

# Optimizing control in a transportation network when users may choose their OD-path

Diplomarbeit bei  
Prof. Dr. Dr. h.c. mult. Martin Grötschel

vorgelegt von Julia Buwaya  
am Fachbereich Mathematik der  
Technische Universität Berlin



KONRAD-ZUSE-ZENTRUM  
FÜR INFORMATIONSTECHNIK  
BERLIN

Berlin, 7. Juni 2013

---

Hiermit versichere ich die selbstständige und eigenhändige Anfertigung dieser Arbeit an Eides statt.

---

(Ort, Datum)

---

(Unterschrift)

# Danksagung

Besonderer Dank gilt Dr. Guillaume Sagnol, der mir während des Schreibens der vorliegenden Arbeit stets mit Rat und Tat zur Seite stand. Ich danke außerdem Elmar Swarat, dem Herzstück des TollControlOpt-Projekts am ZIB, für die Möglichkeit im Projekt mitzuwirken und für seine Unterstützung. Vor allem möchte ich mich bei Prof. Dr. Dr. h.c. mult. Martin Grötschel für die Betreuung dieser Arbeit und seine immer ehrliche und konstruktive Kritik bedanken, sowie bei Prof. Dr. Ralf Borndörfer, der diese Arbeit als Zweitkorrektor betreut. Des Weiteren danke ich Steffen Przybylowicz für die vielen heiteren Stunden am ZIB und Bertrand Omont für die gemeinsame Zeit mit viel Input, sowie Jonatan Krolikowski und Stephan Schwartz, die noch einmal über meine Arbeit geschaut haben und meiner Familie und meinen Freunden, die stets für mich da waren.



# Zusammenfassung in deutscher Sprache

In dieser Arbeit wird die Verteilung von Kontrolleuren in einem Verkehrsnetz spieltheoretisch untersucht. Es wird ein Stackelbergmodell zwischen Kontrolleuren und Nutzern des Verkehrsnetzes formuliert, welches Spot-checking game genannt wird und es werden mathematische Programme und Methoden präsentiert, um relevante Gleichgewichte zu berechnen. Ziel ist es Strategien zu berechnen, die einen guten Kompromiss zwischen Rechenzeit und Einnahmen aus den Nutzungsgebühren und den Strafen für gefasste Schwarzfahrer darstellen. Obwohl das Spot-checking game im Allgemeinen kein Nullsummenspiel ist, lässt sich ein lineares Programm (LP) finden, mit dem Nash-Gleichgewichte berechnet werden können. Die Anzahl der Nutzer-Strategien, die in einer ersten Formulierung der Anzahl der Möglichkeiten von Routen durch das Netz entspricht, ist in der Regel sehr groß. Ein Schnittebenenverfahren zur Lösung des aufgestellten sehr großen LP und eine alternative kompakte Repräsentation des LP mit Hilfe von Netzwerkflüssen wird vorgestellt. Das Finden der optimalen Strategie in einem Spot-checking game ist wesentlich schwieriger. Hier wird für die Berechnung ein gemischt-ganzzahliges Programm (MIP) verwendet.

Es werden einige theoretische Ergebnisse zu Abweichungen zwischen Strategien eines Nash-Gleichgewichts und eines Stackelberg-Gleichgewichts präsentiert und das Modell wird auf vereinfachte Instanzen des deutschen LKW-Maut-pflichtigen Autobahnnetzes, sowie einige seiner Teilnetze unter Verwendung realer Daten angewandt.

Wenn bestimmte Eigenschaften erfüllt sind, kann gezeigt werden, dass die optimale Strategie für die Kontrolleure auch eine Strategie eines Nash-Gleichgewichts ist. Es gibt jedoch einfache Beispiele für die dies nicht der Fall ist. Interessanterweise ergeben realistische Werte für die gefundenen einfachen Gegenbeispiele jedoch eine Abweichung in Bezug auf die Einnahmen für die Kontrolleure, die relativ klein ist. Diese Beobachtung entspricht den experimentellen Ergebnissen für die Anwendung auf deutsche Autobahnen. Hier werden in dem spieltheoretischen Modell die optimale Verteilung von Kontrolleuren im Verkehrsnetz mit der Verteilung entsprechend eines Nash-Gleichgewichts und der Verteilung von Kontrolleuren proportional zur Verkehrsdichte miteinander verglichen. Insbesondere lassen experimentelle Untersuchungen vermuten, dass die Strategie eines Nash-Gleichgewichts für die Kontrolleure häufig ein guter Kompromiss zwischen Berechnungszeit und Effizienz der Kontrollen ist.

---

Außerdem wird eine Anwendung, in der zeitliche und örtliche Beschränkungen beachtet werden, präsentiert und ein probabilistischer Dienstplan extrahiert. Ergebnisse dieser Arbeit können möglicherweise in einem aktuellen ganzzahligen Programm zur Berechnung eines integrierten Dienst- und Kontrollplanes für reale Anwendungen verwendet werden.

# Preface

This thesis represents a game-theoretic investigation of the allocation of inspectors in a transportation network, comparing Nash and Stackelberg equilibrium strategies to a strategy in which inspections are conducted proportional to the traffic volume. Parts of the model were presented at the International Network Optimization Conference (INOC) 2013. We furthermore present some theoretical results concerning the deviations of Nash and Stackelberg equilibrium strategies and an application integrating space and time dependencies. This work contains extensive experimental tests for the application in the transportation network of German motorways relying on real data.

In general we will assume a worst-case situation in which users of the network and the inspectors play a Stackelberg game, i.e. the users can observe the strategy chosen by the inspectors before choosing their best-response strategy in order to minimize their costs. We formulate a game between inspectors and users, which we call a *spot-checking game* and find that this game is *best-response equivalent* to a zero-sum game and hence, a Nash equilibrium can be computed by solving a linear program (LP). As the number of strategies for the users is potentially huge, a cutting-plane method to solve the large-scale LP and an alternative, efficient user-flow formulation is presented. Finding the optimal strategy to commit to for the inspectors in the Stackelberg game (i.e. finding a Stackelberg equilibrium strategy) is much harder. Here we use a mixed-integer program (MIP) for the computation. If certain properties are satisfied, we can prove that any inspectors' Stackelberg equilibrium strategy is also a Nash equilibrium strategy. We will then present simple examples in which Nash equilibrium strategies may exist, and which are not a Stackelberg equilibrium strategy, or in which a Stackelberg equilibrium strategy might not be a Nash equilibrium strategy at all. Interestingly, on the other hand, realistic values for the input parameters of these examples yield a gap between the inspectors' payoff for playing with a Nash equilibrium strategy and the inspectors' payoff in a Stackelberg equilibrium that is relatively small. This observation corresponds to the experimental results for the application on German motorways, that will be presented afterward. We will see that the Nash equilibrium strategy computed is often close to the Stackelberg equilibrium strategy in terms of inspectors' payoff, i.e. experimental results suggest that instead of solving an MIP to compute a Stackelberg equilibrium, one may often use an LP computing a Nash equilibrium strategy in order to have a good approximation and a much faster computation time.





# Contents

<b>1</b>	<b>Motivation</b>	<b>1</b>
1.1	Optimizing toll enforcement on German motorways . . . . .	1
1.2	Advantageous routes and strategic users . . . . .	2
1.3	Objective and related works . . . . .	3
<b>2</b>	<b>A game-theoretic approach</b>	<b>5</b>
2.1	A spot-checking game . . . . .	5
2.2	Game settings and equilibria . . . . .	7
2.2.1	Inspectors' Nash vs. Stackelberg equilibrium strategies . . . .	10
2.3	Computation of an inspectors' Nash equilibrium strategy . . . . .	17
2.3.1	Best-response equivalent zero-sum game . . . . .	18
2.3.2	User-paths Nash LP and cutting-plane method . . . . .	21
2.3.3	User-flow Nash LP . . . . .	23
2.3.4	Computation of the inspectors' payoff in a Stackelberg game .	24
2.4	Computation of an inspectors' Stackelberg equilibrium strategy . . .	25
<b>3</b>	<b>Application on German motorways</b>	<b>29</b>
3.1	Two-level transportation network . . . . .	30
3.1.1	Experimental results I . . . . .	35
3.2	Single-pay-path and variations of the inspectors' objective . . . . .	42
3.2.1	Experimental results II . . . . .	45
3.3	Inspections regarding space and time . . . . .	48
3.3.1	Introducing a cyclic duty digraph . . . . .	48
3.3.2	Extracting a probabilistic duty roster . . . . .	51
3.3.3	Experimental results III . . . . .	52
3.3.4	Extensions . . . . .	57
<b>4</b>	<b>Conclusion</b>	<b>61</b>
<b>A</b>	<b>Spot-checking games as superclass of security games</b>	<b>63</b>
<b>B</b>	<b>German motorways control system and the probability to be in-</b>	
	<b>spected when an inspector is present</b>	<b>67</b>
	<b>Bibliography</b>	<b>71</b>



# Chapter 1

## Motivation

### 1.1 Optimizing toll enforcement on German motorways

The revenues from the heavy goods vehicle toll (HGV toll or German: Lkw-Maut) for commercial trucks of twelve tonnes or more on German federal motorways serve to maintain and upgrade the said transportation system, grossing about 4.5 billion euros annually [Fed13]. The distance-based toll, introduced in 2005 and extended to parts of major federal highways (Bundesstrassen) in 2012, may be paid using an automated GPS/GSM system (through a so-called On-Board-Unit, or OBU) or by manually purchasing a ticket. Levying, administration, control and surveillance of the HGV-toll is a sovereign task of the German Federal Office for Goods Transportation (BAG). To this end, road inspections are conducted. The alternative of building entry-barriers would be rather costly due to the large number of entry points. Instead a combination of 300 stationary control gantries (which is not enough for a network-wide control), random local inspections at turn-off or exit points, and random tours of about 300 mobile inspection units is deployed.

In a current project with the BAG, the Konrad-Zuse-Institute Berlin (ZIB) is optimizing the distribution of the mobile inspection units on subnetworks of the German toll network. The aim is to optimize toll enforcement and to assure a network-wide control whose intensity is proportional to given spatial and time-dependent traffic distributions. While doing so, legal regulations for the staff have to be fulfilled. I.e. a feasible duty roster and an optimal tour plan for the mobile inspection units has to be produced. A mathematical approach, called the *Toll Enforcement Problem (TEP)* [BSS11], was developed to depict this problem and was formulated as a large-scale integer program (IP). This IP could be solved to optimality for real-world instances associated with a control area of a German subnetwork [BSS12].

While the TEP is dealing with the complexity of creating a feasible integrated control and duty roster, this thesis corresponds to a theoretical investigation of the allocation of inspectors in a transportation network.

## 1.2 Advantageous routes and strategic users

It is quite natural to say that inspections imply that potential toll evaders exist. If the aim of inspections is to enforce the payment of a toll, the practicality of the methodology to distribute inspection intensities proportional to recorded traffic volumes needs to be evaluated. A recent report on behalf of the German government states that on a distance equivalent to one fourth of the distance covered by all federal motorways, there was an average increase from 2005 to 2007 of 50 or more trucks per workday through toll avoidance[Bun09]. One reason why the HGV toll was extended to certain federal highways. The following figure 1.1 shows results of a route simulation from the latter report. The blue lines represent toll liable motorways, the gray lines represent toll free trunk roads with an average increase from 2005 to 2007 of less than 50 trucks per workday, while the yellow, orange and red lines show trunk roads with an average increase of 50 or more trucks.

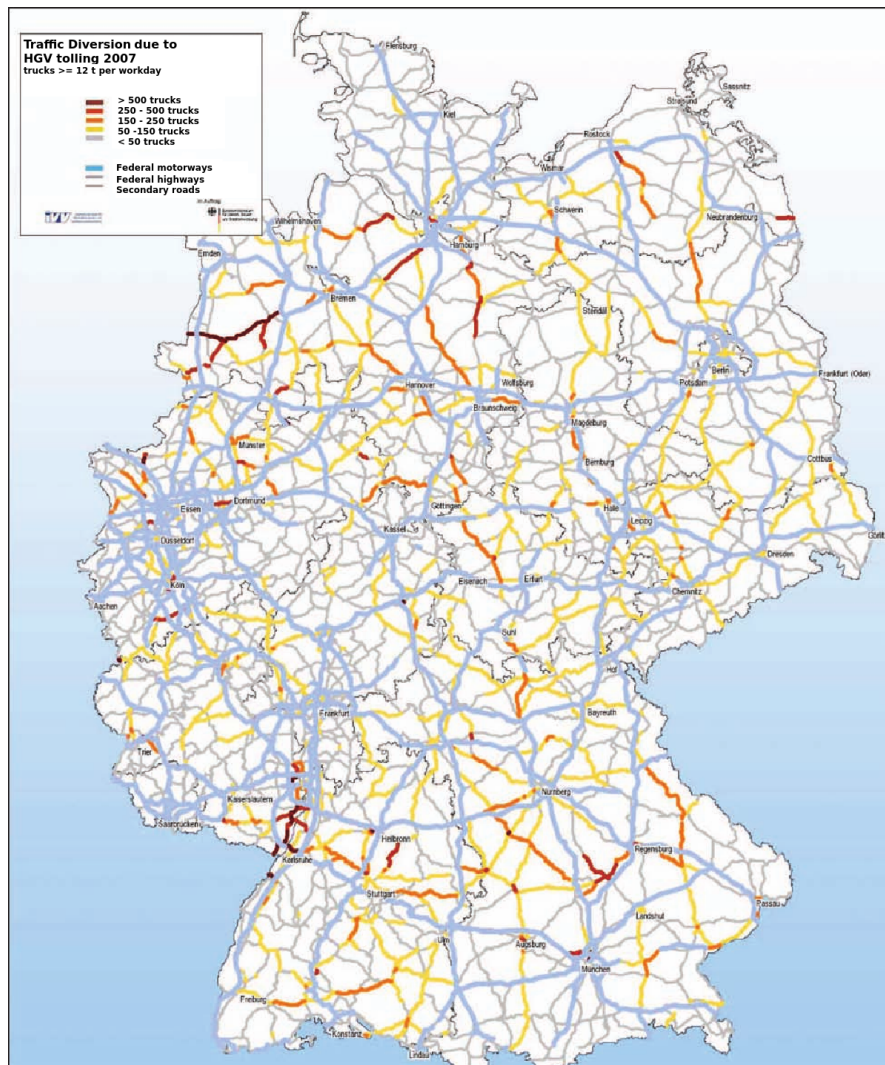


Figure 1.1: Increase of traffic diversion to avoid paying HGV toll in 2007

In this thesis, we will focus our attention on users who strategically choose their Origin-Destination path (short: OD-path) through the network in order to minimize their expenses and are willing to evade the toll if this leads to a cheaper trip.

## 1.3 Objective and related works

We will use a game-theoretic approach to investigate the allocation of inspectors in a transportation network. This thesis extends basic ideas of a problem formulation by Borndörfer et al. from 2012 [BOSS12]. In the said article, the strategic interaction between inspectors and users of a transportation network is studied when a user's route is fixed. A variety of related problems in which the objective is to randomize patrols in a strategical way, has been studied using game theory. In [JKK<sup>+</sup>10] US air marshals are allocated to a list of flights. In [PJM<sup>+</sup>08] a game-theoretic model is applied to schedule security forces to protect the Los Angeles Airport towards adversaries and, in [YJJ<sup>+</sup>12], an approach to compute an optimal duty roster for fare inspectors for the Los Angeles Metro is presented.

In contrast to previous works, we consider a network with arbitrary topology (in [YJJ<sup>+</sup>12] single metro lines are considered), and users are free to choose their OD-path and may take a detour to avoid inspectors. A wise modeling approach has to be chosen, as the number of user strategies is exponential. [JKK<sup>+</sup>10] deals with a huge number of strategies for the defenders and gives a branch-and-price algorithm. We will present a simple cutting-plane method and an alternative network flow formulation to represent strategies of the users which will simplify matters. Parts of this thesis, i.e. the results referred to in subsection 3.2, were presented on the INOC 2013 in a cooperative work with Borndörfer et al. [BBSS12]. Moreover, we find similar applications of game theory in studies of arms control and disarmament, usually relating to an arms control treaty that has been signed, problems in accountancy and auditing, enforcement of environmental regulations, crime control and related areas [AvSZ02].

In the following, we will introduce a model which we call spot-checking game and present mathematical programs and methods to compute its relevant equilibria. Our aim is to find equilibrium strategies with a good balance between computation time and inspectors' payoff. We will present some theoretical results and afterward apply the model to simplified instances of the transportation network of German motorways using real data. Computed optimal distributions of inspectors are compared to distributions proportional to recorded traffic volumes in a spot-checking game. We will present an application where we are considering inspections regarding spacial restrictions and time dependencies and extract a probabilistic duty roster. Results of this thesis may be used as input for the current large-scale integer program to compute an integrated control and staff roster for the real-world problem mentioned at the beginning of this section.



# Chapter 2

## A game-theoretic approach

We are using a game-theoretic approach to model the strategic interaction between inspectors and users of a transportation network. We start off by creating a general model, where the users face some costs, independent of inspections, for using arcs of the network and in addition a fine if an inspector is present. The inspectors receive a reward for every arc taken by a user and the profit from fines. Later on in chapter 3, we will apply this model to the transportation network of German motorways where the users are legally obliged to pay a toll. In contrast to previous works, a user will have multiple Origin-Destination-path (OD-path) options. The number of OD-path options can be huge. To receive results in reasonable amount of time, a suitable problem formulation has to be chosen. In the following we present a spot-checking game and methods to compute its equilibria. We give a brief overview on the related class of security games and their general properties and present some theoretical results specific to our model. *Algorithmic Game Theory*, written by Noam Nisan et al. in 2007 [NRT07], and *Game Theory*, written by Fudenberg and Tirole in 1991 [FT91], served as reference books for the basic game-theoretic concepts used in the following. As a reference book for the basic concepts of linear and integer programming, Schrijver's *Theory of linear and integer programming* from 1998 [Sch98] was used.

### 2.1 A spot-checking game

**Transportation network** A transportation network is represented by a weighted digraph  $D = (V, E, \mathbf{w}, \mathbf{f}, \boldsymbol{\sigma})$ . Traveling costs for a network user for taking arc  $e \in E$  are given by  $w_e$ , the reward for the inspectors if a user takes  $e \in E$  is  $f_e$ , and  $\sigma_e$  is an additional fine opposed by the inspectors on a user for taking an arc  $e \in E$  when an inspector is present on  $e$ .

The following relations hold:

$$f_e \leq w_e \quad \text{and} \quad 0 < \sigma_e, \quad \forall e \in E.$$

**Players, strategies, payoffs** We assume that all users are distributed over a set  $\mathcal{K}$  of commodities. If we do not consider time, a commodity corresponds to an Origin-Destination node pair (short: OD-pair) in  $D$ . If we consider time and a set of time intervals  $\mathcal{T}$  is given, then a commodity is a pair  $(k, t)$ , where  $k$  is an OD-pair in  $D$  and  $t \in \mathcal{T}$ . The number  $x_k$  of users of a commodity  $k \in \mathcal{K}$  is given. We may also refer to  $x_k$  as the demand for commodity  $k$ . Let  $\mathcal{R}_k$  denote the set of all OD-paths of  $k \in \mathcal{K}$ . The user's costs for traveling on a commodity  $k$  are bounded from above by some value  $w_{r_k^*}$  for taking some path  $r_k^* \in \mathcal{R}_k$ . We slightly abuse the notation and use  $w_{r_k^*} := \sum_{e \in r_k^*} w_e$ . No inspections can be conducted on any arc in  $r_k^*$  and the inspectors' reward if a user takes  $r_k^*$  is given by  $f_{r_k^*} := \sum_{e \in r_k^*} f_e$ . (In practice,  $r_k^*$  corresponds to a path of  $k$  on which users pay the toll fare and do not face any fine).

We will only consider strategic interactions between a network user and the inspectors, i.e. we do not consider congestions or similar and instead, we formulate a non-cooperative multi-player game  $G$  between one *inspectors player* (also referred to as the *inspectors*) representing the team of inspectors and, for each commodity  $k \in \mathcal{K}$ , one *player  $k$*  representing all users of commodity  $k$ .

The number of inspectors in a transportation network is given and denoted by  $\gamma$ . The set of pure strategies of the inspectors player, which we denote by  $\mathcal{L}$ , is defined as the set of subsets of  $E$  of cardinality less than or equal to  $\gamma$ . Every  $l \in \mathcal{L}$  fulfills the requirement that  $\forall k \in \mathcal{K}$ , a path  $r_k^*$ , on which no inspections can be conducted, exists. A mixed strategy of the inspectors player corresponds to a probability distribution over all sets in  $\mathcal{L}$ . For any mixed strategy and  $\forall e \in E$ , let  $q_e$  denote the marginal probability that an inspector is present on  $e$ , i.e. let  $q_e$  be the sum over all pure strategies  $l \in \mathcal{L}$  which cover arc  $e$  of the probability that  $l$  is chosen in the mixed strategy. Our model will only depend on the vector  $\mathbf{q}$  holding these marginal probabilities, where  $0 \leq q_e \leq 1$  and  $\sum_{e \in E} q_e \leq \gamma$  and  $\mathbf{q} \in \mathcal{Q}$  for some polytope  $\mathcal{Q}$ . Whenever we refer to  $\mathbf{q}$  as a strategy of the inspectors, we actually mean the vector holding the marginal probabilities for the presence of inspectors on arcs. We point out that for every vector  $\mathbf{q}$  satisfying the conditions, we can find a probability distribution over sets in  $\mathcal{L}$  whose marginal equals  $\mathbf{q}$ .

For all  $k \in \mathcal{K}$ , the set of pure strategies of player  $k$  consists of all paths  $r \in \mathcal{R}_k$ . A mixed strategy of player  $k$  is given by any probability distribution over all of his pure strategies and is denoted by  $\mathbf{p}^k$ . The set of mixed strategies of player  $k$  is denoted by  $\Omega^k$ . Let  $\mathbf{p} := ((\mathbf{p}^k)_{k \in \mathcal{K}})$  denote the finite sequence of mixed strategies  $\mathbf{p}^k$  of all player  $k$ ,  $k \in \mathcal{K}$ . The cartesian product of the sets of mixed strategies of all player  $k$ ,  $k \in \mathcal{K}$ , is denoted by  $\Omega$ .

The expected payoff of player  $k$  is given by

$$\pi_k(\mathbf{p}, \mathbf{q}) := \pi_k(\mathbf{p}^k, \mathbf{q}) := - \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e).$$



The expected payoff of the inspectors is given by

$$\pi_c(\mathbf{p}, \mathbf{q}) := \sum_{k \in \mathcal{K}} x_k \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + f_e).$$

## 2.2 Game settings and equilibria

Let game  $G$  be of complete information (i.e. every player knows the payoffs and strategies available to himself and all other players). Each player is risk-neutral and trying to advance his self-interest (i.e. the preferences of a player over the different outcomes of  $G$  are determined by his expected payoff for the mixed strategy profiles  $(\mathbf{p}, \mathbf{q})$ , which he wishes to maximize). Note that in general game  $G$  is not zero-sum.

Some research has been done on the suitability of game settings and equilibria describing the strategic interaction between inspectors or security patrols, respectively, and adversaries on certain targets; see [AvSZ02], [vSZ04], [CS06], [PJO<sup>+</sup>09], [YKK<sup>+</sup>10]. In reality, despite the limited rationality of a user, we are uncertain about the user's ability to observe the inspectors' action (i.e. the strategy that is chosen by the inspectors). If adversaries, or users, act without acquiring (possibly costly) information, a simultaneous game setting may be suitable [YKK<sup>+</sup>10], [PJO<sup>+</sup>09]. Here, a stable outcome is described by a Nash equilibrium. The famous Nash theorem from 1951 [Nas51] states that every (non-cooperative) game with a finite set of players and a finite set of strategies has a Nash equilibrium of mixed strategies, hence a Nash equilibrium of  $G$  exists. But it seems reasonable to assume that the users monitor the inspectors to some extent. A setting, in which the users have perfect information about the inspectors' action is described by a Stackelberg game, which was introduced by Heinrich von Stackelberg in 1934 [vS34]. Here one player, the *leader* (in  $G$ , the inspectors), commits to a strategy knowing that the other player(s), the *follower(s)* (in  $G$ , every player  $k$ ), will observe it perfectly and then react to it with a best response. Being aware of that the followers will play optimally, an outcome where the leader commits to a strategy that maximizes his payoff is called a Stackelberg equilibrium. As pointed out in [vSZ04], for zero-sum games a Stackelberg game was explicitly defined by von Neumann and Morgenstern (p. 100 ff., calling it a *minorant*, resp. *majorant* game) [vNM47] in order to introduce the maxmin value, respectively minmax value, of the game. For finite two-person zero-sum games, the different game-theoretic solution concepts of maxmin, minmax, Nash equilibrium and Stackelberg equilibrium all yield the same optimal strategies for the leader (see e.g. [YKK<sup>+</sup>10]). In general, the ability to commit to a strategy can often be exploited to influence the outcome of the game in the leader's favor, e.g. the leader may be best off playing a strategy that is strictly dominated by another strategy [CS06], such a strategy would not be chosen in a Nash equilibrium. Consider the following example of a bimatrix game taken from [CS06]: For the table given below, we have a row player whose pure strategies are the row on the left and the row on the right and we have a column player whose pure strategies are the upper column and the lower column. An entry (r,c) in the table holds the payoffs

of the players for the corresponding pure strategy profile, i.e.  $r$  is the payoff for the row player and  $c$  is the payoff for the column player. The row player is the leader and the column player is the follower.

