with mixed leadership: The open-loop solution. *Applied Mathematics and Computation*, 217(3):972–979, 2010.

[18] C.B. Barber, D.P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, 1996.

[19] D. Bauso, L. Giarre, and R. Pesenti. Consensus in noncooperative dynamic games: A multiretailer inventory application. *IEEE Transactions on Automatic Control*, 53(4): 998–1003, 2008.

[20] T. Bellemans, B. De Schutter, and B. De Moor. Model predictive control for ramp metering of motorway traffic: A case study. *Control Engineering Practice*, 14(7):757–767, 2006.

[21] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.

[22] R.E. Bixby, M. Fenelon, Z. Gu, E. Rothberg, and R. Wunderling. MIP: Theory and practice - Closing the gap. In M. J. D. Powell and S. Scholtes, editors, *System Modelling and Optimization: Methods, Theory, and Applications*, volume 174 of *IFIP Conference Proceedings*, pages 19–50, Cambridge, UK, 2000. Kluwer Academic Publishers.

[23] T. Blayac and A. Causse. Value of travel time: a theoretical legitimization of some nonlinear representative utility in discrete choice models. *Transportation Research Part B: Methodological*, 35(4):391–400, 2001.

[24] S. Boyd and L. Vandenberge. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[25] D. Braess. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, 12 (1):258–268, 1968.

[26] E.J. Bredensteiner and K.P. Bennett. Multicategory classification by support vector machines. *Computational Optimizations and Applications*, 12(1–3):53–79, 1999.

[27] M.D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, UK, 2003.

[28] E.F. Camacho and C. Bordons. *Model Predictive Control in the Process Industry*. Advances in Industrial Control. Springer-Verlag, London, UK, 1995.

[29] D.H. Cansever and T. Başar. A minimum sensitivity approach to incentive design problems. *Large Scale Systems*, 5(3):233–244, 1983.

[30] D.H. Cansever and T. Başar. Optimum/near optimum incentive policies for stochastic decision problems involving parametric uncertainty. *Automatica*, 21(5):575–584, 1985.

[31] D.H. Cansever and T. Başar. On stochastic incentive control problems with partial dynamic information. *Systems & Control Letters*, 6(1):69–75, 1985.

[32] R.C. Carlson, I. Papamichail, M. Papageorgiou, and A. Messmer. Optimal mainstream traffic flow control of large-scale motorway networks. *Transportation Research Part C*, 18(2):193–212, 2010.

[33] R.C. Carlson, I. Papamichail, M. Papageorgiou, and A. Messmer. Optimal motorway traffic flow control involving variable speed limits and ramp metering. *Transportation Science*, 44(2):238–253, 2010.

[34] T.-S. Chang and Y.-C. Ho. Incentive problems: A class of stochastic Stackelberg closed-loop dynamic games. *Systems and Control Letters*, 1(1):16–21, 1981.

[35] H.-K. Chen. *Dynamic Travel Choice Models: A Variational Inequality Approach*. Springer, Berlin, Heidelberg, Germany, 1988.

[36] A. Chinchuluun, P.M. Pardalos, and H.-X. Huang. Multilevel (hierarchical) optimization: Complexity issues, optimality conditions, algorithms. In D.Y. Gao and H.D. Sherali, editors, *Advances in Applied Mathematics and Global Optimization*, volume 17 of *Advances in Mechanics and Mathematics*, chapter 6, pages 197–221. Springer, New York, NY, USA, 2009.

[37] A.H.F. Chow. Properties of system optimal traffic assignment with departure time choice and its solution method. *Transportation Research Part B: Methodological*, 43(3): 325–344, 2009.

[38] F.H. Clarke. *Optimization and Nonsmooth Analysis*. Wiley-Interscience, New York, NY, USA, 1983.

[39] B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of Operations Research*, 153(1):235–256, 2007.

[40] G. Como, K. Savla, D. Acemoglu, M.A. Dahleh, and E. Frazzoli. Robust distributed routing in dynamical networks - part i: Locally responsive policies and weak resilience. *IEEE Transactions on Automatic Control*, 58(2):317–332, 2013.

[41] G. Como, K. Savla, D. Acemoglu, M.A. Dahleh, and E. Frazzoli. Robust distributed routing in dynamical networks -part ii: Strong resilience, equilibrium selection and cascaded failures. *IEEE Transactions on Automatic Control*, 58(2):333–348, 2013.

[42] Z. Cong, B. De Schutter, and R. Babuška. Ant colony routing algorithm for freeway networks. *Submitted to a journal*, 2013.

[43] J.S. Cramer. *The Logit Model: An Introduction for Economists*. Edward Arnold, London, UK, 1991.

[44] J.B. Cruz, Jr. Leader-follower strategies for multilevel systems. *IEEE Transactions on Automatic Control*, 23(2):244–255, 1978.

[45] J.B. Cruz, Jr. and M.A. Simaan. Ordinal games and generalized Nash and Stackelberg solutions. *Journal of Optimization Theory and Applications*, 107(2):205–222, 2000.

[46] C.F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B*, 28(4): 269–287, 1994.

[47] C. de Boor and A. Ron. On multivariate polynomial interpolation. *Constructive Approximation*, 6(3):287–302, 1990.

[48] K.A. De Jong. *Evolutionary Computation: A Unified Approach*. A Bradford Book. The MIT Press, Cambridge, MA, USA, 2006.

[49] K.A. De Jong. *An Analysis of the Behavior of a Class of Genetic Adaptive Systems.* PhD dissertation, University of Michigan, 1975.

[50] B. De Schutter, W.P.M.H. Heemels, and A. Bemporad. On the equivalence of linear complementarity problems. *Operations Research Letters*, 30(4):211–222, 2002.

[51] S. Defermos and F.T. Sparrow. Optimal resource allocation and toll patterns in user-optimized transport networks. *Journal of Transport Economics and Policy*, 5:184–200, 1971.

[52] S. Dempe. *Foundations of Bilevel Programming*, volume 61 of *Nonconvex Optimization and Its Applications*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[53] A.J. DeSerpa. Microeconomic theory and the valuation of traveltime: some clarifications. *Regional Urban Economics*, 2(4):401–410, 1973.

[54] T.A. Edmunds and J.F. Bard. Algorithms for nonlinear bilevel mathematical programming. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(1):83–89, 1991.

[55] H. Ehtamo and R.P. Hämäläinen. Construction of optimal affine incentive strategies for linear-quadratic Stackelberg games. In *Proceedings of the 24th IEEE Conference on Decision and Control*, pages 1093–1098, Fort Lauderdale, FL, USA, 1985.

[56] H. Ehtamo and R.P. Hämäläinen. Incentive strategies and equilibria for dynamic games with delayed information. *Journal of Optimization Theory and Applications*, 63(3):355–369, 1989.

[57] H. Ehtamo, M. Kitti, and R.P. Hämäläinen. Recent studies on incentive design problems in game theory and management science. In G. Zaccour, editor, *Optimal Control and Differential Games*, volume 5 of *Advances in Computational Management Science*, chapter 8, pages 121–134. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[58] J.C. Engwerda. *LQ Dynamic Optimization and Differential Games*. John Wiley & Sons, Chichester, UK, 2005.

[59] N.P. Faísca, P.M. Saraiva, Berç Rustem, and E.N. Pistikopoulos. A multi-parametric programming approach for multilevel hierarchical and decentralised optimisation problems. *Computational Management Science*, 6(4):377–397, 2009.

[60] G. Ferrari-Trecate. Hybrid identification toolbox (HIT). http://www-rocq.inria.fr/who/Giancarlo.Ferrari-Trecate/HIT_toolbox.html, 2005.

[61] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. *Automatica*, 39(2):205–217, 2003.

[62] C. Fisk. Some developments in equilibrium traffic assignment. *Transportation Research Part B: Methodological*, 14(3):243–255, 1980.