2, 1	4, 0
1, 0	3, 1

The lower strategy for the row player is strictly dominated by the upper strategy. If the row player chooses the upper strategy, the column player will choose the left strategy. This outcome corresponds to the unique Nash equilibrium of this game with a payoff of 2 for the row player. Now, in a Stackelberg game, if the row player would commit to the lower strategy instead, the column player would prefer the right strategy, giving the row player a higher payoff of 3. For any mixed strategy where the row player chooses the lower strategy with probability  $0.5 + \epsilon$  with  $\epsilon > 0$ , the column player prefers the right-hand side. The smaller the  $\epsilon$ , the higher the payoff will be for the row player. It is a standard assumption and sufficient for a Stackelberg equilibrium (of mixed strategies) to exist [CS06], to assume that, if the follower is indifferent regarding a set of strategies to respond with, he will choose a strategy which favors the leader the most, i.e. the optimal strategy to commit to for the row player is to play the lower strategy with probability of 0.5. Here, the column player prefers the right-hand side and the expected payoff for the row player is 3.5.

We see that in general, Nash and Stackelberg equilibrium may not coincide and the corresponding leader's strategies and payoffs may be different. The leader's payoff of a Stackelberg equilibrium (of mixed strategies) in two-player games is always greater than or equal to the payoff of a Nash equilibrium [vSZ04]. On the other hand, in certain cases it is much easier to compute a Nash equilibrium, and the deviation of the payoffs of a Nash strategy and a Stackelberg strategy might be very small. Definitions, some properties and relations between the different equilibria that are specific to spot-checking game  $G$  will be given and analyzed in the following:

**Nash equilibrium** A Nash equilibrium of spot-checking game  $G$  is a strategy profile  $(\mathbf{p}, \mathbf{q})$  satisfying

$$\begin{aligned} \pi_c(\mathbf{p}, \mathbf{q}) &\geq \pi_c(\mathbf{p}, \mathbf{q}'), & \forall \mathbf{q}' \in \mathcal{Q}, \text{ and} \\ \pi_k(\mathbf{p}^k, \mathbf{q}) &\geq \pi_k(\mathbf{p}^{k'}, \mathbf{q}), & \forall \mathbf{p}^{k'} \in \Omega^k, \forall k \in \mathcal{K}. \end{aligned}$$

In the following, we may also use the term *Nash strategy* to refer to a Nash equilibrium strategy.

**Stackelberg equilibrium** For every  $k \in \mathcal{K}$ , let the set of *best responses* for player  $k$  to  $\mathbf{q} \in \mathcal{Q}$  be given by

$$BR_k(\mathbf{q}) := \{\mathbf{p}^k \in \Omega^k : \pi_k(\mathbf{p}^k, \mathbf{q}) \geq \pi_k(\mathbf{p}^{k'}, \mathbf{q}), \forall \mathbf{p}^{k'} \in \Omega^k\}.$$

Respectively, the set of best responses for all player  $k$ ,  $k \in \mathcal{K}$ , to  $\mathbf{q} \in \mathcal{Q}$  (short: the set of best responses to  $\mathbf{q} \in \mathcal{Q}$ ) is given by

$$BR(\mathbf{q}) := \{\mathbf{p} \in \Omega : \pi_k(\mathbf{p}^k, \mathbf{q}) \geq \pi_k(\mathbf{p}^{k'}, \mathbf{q}), \forall \mathbf{p}^{k'} \in \Omega^k, \forall k \in \mathcal{K}\}.$$

The set of best responses to  $\mathbf{p} \in \Omega$  is

$$BR(\mathbf{p}) := \{\mathbf{q} \in \mathcal{Q} : \pi_c(\mathbf{p}, \mathbf{q}) \geq \pi_c(\mathbf{p}, \mathbf{q}'), \forall \mathbf{q}' \in \mathcal{Q}\}.$$

A Stackelberg equilibrium of  $G$  is a strategy profile  $(\mathbf{p}, \mathbf{q})$  satisfying

$$\pi_c(\mathbf{p}, \mathbf{q}) = \max_{\mathbf{q}' \in \mathcal{Q}} \max_{\mathbf{p}' \in BR(\mathbf{q}')} \pi_c(\mathbf{p}', \mathbf{q}').$$

A Stackelberg equilibrium is a profile  $(\mathbf{p}, \mathbf{q})$  which maximizes the leader's payoff among the set of all profiles, where the followers' strategies are best responses to the leader's strategy. If a follower has several best responses available, he will select the one that favors the leader the most. We may also use the term *Stackelberg strategy* to refer to a Stackelberg equilibrium strategy.

In the following proposition, we prove an equivalent representation of the best responses of player  $k$ ,  $k \in \mathcal{K}$ , as specific shortest paths.

**Proposition 2.2.1.** *The vector  $\mathbf{p}^k$  is a best response to  $\mathbf{q}$  for a player  $k$  if and only if it is a probability distribution assigning positive values only to the shortest paths of  $\mathcal{R}_k$  w.r.t. the total expected costs for player  $k$  w.r.t.  $\mathbf{q}$ .*

**Proof.** Let  $\mathbf{p}^k$  be a best response to  $\mathbf{q}$ . By definition of  $BR_k(\mathbf{q})$ ,  $\forall \mathbf{p}^{k'} \in \Omega^k$ , we have

$$\pi_k(\mathbf{p}^k, \mathbf{q}) = - \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e) \geq \pi_k(\mathbf{p}^{k'}, \mathbf{q}) = - \sum_{r \in \mathcal{R}_k} p_r^{k'} \sum_{e \in r} (\sigma_e q_e + w_e). \quad (2.1)$$

Equivalently,  $\forall \mathbf{p}^{k'} \in \Omega^k$ ,

$$-\pi_k(\mathbf{p}^k, \mathbf{q}) = \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e) \leq -\pi_k(\mathbf{p}^{k'}, \mathbf{q}) = \sum_{r \in \mathcal{R}_k} p_r^{k'} \sum_{e \in r} (\sigma_e q_e + w_e). \quad (2.2)$$

The term  $\sum_{e \in r} (\sigma_e q_e + w_e)$  corresponds to the total expected costs, when  $\mathbf{q}$  is fixed, for taking a path  $r \in \mathcal{R}_k$ . Let us assume that there exists a path  $\tilde{r} \in \mathcal{R}_k$  with  $p_{\tilde{r}}^k > 0$  that is not a shortest path. Then a path  $\hat{r} \in \mathcal{R}_k$  that is shorter than path  $\tilde{r}$  exists and we can construct a probability distribution

$$p_r^{k'} := \begin{cases} 0 & \text{if } r = \tilde{r}, \\ p_{\hat{r}}^k + p_{\tilde{r}}^k & \text{if } r = \hat{r}, \\ p_r^k & \text{else} \end{cases}$$

with

$$-\pi_k(\mathbf{p}^k, \mathbf{q}) = \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e) > -\pi_k(\mathbf{p}^{k'}, \mathbf{q}) = \sum_{r \in \mathcal{R}_k} p_r^{k'} \sum_{e \in r} (\sigma_e q_e + w_e). \quad (2.3)$$

As every probability distribution over  $\mathcal{R}_k$  is a mixed strategy for player  $k$ ,  $\mathbf{p}' \in \Omega^k$ , (2.3) is a contradiction of (2.2).

Conversely, let  $\mathbf{p}^k$  be a probability distribution over the elements of  $\mathcal{R}_k$  assigning positive values only to the shortest paths of  $\mathcal{R}_k$  w.r.t. their total expected costs for given  $\mathbf{q}$ . Then, as  $\sum_{e \in r} (\sigma_e q_e + w_e)$  corresponds to the total expected costs w.r.t.  $\mathbf{q}$  for using a path  $r \in \mathcal{R}_k$ , for every other probability distribution  $\mathbf{p}^{k'}$  over  $\mathcal{R}_k$ ,  $\sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e) \leq \sum_{r \in \mathcal{R}_k} p_r^{k'} \sum_{e \in r} (\sigma_e q_e + w_e)$  holds. Every probability distribution over  $\mathcal{R}_k$  is a mixed strategy for player  $k$ , so  $\mathbf{p}, \mathbf{p}' \in \Omega^k$  and the latter inequality can be written as  $\pi_k(\mathbf{p}^k, \mathbf{q}) \leq \pi_k(\mathbf{p}^{k'}, \mathbf{q}), \forall \mathbf{p}^{k'} \in \Omega^k$ , hence by definition of  $BR_k(\mathbf{q})$ ,  $\mathbf{p}$  is a best response for player  $k$  to  $\mathbf{q}$ .  $\square$

A spot-checking game is a Stackelberg game. Below, we formally define the payoff for the inspectors in a Stackelberg game when playing with some mixed strategy  $\mathbf{q} \in \mathcal{Q}$ .

**Inspectors' payoff in a Stackelberg game** The inspectors' payoff in a Stackelberg game when playing with strategy  $\mathbf{q}$  is determined by the following function:

$$v_{stack} : \mathcal{Q} \longrightarrow \mathbb{R}_+, \mathbf{q} \longmapsto \max_{\mathbf{p}' \in BR(\mathbf{q})} \pi_c(\mathbf{p}', \mathbf{q}).$$

### 2.2.1 Inspectors' Nash vs. Stackelberg equilibrium strategies

Spot-checking games form a special superclass of security games (e.g. formally defined in [YKK<sup>+</sup>10]; reductions are given in appendix A). While in security games the interaction between inspectors and adversaries are studied, we are considering inspectors and users of a transportation network. A striking feature of spot-checking games is the often very likely exponential number of strategies for the users. We have to develop a good formulation and approaches to handle this difficulty. In the comprehensive study by Yin et al. [YKK<sup>+</sup>10] about the relation of Stackelberg and Nash equilibria in security games some positive results are proven. We will show in subsection 2.3.1 that, as in all security games, the strategies in different Nash equilibria of a spot-checking game are interchangeable<sup>1</sup>. We will present a much easier proof than the one presented in [YKK<sup>+</sup>10]. For security games, Yin et al. showed that under certain conditions an inspectors' Stackelberg strategy is also a Nash strategy. In general, these conditions do not hold for spot-checking game  $G$ . But just as in the experimental tests of Yin et al., our experimental results for the application in the transportation network of German motorways will yield that the

<sup>1</sup>Given any two Nash equilibria  $(\mathbf{p}, \mathbf{q})$  and  $(\mathbf{p}', \mathbf{q}')$  of  $G$ ,  $(\mathbf{p}, \mathbf{q}')$  and  $(\mathbf{p}', \mathbf{q})$  are also Nash equilibria of  $G$ .

inspectors' payoff in a Stackelberg equilibrium is often close or equal to the payoff in a Stackelberg game when playing with a Nash strategy. In particular, instead of solving a mixed-integer problem (MIP) to compute a Stackelberg equilibrium (subsection 2.4), we may use a linear program (LP) to compute a Nash strategy (subsection 2.3) and drastically reduce the computation time in return for a good approximation. We will now present some theoretical results for spot-checking game  $G$ .

Let us begin with two trivial lemmas that give a general idea of the relation between the inspectors' Nash strategies and Stackelberg strategies in terms of the inspectors' payoff. Let  $(\mathbf{p}, \mathbf{q})$  be some Nash equilibrium and  $\mathbf{q}^*$  be an inspectors' Stackelberg strategy of  $G$ . Then

$$\begin{aligned} \pi_c(\mathbf{p}, \mathbf{q}) & \text{ is the inspectors' profit for the Nash} \\ & \text{equilibrium } (\mathbf{p}, \mathbf{q}), \\ v_{stack}(\mathbf{q}) = \max_{\mathbf{p}' \in BR(\mathbf{q})} \pi_c(\mathbf{p}', \mathbf{q}) & \text{ is the inspectors' profit in a Stackelberg game} \\ & \text{when playing with Nash strategy } \mathbf{q}, \\ v_{stack}(\mathbf{q}^*) = \max_{\mathbf{q}' \in \mathcal{Q}} v_{stack}(\mathbf{q}') & \text{ is the inspectors' profit for a Stackelberg} \\ & \text{equilibrium.} \end{aligned}$$

**Lemma 2.2.2.** *Let  $(\mathbf{p}, \mathbf{q})$  be a Nash equilibrium and let  $\mathbf{q}^*$  be an inspectors' Stackelberg strategy of  $G$ . Then*

$$\pi_c(\mathbf{p}, \mathbf{q}) \leq v_{stack}(\mathbf{q}) \leq v_{stack}(\mathbf{q}^*).$$

**Proof.**  $(\mathbf{p}, \mathbf{q})$  is a Nash equilibrium, i.e.  $\mathbf{q} \in BR(\mathbf{p})$  and  $\mathbf{p} \in BR(\mathbf{q})$ . Using above definitions, we have  $\pi_c(\mathbf{p}, \mathbf{q}) \leq \max_{\mathbf{p}' \in BR(\mathbf{q})} \pi_c(\mathbf{p}', \mathbf{q}) = v_{stack}(\mathbf{q}) \leq \max_{\mathbf{q}' \in \mathcal{Q}} v_{stack}(\mathbf{q}') = v_{stack}(\mathbf{q}^*)$ .  $\square$

**Lemma 2.2.3.** *Let  $(\mathbf{p}, \mathbf{q})$  be a Nash equilibrium and let  $\mathbf{q}^*$  be an inspectors' Stackelberg strategy of  $G$ . Furthermore, let  $(\tilde{\mathbf{p}}, \tilde{\mathbf{q}}) := \arg \max_{\mathbf{p}', \mathbf{q}'} \{\pi_c(\mathbf{p}', \mathbf{q}') : \mathbf{p}' \in BR(\mathbf{q}'), \mathbf{q}' \in BR(\mathbf{p}')\}$  be a best Nash equilibrium of  $G$  from the inspectors' point of view. Then*

$$\pi_c(\mathbf{p}, \mathbf{q}) \leq \pi_c(\tilde{\mathbf{p}}, \tilde{\mathbf{q}}) \leq v_{stack}(\mathbf{q}^*).$$

**Proof.**  $(\mathbf{p}, \mathbf{q})$  is a Nash equilibrium, i.e.  $\mathbf{q} \in BR(\mathbf{p})$  and  $\mathbf{p} \in BR(\mathbf{q})$ , so  $\pi_c(\mathbf{p}, \mathbf{q}) \leq \max_{\mathbf{p}', \mathbf{q}'} \{\pi_c(\mathbf{p}', \mathbf{q}') : \mathbf{p}' \in BR(\mathbf{q}'), \mathbf{q}' \in BR(\mathbf{p}')\} = \pi_c(\tilde{\mathbf{p}}, \tilde{\mathbf{q}})$ . The second inequality is obtained by applying lemma 2.2.2.  $\square$

Note that an optimal Nash equilibrium from the inspectors' point of view (which is a pair of best response strategies) is not the same as the best of all inspectors' Nash strategies in a Stackelberg game.

In the following, we will present a proposition that states a restriction on  $G$  which implies that the inspectors' payoff in a Stackelberg game for playing with a Nash strategy and the inspectors' payoff of a Stackelberg equilibrium are identical.

**Proposition 2.2.4.** *If*

1. *the Nash equilibrium  $(\mathbf{p}, \mathbf{q})$  of game  $G$  is unique, and*
2. *for all inspectors' Stackelberg strategies  $\mathbf{q}^*$ , there exists a followers' strategy  $\mathbf{p}^*$ , such that  $(\mathbf{p}^*, \mathbf{q}^*)$  is a Nash equilibrium,*

*then  $v_{stack}(\mathbf{q}) = v_{stack}(\mathbf{q}^*)$ .*

**Proof.** Let  $(\mathbf{p}, \mathbf{q})$  be the unique Nash equilibrium of  $G$  and for all Stackelberg strategies  $\mathbf{q}^*$  of  $G$ , let there exist a followers' strategy  $\mathbf{p}^*$ , such that  $(\mathbf{p}^*, \mathbf{q}^*)$  is a Nash equilibrium. Then  $(\mathbf{p}^*, \mathbf{q}^*)$  must be the Nash equilibrium  $(\mathbf{p}, \mathbf{q})$  and i.e.  $\mathbf{q}^* = \mathbf{q}$  and hence  $v_{stack}(\mathbf{q}^*) = v_{stack}(\mathbf{q})$ .  $\square$

In low dimensions, practical examples where the Nash equilibrium is not unique seem to correspond to the cases that are uninteresting: for example, if all users pay the toll. But examples exist and we will present such an example below. The second condition of proposition 2.2.4 is always satisfied for a restricted subclass of spot-checking games. We formulate this result in the following theorem 2.2.5. In other words, for a special case, we can prove, that any inspectors' Stackelberg strategy is a Nash strategy by using some results of [YKK<sup>+</sup>10].

**Theorem 2.2.5.** *If all OD-paths in transportation network  $D$  are disjoint, then every inspectors' Stackelberg strategy in spot-checking game  $G$  is also a Nash strategy.*

**Proof.** Recall that  $\mathcal{L}$  denotes the set of pure strategies of the inspectors, i.e. all subsets of cardinality less or equal to  $\gamma$  in arc set  $E$  of the transportation network  $D = (V, E)$  (w.r.t. to some linear constraints). Let  $\mathcal{R}$  denote the union over  $k \in \mathcal{K}$  of all paths  $r \in \mathcal{R}_k$  (i.e.  $\mathcal{R}$  is the set of all OD-paths in spot-checking game  $G$ ). We define a new set that holds information about which OD-paths are covered by a pure strategy  $l \in \mathcal{L}$  of the inspectors, i.e.  $\forall l \in \mathcal{L}$ , let  $\mathcal{R}^l := \{r \in \mathcal{R} : \exists e \in (r \cap l)\}$ .

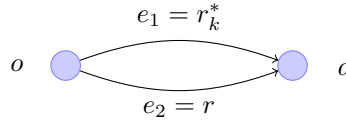
Yin et al. introduced a restriction called an SSAS property for security games (see equation (2) in [YKK<sup>+</sup>10]). For spot-checking game  $G$ , this property corresponds to the property that

$$\forall l \in \mathcal{L}, \forall S \subseteq \mathcal{R}^l, \exists l' \in \mathcal{L} : S = \mathcal{R}^{l'}.$$

Now, if all OD-paths are disjoint and a certain number of paths are controlled when choosing a pure strategy, then also for every subset of these paths we can find a pure strategy controlling these paths exclusively, because an inspectors pure strategy corresponds to a subset of arcs of cardinality smaller than or equal to  $\gamma$ . I.e. the SSAS property is satisfied. We apply corollary 4.9 of [YKK<sup>+</sup>10], which states that in security games with the SSAS property any defender's (i.e. inspectors') Stackelberg strategy is also a Nash strategy.  $\square$

In the following, we will present an example where the Nash equilibrium is not unique, but the conditions of theorem 2.2.5 are satisfied and hence any Stackelberg strategy is also a Nash strategy. The maximal gap between the payoffs for playing with a Nash strategy can be determined w.r.t. fixed users' traveling costs  $\mathbf{w}$  and inspectors' rewards  $\mathbf{f}$ . Afterwards, we present an example, where the Stackelberg strategy is not a Nash strategy.

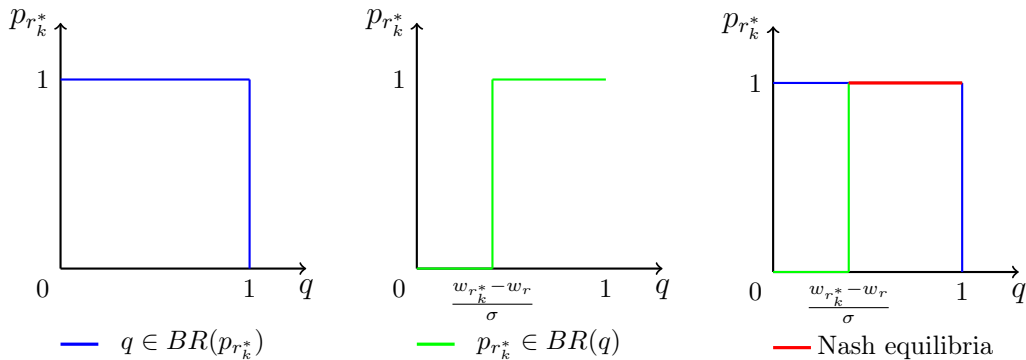
**Example I** We consider a spot-checking game between one inspector and a single user of commodity  $k = (o, d)$  in the transportation network given by



We assume that the inspector may only control on arc  $e_2$  and has the ability to control with a certain probability. Her mixed strategy is denoted by  $q$ . The user of  $k$  (player  $k$ ) has two options to travel, either using  $e_1$  corresponding to the path  $r_k^*$  where no inspections may be conducted or taking  $e_2$  which is denoted as path  $r$ . His mixed strategy is  $\mathbf{p}$ . The costs for player  $k$  are  $w_{r_k^*}$  for taking  $r_k^*$  and  $w_r + \sigma q$  for taking path  $r$ , respectively, the payoff for the inspector if player  $k$  takes  $r_k^*$  is  $f_{r_k^*}$  and if player  $k$  takes  $r$  it is  $f_r + \sigma q$ .

If  $w_r > w_{r_k^*}$ , player  $k$  will always choose path  $r_k^*$  and if  $w_{r_k^*} > w_r + \sigma$ , player  $k$  will always choose path  $r$  independent of the inspector's strategy. Hence let's assume  $w_r \leq w_{r_k^*} \leq w_r + \sigma$ . Here the best response of player  $k$  depends on the value of  $q$ .

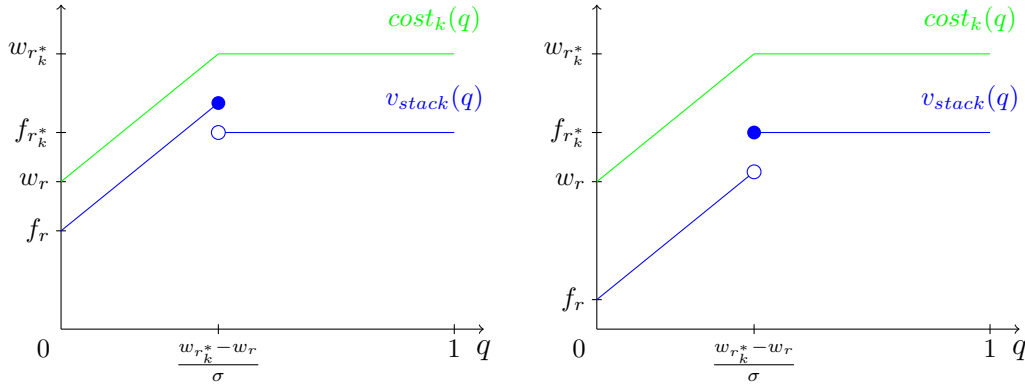
It is easy to see that the pure strategy profile, where player  $k$  plays  $r_k^*$  and the inspector plays  $e_2$ , is a Nash equilibrium ( $r_k^*$  is a best responses to  $e_2$  and  $e_2$  is a best responses to  $r_k^*$ ). The following graphic shows the best response of the inspector to  $\mathbf{p}$  in blue and the best response of player  $k$  to  $q$  in green (on the y-axes we have the values of  $p_{r_k^*}$  which implicitly gives the value of  $p_r$  as the sum of  $p_{r_k^*}$  and  $p_r$  must equal one). All points of the intersection of best responses (below in red) correspond to Nash equilibria of  $G$ :



Player  $k$  is indifferent between taking path  $r_k^*$  and  $r$  if  $w_{r_k^*} = w_r + q\sigma$  equivalent to  $q = \frac{w_{r_k^*} - w_r}{\sigma}$ . A Nash equilibrium is obtained for all  $q \geq \frac{w_{r_k^*} - w_r}{\sigma}$ . If in a Nash equilibrium  $w_{r_k^*} < w_r + q\sigma$ , then player  $k$ 's best response is to use path  $r_k^*$ . The inspector's payoff is  $f_{r_k^*}$ . Else, if  $w_{r_k^*} = w_r + q\sigma$ , player  $k$  will be indifferent between  $r$  and  $r_k^*$ . Depending on player  $k$ 's choice, the inspector's payoff will be  $f_{r_k^*}$  or  $f_r + q\sigma$ . So the maximal difference between the inspector's payoffs for the best Nash equilibrium and any other Nash equilibrium is  $|f_r + q\sigma - f_{r_k^*}|$  with  $q = \frac{w_{r_k^*} - w_r}{\sigma}$ , i.e. the gap is given by

$$\left| f_r + \frac{w_{r_k^*} - w_r}{\sigma} \sigma - f_{r_k^*} \right| = |(w_{r_k^*} - f_{r_k^*}) - (w_r - f_r)|.$$

In the following graphs the inspectors' payoff in a Stackelberg game  $v_{stack}(q)$  and player  $k$ 's costs (labeled  $cost_k(q)$ ) in a Stackelberg game for given  $\mathbf{q}$  are sketched for the two cases  $f_{r_k^*} < f_r + q\sigma$  when  $q = \frac{w_{r_k^*} - w_r}{\sigma}$  (on the left) and  $f_{r_k^*} > f_r + q\sigma$  when  $q = \frac{w_{r_k^*} - w_r}{\sigma}$  (on the right). Recall that every maximum of  $v_{stack}(q)$  is an inspectors' Stackelberg equilibrium payoff (see definition at the beginning of this section).



On the left, the inspectors' Stackelberg strategy is  $q = \frac{w_{r_k^*} - w_r}{\sigma}$  and unique. In the case displayed on the right, all  $q > \frac{w_{r_k^*} - w_r}{\sigma}$  are inspectors' Stackelberg strategies. In a third case, when  $f_{r_k^*} = f_r + \frac{w_{r_k^*} - w_r}{\sigma} \sigma$ , all  $q \geq \frac{w_{r_k^*} - w_r}{\sigma}$  are Stackelberg strategies. Therefore the above presented gap also corresponds to the maximal gap between the inspectors' payoff in a Stackelberg game for playing with a Nash strategy and the inspectors' payoff for a Stackelberg equilibrium.

Realistic values for the transportation network in the application on German motorways are a toll fare of 0.17 euro cents per driven km and a fine of 200 euros. In this simple spot-checking game, let for example,  $r$  and  $r_k^*$  both have a length of 100 km. We can interpret  $r_k^*$  as a path where the user pays the toll and  $r$  as a path where he evades it. Let us assume that there are no other traveling costs than the toll fare, i.e.



$$\begin{aligned} w_{r_k^*} &= 17, f_{r_k^*} = 17, \\ w_r &= 0, f_{e_2} = 0, \\ \sigma &= 200. \end{aligned}$$

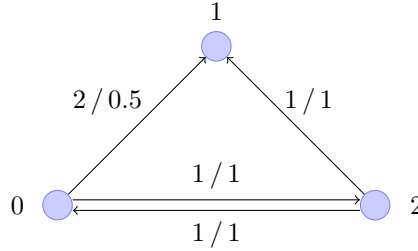
Then the maximal gap between the inspector's payoff for the Stackelberg strategy and any Nash strategy is

$$|(w_{r_k^*} - f_{r_k^*}) - (w_r - f_r)| = |(17 - 17) - (0 - 0)| = 0.$$

If we adjust the values a bit, for example adding additional traveling costs for the player  $k$  on path  $r$  and on path  $r_k^*$  and varying their lengths, we still receive a relatively small maximal gap. (By relative gap, we mean the maximal gap divided by the the maximal user's cost  $w_{r_k^*}$ ).  $\square$

We now present an example where the inspectors' Stackelberg strategy in a spot-checking game  $G$  is not part of any Nash equilibrium.

**Example II** Consider the following transportation network  $D$  (labels represent  $w_e / f_e$ ):

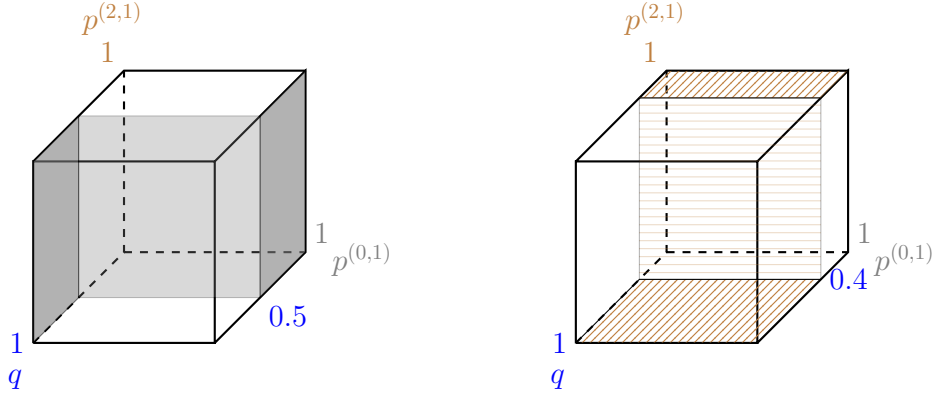


The fine for being caught evading is  $\sigma = 10$  on every arc and we have one inspector that can control arc  $(0, 1)$  or arc  $(2, 1)$ . There are two different commodities to be considered: Commodity  $(0, 1)$  with demand  $x_{(0,1)} = 5$  and two path options in  $D$ , i.e. path  $\{(0, 1)\}$  or path  $\{(0, 2), (2, 1)\}$ , and commodity  $(2, 1)$  with demand  $x_{(2,1)} = 10$  also also two path options, path  $\{(2, 1)\}$  or path  $\{(2, 0), (0, 1)\}$ . The OD-paths for every commodity on which no controls can be conducted are not drawn in the picture, we assume that the costs for taking those paths are very high and therefore, they will never be chosen.

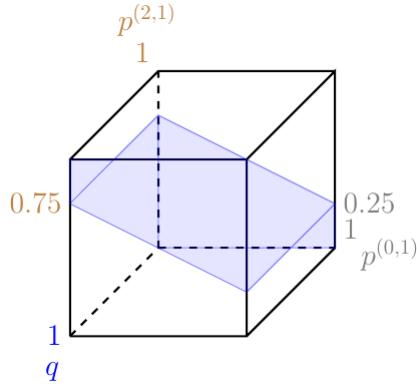
As all players have two pure strategies, we can represent the mixed strategies using a one-dimensional variables:  $p^{(0,1)}$  is the probability that the users of  $(0, 1)$  take path  $\{(0, 1)\}$ ,  $p^{(2,1)}$  is the probability that the users of  $(2, 1)$  take path  $\{(2, 1), (1, 0)\}$  and  $q$  is the probability that the inspector is inspecting arc  $(0, 1)$ .

Player  $(0, 1)$  is indifferent between choosing path  $\{(0, 1)\}$  and path  $\{(0, 2), (2, 1)\}$  if  $q = (1 - q)$  which is equivalent to  $q = 0.5$ . If  $q < 0.5$ , she will choose path  $\{(0, 1)\}$  and if  $q > 0.5$ , she will choose path  $\{(0, 2), (2, 1)\}$ . Player  $(2, 1)$  is indifferent between choosing path  $\{(2, 1)\}$  and path  $\{(2, 0), (0, 1)\}$  if  $3 + 10q = 1 + 10(1 - q)$  which is

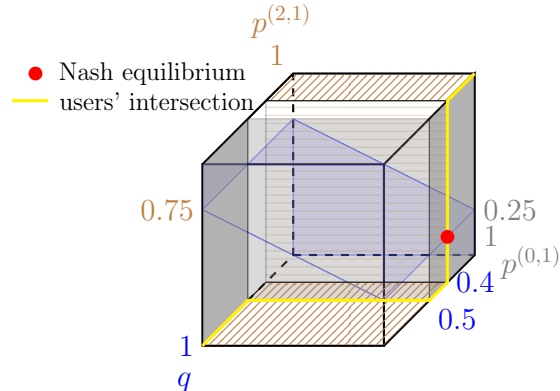
equivalent to  $q = 0.4$ . If  $q < 0.4$ , he will choose path  $\{(2, 0), (0, 1)\}$  and if  $q > 0.4$ , he will choose path  $\{(2, 1)\}$ . Below, in the picture on the left the value of  $p^{(0,1)}$  for given  $q$  is shown in gray color. On the right the strategy of player  $(2, 1)$  for given  $q$  is displayed in orange.



Doing the math for the inspector, we find that she is indifferent between inspecting arc  $(0, 1)$  and arc  $(2, 1)$  if  $p^{(0,1)} + 2p^{(2,1)} = 1.5$ . If  $p^{(0,1)} + 2p^{(2,1)}$  is smaller than 1.5, she will control arc  $(0, 1)$  and if it is greater, she will control arc  $(2, 1)$ . The inspector's strategy when the strategies chosen by the other players are given is displayed below in blue:

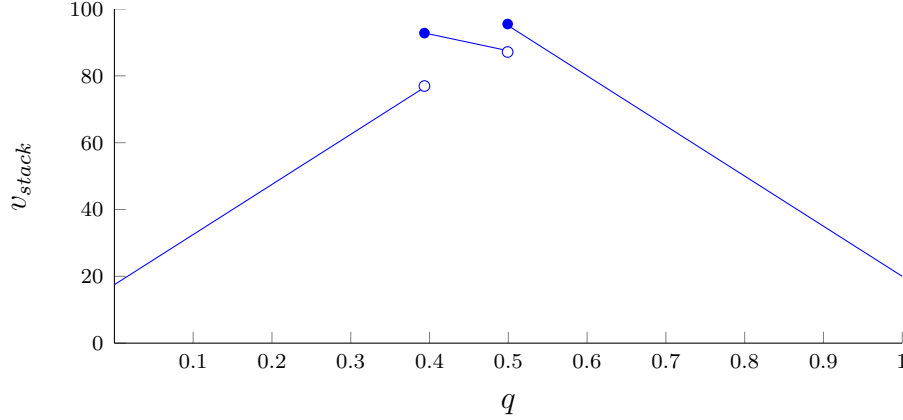


A Nash equilibrium corresponds to a point in the cube, where the strategies of all players intersect:



We have a unique inspectors' Nash strategy at  $q_{(0,1)} = 0.4$ ,  $q_{(2,1)} = 0.6$ .

Now in the following picture, the inspectors' payoff in a Stackelberg game w.r.t to  $\mathbf{q}$  is shown:



The maximum value for  $v_{stack}(\mathbf{q})$ , i.e. the inspectors' payoff in a Stackelberg equilibrium, is reached at  $q_{(0,1)} = 0.5$  with an inspectors' profit of 95. For the Nash strategy  $q_{(0,1)} = 0.4$  the payoff is only 92.5.  $\square$

In conclusion, we can state that for a restricted subclass of spot-checking games, we could prove that inspectors' Stackelberg and Nash strategies coincide. In theory, we can find simple counterexamples for which this is not true, but realistic input parameters yield a gap between the inspectors payoffs that is relatively small. This observation resembles experimental results for the application on German motorways that will be presented later on.

We will now present programs to compute an inspectors' Nash and an inspectors' Stackelberg strategy. We will see that it is much easier to compute a Nash strategy.

## 2.3 Computation of an inspectors' Nash equilibrium strategy

A polymatrix game is a multi-player game in which a finite number of players is each playing a finite number of bimatrix games simultaneously; the payoff of a player is the sum of the payoffs from every bimatrix game she plays. A zero-sum polymatrix game is a polymatrix game in which every bimatrix game is zero-sum. As pointed out by Berndörfer et al. [BOSS12], in 2009 Daskalakis and Papadimitriou generalized the famous Minmax theorem of John von Neumann from 1928 [vN28] to zero-sum polymatrix games. As a corollary, a mixed Nash equilibrium can be computed efficiently in polynomial time using linear programming [DP09].

Our spot-checking game is a polymatrix game, but it is not zero-sum and the usage of a linear program to compute a Nash equilibrium is not straight forward as done in [BOSS12]. This is due to the multiple OD-path options for every user accompanied by varying traveling costs. We will overcome this obstacle by proving a *best-response equivalence* of  $G$  and a zero-sum game  $G'$  for which Daskalakis and Papadimitriou's results hold. Two games are *best-response equivalent* if they have corresponding best responses; therefore their Nash equilibria coincide. The terminology was introduced by Rosenthal in 1974 [Ros74]. We then present a linear program formulation to find a Nash strategy for the inspectors in  $G'$  corresponding to the formulation presented in the introductory example in [DP09].

### 2.3.1 Best-response equivalent zero-sum game

For a nicer display in the following proof and in the derivation of our linear program formulation, we will represent spot-checking game  $G$  explicitly as a polymatrix game: in  $G$  the inspectors player is playing a bimatrix game against every player  $k$ ,  $k \in \mathcal{K}$ , and every player  $k$  is playing one bimatrix game against the inspectors player. Recall that the expected payoff of player  $k$  is given by

$$\pi_k(\mathbf{p}, \mathbf{q}) = - \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e),$$

and the expected payoff of the inspectors is

$$\pi_c(\mathbf{p}, \mathbf{q}) = \sum_{k \in \mathcal{K}} x_k \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + f_e).$$

For simplicity reasons, let us assume that we have exactly  $\gamma$  inspectors. Less than  $\gamma$  inspectors could be modeled by adding artificial arcs to the network that can be controlled, but that are never taken by the users. Now consider the following matrices:

$$(B_k)_{r \in \mathcal{R}_k, e \in E} := \frac{\sum_{e' \in r} (w_{e'} - f_{e'})}{\gamma}, \text{ and}$$

$$(A_k)_{r \in \mathcal{R}_k, e \in E} := \begin{cases} \sigma_e + \frac{\sum_{e' \in r} f_{e'}}{\gamma} & \text{if } r \in \mathcal{R}_k \text{ and } e \in r, \\ \frac{\sum_{e' \in r} f_{e'}}{\gamma} & \text{if } r \in \mathcal{R}_k \text{ and } e \notin r. \end{cases}$$

Then, using the standard representation for polymatrix games, the expected payoff of a player  $k$ ,  $k \in \mathcal{K}$ , is equivalently given by

$$\pi_k(\mathbf{p}, \mathbf{q}) = -(\mathbf{p}^k)^T (A_k + B_k) \mathbf{q},$$

because

$$\begin{aligned}
 -(\mathbf{p}^k)^T(A_k + B_k)\mathbf{q} &= -\sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} (\sigma_e + \frac{\sum_{e' \in r} w_{e'}}{\gamma}) q_e + \sum_{e \notin r} (\frac{\sum_{e' \in r} w_{e'}}{\gamma}) q_e \right) \\
 &= -\sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} \sigma_e q_e + \sum_{e \in E} (\frac{\sum_{e' \in r} w_{e'}}{\gamma}) q_e \right) \\
 &= -\sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} \sigma_e q_e + (\frac{\sum_{e' \in r} w_{e'}}{\gamma}) \sum_{e \in E} q_e \right) \\
 &= -\sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} \sigma_e q_e + (\frac{\sum_{e' \in r} w_{e'}}{\gamma}) \gamma \right) \\
 &= -\sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e + w_e) = \pi_k(\mathbf{p}, \mathbf{q}).
 \end{aligned}$$

Respectively, for the expected payoff of the inspectors we have

$$\pi_c(\mathbf{p}, \mathbf{q}) = \sum_{k \in \mathcal{K}} x_k (\mathbf{p}^k)^T A_k \mathbf{q}.$$

We define zero-sum polymatrix game  $G'$  to have the same players as game  $G$  with the same strategy sets, but a different payoff of the inspectors player:

The the expected payoff of a player  $k$ ,  $k \in \mathcal{K}$ , in  $G'$  is given by

$$\pi'_k(\mathbf{p}, \mathbf{q}) := \pi_k(\mathbf{p}, \mathbf{q}) = -(\mathbf{p}^k)^T(A_k + B_k)\mathbf{q},$$

and the expected payoff of the inspectors in  $G'$  is

$$\pi'_c(\mathbf{p}, \mathbf{q}) := \sum_{k \in \mathcal{K}} x_k (\mathbf{p}^k)^T(A_k + B_k)\mathbf{q}.$$

Note that the payoff of the inspectors from every partial bimatrix game played against a player  $k$  is weighted by the demand  $x_k$  of commodity  $k$ . Game  $G'$  is zero-sum as we could interpret the inspectors' payoff as if  $\forall k \in \mathcal{K}$ , the inspectors player would play the same bimatrix game against  $x_k$  player  $k$ , which will all choose the same strategy for a given  $\mathbf{q}$ .

I.e. we create a zero sum game  $G'$  out of  $G$  by replacing  $f_e$  by  $w_e$  for all  $e \in E$ .

**Proposition 2.3.1.** *Spot-checking game  $G$  and zero-sum game  $G'$  defined above are best-response equivalent.*

**Proof.** As all player  $k$ ,  $k \in \mathcal{K}$ , have the same set of strategies with the same expected payoffs in  $G$  and  $G'$ .

Now let  $\mathbf{q} \in \mathcal{Q}$  be a best response of the inspectors to  $\mathbf{p} \in \Omega^k$  in  $G'$ , i.e.

$$\sum_{k \in \mathcal{K}} x_k (\mathbf{p}^k)^T(A_k + B_k)\mathbf{q} \geq \sum_{k \in \mathcal{K}} x_k (\mathbf{p}^k)^T(A_k + B_k)\mathbf{q}', \quad \forall \mathbf{q}' \in \mathcal{Q}.$$

$B_k$  is a matrix that holds the same values in every column, i.e. let  $\mathbf{b}_k := (\frac{\sum_{e' \in r} (w_{e'} - f_{e'})}{\gamma})_{r \in \mathcal{R}_k}$  be a vector of dimensions  $|\mathcal{R}_k|$ , then  $B_k = [\mathbf{b}_k \mathbf{b}_k \dots \mathbf{b}_k]$  and it is easy to see that

$$(\mathbf{p}^k)^T B_k \mathbf{q}' = (\mathbf{p}^k)^T \mathbf{b}_k \gamma, \quad \forall \mathbf{q}' \in \mathcal{Q}.$$

Therefore:

$$\begin{aligned}
 & \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T (A_k + B_k) \mathbf{q} \geq \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T (A_k + B_k) \mathbf{q}', \quad \forall \mathbf{q}' \in \mathcal{Q} \\
 \Leftrightarrow & \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T (A_k) \mathbf{q} + \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T \mathbf{b}_k \gamma \geq \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T A_k \mathbf{q}' + \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T \mathbf{b}_k \gamma, \quad \forall \mathbf{q}' \in \mathcal{Q} \\
 \Leftrightarrow & \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T (A_k) \mathbf{q} \geq \sum_{k \in \mathcal{K}} x_k(\mathbf{p}^k)^T A_k \mathbf{q}', \quad \forall \mathbf{q}' \in \mathcal{Q}
 \end{aligned}$$

I.e.  $\mathbf{q}$  is a best response of the inspectors to  $\mathbf{p}$  in spot-checking game  $G$ .

We presented an equivalence transformation, hence the transformation holds in both directions.  $\square$

Recall that in zero-sum games, all Nash equilibrium strategies are interchangeable, i.e. replacing a Nash strategy in a Nash equilibrium with any other Nash strategy also yields a Nash equilibrium. Furthermore, in zero-sum games, all Nash equilibria yield the same payoff for every player. The latter property cannot be transferred to the payoffs of several Nash equilibria in  $G$ , e.g. as we have seen in the last subsection 2.2.1 in example I.

For the best-response equivalence transformation of  $G$  and  $G'$ , we adopted a simple transformation presented in an article by Kannan and Theobald in 2010 (see section 2.1 in [KT10]) dealing with bimatrix games  $(A, B)$  where the sum of the matrices  $A$  and  $B$  has a fixed rank, and where polynomial algorithms for finding an  $\epsilon$ -approximation of a Nash equilibrium are presented.

In particular, our spot-checking game  $G$  belongs to the class of strategically zero-sum games introduced by Moulin and Vial in 1978 [MV78]. A bimatrix game  $(A, B)$  is strategically zero-sum if and only if there exist scalars  $w > 0$  and  $v > 0$  such that  $wA + vB = W + V$  where  $W$  is a matrix with equal columns and  $V$  is a matrix with equal rows. In our case, for the bimatrix games between the inspectors and player  $k$ , we have  $A = A_k$ ,  $B = -(A_k + B_k)$  and  $w = v = 1$ .  $B_k$  is a matrix with equal columns. The equivalence can be generalized straightforwardly to multi-player games (see statement in section 3.4 in [MV78]). Strategically zero-sum games are the games for which no completely<sup>2</sup> mixed Nash equilibrium can be improved upon. Every strategically zero-sum game is best-response equivalent to a zero-sum game (see example 5 and proof in [MV78]). But as we are not sure whether a completely mixed Nash equilibrium exists, we cannot be sure that a Nash equilibrium we compute will be optimal (optimal in terms of yielding the highest possible payoff for a Nash equilibrium profile). Furthermore, we do not know whether the optimal Nash equilibrium strategy corresponds to a Stackelberg strategy or at least to the best of all Nash strategies in a Stackelberg game.

We will now present two equivalent approaches to compute an inspectors' Nash strategy, one using user path variables and one using user flow variables.

<sup>2</sup>The support of the mixed strategies includes all pure strategies.

### 2.3.2 User-paths Nash LP and cutting-plane method

The above result yields that we can compute a Nash equilibrium for  $G$  by computing a Nash equilibrium for  $G'$ . We will formulate a linear program to calculate a Nash equilibrium from the inspectors player's point of view, using the well-known ideas of the Minmax theorem as proposed by Daskalakis and Papadimitriou [DP09].

First of all,  $\forall k \in \mathcal{K}$ , let  $A'_k := A_k + B_k$  denote the cost matrix of player  $k$  of the bimatrix game against the inspectors player, respectively,  $A'_k$  is the payoff matrix of the inspectors in  $G'$  from the bimatrix game against player  $k$ . Furthermore let  $\lambda_k := (\mathbf{p}^k)^T (A_k + B_k) \mathbf{q}$  denote the expected costs of player  $k$  (equivalent to the negative expected payoff of player  $k$ ).

In zero-sum game  $G'$ , when  $\mathbf{q}$  is fixed player  $k$ ,  $k \in \mathcal{K}$  solves

$$\min_{\mathbf{p}^k \in \Omega^k} (\mathbf{p}^k)^T A'_k \mathbf{q},$$

while the inspectors player when  $\mathbf{p}^k$  is fixed solves

$$\max_{q \in \mathcal{Q}} \sum_{k \in \mathcal{K}} x_k (\mathbf{p}^k)^T A'_k \mathbf{q}.$$

We can combine both problems to compute a Nash equilibrium, i.e. we solve

$$\max_{q \in \mathcal{Q}} \sum_{k \in \mathcal{K}} x_k \min_{\mathbf{p}^k \in \Omega^k} (\mathbf{p}^k)^T A'_k \mathbf{q}.$$

Since  $\mathbf{p}^k$  is a probability distribution, we have

$$\min_{\mathbf{p}^k \in \Omega^k} (\mathbf{p}^k)^T A'_k \mathbf{q} = \min_{r \in \mathcal{R}_k} (A'_k \mathbf{q})_r, \quad \forall k \in \mathcal{K}.$$

Hence the following LP determines an inspectors' Nash strategy for  $G'$ :

$$(P) \quad \begin{aligned} & \max_{\substack{\mathbf{q} \in \mathcal{Q} \\ \lambda_k \in \mathbb{R}}} \sum_{k \in \mathcal{K}} x_k \lambda_k \\ & \lambda_k \leq (A'_k \mathbf{q})_r, \quad \forall r \in \mathcal{R}_k, \forall k \in \mathcal{K} \end{aligned}$$

We will also refer to  $(P)$  as the *user-paths Nash LP* of game  $G$ . A corresponding Nash strategy for all player  $k$ ,  $k \in \mathcal{K}$ , can be determined from the optimal dual variables of the constraints in  $(P)$  (see e.g. [FT91]).

In contrast to [BOSS12], we have multiple OD-path options for every commodity  $k \in \mathcal{K}$  and, accordingly, the number of pure strategies for a player  $k$  has increased drastically, i.e.  $\mathcal{R}_k$  is very large. In a complete digraph the number of OD-paths is  $O(n!)$ . Even if in general a transportation network will not be complete, in many cases  $\mathcal{R}_k$  will be huge. Therefore,  $(P)$  has a very large number of constraints. Thus we need to find a suitable solving algorithm. A standard solving algorithm in such a case is a cutting-plane method.

We will apply a method proposed by Ford and Fulkerson in 1958 [FF58]. Here, an equivalent “arc-chain formulation” of the multi-commodity max-flow problem is solved “by replacing the pricing operation of the simplex method [...] with several applications of a combinatorial algorithm for finding a shortest chain joining a pair of points in the network”. Their argument is that “the enumeration of all chains from commodity sources to sinks in a network of moderate size would be a lengthy task, to say at least”.

At the beginning, we consider only subsets of paths for every commodity  $k$ , i.e. subsets  $\bar{\mathcal{R}}_k \subseteq \mathcal{R}_k$ , and receive the following restricted LP:

$$\begin{aligned}
 (\bar{P}) \quad & \max_{\substack{\mathbf{q} \in \mathcal{Q} \\ \lambda_k \in \mathbb{R}}} \sum_{k \in \mathcal{K}} x_k \lambda_k \\
 & \lambda_k \leq (A'_k \mathbf{q})_r, \quad \forall r \in \bar{\mathcal{R}}_k, \forall k \in \mathcal{K}
 \end{aligned}$$

The optimal value for an optimal restricted solution  $\bar{\lambda}_k, \bar{\mathbf{q}}$  of  $(\bar{P})$  is an upper bound for the optimal value of  $(P)$  (and we know that a feasible solution for  $(P)$  exists, e.g. the users may choose any of their OD-paths and the inspectors may randomly check arcs). Hence, if  $\bar{\lambda}_k, \bar{\mathbf{q}}$  is feasible for  $(P)$ , it is optimal. Now, as proposed in [FF58], instead of  $\forall k \in \mathcal{K}, \forall r \in \bar{\mathcal{R}}_k$ , checking inequality  $\lambda_k \leq (A'_k \bar{\mathbf{q}})_r$ , we solve our so-called *pricing problem*:  $\forall k \in \mathcal{K}$ , we check if the inequality is satisfied for  $\arg \min_{r \in \bar{\mathcal{R}}_k} (A'_k \bar{\mathbf{q}})_r$

by solving a shortest-path problem. For this purpose we construct a copy  $\bar{D}$  of network  $D$  with modified weights  $\sigma_e \bar{q}_e + w_e$  on every arc  $e \in E$ . In  $\bar{D}$  the length of path  $r \in \bar{\mathcal{R}}_k$  equals the value of  $(A'_k \bar{\mathbf{q}})_r$ . All weights are positive, so the shortest paths can easily be computed (e.g. using one of the methods proposed in [FF58] or simply Dijkstra’s algorithm that was published one year later in 1959 [Dij59]). If the shortest paths satisfy the inequalities, all inequalities are satisfied and we have optimality, otherwise we add the *cutting planes* found (i.e. an inequality for every shortest path that did not satisfy the inequality) to the restricted LP and repeat the procedure, see Algorithm 1 below.