[63] R. Fletcher. *Practical Methods of Optimization, Volume 1: Unconstrained Optimization*. John Wiley & Sons, Chichester, UK, 1980.

[64] G.F. Franklin, J.D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Prentice-Hall, Upper Saddle River, NJ, USA, 6th edition, 2010.

[65] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA, USA, 2nd edition, 1991.

[66] G. Fung and O.L. Mangasarian. Proximal support vector machine classifiers. In F. Provost and R. Srikant, editors, *Proceedings of KDD-2001: Knowledge Discovery and Data Mining*, pages 77–86. Asscociation for Computing Machinery, San Francisco, CA, USA, 2001.

[67] C.E. Garcia, D.P. Prett, and M. Morari. Model predictive control: Theory and practice: A survey. *Automatica*, 25(3):335–348, 1989.

[68] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, San Francisco, CA, USA, 1979.

[69] W. Gellert, S. Gottwald, M. Hellwich, H. Kästner, and H. (Eds.) Künstner. *VNR Concise Encyclopedia of Mathematics*. Van Nostrand Reinhold, New York, NY, USA, 2nd edition, 1989.

[70] A.H. Ghods, L. Fu, and A. Rahimi-Kian. An efficient optimization approach to real-time coordinated and integrated freeway traffic control. *IEEE Transactions on Intelligent Transportation Systems*, 11(4):873–884, 2010.

[71] R. Gibbons. *Game Theory for Applied Economists*. Princeton University Press, Princeton, NJ, USA, 1992.

[72] E.G. Gilbert, D.W. Johnson, and S.S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Transactions on Robotics and Automation*, 4:193–203, 1988.

[73] M. Goh. Congestion management and electronic road pricing in singapore. *Journal of Transport Geography*, 10(1):29–38, 2005.

[74] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, MD, USA, 2nd edition, 1989.

[75] N. Groot, B. De Schutter, S. Zegeye, and H. Hellendoorn. Model-based predictive traffic control: A piecewise-affine approach based on METANET. In *Proceedings of the 18th IFAC World Congress*, pages 10709–10714, Milan, Italy, August–September 2011.

[76] N. Groot, B. De Schutter, S.K. Zegeye, and H. Hellendoorn. Model-based traffic and emission control using PWA models: A mixed-logical dynamic approach. In *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems*, pages 2142–2147, Washington, DC, USA, October 2011.

[77] N. Groot, B. De Schutter, and H. Hellendoorn. Existence conditions for an optimal affine leader function in the reverse Stackelberg game. In *Proceedings of the 15th IFAC Workshop on Control Applications of Optimization*, Rimini, Italy, September 2012. Paper 16.

[78] N. Groot, B. De Schutter, and H. Hellendoorn. A full characterization of the set of optimal affine leader functions in the reverse Stackelberg game. In *Proceedings of the 51st IEEE Conference on Decision and Control*, pages 6995–7000, Maui, HI, USA, December 2012.

[79] N. Groot, B. De Schutter, and H. Hellendoorn. Achieving system-optimal splitting rates in a freeway network using a reverse Stackelberg approach. In *Proceedings of the 13th IFAC Symposium on Control in Transportation Systems*, pages 132–137, Sofia, Bulgaria, September 2012.

[80] N. Groot, B. De Schutter, and H. Hellendoorn. Dynamic optimal routing based on a reverse Stackelberg game approach. In *Proceedings of the 15th International IEEE Conference on Intelligent Transportation Systems*, pages 782–787, Anchorage, AK, USA, September 2012.

[81] N. Groot, B. De Schutter, and H. Hellendoorn. Reverse Stackelberg games, part 1: Basic framework. In *Proceedings of the 2012 IEEE Multi-Conference on Systems and Control*, pages 421–426, Dubrovnik, Croatia, October 2012.

[82] N. Groot, B. De Schutter, and H. Hellendoorn. Reverse Stackelberg games, part 2: Results and open issues. In *Proceedings of the 2012 IEEE Multi-Conference on Systems and Control*, pages 427–432, Dubrovnik, Croatia, October 2012.

[83] N. Groot, B. De Schutter, and H. Hellendoorn. Optimal affine leader functions in reverse Stackelberg games: Existence conditions and characterization. *Submitted to a journal*, 2012.

[84] N. Groot, B. De Schutter, and H. Hellendoorn. Integrated model predictive traffic and emission control using a piecewise-affine approach. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):587–598, 2013.

[85] N. Groot, B. De Schutter, and H. Hellendoorn. Optimal leader functions for the reverse Stackelberg game: Splines and basis functions. In *Proceedings of the 2013 European Control Conference*, pages 696–701, Zürich, Switzerland, July 2013.

[86] N. Groot, B. De Schutter, and H. Hellendoorn. On systematic computation of optimal nonlinear solutions for the reverse Stackelberg game. *Submitted to a journal*, 2013.

[87] N. Groot, B. De Schutter, and H. Hellendoorn. Towards system-optimal routing in urban and freeway traffic networks: A reverse Stackelberg game approach. *Submitted to a journal*, 2013.

[88] T. Groves. Incentives in teams. *Econometrica*, 41(4):617–631, 1973.

[89] P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on Scientific and Statistical Computing*, 13(5):1194–1217, 1992.

[90] B. Hasselblatt and A. Katok. *A First Course in Dynamics*. Cambridge University Press, Cambridge, UK, 2003.

[91] X. He, A. Prasad, S.P. Sethi, and G.J. Gutierrez. A survey of Stackelberg differential game models in supply chain and marketing channels. *Journal of Systems Science and Systems Engineering*, 16(4):385–413, 2007.

[92] A. Hegyi. *Model Predictive Control for Integrating Traffic Control Measures*. Phd dissertation, Delft University of Technology, 2004.

[93] A. Hegyi, B. De Schutter, and H. Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C*, 13(3):185–209, 2005.

[94] A. Hegyi, B. De Schutter, and J. Hellendoorn. Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):102–112, 2005.

[95] D. Helbing. Theoretical foundation of macroscopic traffic models. *Physica A*, 219(3–4): 375–390, 1995.

[96] P.S.C. Heuberger, P.M.J. Van den Hof, and B. (Eds.) Wahlberg. *Modelling and Identification with Rational Orthogonal Basis Functions*. Springer-Verlag, London, UK, 2005.

[97] Y.-C. Ho, P.B. Luh, and R. Muralidharan. Information structure, Stackelberg games, and incentive controllability. *IEEE Transactions on Automatic Control*, 26(2):454–460, 1981.

[98] Y.-C. Ho, P.B. Luh, and G.J. Olsder. A control-theoretic view on incentives. *Automatica*, 18(2):167–179, 1982.

[99] Y.C. Ho. On incentive problems. *Systems & Control Letters*, 3(2):63–68, 1983.

[100] B. Hobbs and S. Nelson. A nonlinear bilevel model for analysis of electric utility demand-side planning issues. *Annals of Operations Research*, 34(1):255–274, 1992.

[101] S.P. Hoogendoorn and P.H.L. Bovy. State-of-the-art of vehicular traffic flow modelling. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 215(4):283–303, 2001.

[102] R.G. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32(4):146–164, 1985.

[103] Y.-w. Jing and S.-y. Zhang. The solution to a kind of Stackelberg game systems with multi-follower: Coordinative and incentive. In A. Bensoussan and J.L. Lions, editors, *Analysis and Optimization of Systems*, volume 111 of *Lecture Notes in Control and Information Science*, pages 593–602. Springer-Verlag, Berlin, Germany, 1988.

[104] O. Johansson. Optimal road-pricing: Simultaneous treatment of time losses, increased fuel consumption, and emissions. *Transportation Research Part D: Transport and Environment*, 2(2):77–87, 1997.

[105] D. Joksimovic, M.C.J. Bliemer, and P.H.L. Bovy. Optimal toll design problem in dynamic traffic networks with joint route and departure time choice. *Transportation Research Record: Journal of the Transportation Research Board*, pages 61–72, 2004.