---

**Algorithm 1** Cutting-plane method for user-paths Nash LP  $(P)$

---

- 1: Initialize  $\bar{\mathcal{R}}_k, \forall k \in \mathcal{K}$ .
  - 2: **repeat**
  - 3:     Solve  $(\bar{P})$ .
  - 4:     **for**  $k \in \mathcal{K}$  **do**
  - 5:         Create  $\bar{D}$  for computed solution  $\bar{\mathbf{q}}$  of  $(\bar{P})$ .
  - 6:         Determine the shortest path  $r^*$  in  $\bar{D}$ .
  - 7:         **if**  $\bar{\lambda}_k > r^*$  **then**
  - 8:             Add  $r^*$  to  $\bar{\mathcal{R}}_k$ .
  - 9:         **end if**
  - 10:     **end for**
  - 11: **until** No path has been added to  $(\bar{P})$ .
- 

We point out that we can compute shortest paths simultaneously for commodities with same same origin, which reduces the computation time.



### 2.3.3 User-flow Nash LP

The above approach using path variables solves our problem. But what could be a different, a better modeling approach? An approach that is more compact, with less variables, and for which we could get faster results? In other words, we are looking for an efficient representation of the shortest paths of the network users w.r.t. the costs for taking a path, when  $\mathbf{q}$  is fixed, as those correspond to the best responses of player  $k$  to  $\mathbf{q}$  (see proposition 2.2.1). A shortest path in a digraph w.r.t. positive weights can always be found by solving a special min-cost flow problem with a simple linear program as, for example, presented in [AMO93]. In the following, instead of using one separate variable for every path and commodity, we will use a compact flow representation of the paths. By doing so, we drastically reduce the number of variables. Here we may use a standard LP solver instead of a cutting-plane method as discussed above.

An application of the following user-flow problem formulations to compute Nash and Stackelberg strategies for the inspectors was also presented at the INOC Conference 2013 in a corporate work together with Borndörfer et al. [BBSS12]. This particular application will be presented in this thesis in section 3.2.

Let us see how the strategy set of player  $k \in \mathcal{K}$  and the payoffs when using a flow formulation change.

**Player  $k$ 's strategy** In the user-flow formulation of  $G$ , a mixed strategy  $\mathbf{p}^k$  of player  $k$  is given by flow variables  $0 \leq p_e^k \leq 1, \forall e \in E, \forall k = (o, d) \in \mathcal{K}$ , satisfying the following flow conservation:

$$\sum_{e \in \delta^-(v)} p_e^k - \sum_{e \in \delta^+(v)} p_e^k = \begin{cases} -1 & \text{if } v = o, \\ 1 & \text{if } v = d, \\ 0 & \text{else if } v \in V. \end{cases}$$

The flow variables  $p_e^k$  corresponds to the probability that player  $k \in \mathcal{K}$  uses an arc  $e \in E$ .

The expected payoff of player  $k, k \in \mathcal{K}$ , is now given by

$$\pi_k(\mathbf{p}, \mathbf{q}) = \pi_k(\mathbf{p}^k, \mathbf{q}) = -\lambda_k = -\sum_{e \in E} p_e^k (\sigma_e q_e + w_e).$$

The expected payoff of the inspectors in spot-checking game  $G$  is

$$\pi_c(\mathbf{p}, \mathbf{q}) = \sum_{k \in \mathcal{K}} x_k \sum_{e \in E} p_e^k (\sigma_e q_e + f_e),$$

and the expected payoff in the best-response equivalent game  $G'$  is

$$\pi'_c(\mathbf{p}, \mathbf{q}) = \sum_{k \in \mathcal{K}} x_k \lambda_k = \sum_{k \in \mathcal{K}} x_k \sum_{e \in E} p_e^k (\sigma_e q_e + w_e).$$

Taking a closer look at the inspectors' payoffs in  $G$  and in  $G'$  we see that the second term in the parenthesis, i.e.  $f_e$  in  $G$  and  $w_e$  in  $G'$ , does not depend on the value of  $q_e$ ,  $\forall e \in E$ . Following the concept of the best-response equivalence transformation presented in subsection 2.3.1, we can conclude that  $G$  and  $G'$  are best-response equivalent.

We use the following modified linear program formulation, which we will call *Nash LP*, to compute a Nash strategy  $\mathbf{q}$  for the inspectors:

Let  $O := \{o : \exists k = (o, d) \in \mathcal{K}\} \subseteq V$  be the set of origins of commodities  $k \in \mathcal{K}$ . The set of origins  $O$  will be used again to reduce our number of variables: because of the optimal substructure of shortest paths, we can use a single-source (i.e. "single-origin") shortest path problem formulation for all players starting at the same origin instead of solving the shortest path problem for every commodity. The Nash LP is given by

$$\max_{\mathbf{q}, \mathbf{y}, \lambda} \quad \sum_{k \in \mathcal{K}} x_k \lambda_k \quad (2.4a)$$

$$y_v^o - y_u^o \leq \sigma_{(u,v)} q_{(u,v)} + w_{(u,v)}, \quad \forall o \in O, \forall (u,v) \in E \quad (2.4b)$$

$$y_o^o = 0, \quad \forall o \in O \quad (2.4c)$$

$$\lambda_k = y_d^o, \quad \forall k \in \mathcal{K}, k = (o, d) \quad (2.4d)$$

$$\mathbf{q} \in \mathcal{Q} \quad (2.4e)$$

(2.4b)-(2.4c) form the single-source shortest path linear program formulation for the users, where the variables  $y_v^o$  are the so-called node potentials which are bounded from above by the length of a shortest path from the origin  $o$  to any node  $v \in V$ . The length of a shortest path for commodity  $k$  corresponds to the costs  $\lambda_k$  (2.4d). (2.4e) ensures a feasible marginal inspectors' strategy  $\mathbf{q}$ .

Note that the optimal dual variables of constraint (2.4b) define a flow in  $D$  from which Nash strategies for the player  $k$ ,  $k \in \mathcal{K}$ , can be inferred.

### 2.3.4 Computation of the inspectors' payoff in a Stackelberg game

To compute the payoff of the inspectors in a Stackelberg game when playing with Nash strategy  $\mathbf{q}$ , we have to determine a best response to  $\mathbf{q}$  for all player  $k$ ,  $k \in \mathcal{K}$ , which favors the inspectors the most (see definition of  $v_{stack}$  in section 2.2). Let  $D_{SP_k(\mathbf{q})}$  denote the shortest path arborescence w.r.t. the costs for a player  $k$  when  $\mathbf{q}$  is given.  $D_{SP_k(\mathbf{q})}$  can be inferred from the saturated inequalities of the user-paths Nash LP for the optimal solution  $\mathbf{q}$  (and corresponding  $\lambda$ ) or respectively from the saturated inequalities of the constraints (2.4b) of the user-flow Nash LP for the optimal solution. As the user's costs for a given  $\mathbf{q}$  on every arc  $e \in E$  of the transportation network  $D$  are non-negative,  $D_{SP_k(\mathbf{q})}$  is a acyclic directed graph, i.e. it contains no cycles (or cycles of weight zero). The reason for this is that traveling on a cycle of positive weight would always increase the costs of a user and cannot be part of a shortest path. Now, to determine a best response to  $\mathbf{q}$  of

player  $k$  that favors the leader the most, we look for a longest path in  $D_{SP_k(\mathbf{q})}$  w.r.t. the inspectors' payoff from rewards and fines. We do so by assigning non-positive weight  $-(f_e + \sigma_e q_e)$  to every arc  $e$  that is in the arc set of  $D_{SP_k(\mathbf{q})}$ .  $D_{SP_k(\mathbf{q})}$  with non-positive weights  $-(f_e + \sigma_e q_e)$  has no negative cycles and we can compute a shortest path of commodity  $k$ , e.g. using the Bellman–Ford–Moore algorithm (published by Bellman in 1958 [Bel58], Ford in 1956 [For56] and Moore in 1959 [Moo59] and equally credited to all of them). Such a shortest path w.r.t. non-positive weights  $-(f_e + \sigma_e q_e)$  in  $D_{SP_k(\mathbf{q})}$  corresponds to a longest path w.r.t. non-negative weights  $(f_e + \sigma_e q_e)$  in  $D_{SP_k(\mathbf{q})}$ , which is a best response of player  $k$  in favor of the inspectors. The length of such a longest path is the inspectors' payoff from the partial game played against player  $k$  in the Stackelberg game. To compute the total inspectors' payoff for Nash strategy  $\mathbf{q}$  in a Stackelberg game, we have to determine the lengths of such a longest path for every  $k \in \mathcal{K}$  and then compute their sum over  $k \in \mathcal{K}$ .

## 2.4 Computation of an inspectors' Stackelberg equilibrium strategy

To compute a Stackelberg equilibrium of  $G$ , we will use a simplified complementary slackness condition to integrate an optimality condition for the follower in the optimization problem of the leader similar as done in a paper by Paruchuri et al. in 2008 [PPM<sup>+</sup>08]. In the latter paper, a mixed integer problem (MIP) is presented to compute a Stackelberg equilibrium for a Bayesian Stackelberg game<sup>3</sup>, optional computation method can be found for example in [CS06]. The MIP formulation was developed to efficiently solve the related problem of randomizing patrols and selecting checkpoints in a strategical way to protect the Los Angeles International Airport towards adversaries. That problem was also modeled using game theory.

Now let us consider a user-flow formulation. To formulate a complementary slackness condition, let us explicitly write down the primal and dual linear program of player  $k = (o, d)$ ,  $k \in \mathcal{K}$ , to compute a best response w.r.t.  $\mathbf{q}$ . The primal LP is given by

$$\min_{\mathbf{p}^k} \sum_{e \in E} (\sigma_e q_e + w_e) p_e^k \quad (2.5a)$$

$$\sum_{e \in \delta^-(v)} p_e^k - \sum_{e \in \delta^+(v)} p_e^k = \begin{cases} -1 & \text{if } v = o \\ 1 & \text{if } v = d \\ 0 & \text{else if } v \in V \end{cases} \quad (2.5b)$$

$$p_e^k \geq 0, \quad \forall e \in E \quad (2.5c)$$

The dual LP corresponds to the shortest path problem formulation using node po-

<sup>3</sup>A Bayesian Stackelberg game is a Stackelberg game between a leader and a follower of unknown type. Each type of follower has its own pure strategy set. A probability distribution over possible types of followers is given and used to optimize the expected payoff of the leader. Our Stackelberg game with several followers and a concrete distribution over those followers (given by the demand for the commodities) can be transformed into such a Bayesian Stackelberg game.

tentials  $y_v^k$  as presented in Nash LP 2.4, i.e. player  $k$  is solving

$$\max_{\mathbf{y}^k} y_d^k \quad (2.6a)$$

$$y_v^k - y_u^k \leq \sigma_{(u,v)} q_{(u,v)} + w_{(u,v)}, \quad \forall (u,v) \in E \quad (2.6b)$$

$$y_o^k = 0 \quad (2.6c)$$

If primal and dual LP have a feasible solution, then an optimality criterion for the solutions is given by the complementary slackness theorem (see e.g. [Sch98]), i.e. if  $\mathbf{p}^k$  is feasible for the primal LP 2.5 and  $\mathbf{y}^k$  is feasible for the dual LP 2.6, then they are optimal for their corresponding programs if and only if

$$(\sigma_{(u,v)} q_{(u,v)} + w_{(u,v)} - (y_v^k - y_u^k)) p_{(u,v)}^k = 0, \quad \forall (u,v) \in E.$$

To integrate the best response of player  $k$  to  $\mathbf{q}$  in a Stackelberg game into the optimization problem solved by the inspectors, we simplify the complementary slackness condition by only considering one optimal pure strategy of player  $k$  which favors the inspectors the most (see definition of a Stackelberg equilibrium in section 2.2). This is realized by using a formulation with a big constant  $M$ . A further specialty of our problem formulation will be a single-source-multi-sink formulation of the user flow, which will again reduce the number of variables. For every origin  $o \in O$ , we define  $\mathfrak{D}_o := \{d : (o,d) \in \mathcal{K}\}$ , its set of destinations. The inspectors' Stackelberg equilibrium strategy  $\mathbf{q}$  in spot-checking game  $G$  is an optimal solution of the following MIP:

$$\max_{\mathbf{q}, \mathbf{y}, \lambda, \mu, \mathbf{p}} \sum_{k \in \mathcal{K}} x_k \lambda_k - \sum_{o \in O} \sum_{e \in E} p_e^o w_e + \sum_{o \in O} \sum_{e \in E} p_e^o f_e \quad (2.7a)$$

$$0 \leq \sigma_{(u,v)} q_{(u,v)} + w_{(u,v)} - (y_v^o - y_u^o) \leq M(1 - \mu_{(u,v)}^o), \quad \forall o \in O, \forall (u,v) \in E \quad (2.7b)$$

$$y_o^o = 0, \quad \forall o \in O \quad (2.7c)$$

$$\lambda_k = y_d^o, \quad \forall k \in \mathcal{K}, k = (o,d) \quad (2.7d)$$

$$\sum_{e \in \delta^-(v)} p_e^o - \sum_{e \in \delta^+(v)} p_e^o = \begin{cases} - \sum_{d \in \mathfrak{D}_o} x_{(o,d)} & \text{if } v = o, \\ x_{(o,v)} & \text{if } v \in \mathfrak{D}_o, \\ 0 & \text{else,} \end{cases} \quad \forall v \in V, \forall o \in O \quad (2.7e)$$

$$0 \leq p_e^o \leq M \mu_e^o, \quad \forall o \in O, \forall e \in E \quad (2.7f)$$

$$\mu_e^o \in \{0, 1\}, \quad \forall o \in O, \forall e \in E \quad (2.7g)$$

$$\mathbf{q} \in \mathcal{Q} \quad (2.7h)$$

To avoid a quadratic term in the objective of the leader (2.7a), we do not use the explicit change of variables as proposed in [PPM<sup>+</sup>08], but represent the payoff of the inspectors in linear terms with help of the previously defined costs  $\lambda_k$  of player  $k$ . In (2.7e) a feasible user-flow for the player  $k$ ,  $k \in \mathcal{K}$ , is defined as single-source-multi-sink flow problem for every origin  $o \in O$ . We introduce binary variables  $\mu_e^o$  (2.7g), which indicate whether an arc belongs to a shortest path arborescence rooted in origin  $o$  and a big constant  $M$  that helps us to choose an optimal pure strategy for the player  $k$ ,  $k \in \mathcal{K}$ . (2.7b) and (2.7f) represent the simplified complementary slackness conditions and together with (2.7c) form the single-source shortest path formulation for the player  $k$ ,  $k \in \mathcal{K}$ , where the length of a shortest path for commodity  $k$

corresponds to the costs  $\lambda_k$  (2.7d). (2.7h) ensures a feasible marginal inspectors' strategy  $\mathbf{q}$ .

We will now present some experimental results for the spot-checking game on an application on German motorways.



# Chapter 3

## Application on German motorways

We apply the established spot-checking game to the transportation network of German motorways. The users of the network are the truck drivers which are legally obliged to pay a toll. Instead of paying the toll, a cheaper option might be to evade by taking a detour to avoid inspectors. The aim is to optimize the allocation of inspectors on motorways to enforce the payment of the toll. There are several ways for the application to represent the German motorway network. In the following we present two approaches, a two-level network in which the users can switch between paying and evading during their trip and a single-pay-path network, where users take either the shortest path when paying the toll or evade on some route in the motorway network. Afterward, we consider inspections regarding space and time, i.e. we consider duties for the inspectors of a fixed length that are distributed over a time horizon. A transition from one spot in the motorway network to another within an inspection duty is only possible if those spots are connected.

In a spot checking game applied to German motorways, a transportation network corresponds to an extended representation of a simplified real-world motorway network. A simplified real-world German motorway network, respectively one of its subregions, is given by a digraph holding toll liable arcs and toll-free trunk roads. An arc represents a part of a motorway from one entry, turning or exit point to another. For experimental tests, lengths and connections of toll liable arcs corresponding to the topology of the real German motorway network as well as real traffic data were available. The number of nodes, arcs and commodities was reduced for the sake of simplicity. Almost all free trunk roads were generated artificially, because the data was not available, by creating a certain number arcs from one node to another node when their geographical distance was between 30 to 50 km.

In section 3.1, we present the most advanced model to represent a transportation network in which users can decide on which parts of their trip they are paying toll and which parts they are evading. We then present a model in 3.2 with a reduced number of arcs, where users have exactly one path on which they can pay the toll and other paths on which they evade. The model presented in 3.3 extends the model in 3.1 integrating spatial and time dependencies.



Figure 3.1: Geographical map of Germany (source: OpenStreetMap).

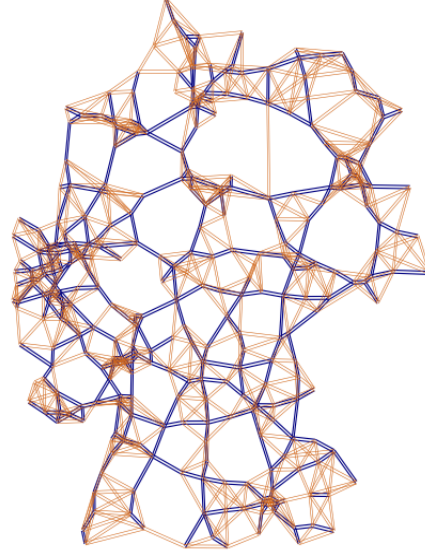


Figure 3.2: Simplified motorway network with free trunk roads (orange).

### 3.1 Two-level transportation network

To realize a transportation network in which drivers can pay the toll for one part of their trip and evade on the other part, we construct a digraph  $D = (V, E)$  which we call *two-level network*, consisting of two copies of the simplified real-world motorway network: one representing the network in which drivers pay the toll and one in which drivers evade. Both copies are connected by so-called transition arcs. For every driver's origin, we add an artificial origin node with arcs without costs to the corresponding nodes in the copies, respectively we add an artificial destination node to be the head of free arcs from its corresponding nodes in the copies.

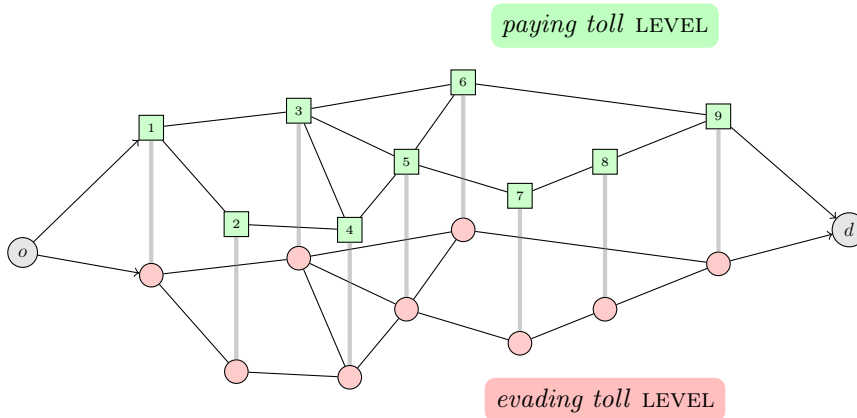


Figure 3.3: Sketch of an excerpt of two-level network  $D$



I.e., the arc set  $E$  is partitioned into the set  $E_p$  of arcs on which drivers are paying the toll, the set  $E_c$  of arcs on which drivers evade the toll, the set  $E_d$  of toll-free trunk roads and a set  $E_u$  representing transitions between a paid arc and an evaded arc and vice versa. In order to simplify matters, the cost-free arcs connecting the origin and destination nodes to the two levels have lengths equal to zero and belong to the set  $E_d$ .

The users of the network, the truck drivers, are distributed over a set of commodities  $\mathcal{K}$  in  $D$ . The demand for a commodity  $k$ , i.e. the number of drivers who make a trip on  $k$ , during a given period of time is known and denoted by  $x_k$ .  $\mathcal{R}_k$  denotes the set of all OD-paths of a driver on a commodity  $k$ .

The distance-based toll fare per km is  $f$  and the fine<sup>1</sup> for evading the toll if being inspected is  $P$ . The number of inspectors in  $D$  is limited by a capacity  $\gamma$ .  $q_e$  denotes the probability that an inspector is present on  $e \in E$  and  $\sigma_e$  is the probability to be inspected when an inspector is present. In the application, the variables  $\sigma_e$  of chapter 2 are replaced by  $\sigma_e P$  for all  $e \in E$ .

The driver's costs excluding fines for using an arc  $e$  are given by

$$w_e = \begin{cases} b l_e & \forall e \in E_c \\ (b + f) l_e & \forall e \in E_p \\ (b + a) l_e & \forall e \in E_d \\ \theta & \forall e \in E_u \end{cases}$$

with

- $l_e$  : length of arc  $e \in E$
- $b$  : basic costs per km (fuel, salary, etc.)
- $a$  : additional costs per km for taking a trunk road
- $\theta$  : costs for a switch between paying and evading

The driver's total expect costs for using an arc  $e \in E$  are

$$\sigma_e q_e P + w_e.$$

A justification for using  $\sigma_e$  as an input parameter is given in appendix B.

Whenever a driver switches from paying to evading (or evading to paying) he has to take a transition arc and pays an additional transition cost, as we assume switching would be an organizational expenditure. These transition costs are hard to determine in reality, but in theory it can give an idea of how far these influence the behavior of the drivers. The two-level network gives furthermore a nice representation of parallel behavior on a single road as we will see later on in the experimental results.

<sup>1</sup> In Germany the amount of the fine for evading varies (on average between 200 and 400 Euro). In addition to the fine, a driver caught evading would have to pay the toll fare for his trip. Instead we use the simplified assumption that the fine is a constant.

**Inspectors' strategy** Inspections can be conducted on all arcs  $e \in E_c$ . The inspectors do not have an interest to place more inspectors in the network than needed to maximize their payoff, hence the pure strategy set of the inspectors corresponds all subsets of  $E_c$  of cardinality smaller or equal to  $\gamma$ . As discussed in the last chapter (see section 2.1) our model will only depend on the marginal probabilities  $q_e$  that an inspector is present on arc  $e$ . The set of marginal probabilities is given by

$$\mathcal{Q} := \{\mathbf{q} \in [0, 1]^{|E_c|} : \sum_{e \in E_c} q_e \leq \gamma\}.$$

To simplify the notation, we introduce constants  $q_e = 0$  for all arcs  $e \in E \setminus E_c$  where no inspections will be conducted.

**Player  $k$ 's strategy** A driver on commodity  $k$  is free to choose a OD-path  $r \in \mathcal{R}_k$  through the two-level network  $D$ . By choosing his paths, he chooses which portions of the real motorway network where he will be evading the toll and which portions where he will be paying. In addition, he is allowed to take toll-free trunk roads. Switching between evading and paying during the trip will be coupled with additional transition costs as explained above. The set of pure strategies of player  $k$  consists of all paths  $r \in \mathcal{R}_k$ . Using the user-paths formulation introduced in section 2.3 of the last chapter, the set of mixed strategies of player  $k$  is given by

$$\Omega^k := \{\mathbf{p}^k \in [0, 1]^{|\mathcal{R}_k|} : \sum_{r \in \mathcal{R}_k} p_r^k = 1\}.$$

$p_r^k$  may also be interpreted as the proportion of drivers associated with commodity  $k \in \mathcal{K}$  that are using path  $r \in \mathcal{R}_k$ .

In the user-flow formulation mixed strategy  $\mathbf{p}^k \in \Omega^k$ ,  $\forall k \in \mathcal{K}$ , is a flow in two-level-network  $D$  with  $0 \leq p_e^k \leq 1$ ,  $\forall e \in E, \forall k \in \mathcal{K}$ , satisfying the following flow conservation  $\forall v \in V, \forall k = (o, d) \in \mathcal{K}$ :

$$\sum_{e \in \delta^-(v)} p_e^k - \sum_{e \in \delta^+(v)} p_e^k = \begin{cases} -1 & \text{if } v = o, \\ 1 & \text{if } v = d, \\ 0 & \text{else.} \end{cases}$$

$p_e^k$  is the probability that a driver associated with commodity  $k \in \mathcal{K}$  is using an arc  $e \in E$ .

Recall that a best response of player  $k$  to the inspectors' strategy  $\mathbf{q}$  is to take a shortest OD-path with respect to arcs weights  $\sigma_e q_e P + w_e$ ,  $\forall e \in E$  (see proposition 2.2.1).

**Scenario** We assume that every driver wants to minimize his total expected costs<sup>2</sup>. For the inspectors, we assume that they are trying to maximize the profits they

---

<sup>2</sup> We do not consider other drivers' behavior (for example being honest and always paying the toll). In order to enforce the payment of a toll by means of mobile inspection units, users which strategically minimize their total costs are the incentive of inspections.

can gain from toll fares and fines. We will call this scenario MAXPROFIT. This objective might not be equivalent to enforcing payment of tolls. In this section where we use a two-level network to represent the transportation network, we will limit our attention to the latter scenario. In the next section 3.2, we will introduce a different transportation network, the *single-pay-path-network*. The topology of that network will be easier and we will consider further scenarios, i.e. a scenario MAXTOLL, where the inspectors' objective is to maximize the profit gained from toll fares and a scenario MAXPAYERS, where they wish to minimize the number of users evading tolls.