[106] D. Joksimovic, M.C.J. Bliemer, P.H.L. Bovy, and Z. Verwater-Lukszo. Dynamic road pricing for optimizing network performance with heterogeneous users. In *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, pages 407–412, Tucson, AZ, USA, 2005.

[107] H.R. Kashani and G.N. Saridis. Intelligent control for urban traffic systems. *Automatica*, 19(2):191–197, 1983.

[108] E. Kohlberg and J.-F. Mertens. On the strategic stability of equilibria. *Econometrica*, 54 (5):1003–1037, 1986.

[109] A. Kotsialos and M. Papageorgiou. Nonlinear optimal control applied to coordinated ramp metering. *IEEE Transactions on Control Systems Technology*, 12(6):920–933, 2004.

[110] A. Kotsialos and M. Papageorgiou. Efficiency and equity properties of freeway network-wide ramp metering with AMOC. *Transportation Research Part C*, 12(6):401–420, 2004.

[111] A. Kotsialos, M. Papageorgiou, and A. Messmer. Integrated optimal control of motorway traffic networks. In *Proceedings of the 18th American Control Conference*, pages 2183–2187, San Diego, CA, USA, June 1999.

[112] A. Kotsialos, M. Papageorgiou, and A. Messmer. Optimal coordinated and integrated motorway network traffic control. In *Proceedings of the 14th International Symposium of Transportation and Traffic Theory (ISTTT)*, pages 621–644, Jerusalem, Israel, July 1999.

[113] A. Kotsialos, M. Papageorgiou, and F. Middelham. Optimal coordinated ramp metering with advanced motorway optimal control. *Proceedings of the 80th Annual Meeting of the Transportation Research Board*, 1748:55–65, 2001.

[114] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET. *IEEE Transactions on Intelligent Transportation Systems*, 3(4):282–292, 2002.

[115] A. Kotsialos, M. Papageorgiou, M. Mangeas, and H. Haj-Salem. Coordinated and integrated control of motorway networks via non-linear optimal control. *Transportation Research Part C*, 10(1):65–84, 2002.

[116] A. Kotsialos, M. Papageorgiou, and F. Middelham. Local and optimal coordinated ramp metering for freeway networks. *Journal of Intelligent Transportation Systems*, 9 (4):187–203, 2005.

[117] M. Kvasnica, P. Grieder, and M. Baotić. Multi-parametric toolbox (mpt). http://control.ee.ethz.ch/~mpt/, 2004.

[118] M. Kvasnica, A. Szücs, and M. Fikar. Automatic derivation of optimal piecewise affine approximations of nonlinear systems. In *Proceedings of the 18th IFAC World Congress*, pages 8675–8680, Milano, Italy, August–September 2011.

[119] M. Labbé, P. Marcotte, and G. Savard. A bilevel model of taxation and its application to optimal highway pricing. *Management Science*, 44(12):1608–1622, 1998.

[120] J.J. Laffont and D. Martimort. *The Theory of Incentives: The Principal-Agent Model*. Princeton University Press, Princeton, NJ, USA, 2002.

[121] C. Langbort. A mechanism design approach to dynamic price-based control of multi-agent systems. In A. Rantzer and R. Johansson, editors, *Distributed Decision Making and Control*, volume 417 of *Lecture Notes in Control and Information Sciences*, pages 113–129. Springer Verlag, London, UK, 2012.

[122] C. Lau, editor. *Neural Networks: Theoretical Foundations and Analysis*. IEEE Press, Piscataway, NJ, USA, 1992.

[123] G. Leitmann. On generalized Stackelberg strategies. *Journal of Optimization Theory and Applications*, 26(4):637–643, 1978.

[124] J. Lemaire. An application of game theory: cost allocation. *Astin Bulletin*, 14(1):61–81, 1984.

[125] M. Li, J.B. Cruz, Jr., and M.A. Simaan. An approach to discrete-time incentive feedback Stackelberg games. *IEEE Transactions on Systems, Man, and Cybernatics - Part A: Systems and Humans*, 32(4):472–481, 2002.

[126] E.H. Lieb and M. Loss. *Analysis*. Graduate Studies in Mathematics, Vol. 14. American Mathematical Society, Providence, RI, USA, 1997.

[127] M.B. Lignola and J. Morgan. Topological existence and stability for Stackelberg problems. *Journal of Optimization Theory and Applications*, 84(1):145–169, 1995.

[128] S. Lin, B. De Schutter, S.K. Zegeye, H. Hellendoorn, and Y. Xi. Integrated urban traffic control for the reduction of travel delays and emissions. In *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 677–682, September 2010.

[129] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn. Fast model predictive control for urban road networks via MILP. *IEEE Transactions on Intelligent Transportation Systems*, 12(3):846–856, 2011.

[130] J.T. Linderoth and T.K. Ralphs. Noncommercial software for mixed-integer linear programming. In J. Karlof, editor, *Integer Programming: Theory and Practice*, pages 253–303. CRC Press Operations Research Series, 2005.

[131] B. Liu. Stackelberg-Nash equilibrium for multilevel programming with multiple followers using genetic algorithms. *Computers and Mathematics with Applications*, 36(7): 79–89, 1998.

[132] X. Liu and S. Zhang. Optimal incentive strategy for leader-follower games. *IEEE Transactions on Automatic Control*, 37(12):1957–1961, 1992.

[133] D.G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, New York, NY, USA, 1969.

[134] P.B. Luh, Y.-C. Ho, and R. Muralidharan. Load adaptive pricing: An emerging tool for electric utilities. *IEEE Transactions on Automatic Control*, 27(2):320–329, 1982.

[135] I. Macho-Stadler and J.D. Pérez-Castrillo. *An Introduction to the Economics of Information: Incentives and Contracts*. Oxford University Press, Oxford, UK, 1997.

[136] J.K. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, Harlow, UK, 2002.

[137] J.R. Marden, G. Arslan, and J.S. Shamma. Cooperative control and potential games. *IEEE Transacions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(6):1393–1407, 2009.

[138] J. Marschak and R. Radner. *Economic Theory of Teams*. Yale University Press, New Haven, CT, USA, 1972.

[139] G. Martín-Herrán and G. Zaccour. Credible linear-incentive equilibrium strategies in linear-quadratic differential games. In O. Pourtallier, V. Gaitsgory, and P. Bernhard, editors, *Advances in Dynamic Games and Their Applications*, volume 10 of *Annals of the International Society of Dynamic Games*, pages 1–31. Birkhäuser Boston, 2009.

[140] A.D. May. *Traffic Flow Fundamentals*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1990.

[141] A.D. May and D.S. Milne. Effects of alternative road pricing systems on network performance. *Transportation Research Part A: Policy and Practice*, 34(6):407–436, 2000.

[142] D.K. Merchant and G.L. Nemhauser. A model and an algorithm for the dynamic traffic assignment problems. *Transportation Science*, 12(3):183–199, 1978.

[143] M.D. Mesarovic, D. Macko, and Y. Takahara, editors. *Theory of Hierarchical Multilevel Systems*. Academic Press, New York, NY, USA, 1970.

[144] A. Messmer and M. Papageorgiou. METANET: A macroscopic simulation program for motorway networks. *Traffic Engineering and Control*, 31(9):466–470, 1990.

[145] M. Morari and J.H. Lee. Model predictive control: Past, present and future. *Computers & Chemical Engineering*, 23(4):667–682, 1999.

[146] S.A. Morrison. A survey of road pricing. *Transportation Research A*, 20A(2):87–97, 1986.

[147] T.S. Motzkin, H. Raiffa, G.L. Thompson, and R.M. Thrall. The double description method. In H. Kuhn and A. Tucker, editors, *Contributions to the Theory of Games*, volume 28 of *Annals of Mathematics Studies*, pages 51–73. Princeton University Press, Princeton, NJ, USA, 1953.

[148] J.F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

[149] R. Negenborn. *Multi-Agent Model Predictive Control with Applications to Power Networks*. PhD dissertation, Delft University of Technology, 2007.