The approximate probability to be inspected on arc  $e \in E$  is given by  $\sigma_e q_e$  and the expected number of times player  $k$  is subject to inspections on path  $r \in \mathcal{R}_k$  is  $\sum_{e \in r} \sigma_e q_e$ . In general, we will have a model, where a driver can be fined several times for being caught evading. Experimental results of previous a previous work [BOSS12] suggest, that the risk to be caught evading more than once is very small. Hence, we will approximate the probability  $1 - \prod_{e \in r} (1 - \sigma_e q_e)$  to be inspected on  $r$  with  $\sum_{e \in r} \sigma_e q_e$ .

**User-paths formulation** In the user-paths formulation the expected payoff of player  $k$  is given by

$$-\lambda_k := - \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (\sigma_e q_e P + w_e),$$

the expected payoff of the inspectors player is

$$\sum_{k \in \mathcal{K}} x_k \sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} \sigma_e q_e P + \sum_{e \in r \cap E_p} f l_e \right),$$

and the expected payoff of the inspectors' in the best-response equivalent zero-sum game is

$$\sum_{k \in \mathcal{K}} x_k \sum_{r \in \mathcal{R}_k} p_r^k \left( \sum_{e \in r} \sigma_e q_e P + \sum_{e \in r} w_e \right) = \sum_{k \in \mathcal{K}} x_k \lambda_k.$$

The user-paths Nash LP, which we will solve using the cutting-plane method presented in 2.3.2, is given by

$$\max_{\mathbf{q}, \lambda} \quad \sum_{k \in \mathcal{K}} x_k \lambda_k \tag{3.1a}$$

$$\lambda_k \leq \sum_{e \in r} (\sigma_e q_e P + w_e), \quad \forall r \in \mathcal{R}_k, \forall k \in \mathcal{K} \tag{3.1b}$$

$$0 \leq q_e \leq 1 \quad \forall e \in E_c \tag{3.1c}$$

$$\sum_{e \in E_c} q_e \leq \gamma \tag{3.1d}$$

Recall that we have defined constants  $q_e = 0, \forall e \in E \setminus E_c$ .

**User-flow formulation** Following previous thoughts, we approximate the probability to be inspected in the user-flow formulation by  $\sum_{e \in E} \sigma_e q_e p_e^k$ . The expected payoff for player  $k$  is given by

$$-\lambda_k := - \sum_{e \in E} (\sigma_e q_e P + w_e) p_e^k.$$

Integrating the flow representation of player  $k$ 's strategies, the expected payoff of the inspectors is given by

$$\sum_{k \in \mathcal{K}} x_k \left( \sum_{e \in E} \sigma_e q_e P p_e^k + \sum_{e \in E_p} f l_e p_e^k \right),$$

and the inspectors' payoff in the best-response equivalent zero-sum game is

$$\sum_{k \in \mathcal{K}} x_k \lambda_k.$$

Let  $O := \{o : \exists k = (o, d) \in \mathcal{K}\} \subseteq V$  be the set of origins for commodity in  $\mathcal{K}$ . According to the construction presented in subsection 2.3.3, we may use the following user-flow Nash LP to compute an inspectors' Nash strategy:

$$\max_{\mathbf{q}, \mathbf{y}, \lambda} \quad \sum_{k \in \mathcal{K}} x_k \lambda_k \tag{3.2a}$$

$$y_v^o - y_u^o \leq \sigma_{(u,v)} q_{(u,v)} P + w_{(u,v)}, \quad \forall o \in O, \forall (u, v) \in E \tag{3.2b}$$

$$y_o^o = 0, \quad \forall o \in O \tag{3.2c}$$

$$\lambda_k = y_d^o, \quad \forall k \in \mathcal{K}, k = (o, d) \tag{3.2d}$$

$$0 \leq q_e \leq 1, \quad \forall e \in E_c \tag{3.2e}$$

$$\sum_{e \in E_c} q_e \leq \gamma \tag{3.2f}$$

Again,  $q_e = 0, \forall e \in E \setminus E_c$ , is a constant.

To use the single-source-multi-sink formulation to compute an inspectors' Stackelberg strategy presented in section 2.4, let furthermore  $\mathfrak{D}_o := \{d : (o, d) \in \mathcal{K}\}$  be the set of destinations to an origin  $o$  in two-level network  $D$ . The Stackelberg MIP is given by:

$$\max_{\lambda, \mathbf{y}, \mathbf{q}, \mathbf{p}, \mu} \sum_{k \in \mathcal{K}} x_k \lambda_k - \sum_{o \in \mathcal{O}} \sum_{e \in E} p_e^o w_e + \sum_{o \in \mathcal{O}} \sum_{e \in E} p_e^k f_e \quad (3.3a)$$

$$0 \leq \sigma_{(u,v)} q_{(u,v)} P + w_{(u,v)} - (y_v^o - y_u^o) \leq M(1 - \mu_{(u,v)}^o), \quad \forall o \in \mathcal{O}, \forall (u,v) \in E \quad (3.3b)$$

$$y_o^o = 0, \quad \forall o \in \mathcal{O} \quad (3.3c)$$

$$\lambda_k = y_d^o, \quad \forall k \in \mathcal{K}, k = (o, d) \quad (3.3d)$$

$$\sum_{e \in \delta^-(v)} p_e^o - \sum_{e \in \delta^+(v)} p_e^o = \begin{cases} - \sum_{d \in \mathfrak{D}_o} x_{(o,d)} & \text{if } v = o \\ x_{(o,v)} & \text{if } v \in \mathfrak{D}_o \\ 0 & \text{else} \end{cases} \quad \forall v \in V, o \in \mathcal{O} \quad (3.3e)$$

$$0 \leq p_e^o \leq M \mu_e^o, \quad \forall o \in \mathcal{O}, \forall e \in E \quad (3.3f)$$

$$\mu_e^o \in \{0, 1\}, \quad \forall o \in \mathcal{O}, \forall e \in E \quad (3.3g)$$

$$0 \leq q_e \leq 1, \quad \forall e \in E_c \quad (3.3h)$$

$$\sum_{e \in E_c} q_e \leq \gamma \quad (3.3i)$$

### 3.1.1 Experimental results I

For experimental tests we employ real traffic data. Implementations were done in the Python programming language and all computations were made on a PC with 4 processors at 3 GHz using the CPLEX solver. The 3-dimensional visualization was done using MayaVi. As said at the beginning of this chapter, the motorway networks of Germany and its subregions presented in the following correspond to simplified depictions of the real world. The toll-free trunk roads were generated artificially. We do not consider any costs for conducting an inspection and in the following set the basic costs  $b$  for the truck drivers per km (fuel, salary, etc.) constantly to one.

We will choose values for input parameters like the toll fare per km and the fine according to average values in the real world. The probability to be inspected when an inspector is present corresponds to an estimated approximation. We solve instances of different sizes and topologies and compare results and solving approaches with respect to computation time, number of inspectors deployed and the inspectors' payoff. As e.g. explained in the report about traffic diversion on behalf of the German Government [Bun09] mentioned in chapter 1, we can not be sure what price reflects the additional costs for taking a trunk road as well as we do not know the price for switching between evading and paying. We will choose values and with respect to those values check to what extent the number of inspectors deployed influences the usage of trunk roads.

#### Results for the subregion Sachsen-Anhalt

We will start with experimental results for the German subregion Sachsen-Anhalt. The simplified motorway network and the corresponding two-level network have the following properties:

**Motorway network**

number of nodes	: 26
total number of arcs	: 74
number of trunk road arcs	: 20
number of commodities	: 192
number of truck drivers	: 342,095

**Two-level network**

number of nodes	: 104
number of arcs	: 304

We create instance *ST-6* of Sachsen-Anhalt, with the following input parameters:

$f$ (toll per km)	: 0.176
$a$ (additional trunk road costs per km)	: 0.4
$P$ (fine)	: 200
$\theta$ (costs for switch between paying and evading)	: 0.01
$\sigma_e, \forall e \in E$ (prob. to be inspected when insp. present)	: 0.15
$\gamma$ (number of inspectors)	: 6

User-paths Nash LP and user-flow Nash LP yield the same result for an inspectors' Nash strategy  $\mathbf{q}$ . The level of control on every arc of the motorway network is given in the following figure 3.4. Parallel arcs correspond to opposite direction on the same motorway. The trunk roads are drawn as dotted lines in gray color. Red indicates a high control level, i.e. a high probability that an inspector is present on an arc. The highest control level on an arc is about 0.18. Blue indicates a low control rate.

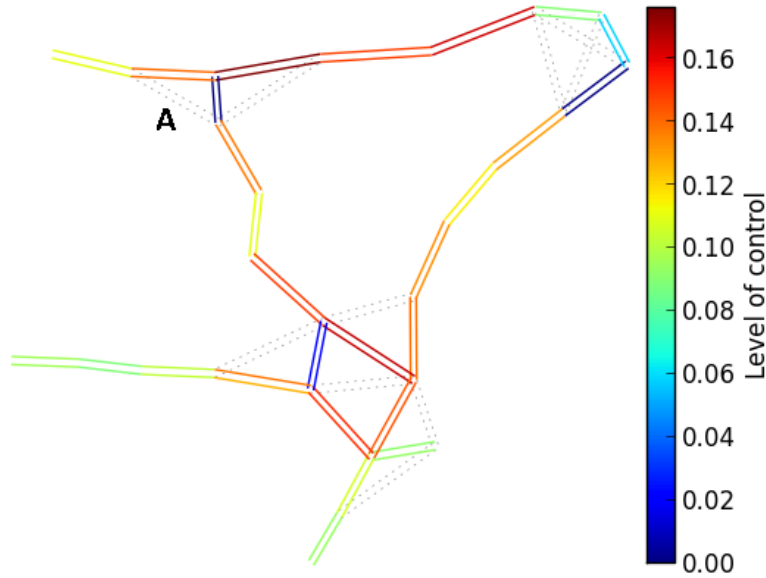


Figure 3.4: Inspectors' Nash strategy for instance *ST-6*

The difference between the user-paths and user-flow formulation lies in computation time of a solution. We initialized the cutting-plane method to solve the user-paths LP considering the shortest path in the paid level of the two-level network and the

shortest path in the evaded level when no inspectors are present. While the cutting-plane method needs 5 iterations and 19 seconds to solve the paths LP, the flow LP can be solved in 7 seconds.

The Stackelberg strategy is almost identical to the computed Nash strategy with small differences between 0.044 and 0.001 on some arcs, while the computation time to solve the Stackelberg MIP of 75 seconds is much longer than the computation time to compute a Nash strategy. The inspectors' profit from toll and fines when playing with the computed Nash strategy is 4,869,105 corresponding to 99,3% (-34,333) of the Stackelberg equilibrium profit of 4,903,438. The computed proportion of driven km in the toll liable network on which the toll was paid is 64%. Figure 3.5 shows the drivers flow in the computed Stackelberg equilibrium for instance *ST-6*.

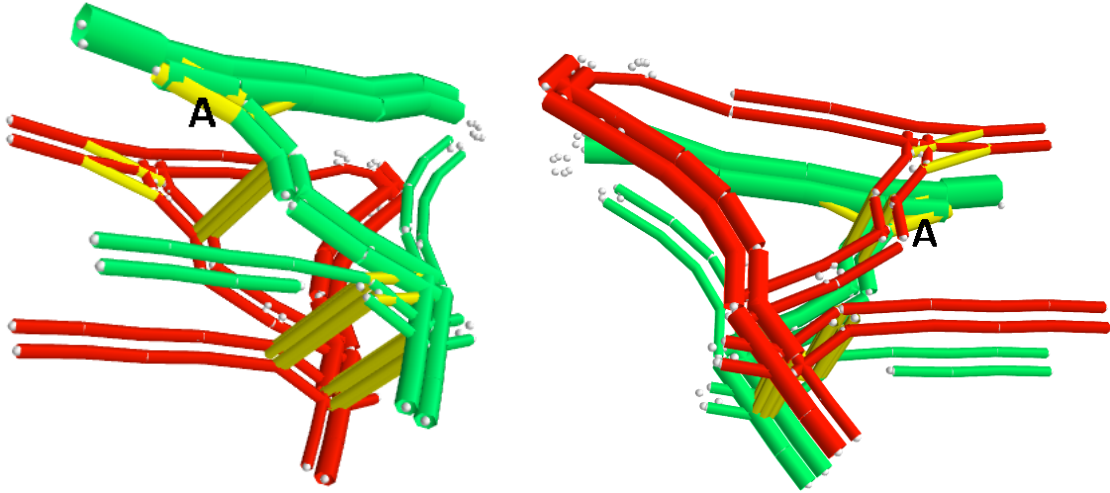


Figure 3.5: Drivers flow in Stackelberg equilibrium of instance *ST-6*

The width of the tubes in figure 3.5 give an impression of the distribution of drivers. The greater the width of an arc, the more drivers are using this arc. The maximal number of drivers on an arc in figure 3.5 is 55,165. Again, parallel arcs correspond to opposite direction on the same motorway. Red arcs represent arcs on which drivers evade the toll and green arcs represent arcs on which drivers pay the toll. Yellow arcs are either toll-free trunk roads or transition arcs between the paid level and the evaded level of the two-level network. Note that a driver may have several best responses in favor of the inspector, the paid and evaded km in a best response may vary. Figure 3.5 shows one option for best responses in favor of the inspectors. Recall that the objective of the inspectors in this game setting was to maximize their profit from toll and fines. As a result of chapter 2, this profit is bounded from above by the sum over all users of the costs when taking a route which is not evaded. The more inspectors, the closer the profit will be to this bound, where one best response to the inspectors strategy corresponds to not evading the toll. The trunk roads in our model often correspond to geographical short cuts. But we assume that the additional costs per km for taking a trunk road are much higher than the toll fare.

At label A in figure 3.4 and figure 3.5, we see that it is a cheaper option to take a trunk road than paying the toll or evading. Other yellow arcs which are connecting red and green arcs represent drivers switching from paying to evading and vice versa.

We have solved instances with the same input values as *ST-6* but different numbers of inspectors  $\gamma$ . We compared the profit gained in a Stackelberg game when playing with a Stackelberg strategy (STACK) to playing with a Nash strategy (NASH) and to playing with a strategy chosen proportional to a recorded traffic distribution (PROP). The result is presented in figure 3.6.

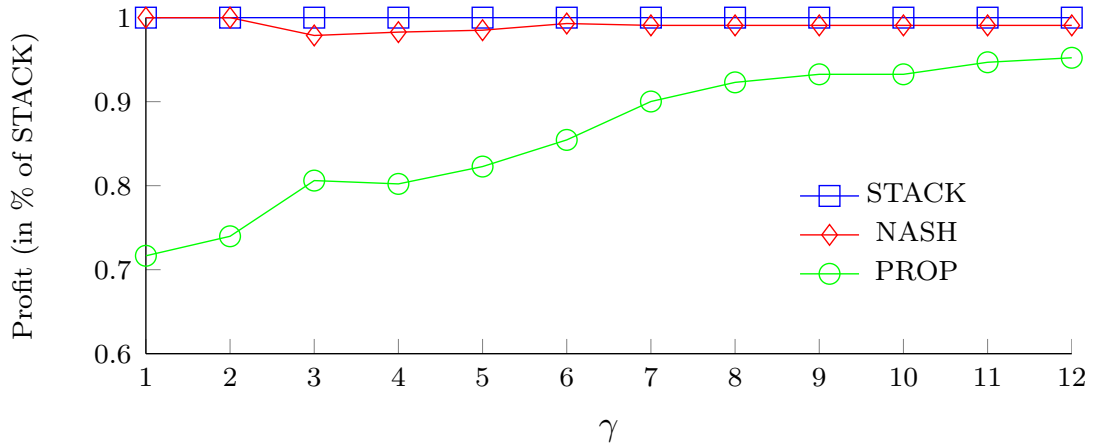


Figure 3.6: Inspectors' profit vs. number of inspectors ( $\gamma$ )

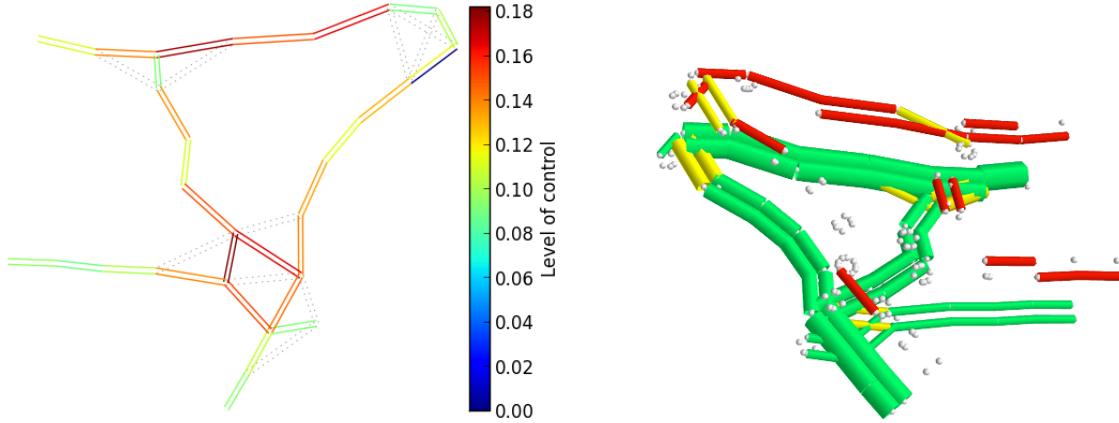
We see that the profit of NASH is often very close to STACK, while PROP yields much less profit for  $\gamma \leq 8$ . While for all  $\gamma \in \{1, \dots, 12\}$ , the user-flow Nash LP is solved in 7 to 8 seconds, for  $\gamma \in \{2, \dots, 6\}$ , the cutting plane method needs 9 to 20 seconds to solve the user-paths Nash LP, but only 5 seconds for  $\gamma = 1$  and  $\gamma \in \{7, \dots, 12\}$ . For all  $\gamma \in \{7, \dots, 12\}$ , the STACK profit is equally 4,915,309, i.e. there is no increase with growing  $\gamma$ . With growing  $\gamma$  the computed paid fraction of km driven in the toll liable motorway network in STACK increases: For one inspector ( $\gamma = 1$ ) we computed 4 % and we reached 100 % for at least 7 inspectors. For  $\gamma \in \{1, \dots, 6\}$ , the total number of kilometers driven on trunk roads is 892,595 km. This number increases to 2,183,955 km for  $\gamma \in \{7, \dots, 12\}$ , here almost all users pay the toll<sup>3</sup> and/or take a trunk road to complete their trip. A picture of the computed drivers flow for  $\gamma = 7$  and the corresponding inspectors Stackelberg strategy is given in figure 3.7.

For the following instance of the subregion of Rheinland-Pfalz, we will present results when the costs for taking a trunk road are larger at  $a = 2$  per km.

Concluding, for this simple instance, we can say that in terms computation time, the user-flow approach is better than the user-paths approach. A Nash strategy

<sup>3</sup>98% of the km driven in the toll liable network are paid.



Figure 3.7: Inspectors strategy (left) and drivers flow (right) in STACK for  $\gamma = 7$ 

is a good trade-off for the Stackelberg strategy between efficiency of controls and computation time.

### Results for subregion Rheinland-Pfalz

Let us now compute results for a much larger instance representing the German subregion Rheinland-Pfalz.

#### Motorway network

number of nodes	:	37
total number of arcs	:	142
number of trunk road arcs	:	56
number of commodities	:	323
number of truck drivers	:	425,082

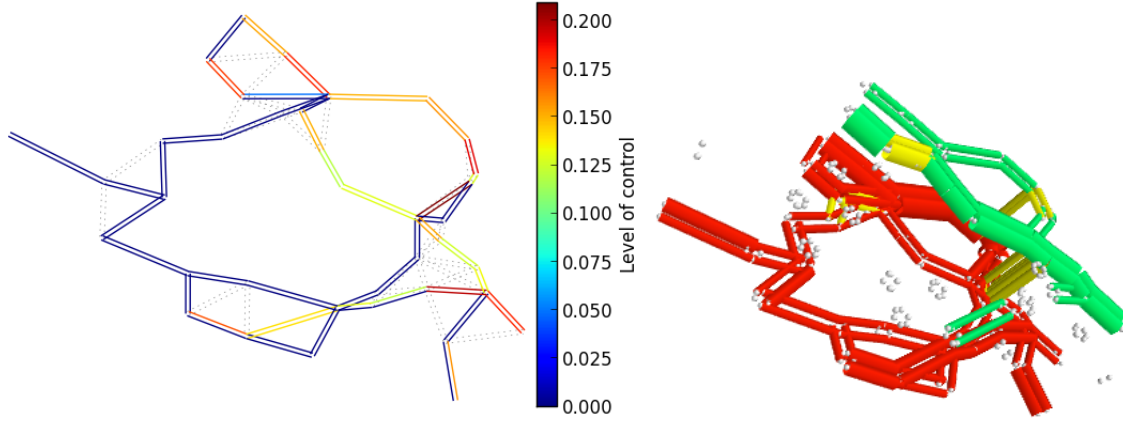
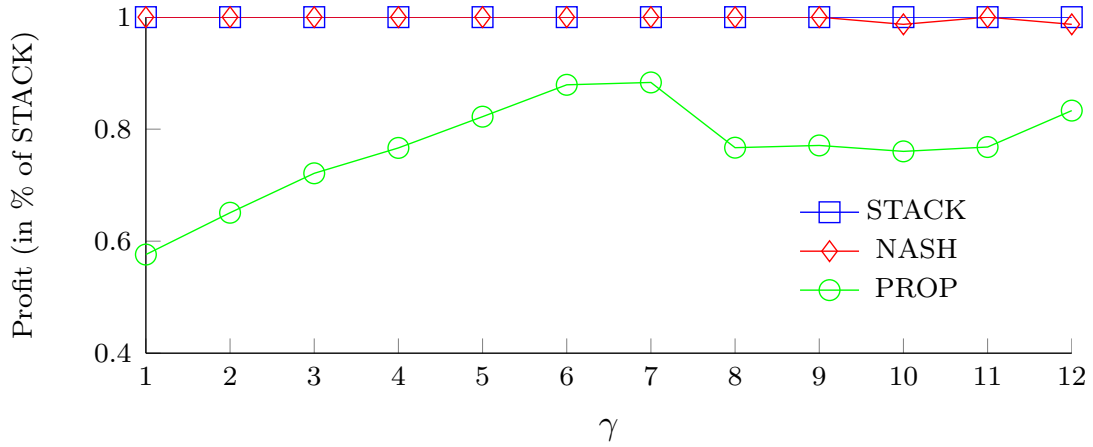
#### Two-level network

number of nodes	:	148
number of arcs	:	506

We create instance *RP-6* of Rheinland-Pfalz, with the following input parameters:

$f$ (toll per km)	:	0.176
$a$ (additional trunk road costs per km)	:	2
$P$ (fine)	:	200
$\theta$ (costs for switch between paying and evading)	:	0.01
$\sigma_e, \forall e \in E$ (prob. to be inspected when insp. present)	:	0.15
$\gamma$ (number of inspectors)	:	6

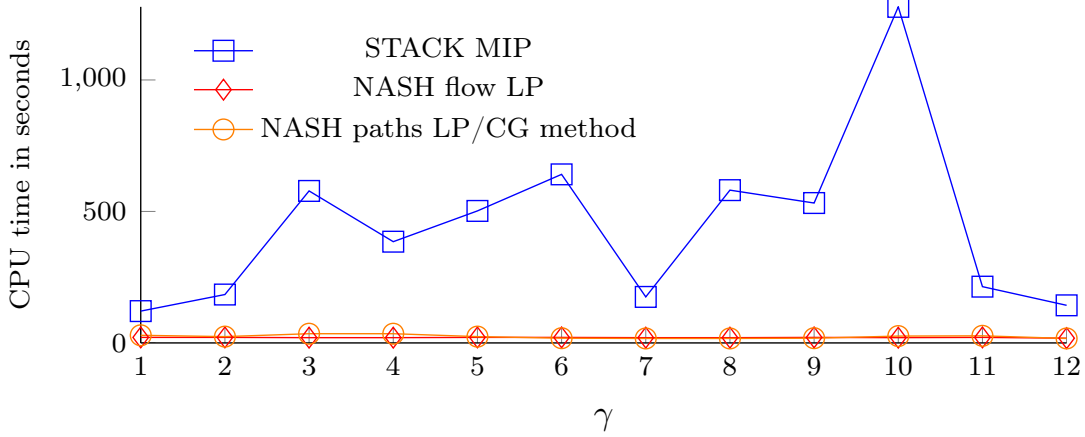
For this larger instance than *ST-6* with additional trunk roads costs per km of 2 and the same number of inspectors, we computed an inspectors' Stackelberg equilibrium payoff of 5,070,128 equivalent to the payoff for the computed Nash strategy. The computed fraction of paid distance of the toll-liable network is 35%. The computed inspectors' Stackelberg strategy and the corresponding drivers flow are shown in figure 3.8. Also here, we compare the inspectors' profit for STACK, NASH and PROP for instances with the same input values as *RP-6* but different numbers of

Figure 3.8: Inspectors strategy (left) and drivers flow (right) in STACK for instance *RP-6*Figure 3.9: Inspectors' profit vs. number of inspectors ( $\gamma$ )

inspectors  $\gamma$ . Results are shown in figure 3.9. While NASH is almost everywhere equal to STACK (and much higher than PROP), the differences in computation times are significant as we can see in the figure 3.10. For  $\gamma \in \{1, \dots, 12\}$  we needed 18 to 21 seconds to solve the user-flow Nash LP and we solved the user-paths Nash LP using the cutting-plane method in 17 to 35 seconds. The solving time of the Stackelberg MIP ranges between 121 and 581 seconds with one outlier at  $\gamma = 10$  with 1278 seconds.