[150] D.M. Newbery. Road damage externalities and road user charges. *Econometrica*, 56 (2):295–316, 1988.

[151] D. Ngoduy and S.P. Hoogendoorn. An automated calibration procedure for macroscopic traffic flow models. In *Proceedings of the 10th IFAC Symposium on Control in Transportation Systems*, pages 295–300, Tokyo, Japan, August 2003.

[152] P.-y. Nie. Dynamic discrete-time multi-leader-follower games with leaders in turn. *Computers and Mathematics with Applications*, 61(8):2039–2043, 2011.

[153] G.J. Olsder. Phenomena in inverse Stackelberg games, part 1: Static problems. *Journal of Optimization Theory and Applications*, 143(3):589–600, 2009.

[154] G.J. Olsder. Phenomena in inverse Stackelberg games, part 2: Dynamic problems. *Journal of Optimization Theory and Applications*, 143(3):601–618, 2009.

[155] P. Olszewski and L. Xie. Modelling the effects of road pricing on traffic in singapore. *Transportation Research Part A: Policy and Practice*, 39(7–9):755–772, 2005.

[156] M.J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, MA, USA, 1994.

[157] M. Papageorgiou, J.-M. Blosseville, and H. Hadj-Salem. Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris: Part I: Modelling. *Transportation Research Part A: General*, 24(5):345–359, 1990.

[158] M. Papageorgiou, J.-M. Blosseville, and H. Hadj-Salem. Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris: Part II: Coordinated on-ramp metering. *Transportation Research Part A: General*, 24(5):361–370, 1990.

[159] G.B. Papavassilopoulos and J.B. Cruz, Jr. Sufficient conditions for Stackelberg and Nash strategies with memory. *Journal of Optimization Theory and Applications*, 31(2): 1233–260, 1980.

[160] M. Patriksson and R.T. Rockafellar. A mathematical model and descent algorithm for bilevel traffic management. *Transportation Science*, 36(3):271–291, 2002.

[161] H.J. Payne. Models of freeway traffic and control. In G.A. Bekey, editor, *Mathematical Models of Public Systems*, volume 1 of *Simulation Council Proceedings Series*, pages 51–61, La Jolla, CA, USA, 1971.

[162] S. Peeta and A.K. Ziliaskopoulos. Foundations of dynamic traffic assignment: The past, the present and the future. *Networks and Spatial Economics*, 1(3–4):233–265, 2001.

[163] J.B. Rawlings and D.Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison, WI, USA, 2009.

[164] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, USA, 1970.

[165] H.H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960.

[166] O. Saeki and K. Tsuji. A method for solving a class of mixed integer linear programming problem with block angular structure. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pages 308–313, Hammamet, Tunisia, October 2002.

[167] M.A. Salman and J.B. Cruz, Jr. An incentive model of duopoly with government coordination. *Automatica*, 17(6):821–829, 1981.

[168] M.A. Salman and J.B. Cruz, Jr. Team-optimal closed-loop Stackelberg strategies for systems with slow and fast modes. *International Journal of Control*, 37(6):1401–1416, 1983.

[169] W.H. Sandholm. Evolutionary implementation and congestion pricing. *Review of Economic Studies*, 69(3):667–689, 2002.

[170] R. Scattolini. Architectures for distributed and hierarchical model predictive control: A review. *Journal of Process Control*, 19(5):723–731, 2009.

[171] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Chichester, UK, 1986.

[172] H. Shen and T. Başar. Incentive-based pricing for network games with complete and incomplete information. In S. Jørgensen, M. Quincampoix, and T.L. Vincent, editors, *Advances in Dynamic Game Theory*, volume 9 of *Annals of the International Society of Dynamic Games*, pages 431–458. Birkhäuser Boston, New York, NY, USA, 2007.

[173] M. Simaan and J.B. Cruz, Jr. A Stackelberg solution for games with many players. *IEEE Transactions on Automatic Control*, 18(3):322–324, 1973.

[174] M. Simaan and J.B. Cruz, Jr. On the Stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Applications*, 11(5):533–555, 1973.

[175] M. Simaan and J.B. Cruz, Jr. Additional aspects of the Stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Applications*, 11(6):613–626, 1973.

[176] J. Maynard Smith. *Evolution and the Theory of Games*. Cambridge University Press, Cambridge, UK, 1982.

[177] E.D. Sontag. Nonlinear regulation: The piecewise linear approach. *IEEE Transactions on Automatic Control*, 26(2):346–357, 1981.

[178] K. Staňková. *On Stackelberg and Inverse Stackelberg Games & Their Applications in the Optimal Toll Design Problem, the Energy Markets Liberalization Problem, and in the Theory of Incentives*. PhD dissertation, Delft University of Technology, 2009.

[179] K. Staňková, H.-J. Von Mettenheim, and G.J. Olsder. Dynamic optimal toll design problem with second-best-flow-dependent tolling solved using neural networks. In *Proceedings of the 10th International Conference on Applications of Advanced Technologies in Transportation*, Athens, Greece, 2008.

[180] K. Staňková, G.J. Olsder, and M.C.J. Bliemer. Comparison of different toll policies in the dynamic second-best optimal toll design problem: Case study on a three-link network. *European Journal of Transport and Infrastructure Research*, 9(4):331–346, 2009.

[181] A. Sumalee, S. Shepherd, and A. May. Road user charging design: dealing with multi-objectives and constraints. *Transportation*, 36(2):167–186, 2009.

[182] J. Supernak, J. Golob, T.F. Golob, , C. Kaschade, C. Kazimi, E. Schreffler, and D. Steffey. San diego's interstate 15 congestion pricing project: Traffic-related issues. *Transportation Research Record: Journal of the Transportation Research Board*, pages 43–52, 2002.

[183] A.F. Timann. *Theory of Approximation of Functions of a Real Variable*. Pergamon Press, Oxford, UK, 1963.

[184] B. Tolwinski. Closed-loop Stackelberg solution to a multistage linear-quadratic game. *Journal of Optimization Theory and Applications*, 34(4):485–501, 1981.

[185] A.W. Tucker. A least-distance approach to quadratic programming. In *Mathematics of the Decision Sciences, Part 1*, volume 11 of *Lectures in Applied Mathematics*. American Mathematical Society, Providence, RI, USA., 1968.

[186] T. Vallée and T. Başar. Incentive Stackelberg solutions and the genetic algorithm. *Paper presented at the International Symposium of Dynamic Games and Applications (ISDGA)*, 1998.

[187] T. Vallée and T. Başar. Off-line computation of Stackelberg solutions with the genetic algorithm. *Computational Economics*, 13(1):201–209, 1999.

[188] S. van Hoesel. An overview of Stackelberg pricing in networks. *European Journal of Operational Research*, 189(3):1393–1402, 2008.

[189] J. van Lint. *Reliable Travel Time Prediction for Freeways*. Phd dissertation, TRAIL thesis series T2004/3, Delft University of Technology, 2004.

[190] V.N. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, NY, USA, 1998.

[191] E.T. Verhoef. Second-best congestion pricing in general networks. Heuristic algorithms for finding second-best optimal toll levels and toll points. *Transportation Research Part B: Methodological*, 36(8):707–729, 2002.

[192] L.N. Vicente and P.H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5(3):291–306, 1994.

[193] W.S. Vickrey. Congestion theory and transport investment. *The American Economic Review*, 59(2):251–260, 1969.

[194] J. von Neumann and O. Morgenstern. *Games and Economic Behavior*. Princeton University Press, Princeton, NJ, USA, 3rd edition, 1953.

[195] H. von Stackelberg. *Marktform und Gleichgewicht*. Julius Springer, Vienna, Austria, 1934.

[196] H. von Stackelberg. *The Theory of Market Economy*. Oxford University Press, Oxford, UK, 1952.

[197] M.D. Vose. *The Simple Genetic Algorithm: Foundations and Theory*. Complex Adaptive Systems. MIT Press, Cambridge, MA, USA, 1999.

[198] J.G. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers, Part II*, 1:325–362, 1952.

[199] H.P. Williams. *Model Building in Mathematical Programming*. Wiley, New York, NY, USA, 3rd edition, 1993.

[200] F. Xiao, H. Yang, and D. Han. Competition and efficiency of private toll roads. *Transportation Research Part B: Methodological*, 41(3):292–308, 2007.

[201] M.B. Yildirim and D.W. Hearn. A first best toll pricing framework for variable demand traffic assignment problems. *Transportation Research Part B: Methodological*, 39(8): 659–678, 2005.

[202] I. Yperman. *The Link Transmission Model for Dynamic Network Loading*. PhD dissertation, Katholieke Universiteit Leuven, 2007.

[203] G. Zaccour. On the coordination of dynamic marketing channels and two-part tariffs. *Automatica*, 44(5):1233–1239, 2008.

[204] S.K. Zegeye, B. De Schutter, H. Hellendoorn, and E. Breunesse. Model-based traffic control for balanced reduction of fuel consumption, emissions, and travel time. In *Proceedings of the 12th IFAC Symposium on Transportation Systems*, pages 149–154, Redondo Beach, CA, USA, September 2009.

[205] S.K. Zegeye, B. De Schutter, and J. Hellendoorn. Model predictive traffic control to reduce vehicular emissions – An LPV-based approach. In *Proceedings of the 2010 American Control Conference*, pages 2284–2289, Baltimore, MD, USA, June–July 2010.

[206] L. Zhang and D.M. Levinson. Road pricing with autonomous links. *Journal of the Transportation Research Board, No. 1932*, pages 147–155, 2005.

[207] S. Zhang, H.M. Zhang, H.-J. Huang, L. Sun, and T.-Q. Tang. Competitive, cooperative and stackelberg congestion pricing for multiple regions in transportation networks. *Transportmetrica*, 7(4):297–320, 2011.

[208] S.-Y. Zhang. A nonlinear incentive strategy for multi-stage Stackelberg games with partial information. In *Proceedings of the 25th IEEE Conference on Decision and Control*, pages 1352–1357, Athens, Greece, December 1986.

[209] Y.-P. Zheng and T. Başar. Existence and derivations of optimal affine incentive schemes for Stackelberg games with partial information: A geometric approach. *International Journal of Control*, 35(6):997–1011, 1982.

[210] Y.P. Zheng, T. Başar, and J.B. Cruz, Jr. Stackelberg strategies and incentives in multiperson deterministic decision problems. *IEEE Transactions on Systems, Man, and Cybernetics*, 14(1):10–24, 1984.

[211] A. Ziegler. *A Game Theory Analysis of Options: Corporate Finance and Financial Intermediation in Continuous Time*. Springer Finance. Springer-Verlag, Berlin Heidelberg, Germany, 2nd edition, 2004.

Symbols and Abbreviations

List of Symbols

Below follows a list of the most important and frequently used symbols that are also introduced formally in the thesis.

Chapter 2–5

$\jmath_{\mathrm{L}}, \jmath_{\mathrm{F}}$	objective functions of leader resp. follower player
$\Omega_{\mathrm{L}}, \Omega_{\mathrm{F}}$	decision spaces of leader resp. follower player
$(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$	leader's desired equilibrium point, $u_{\mathrm{L}}^{\mathrm{d}} \in \Omega_{\mathrm{L}}, u_{\mathrm{F}}^{\mathrm{d}} \in \Omega_{\mathrm{F}}$
Λ_{d}	sublevel set for $\jmath_{\mathrm{F}}(\cdot)$ at $\jmath_{\mathrm{F}}(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$
$\mathcal{L}_c(f)$	contour of functional $f(\cdot)$ at the value $c \in \mathbb{R}$
$n_{\mathrm{L}}, n_{\mathrm{F}}$	number of leader resp. follower decision elements
γ_{L}	leader function, by default $\gamma_{\mathrm{L}} : \Omega_{\mathrm{F}} \to \Omega_{\mathrm{L}}$
Γ_{L}	set of admissible leader functions
Γ_{L}^{*}	set of optimal affine leader functions for $\Omega_{\mathrm{L}}, \Omega_{\mathrm{F}}$ unconstrained
Γ_{L}^{*}	set of optimal affine leader functions for $\Omega_{\mathrm{L}}, \Omega_{\mathrm{F}}$ unconstrained
$\Gamma_{\mathrm{L}}^{*,\mathrm{con}}$	set of optimal affine leader functions for $\Omega_{\mathrm{L}}, \Omega_{\mathrm{F}}$ constrained
$\Pi_X(x)$	supporting hyperplane to X at $x \in X$
Π_x^{t}	tangent n_{F}-dimensional subspace to $x \in X$
$\mathcal{A}_{\mathrm{L}}$	set of affine relations (sets) through $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$
$\mathcal{A}_{\mathrm{L}}^{X}$	set of affine relations (sets) through $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ that are subsets of X
α_{L}	inverse leader function
α_{L}^{X}	inverse leader function that is a subset of X
$R_{\mathrm{L}}, R_{\mathrm{F}}$	matrices that characterize an affine leader function
$\mathcal{R}_{\mathrm{L}}$	set of realizations R_{L} that yield an optimal affine leader function
$\mathcal{R}_{\mathrm{SG}}, \mathcal{R}_{\mathrm{RSG}}$	follower reaction set for the (reverse) Stackelberg game
$\mathcal{V}(X,x)$	generalized normal to X at $x \in X$

Chapter 4

$\alpha \in \mathbb{R}_0^+$	small constant required in distance expression $d(X,Y)$
$\delta \in \mathbb{R}^+$	gridding precision parameter
$\mathcal{C}_{\mathrm{GA}}$	selection set of population values for genetic algorithm approach

$\mathcal{C}_{\mathrm{NN}}$	selection set of coefficient values for neural network approach
$\mathcal{C}$	cone with apex $(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})$ and the largest aperture for which $\mathcal{C} \cap \Lambda_{\mathrm{d}} = \{(u_{\mathrm{L}}^{\mathrm{d}}, u_{\mathrm{F}}^{\mathrm{d}})\}$
$d_1(x,X)$	signed shortest Euclidean distance between x and $\mathrm{bd}(X)$
$d_2(x,y)$	Euclidean distance between x and y
$d(X,Y)$	auxiliary expression of distance between the set X and Y
$\omega(x,X)$	signed shortest Euclidean distance measure between $x \in X$ and $\mathrm{bd}(X)$
$\omega(X,Y)$	auxiliary expression of distance between the set X and Y

Chapter 5

$\mathcal{F}_{\mathrm{s}}, \mathcal{F}_{\mathrm{t}}$	set of followers in the splitting rate resp. travel time-based game
$N_{\mathrm{F,s}}, N_{\mathrm{F,t}}$	number of followers in the splitting rate resp. travel time-based game
N_{OD}	number of OD-pairs
n_i	number of routes for OD-pair index $i \in \{1, \ldots, N_{\mathrm{OD}}\}$
$\tau^{hij}(k)$	desired travel time [h] for player $(h,i,j) \in \mathcal{F}_{\mathrm{t}}$
$\zeta^{hi}(k)$	route splitting rate vector for player $(h,i) \in \mathcal{F}_{\mathrm{s}}$
$t_m(k)$	predicted average travel time [h] on link m at time step k
$q_{m,d}(k)$	flow [veh/h] on link m traveling towards destination d at time step k
$q_{\mathrm{in}}^{o,d}(k)$	effective demand [veh/h] for origin o destination d at time step k
$q_{\mathrm{in}}^{\mathrm{ctrl},o,d}(k)$	vector of effective demand [veh/h] distributed by the road authority over possible routes at time step k
$q_{\mathrm{in}}^{o,d,\mathrm{d}}(k)$	desired, system-optimal vector of effective demand [veh/h] distributed over possible routes at time step k
$\tau_{\mathrm{u}}(k)$	desired total travel time for an urban corridor network
$\tau_{\mathrm{nom},\ell}(k)$	nominal travel time for mainstream link ℓ at time step k
$\tau_{\mathrm{wait},\ell}(k)$	waiting time for mainstream traffic signal ℓ at time step k
$v_{\mathrm{ctrl},\ell}(k)$	speed control variable [km/h] for link ℓ at time step k
$\mathcal{U}$	set of links in an urban corridor