In the next paragraph, we will present result for an representative instance of the complete German motorway network. We will then consider observations concerning the computation time made and try to solve the Stackelberg MIP using the Nash LP solution as a *warm start*.

For our network of the subregion Rheinland-Pfalz and set input values, we observed that the total km driven on trunk roads increases from 1,094,841 km for  $\gamma = 1$  to 1,281,225 km for  $\gamma \geq 3$  up to 2,663,464 km for  $\gamma = 12$ .

Figure 3.10: Computation time vs. number of inspectors ( $\gamma$ )

### Results for Germany

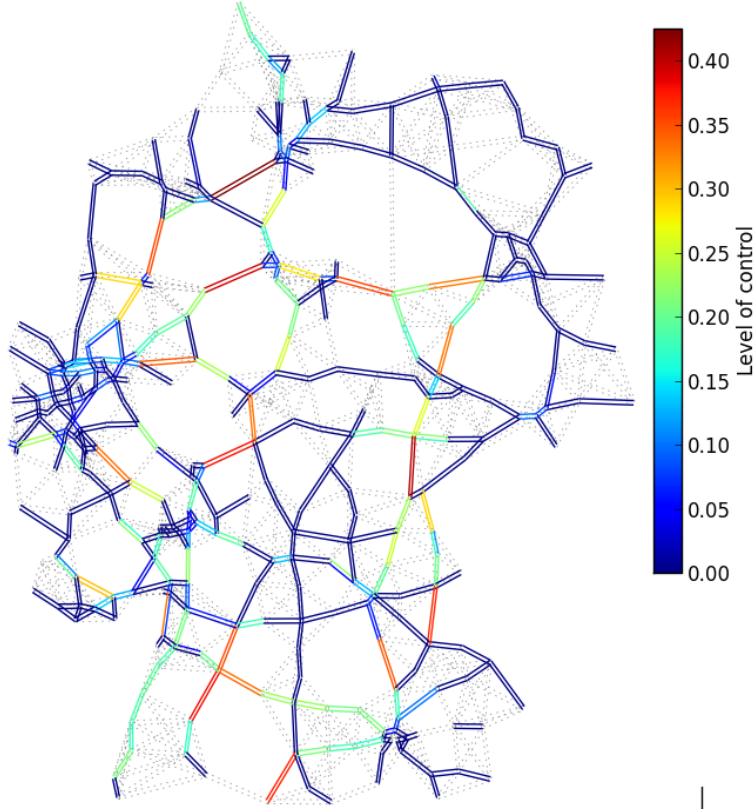
In general control units are assigned to fixed subregions. We will here compute an optimal inspectors strategy in a Stackelberg game for a coalition of control units on the complete German motorway network.

Motorway network		Two-level network	
number of nodes	: 319	number of nodes	: 1,271
total number of arcs	: 2110	number of arcs	: 6,124
number of trunk road arcs	: 1316		
number of commodities	: 5013		
number of truck drivers	: 2,549,743		

We create instance *GER-50* of Germany with the following input values:

$f$ (toll per km)	: 0.176
$a$ (additional trunk road costs per km)	: 0.4
$P$ (fine)	: 200
$\theta$ (costs for switch between paying and evading)	: 0.01
$\sigma_e, \forall e \in E$ (prob. to be inspected when insp. present)	: 0.15
$\gamma$ (number of inspectors)	: 50

Interestingly, for the large instance of Germany it was much faster to solve the user-paths Nash LP using the cutting-plane method than solving the user-flow Nash LP. While the cutting-plane method took only about 20 minutes (i.e. 1,224 seconds), the user-flow LP was solved after about 24 hours. We used the inspectors' Nash strategy as a feasible warm start solution for the Stackelberg MIP and after more than 2 days did not receive results. One optional approach using the inspectors' Nash strategy and the computed best responses for the users in a Stackelberg game as a warm start remains to be done. The inspectors Nash strategy is presented in figure 3.11.

Figure 3.11: Inspectors' Nash strategy for instance *GER-50*

## 3.2 Single-pay-path and variations of the inspectors' objective

In this section, we briefly present a different variant to represent the transportation network of German motorways and experimental results which were presented at the INOC 2013 in a corporate work together with Borndörfer et al. [BBSS12].

Recall in the spot checking game applied to German motorways, a transportation network represents a simplified real-world motorway network consisting of of toll liable arcs and trunk roads. Here we will present a transportation network, where truck drivers take either the shortest path when paying the toll or evade on some route in the motorway network. I.e. we do not allow any switching between paying and evading during a trip.

Using the terminology introduced in the latter section, the transportation network  $D = (V, E)$ , which we call *single-pay-path network*, is holding all arcs  $E_c$  on which drivers evade the toll and that can be controlled, all toll-free trunk road arcs  $E_d$  and for each commodity  $k \in \mathcal{K}$ , an additional arc  $e_k^*$  going from the origin node of  $k$  directly to the destination node of  $k$  on which the drivers pay the toll and no inspection can be conducted. The arc  $e_k^*$  can only be used by drivers of commodity  $k$ . The set  $\mathcal{R}_k$  corresponds to set of OD-paths of commodity  $k$  in  $D$ . The OD-path

on which the users of  $k$  are paying the toll is denoted by  $r_k^*$ . The traveling costs  $w_e$  for using an arc  $e \in E_c \cup E_d$  are defined as given in in the latter section 3.1. The costs  $w_{e_k^*}$  for taking arc  $e_k^*$  correspond to the traveling costs for using the shortest path of  $k$  when paying the toll, i.e. including the legitimate toll fare for the corresponding distance, which we denote by  $f_{e_k^*}$ .

Players and strategies (i.e. strategy  $\mathbf{q}$  for the inspectors' player and  $\forall k \in \mathcal{K}$ , strategy  $\mathbf{p}^k$  for player  $k$ ) are defined as in section 3.1, except that in the user-flow formulation, we have to make sure that arc  $e_k^*$  is only used by player  $k$ : For every commodity  $k \in \mathcal{K}$ , let  $E_k$  denote the set  $E_c \cup E_d \cup \{e_k^*\}$ . Then  $\forall k \in \mathcal{K}$ , flow variables  $p_e^k \in [0, 1]$  are defined for all  $e \in E_k$ . Furthermore,  $\forall k \in \mathcal{K}$ , we add an additional constant  $q_{e_k^*} = 0$  for  $e_k^*$ .

Using a user-flow formulation, the expected payoff of player  $k$ ,  $\forall k \in \mathcal{K}$ , is given by

$$-\lambda_k := - \sum_{e \in E_k} (\sigma_e q_e P + w_e) p_e^k.$$

**Variations in the inspectors' objective** We assume that every driver wants to minimize his total expected costs. In section 3.1 we limited our focus to the objective of inspectors to maximize their profit gained from toll and fines (MAXPROFIT). We exploited the structure of the objective functions of the players in this scenario to transform the non-zero sum game into game for which we could use an LP to compute a Nash equilibrium. We now want to consider different objectives of inspectors. We introduce two optional scenarios: MAXTOLL, where the inspectors' objective is to maximize the profit gained from toll fares and a scenario MAXPAYERS, where the objective is to minimize the number of evaders. MIP Due to the special structure of the single-pay-path-network, we can find easier problem formulations for the new scenarios.

**Inspectors' payoff** We will express the payoff of the inspectors in general terms by introducing parameters  $\alpha \in [0, 1]$  and  $\beta_e \in \mathbb{R}_+$ ,  $\forall e \in E$ .  $\alpha$  denotes the fraction of revenue from fines and  $\beta_e$  is the inspectors' reward if a user takes arc  $e$ . We can consider the different scenarios by selecting the values of these parameters in the expected payoff of the inspectors given by

$$\sum_{k \in \mathcal{K}} x_k \sum_{e \in E_k} p_e^k (\alpha \sigma_e q_e P + \beta_e). \quad (3.4)$$

If we choose  $\alpha = 1$ ,  $\beta_e = 0$ ,  $\forall e \in E_c \cup E_d$  and  $\beta_{e_k^*} = f_{e_k^*}$ ,  $\forall k \in \mathcal{K}$ , then the inspectors' expected payoff for the MAXPROFIT scenario is defined. A MAXTOLL scenario can be constructed by choosing  $\alpha = 0$  instead and a MAXPAYERS scenario with  $\alpha = 0$  and  $\beta_{e_k^*} = 1$ ,  $\forall k \in \mathcal{K}$ .

**User-flow Nash LP for MAXPROFIT** As in section 3.1,  $\mathcal{O}$  is the set of origins to a commodity in  $\mathcal{K}$ . Following the concepts presented in section 2.3.3, the LP to compute an inspectors' Nash strategy for the MAXPROFIT scenario is given by

$$\max_{\mathbf{q}, \mathbf{y}, \lambda} \quad \sum_{k \in \mathcal{K}} x_k \lambda_k \quad (3.5a)$$

$$y_v^o - y_u^o \leq \sigma_{(u,v)} q_{(u,v)} P + w_{(u,v)}, \quad \forall o \in O, \forall (u,v) \in E_c \cup E_d \quad (3.5b)$$

$$y_o^o = 0, \quad \forall o \in O \quad (3.5c)$$

$$\lambda_k \leq y_d^o, \quad \forall k \in \mathcal{K}, k = (o, d) \quad (3.5d)$$

$$\lambda_k \leq w_{e_k^*}, \quad \forall k \in \mathcal{K}, k = (o, d) \quad (3.5e)$$

$$0 \leq q_e \leq 1, \quad \forall e \in E_c \quad (3.5f)$$

$$\sum_{e \in E_c} q_e \leq \gamma \quad (3.5g)$$

Recall that  $q_e = 0, \forall e \in E_d$  is a constant. The LP corresponds to LP (3.2) of section 3.1 applied to the single-pay-path network, except that we are looking for the shortest path length w.r.t  $\mathbf{q}$  for player  $k$  using arcs of  $E_c \cup E_d$  (constraints (3.5b)-(3.5c)). We then bound the costs for player  $k$  from above by node potential  $y_d^o$  and by the costs  $w_{e_k^*}$  for taking *pay-path*  $e_k^*$  ((3.5d)-(3.5e)).

**Stackelberg MIP** As in section 3.1, we have  $\mathfrak{D}_o := \{d : (o, d) \in \mathcal{K}\}$ , the set of destinations to an origin  $o \in \mathcal{O}$  and formulate the following Stackelberg MIP using a big-M formulation to compute an inspectors' Stackelberg strategy:

$$\max_{\mathbf{q}, \mathbf{y}, \lambda, \mu, \mathbf{p}} \quad \sum_{k \in \mathcal{K}} x_k (\alpha \lambda_k + \mu^k (\beta_{e_k^*} - \alpha w_{e_k^*})) + \sum_{o \in O} \sum_{e \in E} p_e^o (\beta_e - \alpha w_e) \quad (3.6a)$$

$$0 \leq \sigma_{(u,v)} q_{(u,v)} P + w_{(u,v)} - (y_v^o - y_u^o) \leq M(1 - \mu_{(u,v)}^o), \quad \forall o \in O, \forall (u,v) \in E_c \cup E_d \quad (3.6b)$$

$$y_o^o = 0, \quad \forall o \in O \quad (3.6c)$$

$$0 \leq y_d^o - \lambda_k \leq M\mu^k, \quad \forall k \in \mathcal{K}, k = (o, d) \quad (3.6d)$$

$$0 \leq w_{e_k^*} - \lambda_k \leq M(1 - \mu^k), \quad \forall k \in \mathcal{K}, \quad (3.6e)$$

$$0 \leq q_e \leq 1 \quad \forall e \in E_c \quad (3.6f)$$

$$q_e = 0 \quad \forall e \in E \setminus E_c \quad (3.6g)$$

$$\sum_{e \in E} q_e \leq \gamma \quad (3.6h)$$

$$\sum_{u: (v,u) \in E_c \cup E_d} p_{(v,u)}^o - \sum_{u: (u,v) \in E_c \cup E_d} p_{(u,v)}^o = \begin{cases} \sum_{d \in \mathfrak{D}_o} x_{(o,d)} (1 - \mu^{(o,d)}) & \text{if } v = o, \\ -x_{(o,v)} (1 - \mu^{(o,v)}) & \text{if } v \in \mathfrak{D}_o, \\ 0 & \text{else,} \end{cases}$$

$$\forall v \in V, \forall o \in O \quad (3.6i)$$

$$0 \leq p_e^o \leq M^o \mu_e^o \quad \forall o \in O, \forall e \in E_c \cup E_d \quad (3.6j)$$

$$\mu_e^o \in \{0, 1\}, \mu^k \in \{0, 1\} \quad \forall o \in O, \forall e \in E_c \cup E_d \quad (3.6k)$$

The objective function (3.6a) corresponds to the expected inspectors' payoff (3.4) introduced in the above paragraph when  $\lambda_k$  and  $p_e^o$  are written in terms of  $p_e^k$ . The rest of the formulation corresponds to MIP (3.3) presented in section 3.1 applied to the single-pay-path network, except that  $\forall k \in \mathcal{K}$ , the shortest OD-path length w.r.t. the user costs for a given  $\mathbf{q}$  is bounded from above by node potential  $y_d^o$  and the costs  $w_{e_k^*}$  for taking *pay-path*  $e_k^*$  ((3.6d)-(3.6e)). Furthermore, we introduce additional binary variables  $\mu_k$  which indicate whether a player  $k$  takes the *pay-path*  $e_k^*$  or not (see constraint (3.6d), (3.6e) and (3.6i)). We set the value of flow variable  $p_{e_k^*}^k := \mu^k$  to receive a user-flow that represents player  $k$ 's strategy. Recall section 2.4, where the basic construction of an MIP to compute an inspectors' Stackelberg strategy was presented.

For the MAXTOLL and MAXPAYERS scenario, i.e. when  $\alpha = 0$  and  $\beta_e = 0, \forall e \in E_c \cup E_d$ , we can simplify the above MIP (3.6) as the variables  $p_e^o$  are no longer involved in the objective function (3.6a). In the computation for an inspectors' Stackelberg strategy for those scenarios, we can remove the second inequality of constraint (3.6b), constraint (3.6i) and constraint (3.6j).

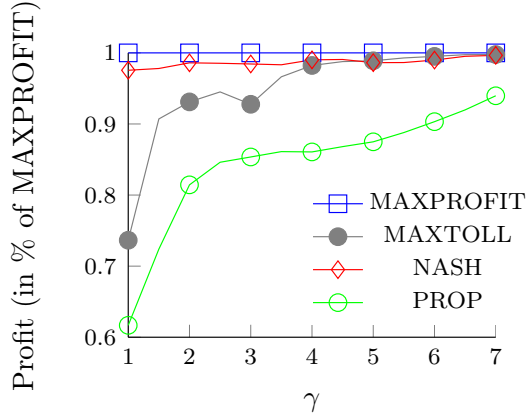
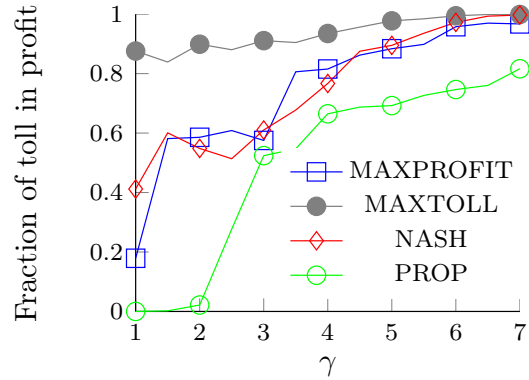
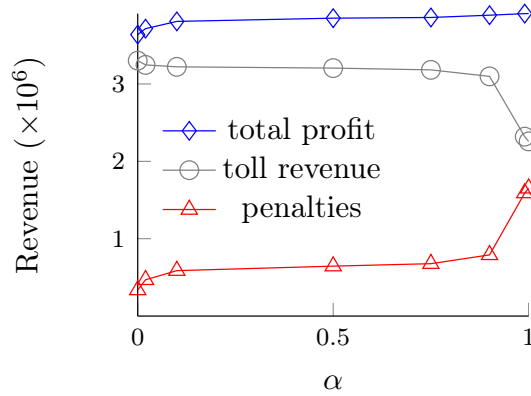
### 3.2.1 Experimental results II

We give a brief overview on experimental results using a single-pay-path network for a simplified motorway network representing the subregion Berlin-Brandenburg and for a network representing complete Germany. Computations were done on a PC with 8 processors at 3.2 GHz using CPLEX.

#### Results for the subregion Berlin-Brandenburg

We computed results for representative network of Berlin-Brandenburg with 45 nodes, 130 arcs and 596 commodities. In the following figures, we compare the inspectors' Stackelberg strategies with respect to the number of inspectors  $\gamma$  of the scenarios MAXTOLL and MAXPROFIT, the inspectors' Nash strategy in the scenario MAXPROFIT and a strategy chosen proportional to recorded traffic volumes (PROP). For both scenarios, we see that the Stackelberg strategies do much better than PROP for  $\gamma \leq 6$  in terms of profit gained (figure 3.12) and evasion rate (figure 3.13). The MAXTOLL scenario yields a higher rate of toll payers than MAXPROFIT. Also here, when using the single-pay-path network representation, results also yield that the inspectors' Nash strategy is a good trade-off between computation time and efficiency of inspections for the Stackelberg strategy in the MAXPROFIT scenario.

In figure 3.14, we present the inspectors' profit for scenarios defined by different input values for parameter  $\alpha$  in the objective function of MIP (3.6) (with  $\beta_e = 0, \forall e \in E_c \cup E_d$  and  $\beta_{e_k^*} = f_{e_k^*}, \forall k \in \mathcal{K}$ ). For  $\alpha = 0.75$ , we receive an inspectors' Stackelberg strategy yielding an inspectors' profit close to the profit in MAXPROFIT, but with less profit coming from fines, i.e. we with more toll payers.


 Figure 3.12: Profit vs.  $\gamma$ 

 Figure 3.13: Fraction of toll in profit vs.  $\gamma$ 

 Figure 3.14: Toll and fines vs.  $\alpha$ 

## Results for Germany

Our representative network for Germany has following has 319 nodes, 2948 arcs and 5013 commodities. We used a computed inspectors' Nash strategy (this took 29 seconds) together with computed best responses for all player  $k$ ,  $k \in \mathcal{K}$  as a warm start solution and where able to compute an inspectors' Stackelberg strategy for the MAXPROFIT scenario with an optimality gap of 1.5% after only 350 seconds. The Nash strategy computed differed from the Stackelberg strategy only on a few arcs and captured 99.7% of the profit. The computed distribution of inspectors in the Stackelberg equilibrium for 50 inspectors is shown in figure 3.15.

## Brief comparison of two-level network and single-pay-path network

In the two-level network we added the option to switch between paying the toll and evading. The choices are depending on the transition costs  $\theta$ . Two-level network and single-pay-path yield different models in terms of the strategy sets for all player  $k$  and different results. As the size of a single-pay-path network is much smaller, we



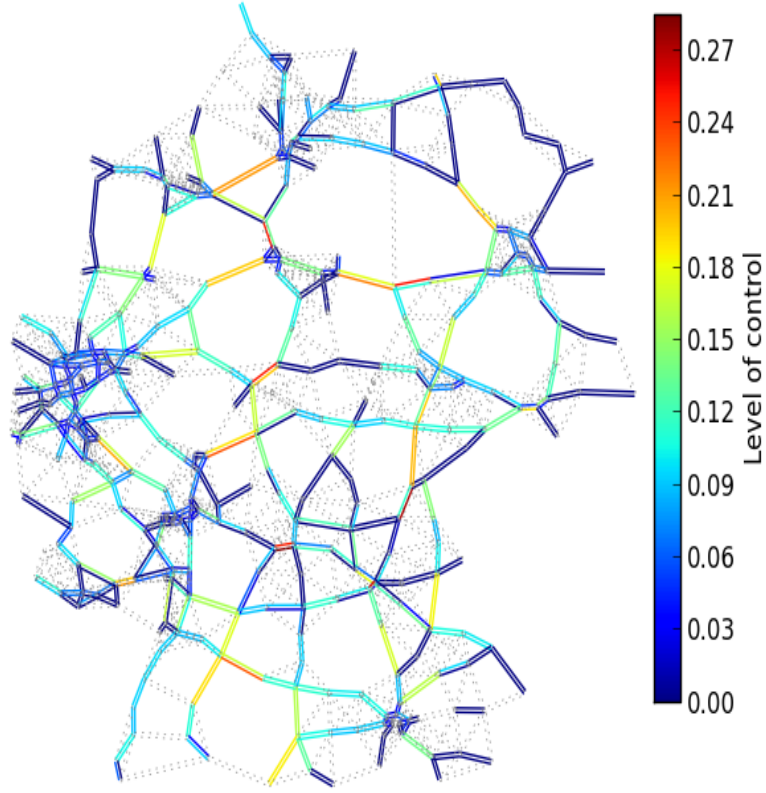


Figure 3.15: Inspectors' Stackelberg strategy in MAXPROFIT for  $\gamma = 50$

were able to get results in shorter time (but we have to consider that computations were done on different PCs with different numbers of processors at different values of GHz). On the other hand, the two-level network depicts users that have much more freedom in choosing to act.

We will now present an extension of the model where users are choosing their strategy in a two-level network and the inspections have to obey certain spatial restrictions and time dependencies.

### 3.3 Inspections regarding space and time

As explained earlier, inspections in reality have to fulfill a variety of regulations and of course they will have to obey the laws of nature such as time and spatial dependencies of consecutive inspection tasks. In the following we will try to integrate these points to get more realistic results. We will start with considering inspections of a fixed duty time lengths and the integration of a real-world regulation to conduct an inspection task within a fixed section of the motorway network. Furthermore, we want to model realistic transitions between different motorway sections, that are actually possible, because of the sections' connections.

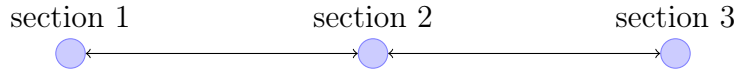
#### 3.3.1 Introducing a cyclic duty digraph

In a recent work from 2012 by Yin et al., schedules for randomized patrols for fare inspection in the Los Angeles Metro Rail System are calculated using a game-theoretic approach [YJJ<sup>+</sup>12]. In contrast to our transportation network, they consider a more trivial topology of a single metro-line. A nice detail in their approach is that feasible inspection duties are modeled as a flow through a network. In the following we will use some ideas of [BSS11] and [YJJ<sup>+</sup>12] and construct a cyclic duty digraph  $\tilde{D} = (\tilde{V}, \tilde{E})$  from which we can extract probabilistic inspections regarding space and time. We introduce the following parameters:

- $\mathcal{H}$  : planning horizon (for example 24 hours)
- $\Delta$  : time discretization (uniform duration of a time interval)
- $\mathcal{T}$  : set of (non-overlapping) time intervals in  $\mathcal{H}$  with  $|\mathcal{T}| = \frac{\mathcal{H}}{\Delta}$
- $L$  : inspection duty length (for example  $2\Delta$ )
- $\mathcal{S}$  : set of predefined inspection sections, where section  $S_i \in \mathcal{S}$  is a subset of  $E$

**Section digraph** We assume that the arc set of the motorway network is divided into inspection section  $S_i \in \mathcal{S}$ . Let digraph  $D_{\mathcal{S}}$ , which we call *section digraph*, be the digraph with nodes corresponding to inspection sections and arcs representing their connections.

**Example 3.3.1.** Our section digraph  $D_{\mathcal{S}}$  is given by three sections with connections as shown in the picture below.



□

**Duty digraph** To construct cyclic duty digraph  $\tilde{D} = (\tilde{V}, \tilde{E})$ , for every time interval  $t \in \mathcal{T}$ , we create a copy of every section  $S_i \in \mathcal{S}$ . Let  $t^\theta \in \mathcal{T}$  denote a duty starting time interval. We replicate above copies again for every starting time interval  $t^\theta \in \mathcal{T}$  and receive triples  $(t^\theta, t, S_i)$  which represent the *inner* nodes of  $\tilde{D}$ .

Furthermore, we have one starting depot node  $d_s$  and one ending depot node  $d_t$  for the inspectors.

Our goal is to have a duty digraph such that every  $(d_s, d_t)$ -path corresponds to a cyclic duty of lengths  $L$  with time interval and section transitions that comply with the real world. This can be realized by creating arcs from starting depot node  $d_s$  to every node  $\tilde{v} = (t^\theta, t, S_i)$  where the duty starting time interval  $t^\theta$  and the time interval  $t$  (corresponding to the conduction time interval of an inspection task) coincide and by creating arcs from every node  $\tilde{v} = (t^\theta, t, S_i)$  where the difference between  $t^\theta$  and  $t$  equals  $\frac{L}{\Delta} \bmod \frac{\mathcal{H}}{\Delta}$  (see footnote<sup>4</sup>) to the ending depot node  $d_t$ . The *inner* nodes (i.e. nodes that are not a depot node) will be connected with each other by arcs w.r.t. possible section transitions and consecutive time interval transitions in a cyclic duty roster, such that every  $(d_s, d_t)$ -path is of lengths  $L$ . Nodes with degree zero should be removed.

**Example 3.3.2.** Let section digraph  $D_S$  be given as in example 3.3.1 and let planning time horizon  $\mathcal{H} = 6$  (hours), time discretization  $\Delta = 2$  and inspection lengths  $L = 4$ . We receive the cyclic duty digraph  $\tilde{D}$  shown in the following figure 3.16. For section 1, 2 and 3 we use the labels  $S_1$ ,  $S_2$  and  $S_3$ .  $\square$

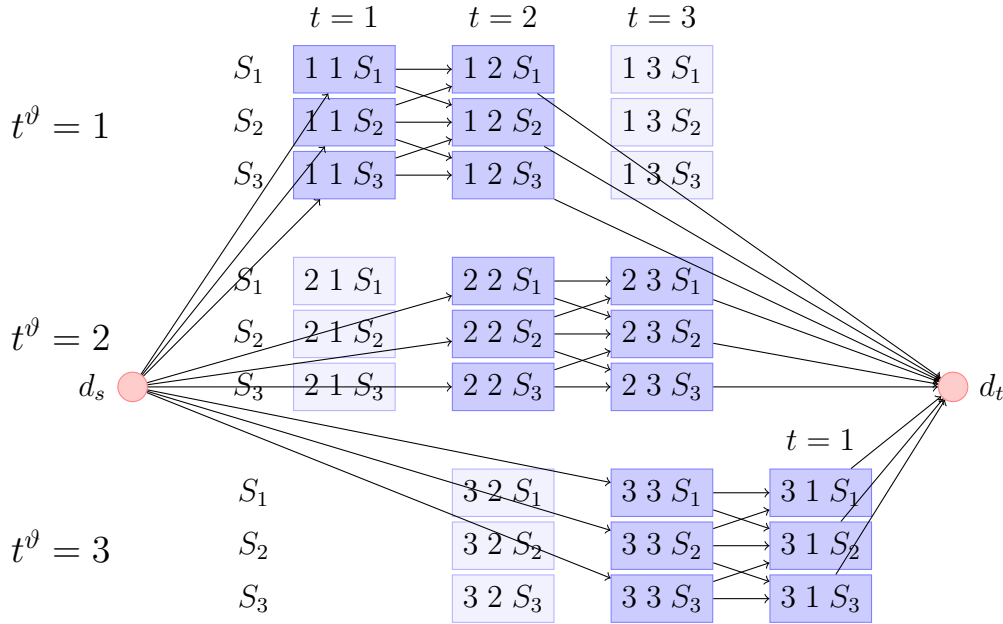


Figure 3.16: Duty digraph  $\tilde{D}$  of example 3.3.2

For now, we will assume that every inspector is allowed to start its duty on every section in any time interval. An extended formulation with several inspection vehicles and varying restrictions will be given in the following section in 3.3.4.

<sup>4</sup>The difference equals the number of time intervals per duty modulo the number of time intervals in the planning horizon.

**Inspectors' strategy** We introduce flow variables  $\tilde{q}_{\tilde{e}} \geq 0$  for all arcs  $\tilde{e} \in \tilde{E}$ . For simplicity reasons, let's assume that we have exactly  $\gamma$  inspectors. A mixed strategy of the inspectors corresponds to a flow through  $\tilde{D} = (\tilde{V}, \tilde{E})$ , i.e.  $\tilde{\mathbf{q}} := (\tilde{q}_{\tilde{e}})_{\tilde{e} \in \tilde{E}}$ , where

$$\sum_{\tilde{e} \in \delta^-(\tilde{v})} \tilde{q}_{\tilde{e}} - \sum_{\tilde{e} \in \delta^+(\tilde{v})} \tilde{q}_{\tilde{e}} = \begin{cases} -\gamma & \text{if } \tilde{v} = d_s, \\ \gamma & \text{if } \tilde{v} = d_t, \\ 0 & \text{else if } \tilde{v} \in \tilde{V}. \end{cases} \quad (3.7)$$

Let us consider a MAXPROFIT scenario, where the network users are choosing their OD-path in a two-level network (see section 3.1).

**Probability that an inspectors is present in the two-level network** The sum of the incoming flow of a node  $(t^\vartheta, t, S_i) \in \tilde{V}$  over all starting time intervals  $t^\vartheta \in \mathcal{T}$  is the fraction of inspections  $q_{(S_i, t)}$  in time interval  $t$  on section  $S_i$ , i.e.

$$q_{(S_i, t)} = \sum_{t^\vartheta \in \mathcal{T}} \sum_{\substack{\tilde{e} \in \delta^-(\tilde{v}): \\ \tilde{v} = (t^\vartheta, t, S_i) \in \tilde{V}}} \tilde{q}_{\tilde{e}}, \quad \forall S_i \in \mathcal{S}, \forall t \in \mathcal{T}. \quad (3.8)$$

Recall that the set  $E_c \subset E$  of the two-level network  $D = (V, E)$  is the set of all arcs on which inspections may be conducted. Let  $l_{S_i} := \sum_{e \in S_i \cup E_c} l_e$  denote the length of section  $S_i$ . ( $\forall S_i \in \mathcal{S}$ , we assume  $l_{S_i} > 0$ ). As an approximation, we will suppose that within a section  $S_i$  and a time interval  $t$  the inspection flow  $q_{(S_i, t)}$  is spread among all arcs on which inspections may be conducted proportional to their length. I.e. let  $q_{(e, t)}$  denote the probability that an inspector is present on arc  $e \in E_c$  in time interval  $t \in \mathcal{T}$ , then the following has to hold:

$$q_{(e, t)} = \min\left(\sum_{S_i \in \mathcal{S}: e \in S_i} q_{(S_i, t)} \frac{l_e}{l_{S_i}}, 1\right) \quad \forall e \in E_c, \forall t \in \mathcal{T} \quad (3.9)$$

Note that several sections may cover the same arc. According to the formulation presented in section 3.1, we introduce constants  $q_{(e, t)} = 0, \forall e \in E \setminus E_c, \forall t \in \mathcal{T}$ .

An optional approximation for the probability that an inspector is present on arcs of the two-level network when using the duty digraph to represent the inspectors strategies is given in the subsection 3.3.4. This alternative is meant to approximate the behavior of experienced inspectors within a section.

Because previous experimental results suggest that the Nash solution is often a very good trade-off for the Stackelberg strategy in the MAXPROFIT scenario w.r.t. computation time and the efficiency of inspections, in the following, we will present the user-flow Nash LP that integrates inspection duties regarding space and time. A user-flow Stack MIP may be constructed similarly. We assume that every user is scheduled to do his trip in a fixed time interval and adjust our notation such that a commodity is given by a pair of Origin-Destination-nodes and a time interval, i.e.

we have  $(k, t) \in \mathcal{K}$ , where  $k = (o, d)$  for selected nodes  $o, d \in V$ ,  $o \neq d$  and  $t \in \mathcal{T}$ . The demand for commodity  $(k, t)$  is given and denoted by  $x_{(k,t)}$ . For simplicity reasons, we assume that every user trip has a duration of  $\Delta$ . The pair  $(v, t)$  is a node in time interval copy  $D_t = (V_t, E_t)$  of two-level network  $D$  corresponding to node  $v \in V$  in time interval  $t$ . Accordingly,  $(e, t)$  is an arc of  $D_t$  and  $O_t$  is the set of all origins of a commodity in time interval  $t$ . We integrate equations (3.7), (3.8) and (3.9) into the user-flow Nash LP (3.2) of section 3.1 and receive the following LP which computes an inspectors' Nash strategy  $\mathbf{q}$  regarding space and time and the corresponding values for  $\tilde{\mathbf{q}}$  from which probabilistic inspection duties can be attained as we will present in next subsection 3.3.2:

$$\max_{\mathbf{q}, \tilde{\mathbf{q}}, \mathbf{y}, \lambda} \sum_{(k,t) \in \mathcal{K}} x_{(k,t)} \lambda_{(k,t)} \quad (3.10a)$$

$$y_{(v,t)}^{(o,t)} - y_{(u,t)}^{(o,t)} \leq \sigma_{((u,v),t)} q_{((u,v),t)} P + w_{((u,v),t)}, \quad \forall (o,t) \in O_t, \forall ((u,v),t) \in E_t, \forall t \in \mathcal{T} \quad (3.10b)$$

$$y_{(o,t)}^{(o,t)} = 0, \quad \forall (o,t) \in O_t, \forall t \in \mathcal{T} \quad (3.10c)$$

$$\lambda_{(k,t)} = y_{(d,t)}^{(o,t)}, \quad \forall (k,t) \in \mathcal{K}, (k,t) = ((o,d),t) \quad (3.10d)$$

$$\sum_{\tilde{e} \in \delta^-(\tilde{v})} \tilde{q}_{\tilde{e}} - \sum_{\tilde{e} \in \delta^+(\tilde{v})} \tilde{q}_{\tilde{e}} = \begin{cases} -\gamma & \text{if } \tilde{v} = d_s, \\ \gamma & \text{if } \tilde{v} = d_t, \\ 0 & \text{else,} \end{cases} \quad \forall \tilde{v} \in \tilde{V} \quad (3.10e)$$

$$0 \leq \tilde{q}_{\tilde{e}}, \quad \forall \tilde{e} \in \tilde{E} \quad (3.10f)$$

$$q_{(S_i,t)} = \sum_{t^\theta} \sum_{\substack{\tilde{e} \in \delta^-(\tilde{v}): \\ \tilde{v} = (t^\theta, t, S_i) \in \tilde{V}}} \tilde{q}_{\tilde{e}}, \quad \forall S_i \in \mathcal{S}, \forall t \in \mathcal{T} \quad (3.10g)$$

$$q_{(e,t)} \leq \sum_{S_i \in \mathcal{S}: e \in \mathcal{S}} q_{(S_i,t)} \frac{l_e}{l_{S_i}}, \quad \forall e \in E_c, \forall t \in \mathcal{T} \quad (3.10h)$$

$$q_{(e,t)} \leq 1, \quad \forall e \in E_c, \forall t \in \mathcal{T} \quad (3.10i)$$

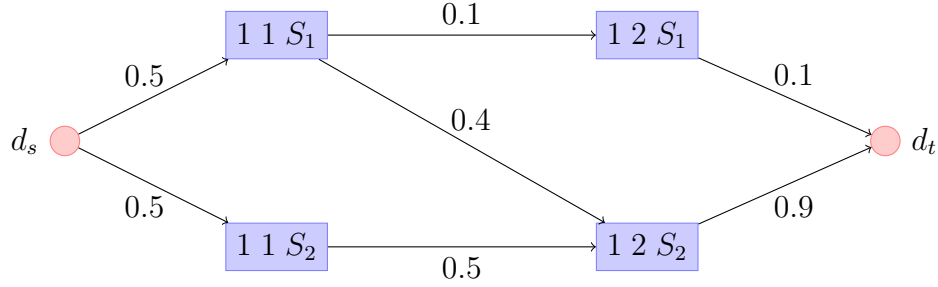
$$q_{(e,t)} = 0, \quad \forall e \in E \setminus E_c, \forall t \in \mathcal{T} \quad (3.10j)$$

(3.10b)-(3.10c) form the single-source shortest path LP formulation for the users of time interval copies  $D_t$  of two-level network  $D$ , where node potentials  $y_{(v,t)}^{(o,t)}$  are bounded from above by the length of a shortest path from origin  $o$  to node  $v$  in  $D_t$ . The length of a shortest path for commodity  $(k, t)$  corresponds to the costs  $\lambda_{(k,t)}$  (3.10d). (3.10e)-(3.10j) ensure a feasible marginal inspectors' strategy  $\mathbf{q}$  subject to a feasible flow  $\tilde{\mathbf{q}}$  in the cyclic duty digraph.

### 3.3.2 Extracting a probabilistic duty roster

From the probability distribution on arcs in duty digraph  $\tilde{D}$  we can extract probabilities to choose a duty. Computed distributions may be used as input for the current large-scale integer program solving the TEP [BSS11], briefly introduced in section 1.1, to compute an integrated control and staff roster. Our approach will be explained using the following example.

**Example 3.3.3.** Let us assume a given motorway network is divided into two sections  $S_1$  and  $S_2$ . We have one inspector and consider two conducting time intervals, just one duty starting time interval and a duty lengths equivalent to the length of two time intervals. Let the computed optimal inspectors flow  $\tilde{\mathbf{q}}$  correspond to the labels of duty digraph  $D_D$  given by:



Arc  $(11S_1, 12S_2)$  has been removed because its flow value is zero. We search for an arc  $\tilde{e}$  with minimal  $\tilde{q}_{\tilde{e}} > 0$ . Let us for example chose arc  $(11S_1, 12S_1)$ . Now we *push* the value 0.1 forward towards  $d_t$  and backward towards  $d_s$ , extracting a  $(d_s, d_t)$ -path. I.e. we compute a  $(12S_1, d_t)$ -path and a  $(d_s, 11S_1)$ -path and connect them with arc  $(11S_1, 12S_1)$ . This path corresponds to a duty that is chosen with probability 0.1. We decrease the flow values on all arcs which belong to the extracted path by 0.1, remove all arcs with  $\tilde{q}_{\tilde{e}} = 0$  and repeat the procedure until all flow values are zero. For this example, we compute the following three duties:

Duty	Description	Chosen with probability
duty 1	on section $S_1$ in $t = 1, 2$	0.1
duty 2	starting on section $S_1$ in $t = 1$ and ending on $S_2$ in $t = 2$	0.4
duty 3	on section $S_2$ in $t = 1, 2$	0.5

□

### 3.3.3 Experimental results III

As in the experimental results I presented in subsection 3.1.1, the computations where done on a PC with 4 processors at 3 GHz using the CPLEX solver.

#### Impact of inspection sections

To get an impression of the impact of assigning inspectors to inspection sections, we compute the optimal Nash strategy for the inspectors for instance *ST-6* presented

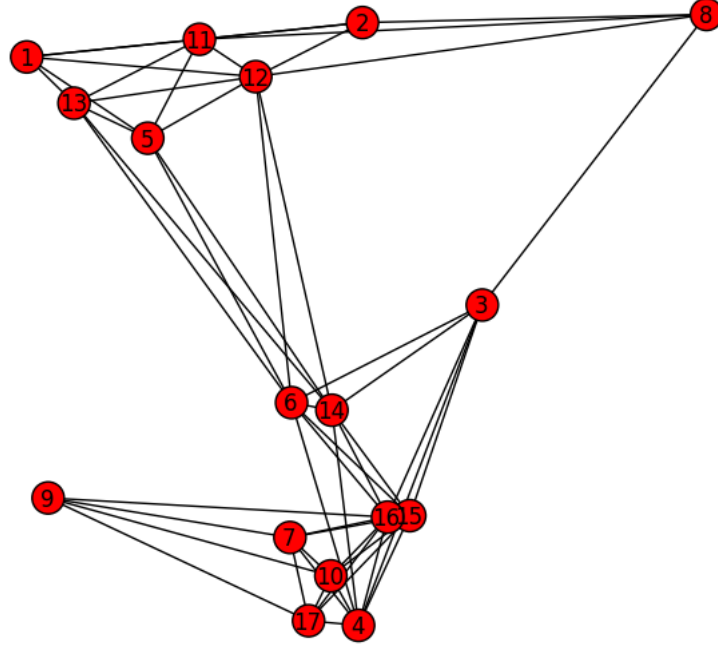
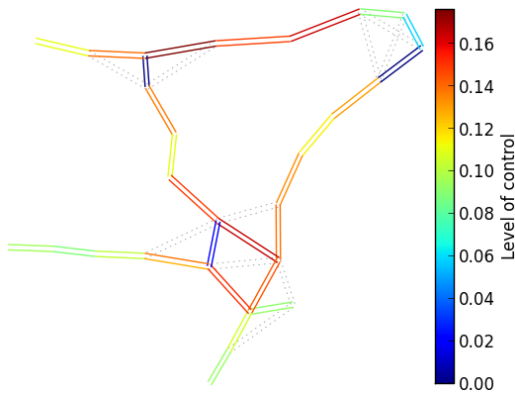
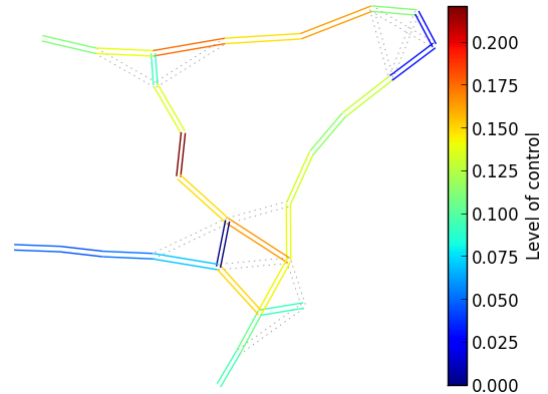


Figure 3.17: BAG inspection sections for Sachsen-Anhalt

in the experimental results I (see 3.1.1). Let us for now neglect temporal restrictions. The following picture shows an approximate spatial distribution of inspection sections of the subnetwork Sachsen-Anhalt of the BAG.

Below, on the left we see the inspectors' Nash strategy for instance  $ST-6$  as presented in 3.1.1, where no inspection sections were considered, which we compare to the picture on the right showing the inspectors' Nash strategy for instance  $ST-6$  subject to the BAG inspection sections.

Figure 3.18: Inspectors' Nash strategy for instance  $ST-6$  (no inspection sections)Figure 3.19: Inspectors' Nash strategy for instance  $ST-6$  subject to BAG inspection sections

When considering BAG inspection sections, the computation time for an inspectors' Nash strategy for instance  $ST-6$  using Nash LP (3.10) with  $|\mathcal{T}| = 1$  is 8 seconds,

which is +1 second compared to the computation time of a Nash strategy with Nash LP (3.2) where no sections are considered. The inspectors' profit from toll and fines in a Stackelberg game for the solution of (3.10) given in figure 3.18 is 4,660,486. This is 2% less (- 9,2024) compared to the profit from the solution of (3.2) given in figure 3.19.

### Inspections for a cyclic schedule

We computed an inspectors' Nash strategy for instance *ST-6* subject to BAG inspection sections for a cyclic schedule with  $\mathcal{H} = 24$  (hours), a time discretization  $\Delta = 2$  and a duty length  $L = 4$ . The result is shown in the series of pictures in figure 3.21. We see that for  $\gamma = 6$ , i.e. 6 inspectors distributed over the whole day, we receive inspections on arcs in the northern region of Sachsen-Anhalt mostly. The inspectors' profit in a Stackelberg game when playing with the computed Nash strategy is 1,296,887. The plot in figure 3.20 shows the quotient inspectors' profit / driven km in the toll liable network, which gives some information about the efficiency of controls in every time interval. Recall that the toll fare per km is 0.176 and corresponds to an upper bound for this quotient.

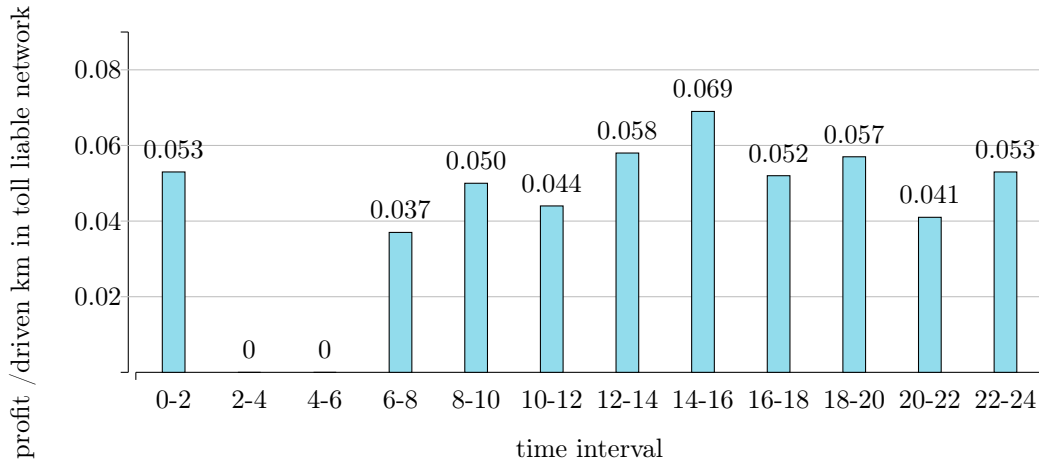


Figure 3.20: inspectors' profit per driven km in the toll liable network vs. time interval

We have also computed an instance of with the same values as SA-6 with  $\gamma = 20$ , presented in figure 3.22, the effect is that inspection are distributed throughout the network, the inspectors' profit in a Stackelberg game is 3,492,790.

The extracted probabilistic duty roster for the computed inspectors' Nash strategy for instance *ST-6* with  $\gamma = 6$  is given in table 3.1. The sections used in the descriptions were given in figure 3.17.



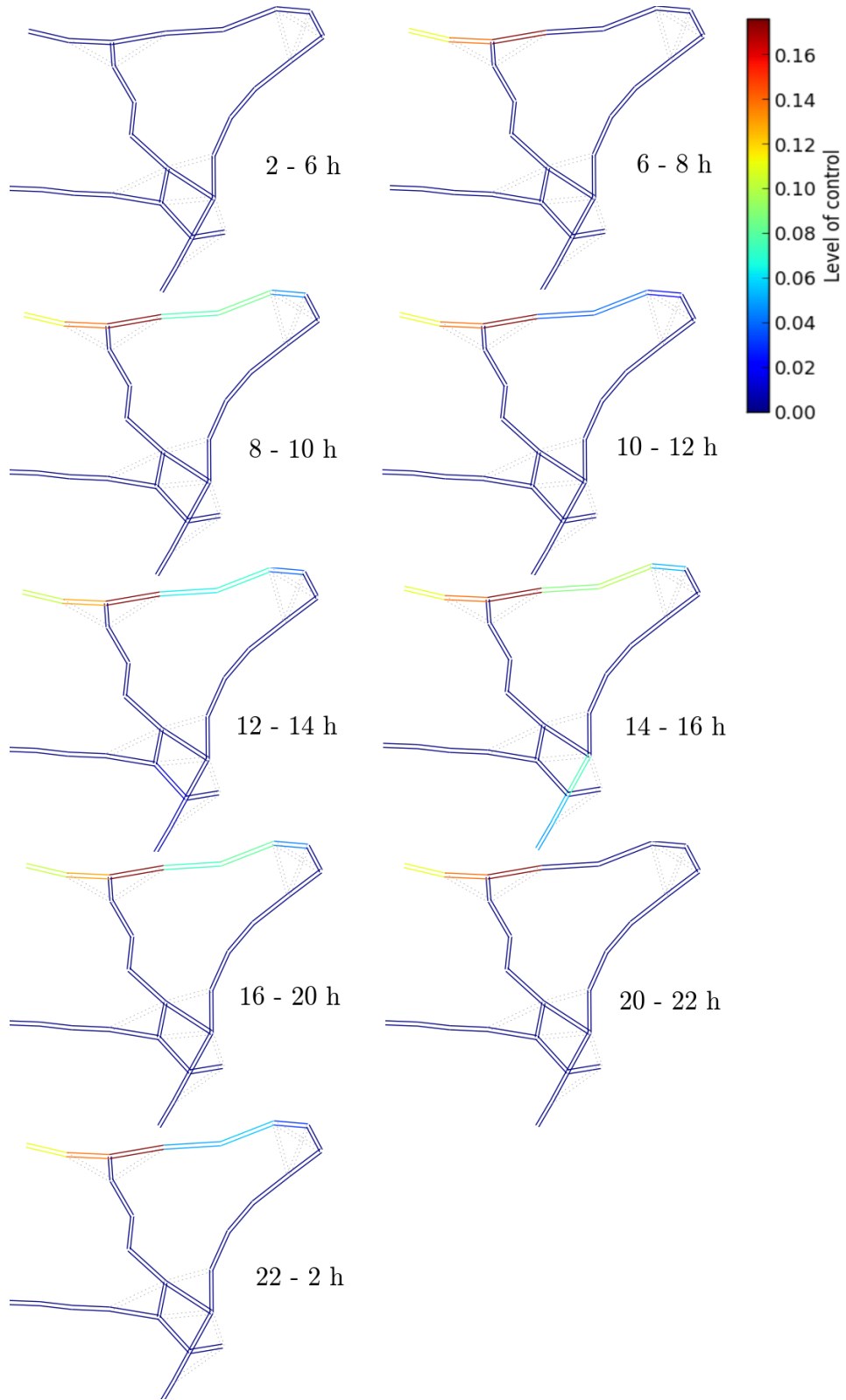


Figure 3.21: Inspectors' Nash strategy for  $ST-6$  for a cyclic schedule

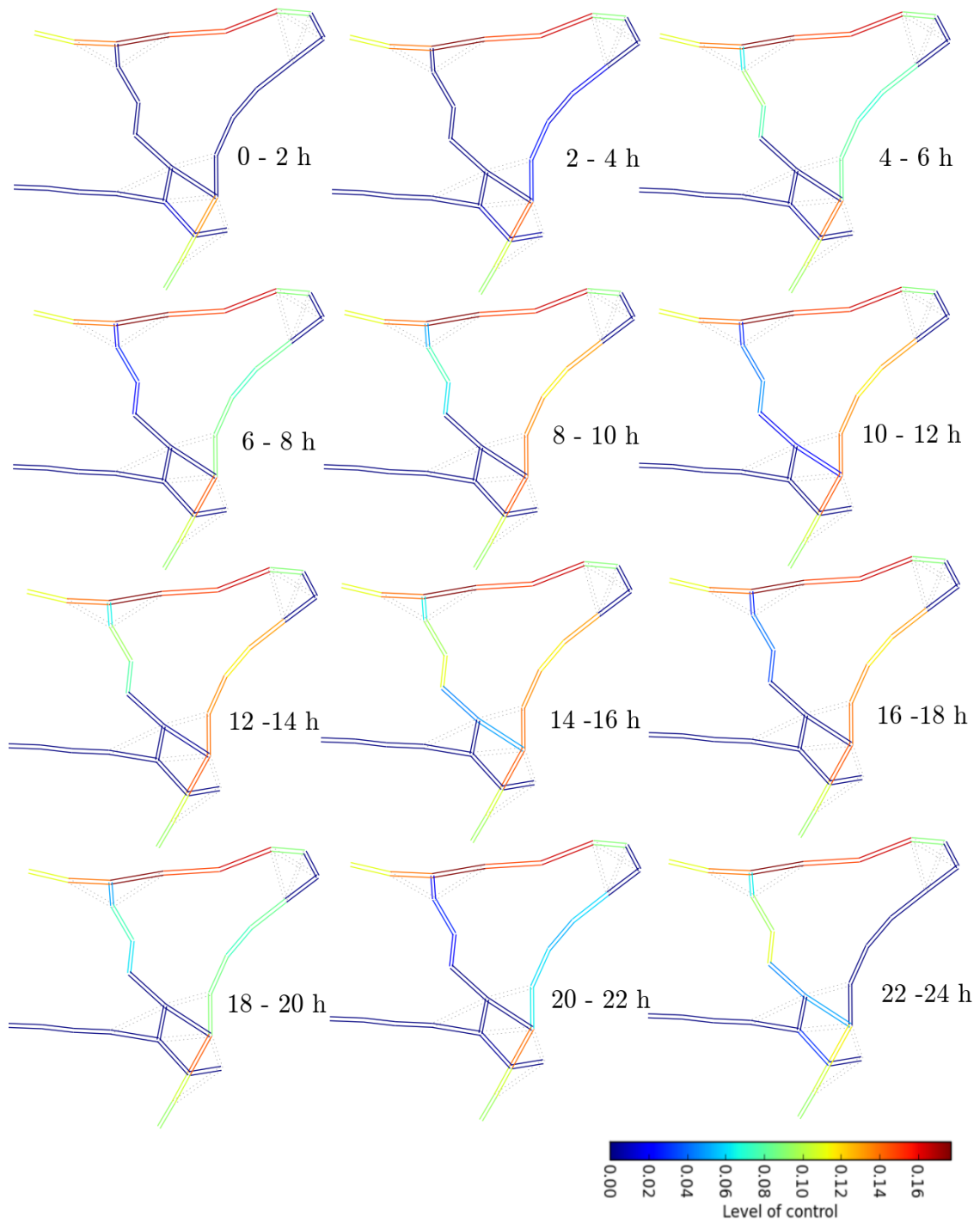


Figure 3.22: Inspectors' Nash strategy for  $ST-6$  with  $\gamma = 20$  for a cyclic day