Chapter 5, APPENDIX A,B

N_{c}	control horizon
N_{p}	prediction horizon
$T_{\mathrm{s}} \in \mathbb{R}^+$	simulation time interval
$T_{\mathrm{c}} \in \mathbb{R}^+$	control time interval
$k_{\mathrm{c}} \in \mathbb{N}^+$	control time step counter
$k \in \mathbb{N}^+$	simulation time step counter
$M \in \mathbb{N}_0^+$	factor relating the simulation and control time interval, $T_{\mathrm{c}} = MT_{\mathrm{s}}$
$\alpha \in \mathbb{R}^+$	non-compliance factor regarding the speed limit
$a_m \in \mathbb{R}^+$	METANET model parameter
η	METANET model parameter [km^2/h]
κ	METANET model parameter [veh/km/lane]

τ	METANET model parameter [h]
$\mathcal{O}$	set of origin nodes
$\mathcal{D}$	set of destination nodes
$\mathcal{N}$	set of internal nodes
$\mathcal{L}$	set of links
I_n	set of links entering n
O_n	the set of outgoing links for node n
N_m	index of the last segment of a link m that enters node n
λ_m	number of lanes in link m
L_m	length of the segments of link m [m]
$\mathcal{R}$	set of available routes
$\mathcal{R}_j$	set of available routes for OD-pair with index $j \in \{1,\ldots,N_{\text{OD}}\}$
P_γ	model parameter for $\gamma \in \Gamma = \{$CO emission, HC emission, NO_x emission, fuel consumption$\}$
I_{all}	set of index pairs (m,i) of all links and segments in the network
$q_{m,i}(k)$	flow [veh/h] on segment i of link m at time step k
$\rho_{m,i}(k)$	density [veh/km/lane] for segment i of link m at time step k
$\rho_{\text{crit},m}(k)$	critical density [veh/km/lane] of a link m at time step k
$v_{m,i}(k)$	space-mean speed [km/h] for segment i of link m at time step k
$v_{\text{ctrl},m,i}(k)$	speed control variable [km/h] for segment i of link m at time step k
$v_{\text{free},m}$	free-flow speed [km/h] for link m
$V(\rho_{m,i}(k))$	desired speed [km/h] for segment i of link m at time step k
$w_o(k)$	queue length [veh] at origin o at time step k
$d_o(k)$	traffic demand [veh/h] of origin o at time step k
$q_o(k)$	outflow [veh/h] of origin o at time step k
$r_o(k)$	ramp-metering rate at origin o at time step k
C_o	capacity [veh/h] of origin o
$w_{o,r}(k)$	queue length at origin o w.r.t. route r [veh] at time step k
$d_{o,r}(k)$	traffic demand [veh/h] of origin o at time step k
$q_{o,r}(k)$	outflow [veh/h] of origin o at time step k
$q_{m,i,r}(k)$	flow [veh/h] on segment i of link m at time step k
$\rho_{\max,m}$	maximum density [veh/km/lane] of a link m
$Q_n(k)$	total flow [veh/h] entering a node n at time step k
$q_{m,0}(k)$	flow [veh/h] leaving node n via link m at time step k
$\beta_{n,m}(k)$	fraction of the total flow to node n leaving via link m at time step k
$\rho_{m,N_m+1}(k)$	virtual downstream density [veh/km/lane] of link m entering a node with multiple outgoing links at time step k
$v_{m,0}(k)$	virtual upstream speed [km/h] of outgoing link m for a node with multiple ingoing links at time step k
$a_{m,i,i+1}(k)$	spatio-temporal acceleration $[m/s^2]$ from segment i to segment $i+1$ of link m at time step k
$n_{m,i,i+1}(k)$	number of vehicles subject to spatio-temporal acceleration at time step k
$a_{m,i}(k)$	temporal acceleration $[m/s^2]$ within the same segment i of a link m at time step k
$n_{m,i}(k)$	number of vehicles subject to temporal acceleration within the same segment i of a link m at time step k

$J_{\text{TTS}}^{\text{MPC}}(k)$	TTS [veh·h] obtained according to the MPC framework at time step k
$J_{\gamma,\text{TEFC}}^{\text{MPC}}(k)$	total emissions and fuel consumption obtained according to the MPC framework at time step k
$J_{\text{pen}}^{\text{MPC}}(k)$	total penalty on control action deviations at time step k
TTS_{norm}	normalization term for TTS
$\text{TEFC}_{\gamma,\text{norm}}$	normalization term for emissions and fuel consumption
pen_{norm}	normalization term for penalty factor

Finally, the following general mathematical symbols have been adopted:

$\mathbb{N}$	set of nonnegative natural numbers $\mathbb{N} = \{0,1,2,\dots\}$
$\mathbb{N}_0$	set of strictly positive natural numbers, excluding $\{0\}$
$\mathbb{R}$	set of real numbers
$\mathbb{R}_0^+$	set of strictly positive real numbers, excluding $\{0\}$
$\mathbb{R}^+$	set of nonnegative real numbers
$\text{null}(X)$	null space of matrix X
$\mathcal{B}(X)$	basis of vector space X
∇	gradient
$\|x\|_2$	Euclidean norm, ℓ^2-norm
$\times_{i=1}^n X_i$	$\times_{i=1}^n X_i = X_1 \times \dots \times X_n$

List of abbreviations

The following abbreviations are used in this thesis:

LQ	linear-quadratic
MLD	mixed logical dynamic
MPC	model predictive control
PWA	piecewise-affine
SVD	singular value decomposition
TEFC	total emissions [g] and fuel consumption [l]
TTS	total time spent [veh·h]

Summary

Reverse Stackelberg Games: Theory and Applications in Traffic Control

In general, the focus of the research presented in this dissertation can be described as the application of game-theoretical elements in the modeling of communication and interaction between agents or controllers on different layers of a large-scale multilevel optimization problem. One of the major challenges in optimization-based control of large-scale infrastructural networks such as traffic networks is to find efficient multilevel optimization schemes through which decisions can be made by controllers of different interacting layers. The hierarchical game on which this thesis is focused can be used to model this interaction.

In particular, research has been conducted on the so-called *reverse Stackelberg game* that can be described as a hierarchical game in which players make decisions sequentially. Moreover, a leader player in this game proposes a so-called *leader function* to the followers, which maps a follower's decision space to the leader's decision space. The leader therefore influences a follower by making her decision dependent on a follower's decision, under the aim to obtain her desired – e.g., globally optimal – pair of leader and follower decision variables. An example of such a relation is to propose different monetary incentives to be associated with different routes in a traffic network, under the objective to induce the drivers to adopt those routes that lead to a system-optimal traffic distribution.

The results of this dissertation can be classified with respect to two main branches: a more fundamental branch aimed at developing existence conditions and solution methods for the general reverse Stackelberg game, and a more applied branch aimed towards application of reverse Stackelberg games in order to mitigate traffic problems like congestion and vehicular emissions.

The main contributions can be summarized as follows:

Existence conditions and characterization of optimal affine solutions

As a first step towards obtaining a systematic solution methods of the reverse Stackelberg game in various settings, which are not yet available in the current literature, necessary and sufficient existence conditions are considered for optimal leader functions with an affine structure. Here, we adopt an *indirect* formulation of the reverse

Stackelberg game, in which the leader's desired equilibrium point is determined *a priori*.