Duty	Description	Chosen with probability
duty 1	6-8 h: section 1, 8-10 h: section 11	0.854
duty 2	8-10: section 1, 10-11 h: section 11	0.414
duty 3	10-12: section 1, 12-14 h: section 11	0.641
duty 4	12-14: section 2, 14-16 h: section 11	0.002
duty 5	12-14: section 17, 14-16 h: section 4	0.358
duty 6	12-14: section 1, 14-16 h: section 11	0.524
duty 7	14-16: section 11, 14-16 h: section 2	0.002
duty 8	14-16: section 2, 14-16 h: section 11	0.002
duty 9	14-16: section 1, 14-16 h: section 11	0.344
duty 10	14-16: section 11, 14-16 h: section 1	0.463
duty 11	16-18: section 11, 18-20 h: section 1	0.414
duty 12	18-20: section 11, 20-22 h: section 1	0.854
duty 13	20-22: section 11, 22-24 h: section 1	0.564
duty 14	20-22: section 11, 22-24 h: section 11	0.564

Table 3.1: Probabilistic duty roster for *ST-6* for a cyclic schedule

### 3.3.4 Extensions

Simple extensions (for example adding a fine for every evader detected to approximate the loss of time in practice for recording an evader) are easy to imagine. Others, like forcing a duty mix over different time intervals may be handled by the IP to solve the TEP [BSS11] (see section 1.1 of this thesis) when creating a real duty roster that is subject to a complex set of legal constraints. In the following, we will present two simple extensions. First, how to integrate multiple depot for the inspectors when computing probabilistic inspections w.r.t. space and time according to subsection 3.3.1 and secondly, we present an optional approach to approximate the distribution of inspectors within a section.

#### Multiple depots

In praxis on German motorways, an inspector will usually begin and end an inspection duty by inspecting a section close to his home. Therefore, to make the model more realistic, we may extend it by integrating multiple depots, one for each inspector, into the duty digraph  $\tilde{D} = (\tilde{V}, \tilde{E})$  introduced in subsection 3.3.1:

Let  $J$  denote the set of inspectors. For every inspector  $j \in J$ , we define non-negative flow variables  $(\tilde{q}_{\tilde{e}}^j)_{\tilde{e} \in \tilde{E}}$ . In cyclic duty digraph  $\tilde{D}$  we replace the starting depot node  $d_s$  and the ending depot node  $d_t$  by starting depot nodes  $d_s^j$  and ending depot nodes  $d_t^j$  for every inspector  $j \in J$ . We add connections to sections according to regulations that have to be given.

Here the sum over all  $j \in J$  and over all starting time intervals  $t^\vartheta$  of the incoming flow of a node  $(t^\vartheta, t, S_i)$  gives the fraction of inspections  $q_{(S_i, t)}$  on section  $S_i$  in time interval  $t$ , i.e.

$$q_{(S_i, t)} = \sum_j \sum_{t^\vartheta} \sum_{\substack{a \in \delta^-(\tilde{v}) \\ \tilde{v} = (t^\vartheta, t, S_i)}} \tilde{q}_a^j \quad (3.11)$$

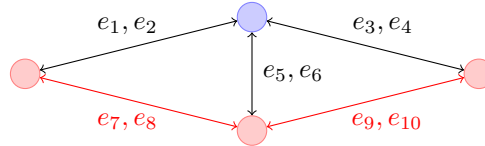
In the MAXPROFIT scenario, we are looking for  $(d_s^j, d_t^j)$ -flows with flow value one which maximize the sum over all inspectors  $j \in J$  of the profit from toll and fines.

### Approximating the distribution of inspectors within a section

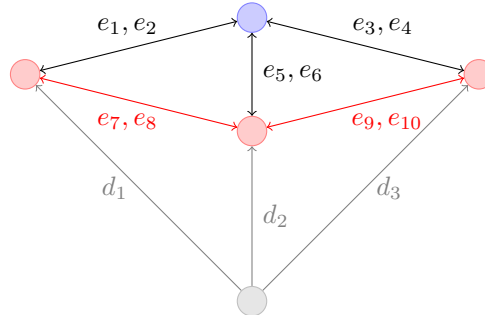
Earlier we approximated inspections within an inspection section and a time interval by distributing inspectors on arcs proportional to their lengths. Here, we will present an optional approach that is meant to approximate the behavior within a section of experienced inspectors, i.e. we allocate inspectors within a section such that, in a spot-checking game, the profit from toll and fines in that section is maximized.

Let us consider two-level network  $D = (V, E)$ . Recall that  $S_i \subseteq E$ . For every section  $S_i \in \mathcal{S}$ , and for every time interval  $t \in \mathcal{T}$ , we create an arc set  $\Xi_{(S_i, t)}$  of arcs from an imaginary node to every node in  $V$  that is incident with an arc of section  $S_i$ .

**Example 3.3.4.** Let two-level network  $D$  be given by the following digraph. The red arcs (arcs  $e_7, e_8, e_9$  and  $e_{10}$ ) belong to a section  $S_i \in \mathcal{S}$ :



Arc set  $\Xi_{(S_i, t)}$  for some time interval  $t \in \mathcal{T}$  consists of new arcs  $d_1, d_2$  and  $d_3$ , which are drawn below in gray color:



□

Now to optimize the allocation of inspectors within a section,  $\forall S_i \in \mathcal{S}$ ,  $\forall t \in \mathcal{T}$ , we introduce new flow variables  $(z_e^{(S_i,t)})_{e \in E \cup \Xi_{(S_i,t)}}$  satisfying:

$$\begin{aligned}
 \sum_{d \in \Xi_{(S_i,t)}} z_d^{(S_i,t)} &= q_{(S_i,t)}, \\
 z_e^{(S_i,t)} &= 0, & \forall e \in E \setminus (S_i \cap E_c), \\
 \sum_{e \in \delta^-(v)} z_e^{(S_i,t)} &= \sum_{e \in \delta^+(v)} z_e^{(S_i,t)}, & \forall v \in V(S_i), \\
 q_{(e,t)} &= \min\left(\sum_{S_i \in \mathcal{S}} z_e^{(S_i,t)}, 1\right), & \forall e \in E, \forall t \in \mathcal{T}.
 \end{aligned} \tag{3.12}$$

Above extensions may be included into the Nash LP regarding space and time (3.10) presented in subsection 3.3.1 by adjusting variable  $\tilde{\mathbf{q}}$  as explained above, adding new flow variable  $\mathbf{z}$  and sets  $\Xi_{(S_i,t)}$  and replacing constraints (3.10g)-(3.10j) with equations (3.11) and (3.12).



# Chapter 4

## Conclusion

We modeled the strategic interaction between the inspectors and the users of a transportation network as a specific Stackelberg game which we call spot-checking game. Our aim was to find equilibrium strategies with a good balance between computation time and inspectors' payoff. We found that a spot-checking game is best-response equivalent to a zero-sum game and hence, a Nash equilibrium can be computed by solving an LP. As the number of strategies for the users is potentially huge, a cutting-plane method to solve the large-scale LP and an alternative efficient user-flow formulation was presented. Finding the optimal strategy to commit to for the inspectors in the Stackelberg game is much harder. Here we used an MIP for the computation. We established a relation between spot-checking games and the class of security games, i.e. spot-checking games form a superclass of the class of security games. Using some results for security games, a theoretical analysis of our spot-checking game yielded some positive conclusions, concerning the deviations of Nash equilibrium and Stackelberg equilibrium strategies. If certain properties are satisfied, we could prove that any inspectors' Stackelberg equilibrium strategy is also a Nash equilibrium strategy. But simple examples can be found for which this is not true. We applied the model to simplified instances of the transportation network of German motorways and some of its subregions using real data. We presented two representations of a transportation network, one in which users are free to pay the toll or evade on parts of their trip and another approach where the users have a single path on which they can pay the toll and further path options on which they evade. The second approach yields faster results due to the smaller size of the network, while the first approach corresponds to the most advanced representation of a transportation network. Computed distributions of inspectors in the equilibria were compared to distributions proportional to recorded traffic volumes in a spot-checking game. Our experimental results suggest that, for a reasonable number of inspectors, the allocation proportional to a recorded traffic distribution yields a much smaller payoff than the optimal payoff for inspectors. While the payoff for distributing inspectors according to a Nash equilibrium strategy is often close the optimal payoff. I.e. experimental results suggest that instead of solving an MIP to compute a Stackelberg equilibrium, one may often use an LP

computing a Nash equilibrium strategy in order to have a good approximation and much less computation time. Furthermore, we presented an application where we were considering inspections regarding spacial restrictions and time dependencies and extracted a probabilistic duty roster. Results of this work may be used as input for the current large-scale integer program to compute an integrated control and staff roster for the real-world problem [BSS11]. In our model, we assumed that the users have full information about the inspectors strategy before they chose their own strategy, an interesting aspect that could integrated into our model would be robustness in terms of uncertainties about the users behavior.



# Appendix A

## Spot-checking games as superclass of security games

Security games are formally defined for example in [YKK<sup>+</sup>10]. In the following, we show that every spot-checking game can be reduced to a security game with possibly an exponential number of targets w.r.t. the size of the transportation network and conversely, we show that every security game can be reduced to a spot-checking game whose size of the transportation network is linear in the number of OD-paths. This yields that spot-checking games form a superclass of the class of security games. For simplicity reasons, we consider a spot-checking game between one player  $k$  and the inspectors player. The formulation can be generalized to a game with multiple followers.

**Reduction of a spot-checking game to a security game** Given a spot-checking game  $G$ , the set of possibly exponentially many OD-paths  $\mathcal{R}_k$  w.r.t. the size of the transportation network corresponds to the set of targets  $T = \{t_1, \dots, t_n\}$  in a security game. The set of defender's resources and the set of defender's schedules in a security game are modeled non-explicitly by assuming that we have a limited number of  $\gamma$  inspectors in spot-checking game  $G$  which can cover subsets of the arc set  $E$  of cardinality up to  $\gamma$  subject to some further linear constraints. In [YKK<sup>+</sup>10] the expected utilities of the players in a security game are defined in terms of utilities, when a target is covered by a defender or not, i.e. by

- $U_d^c(t_i)$ , the defender's utility if  $t_i$  is attacked while  $t_i$  is covered,
- $U_d^u(t_i)$ , the defender's utility if  $t_i$  is attacked while  $t_i$  is not covered,
- $U_a^c(t_i)$ , the attacker's utility if  $t_i$  is attacked while  $t_i$  is covered,
- $U_a^u(t_i)$ , the attacker's utility if  $t_i$  is attacked while  $t_i$  is not covered.

We do not explicitly define player  $k$ 's costs, respectively the inspectors' payoff, if player  $k$  takes a path  $r \in \mathcal{R}_k$  and the inspectors control some subset of arcs belonging to  $r$ , but rather state that the expected payoff for choosing a path  $r$  when the inspectors' marginal probability  $\mathbf{q}$  is given is  $-\sum_{r \in \mathcal{R}_k} (q_e \sigma_e + w_e)$ . This term corresponds to the term  $c_i U_a^c(t_i) + (1 - c_i) U_a^u(t_i)$  in a security game, where  $c_i$  is

the marginal probability that a target  $t_i$  is covered. The expected payoff for the inspectors in a spot-checking game, when player  $k$  takes path  $r$  is  $\sum_{e \in \mathcal{R}_k} (q_e \sigma_e + f_e)$  corresponding to  $c_i U_d^c(t_i) + (1 - c_i) U_d^u(t_i)$  in a security game.

As  $\sigma_e > 0, \forall e \in E$ , the key properties of security games are satisfied. I.e. 1) the payoff for player  $k$  for taking path  $r$ , when no inspector is present on any arc of path  $r$ , is strictly larger than the payoff for player  $k$ , when an inspector is present and 2) the inspectors' payoff gained by covering some arc of  $r$  if player  $k$  takes path  $r$  is strictly larger than the inspectors' payoff if player  $k$  takes  $r$  and no arcs of  $r$  are controlled.

The attacker's strategy  $\mathbf{a}$  in a security game corresponds to player  $k$ 's strategy  $\mathbf{p}^k$ , hence the expected payoff for player  $k$  in a spot-checking game given by

$$- \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (q_e \sigma_e + w_e)$$

can be formulated as

$$\sum_{i=1, \dots, n} a_i (c_i U_a^c(t_i) + (1 - c_i) U_a^u(t_i)),$$

corresponding to the attacker's utility in a security game. Accordingly, the expected payoff for the inspectors

$$\sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (q_e \sigma_e + f_e)$$

can be written as defender's utility in a security game given by

$$\sum_{i=1, \dots, n} a_i (c_i U_d^c(t_i) + (1 - c_i) U_d^u(t_i)).$$

**Reduction of a security game to a spot-checking game** Conversely, we can reduce any given security game between a defender and an attacker to a spot-checking game between inspectors and player  $k$  in a transportation network  $D = (V, E, \mathbf{w}, \mathbf{f}, \boldsymbol{\sigma})$  with exactly one arc from the origin to the destination of player  $k$  for every attacker's target  $t_i \in T = \{t_1, \dots, t_n\}$ . The utilities in a security game, when a target is covered by a defender or not, are given in the spot-checking game as follows (a target  $t_i \in T$  is represented by an arc  $e \in E$ ):

$U_d^c(t_i)$ , the defender's utility if  $t_i$  is attacked while  $t_i$  is covered, is given by  $f_e + \sigma_e$ ,  
 $U_d^u(t_i)$ , the defender's utility if  $t_i$  is attacked while  $t_i$  is not covered, is given by  $f_e$ ,  
 $U_a^c(t_i)$ , the attacker's utility if  $t_i$  is attacked while  $t_i$  is covered, is given by  $w_e + \sigma_e$ ,  
 $U_a^u(t_i)$ , the attacker's utility if  $t_i$  is attacked while  $t_i$  is not covered, is given by  $w_e$ .

---

The additional OD-path on commodity  $k$  in a transportation network, that is never controlled by the inspectors could be modeled by adding another arc, whose costs are very high and hence it is never taken by player  $k$ .

The marginal probability  $c_i$  that  $t_i \in T$  is covered, correspond to  $q_e$  for  $e \in E$ .  $\mathbf{q}$  is subject to some linear constraints, and schedules covering certain targets can be extracted. The attacker's strategy  $\mathbf{a}$  is given by player  $k$ 's strategy  $\mathbf{p}^k$ . Therefore, the expected attacker's utility when the defender is playing with marginal strategy  $\mathbf{c}$  and the attacker is playing with strategy  $\mathbf{a}$  given by

$$\sum_{i=1,\dots,n} a_i (c_i U_a^c(t_i) + (1 - c_i) U_a^u(t_i)),$$

can be represented by the expected payoff for player  $k$  in a spot-checking given by

$$- \sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (q_e \sigma_e + w_e).$$

The expected utility for the defender in a security game for the strategy pair  $(\mathbf{c}, \mathbf{a})$  given by

$$\sum_{i=1,\dots,n} a_i (c_i U_d^c(t_i) + (1 - c_i) U_d^u(t_i)).$$

corresponds to the expected payoff for the inspectors in a spot-checking game given by

$$\sum_{r \in \mathcal{R}_k} p_r^k \sum_{e \in r} (q_e \sigma_e + f_e).$$



## Appendix B

# German motorways control system and the probability to be inspected when an inspector is present

In the application on German motorways, the value of  $\sigma_e$ , the probability to be inspected when an inspector is present on  $e \in E$ , can be approximated independent of the strategies chosen by the players  $k$ .

On German motorways, the inspectors are driving special inspection vehicles and are able to inspect every truck they overtake using an infra-red tool which checks the on-board unit of the truck or by manually checking the number plate in a wireless access data base. As stated in [BBSS12],  $\sigma_e$  can be derived from the ratio of the speed of the inspectors over the trucks' speed.

In addition, in reality we can assume that a large number of users are honest and do not act strategical. Let the total number of drivers (honest drivers, drivers that strategically choose their OD-Path and all other drivers) on an arc  $e \in E$  be denoted by  $y_e$  and let  $\tau$  denote the number of drivers that can be inspected by one inspector, then  $\sigma_e$  can be approximated by  $\min(\frac{\tau}{y_e}, 1)$ . The assumption that there is a negligible small number of strategical users does not contradict our approach to consider strategical acting drivers. Those are the users which are influenced by inspections, the users we would like to deter from evading.



# Bibliography

- [AMO93] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. Network flows: theory, algorithms, and applications. 1993.
- [AvSZ02] Rudolf Avenhaus, Bernhard von Stengel, and Shmuel Zamir. Inspection games. *Handbook of game theory with economic applications*, 3:1947–1987, 2002.
- [BBSS12] Ralf Borndörfer, Julia Buwaya, Guillaume Sagnol, and Elmar Swarat. Optimizing Toll Enforcement in Transportation Networks: a Game-Theoretic Approach. Technical report, ZIB, Takustr.7, 14195 Berlin, 2012.
- [Bel58] Richard Bellman. On a routing problem. *Quarterly of Applied Mathematics*, 16:87–90, 1958.
- [BOSS12] Ralf Borndörfer, Bertrand Omont, Guillaume Sagnol, and Elmar Swarat. A Stackelberg game to optimize the distribution of controls in transportation networks. Technical report, ZIB, Takustr.7, 14195 Berlin, 2012.
- [BSS11] Ralf Borndörfer, Guillaume Sagnol, and Elmar Swarat. An IP approach to toll enforcement optimization on German motorways. Technical Report 11-42, ZIB, Takustr.7, 14195 Berlin, 2011.
- [BSS12] Ralf Borndörfer, Guillaume Sagnol, and Elmar Swarat. A case study on optimizing toll enforcements on motorways. Technical report, ZIB, Takustr.7, 14195 Berlin, 2012.
- [Bun09] Bundesregierung. Bericht über Verkehrsverlagerungen auf das nachgeordnete Straßennetz in Folge der Einführung der Lkw-Maut / source of figure 1.1: Ingenieurgruppe IVV-Aachen. Bundestagsdrucksache 16/13739, June 23, 2009.
- [CS06] Vincent Conitzer and Tuomas Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 82–90. ACM, 2006.
- [Dij59] Edsger Wybe Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

- [DP09] Constantinos Daskalakis and Christos Papadimitriou. On a network generalization of the minmax theorem. *Automata, Languages and Programming*, pages 423–434, 2009.
- [Fed13] German Federal Ministry of Transport, Building and Urban Development. HGV (Heavy Goods Vehicle) toll: Innovative, ecological and fair. <http://www.bmvbs.de/SharedDocs/EN/Artikel/UI/hgv-heavy-goods-vehicle-toll-innovative-ecological-and-fair.html?nn=39896>, Last checked on January 8, 2013.
- [FF58] Lester Randolph Ford, Jr. and Delbert Ray Fulkerson. A suggested computation for maximal multi-commodity network flows. *Management Science*, 5(1):97–101, 1958.
- [For56] Lester Randolph Ford, Jr. Network flow theory. 1956.
- [FT91] Drew Fudenberg and Jean Tirole. *Game theory. 1991*. MIT Press, 1991.
- [JKK<sup>+</sup>10] Manish Jain, Erim Kardes, Christopher Kiekintveld, Fernando Ordóñez, and Milind Tambe. Security games with arbitrary schedules: A branch and price approach. In *Proceedings of The 24th AAAI Conference on Artificial Intelligence*, pages 792–797, 2010.
- [KT10] Ravi Kannan and Thorsten Theobald. Games of fixed rank: A hierarchy of bimatrix games. *Economic Theory*, 42(1):157–173, 2010.
- [Moo59] Edward Forrest Moore. The shortest path through a maze. In *International Symposium on Theory of Switching Part II*, page 285–292. Cambridge, Massachusetts. Harvard University Press, 1959.
- [MV78] Hervé Moulin and Jean-Philippe Vial. Strategically zero-sum games: The class of games whose completely mixed equilibria cannot be improved upon. *International Journal of Game Theory*, 7(3):201–221, 1978.
- [Nas51] John Forbes Nash, Jr. Non-cooperative games. *Annals of mathematics*, 54(2):286–295, 1951.
- [NRT07] Noam Nisan, Tim Roughgarden, and Eva Tardos, and Vijay V. Vazirani (Eds.). *Algorithmic Game Theory*. Cambridge University Press, 2007.
- [PJM<sup>+</sup>08] James Pita, Manish Jain, Janusz Marecki, Fernando Ordóñez, Christopher Portway, Milind Tambe, Craig Western, Praveen Paruchuri, and Sarit Kraus. Deployed ARMOR protection: The application of a game theoretic model for security at the Los Angeles International Airport. In *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems*, pages 125–132. International Foundation for Autonomous Agents and Multiagent Systems, 2008.



- [PJO<sup>+</sup>09] James Pita, Manish Jain, Fernando Ordóñez, Milind Tambe, Sarit Kraus, and Reuma Magori-Cohen. Effective solutions for real-world Stackelberg games: When agents must deal with human uncertainties. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems, volume 1*, pages 369–376. International Foundation for Autonomous Agents and Multiagent Systems, 2009.
- [PPM<sup>+</sup>08] Praveen Paruchuri, Jonathan P Pearce, Janusz Marecki, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. Playing games for security: An efficient exact algorithm for solving Bayesian Stackelberg games. In *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems, volume 2*, pages 895–902. International Foundation for Autonomous Agents and Multiagent Systems, 2008.
- [Ros74] Robert W. Rosenthal. Correlated equilibria in some classes of two-person games. *International Journal of Game Theory*, 3(3):119–128, 1974.
- [Sch98] Alexander Schrijver. *Theory of linear and integer programming*. Wiley, 1998.
- [vN28] John von Neumann. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.
- [vNM47] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1947.
- [vS34] Heinrich von Stackelberg. *Marktform und Gleichgewicht*. Springer, Vienna, 1934.
- [vSZ04] Bernhard von Stengel and Shmuel Zamir. Leadership with commitment to mixed strategies. Technical Report LSE-CDAM-2004-01, CDM Research Report, 2004.
- [YJJ<sup>+</sup>12] Zhengyu Yin, Albert Xin Jiang, M Johnson, Milind Tambe, Christopher Kiekintveld, Kevin Leyton-Brown, Tuomas Sandholm, and J Sullivan. TRUSTS: Scheduling randomized patrols for fare inspection in transit systems. In *Proceedings of The 24th Conference on Innovative Applications of Artificial Intelligence (IAAI)*, 2012.
- [YKK<sup>+</sup>10] Zhengyu Yin, Dmytro Korzhyk, Christopher Kiekintveld, Vincent Conitzer, and Milind Tambe. Stackelberg vs. Nash in security games: Interchangeability, equivalence, and uniqueness. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, volume 1*, pages 1139–1146. International Foundation for Autonomous Agents and Multiagent Systems, 2010.