Moreover, we characterize the full set of such optimal affine solutions, which is useful in case:

(i) the decision spaces are constrained, in which situation it is difficult to derive existence conditions, and in case:

(ii) secondary optimization criteria are employed, such as considering leader functions that are the least sensitive to deviations from the follower's rational, hence optimal, response.

Finally, it is shown how, based on the initial characterization, one can select the subset of optimal affine solutions in case of constrained decision spaces.

Systematic computation of optimal nonlinear solutions

In case no optimal affine solution exists or in case nonlinear leader functions have desirable characteristics such as a certain degree of smoothness, nonlinear leader functions should be derived. The following methods are proposed to systematically compute an optimal nonlinear solution to the indirect reverse Stackelberg game:

- A *continuous multilevel* optimization problem is considered that leads to an optimal leader function written as a linear combination of a given set of basis functions. In some special cases, the complexity of the problem, which is in general NP-hard, is reduced to that of convex programming problems that can be efficiently solved in polynomial time.

- In case the follower's decision space is *discretized in grid points*, the above multilevel optimization problem reduces to a single-level optimization problem. Since optimality of the leader function cannot be guaranteed, an adaptive gridding approach is proposed.

- *Interpolating spline methods* available in the literature can be adopted in the special case the decision spaces of leader and follower together are of at most dimension three.

Additionally, results are provided on a case study conducted to compare the computational requirements for computing an optimal leader function when adopting one of the three indirect methods just mentioned to the direct methods based on evolutionary algorithms available in the literature on reverse Stackelberg games. In the latter methods, leader function coefficients are derived based on either the genetic algorithm or on a neural network approach, in which the leader's performance is optimized and no *a priori* determined equilibrium point is adopted. For the case study at hand, the suggested approaches yield the leader's globally optimal pair of decision variables and moreover consume much less computation time, while the evolutionary approaches for the direct game formulation returned significantly less desirable leader objective function values.

Application of reverse Stackelberg games in traffic control

Next to the development of systematic solution methods for the reverse Stackelberg game, applications are considered that show how this game can be adopted to improve the performance in a traffic system, e.g., by minimizing the total time spent of vehicles in the network. In particular, the game is reformulated in the context of route choice, where the road authority can obtain the system-optimal distribution of traffic in a dynamic setting through the use of leader functions that allocate a positive or negative monetary incentive to each route choice or desired travel time of each homogeneous group of drivers. Upon the choice of the pair of either desired route choice or travel time and the associated monetary incentive that is entered by the follower in an on-board computer, the driver is allocated to a route desired by the road authority. In particular, the following game variants are considered:

- *Route choice in freeway networks*:

  The followers' desired *route splitting rates* are mapped to monetary incentives in order to reach a system-optimal traffic distribution.

- *Route choice in freeway networks*:

  The followers' desired *expected travel times* are mapped to monetary incentives in order to reach a system-optimal traffic distribution.

- *Single-corridor urban networks*:

  The followers' desired expected travel times are mapped to monetary incentives, now under the aim to reduce vehicular emissions while taking into account the urgency and desired travel time of the mainstream drivers. Traffic signals for mainstream traffic as well as for side-street traffic crossing and joining the mainstream are adopted as the road authority's control measure.

As regards fairness issues of allocating monetary incentives to different groups of drivers, it should be noted that the absolute value of the monetary incentives associated with the desired follower decisions is flexible. That is to say, in order for a leader function to solve the reverse Stackelberg game to optimality, the geometry of the leader functions in relation to the followers' sublevel sets is important, rather than the absolute values that compose the image of such a function.

Appendix: Solving a nonconvex model-based predictive traffic control problem through piecewise affine approximation

Finally, a separate topic that is related to traffic control, however not to game theory, has been considered in the beginning of the research period. The approximation of a nonlinear macroscopic traffic and emission model has been studied for the purpose of reducing the computation time of the nonconvex model-based predictive traffic control problem aimed to minimize the total time spent of traffic in the system as well as vehicular emissions and fuel consumption. In particular, the nonlinear model equations are approximated by piecewise affine functions, after which the resulting mixed-logical dynamic problem can be solved by mixed-integer linear programming.

Noortje Groot

Samenvatting

Omgekeerde Stackelbergspellen: Theorie en Toepassingen in Verkeersregeling

Het promotieonderzoek zoals samengevat in dit proefschrift is in brede zin gericht op het toepassen van speltheoretische elementen in het modelleren van communicatie en interactie tussen agenten of regelaars van verschillende lagen van een grootschalig, meerlaags optimalisatieprobleem. Een van de wellicht belangrijkste uitdagingen in de regeling van grootschalige infrastructuren zoals verkeersnetwerken, is het vinden van efficiënte meerlaagse optimalisatieschema's waarin beslissingen gemaakt kunnen worden door regelaars van verschillende interagerende lagen. Het hiërarchische spel waarop deze thesis is gericht, kan worden gebruikt om deze interactie te modelleren.

In het bijzonder wordt het zogenaamde '*omgekeerde Stackelbergspel*' (*reverse Stackelberg game*) onderzocht, wat samengevat kan worden als een hiërarchisch spel waarin spelers op sequentiële wijze beslissingen nemen. Daarbij biedt de leider in dit spel een wiskundige relatie aan de volger aan, de zogenaamde leiderfunctie, die aan het besluit van de volger een waarde van de beslissingsvariabele van de leider toewijst. Derhalve beïnvloedt de leider de volger door haar besluit afhankelijk te maken van dat van de volger, waarbij de leider als doel heeft haar gewenste, bijvoorbeeld haar globaal optimale, koppel van leider en volger beslissingsvariabelen te behalen. Een voorbeeld van een dergelijke relatie is het aanbieden van verschillende monetaire aanmoedigingspremies die geassocieerd zijn met verschillende verkeersroutes, met als doel de verkeersdeelnemers te beïnvloeden om die routes te nemen die leiden tot een optimale verkeersdistributie.

De resultaten van dit onderzoek kunnen worden opgedeeld in twee hoofdtakken: een meer fundamentele tak gericht op het ontwikkelen van bestaansvoorwaarden en oplossingsmethoden voor het algemene omgekeerde Stackelbergspel en een meer toepassingsgerichte tak die gericht is op het actief beïnvloeden van de routekeuze van weggebruikers in verkeersnetwerken om problemen zoals verkeersverstoppingen en de uitstoot van uitlaatgassen te verminderen.

De hoofdbijdragen van deze deelonderwerpen worden in de volgende paragrafen toegelicht.

Bestaansvoorwaarden en karakterisering van optimale affiene oplossingen

Binnen de fundamentele tak is het belangrijk toe te werken naar een systematische oplossingsmethode van het omgekeerde Stackelbergspel in diverse situaties, welke in de huidige literatuur nog niet beschikbaar is. Daartoe zijn eerst bestaansvoorwaarden onderzocht van optimale leiderfuncties met een affiene structuur. Hierbij richten we ons op een *indirecte* formulering van het omgekeerde Stackelbergspel, waarin het gewenste evenwichtspunt van de leider, bestaande uit een koppel van beslissingsvariabelen van de leider en de volger, wordt bepaald voorafgaand aan het opstellen van een leiderfunctie.

Daarbij presenteren we een volledige karakterisering van zulke potentiële optimale affiene oplossingen, welke in het bijzonder nuttig is in het geval:

(i) de beslissingsruimten beperkt zijn door randvoorwaarden, in welke situatie het moeilijk is bestaansvoorwaarden af te leiden, en in het geval:

(ii) secundaire optimalisatiecriteria worden gehanteerd, zoals het beschouwen van leiderfuncties die het minst gevoelig zijn voor afwijkingen van de rationele, dus optimale beslissingsreacties van de volgers.

Tenslotte laten we zien hoe men, gebaseerd op de initiële karakterisering van optimale affiene leiderfuncties, een deelverzameling van optimale affiene oplossingen kan selecteren in het geval van begrenzingen op de beslissingsruimte.

Systematische berekening van optimale niet-lineaire oplossingen

In het geval dat een optimale oplossing van de affiene vorm niet bestaat of in het geval niet-lineaire leiderfuncties gewenste eigenschappen hebben zoals een bepaalde graad van gladheid (*smoothness*), zal het nodig zijn niet-lineaire oplossingen af te leiden. We stellen de volgende methoden voor om systematisch een niet-lineaire optimale oplossing te berekenen voor het indirecte omgekeerde Stackelbergspel:

- We beschouwen een meerlaags continu optimalisatieprobleem dat leidt tot een optimale leiderfunctie welke uitgedrukt wordt als een lineaire combinatie van een gegeven verzameling van basisfuncties. In enkele speciale gevallen is de complexiteit van dit probleem, welke in het algemeen *NP-hard* is, gereduceerd tot dat van een convex optimalisatieprobleem dat efficiënt, in een rekentijd van boven begrensd door een polynoom, opgelost kan worden.

- We beschouwen dit optimalisatieprobleem tevens in het geval de beslissingsruimte van de volger is gediscretiseerd in roosterpunten. Terwijl dit resulteert in een éénlaags optimalisatieprobleem, kan de optimaliteit van een leiderfunctie niet gegarandeerd worden, hetgeen leidt tot een aanpak met een adaptief rooster.

- Spline interpolatoren zoals bekend uit de literatuur kunnen ook worden gebruikt, in het bijzonder voor het speciale geval dat de beslissingsruimten van leider en volger samen op zijn hoogst drie dimensies behelst.

Daarnaast worden de resultaten gepresenteerd van een casus waarin de rekentijd wordt vergeleken wanneer enerzijds de drie zojuist genoemde methoden worden toegepast en anderzijds de *directe* methoden uit de literatuur, gebaseerd op twee evolutionaire algoritmen voor omgekeerde Stackelbergspelen. Bij deze beschouwde directe methoden worden leiderfunctiecoefficiënten afgeleid, gebaseerd op ofwel het genetisch algoritme danwel op een neuraal netwerk methode, waarin de prestatie van de leider geoptimaliseerd wordt en waarin geen vooraf bepaald evenwichtspunt wordt gebruikt. Voor deze casus resulteren de door ons voorgestelde methoden in het voor de leider gewenste globaal optimale koppel van beslissingsvariabelen terwijl deze significant minder rekentijd vergen. Tegelijkertijd resulteren de evolutionaire methoden voor de directe formulering van het spel in minder wenselijke doelfunctiewaarden voor de leider.

Toepassing van omgekeerde Stackelbergspellen in verkeersregeling

Naast het ontwikkelen van systematische oplossingsmethoden voor het omgekeerde Stackelbergspel zijn tevens toepassingen beschouwd die laten zien hoe het omgekeerde Stackelbergspel de verkeerssituatie kan verbeteren, bijvoorbeeld door de totale tijd die voertuigen in het netwerk doorbrengen, te minimaliseren. In het bijzonder hebben we het spel geformuleerd in de context van routekeuze, waarbij de beheerder van een verkeersnetwerk de optimale doorstroming van verkeer in een dynamische setting kan bereiken door middel van het doorgeven van leiderfuncties die een positief dan wel negatief geldbedrag associëren met de routekeuze van een homogene groep automobilisten, dan wel die een bedrag associëren met de gewenste reistijd van een automobilist. Nadat het gewenste koppel van bedrag en reistijd of routekeuze in een boordcomputer in het voertuig ingevoerd is, kan de daaruitvolgende optimale route aan de weggebruiker medegedeeld worden. In het bijzonder worden de volgende varianten van het spel beschouwd:

- *Routekeuze in snelwegnetwerken*:

  Aan de gewenste *routeverdeelbreuken* van de volgers worden monetaire waarden gekoppeld met als doel een systeem-optimale verkeersverdeling te verkrijgen.

- *Routekeuze in snelwegnetwerken*:

  Aan de gewenste *verwachte reistijden* van de volgers worden monetaire waarden gekoppeld met als doel een systeem-optimale verkeersverdeling te verkrijgen.

- *Stedelijke netwerken bestaande uit een enkele corridor*:

  Opnieuw worden aan de gewenste *verwachte reistijden* van de volgers monetaire waarden gekoppeld, echter nu met als doel de doorstroom te optimaliseren, emissie van uitlaatgassen te reduceren en tegelijkertijd rekening te houden met de urgentie en de gewenste reistijd van de wegberijders op de hoofdroute. In dit verband worden verkeerssignalen voor overstekend danwel samenvoegend verkeer en voor het verkeer in de hoofdstroom gebruikt als regelmaatregelen van de wegautoriteit.

Met het oog op de rechtvaardigheid omtrent het toewijzen van monetaire stimuli aan verschillende groepen van automobilisten moet worden beseft dat de absolute waarde van deze monetaire aansporingen zoals geassocieerd met de gewenste beslissingen van de volgers, flexibel is. In plaats van absolute waarden die het beeld (*image*) van een leiderfunctie opmaken, is juist de geometrie van dergelijke functies ten opzichte van de deelniveauverzamelingen van de volgers belangrijk voor het tot in optimaliteit oplossen van het omgekeerde Stackelbergspel.

Appendix: Oplossen van een niet-convex model-gebaseerd voorspellend regelprobleem in verkeersnetwerken door stuksgewijs affiene benadering

Tenslotte is aan het begin van het promotieonderzoek aandacht besteed aan een apart onderwerp dat gerelateerd is aan verkeersregeling maar niet aan speltheorie. De benadering van een niet-lineair macroscopisch verkeers- en emissiemodel is bestudeerd met oog op het verminderen van de rekentijd voor het oplossen van het niet-convexe modelgebaseerd voorspellend verkeersregelingsprobleem, gericht op het gebalanceerd reduceren van de totale tijd die voertuigen doorbrengen in het systeem, de uitstoot en het brandstofverbruik. In het bijzonder worden de niet-linaire modelvergelijkingen benaderd door stuksgewijs affiene functies, waarna het resulterende gemengd-logisch-dynamische probleem opgelost kan worden door een lineair programmeerprobleem met variabelen die reële danwel variabelen die gehele getalswaarden aannemen.

Noortje Groot

About the Author

Noortje Groot was born on December 12th, 1986 in Woerden, the Netherlands. She finished her pre-university education at the Christelijk Gymnasium Utrecht, the Netherlands, in 2005.

In 2008 Noortje obtained the B.Sc. degree focusing mostly on applied mathematics and business studies at the liberal arts college University College Maastricht, Maastricht University, the Netherlands, summa cum laude and first in class. In early 2007, Noortje spent several months as a visiting scholar at Babson College, Wellesley, MA, USA. Her B.Sc. thesis (supervised by dr. Gijs Schoenmakers, Department of Knowledge Engineering, Maastricht University) was on the use of cooperative game theory in the distribution of fishing rights within the European Union.

In 2009 Noortje completed the M.Sc. program in econometrics and operations research at the department of Quantitative Economics, School of Business and Economics, Maastricht University. In her M.Sc. thesis (supervised by dr. Tjark Vredeveld, Department of Quantitative Economics, dr. Tuomo Takkula, INFORM GmbH Aachen, top thesis) an alternative proof of the 'Highway Hierarchies' algorithm devised to solve the shortest path problem was proposed, by using duality theory and a reformulation of the algorithm as a mixed integer linear program.

Since December 2009, Noortje has been employed as a Ph.D. candidate at the Delft Center for Systems and Control, Delft University of Technology, under the supervision of prof.dr.ir. Bart De Schutter and prof.dr.ir. Hans Hellendoorn. In her first year, Noortje obtained the certificate of the Dutch Institute of Systems and Control (DISC) graduate school, after which she worked mostly in the field of hierarchical game theory as a basis for decision making in the context of large-scale networks, with applications in intelligent transportation systems.

Hofstadter's Law: It always takes
longer than you expect, even when you
take into account Hofstadter's Law.