Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

RALF BORNDÖRFER GUILLAUME SAGNOL
THOMAS SCHLECHTE ELMAR SWARAT

Optimal Toll Enforcement - an Integration of Vehicle Routing and Duty Rostering

This work was funded by the German Federal Office for Goods Transport (BAG).

ZIB-Report 13-79 (Oktober 2013)

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7
D-14195 Berlin-Dahlem

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: http://www.zib.de

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Optimal Toll Enforcement - an Integration of Vehicle Routing and Duty Rostering*

Ralf Borndörfer† Guillaume Sagnol† Thomas Schlechte†
Elmar Swarat†

31.10.2013

Abstract

We present the problem of planning mobile tours of inspectors on German motorways to enforce the payment of the toll for heavy good trucks. This is a special type of vehicle routing problem with the objective to conduct as good inspections as possible on the complete network. In addition, the crews of the tours have to be scheduled. Thus, we developed a personalized crew rostering model. The planning of daily tours and the rostering are combined in a novel integrated approach and formulated as a complex and large scale Integer Program. The paper focuses first on different requirements for the rostering and how they can be modeled in detail. The second focus is on a bicriterion analysis of the planning problem to find the balance between the control quality and the roster acceptance. On the one hand the tour planning is a profit maximization problem and on the other hand the rostering should be made in a employee friendly way. Finally, computational results on real-world instances show the practicability of our method.

1 Introduction

One of the most important and basic planning problems in Combinatorial Optimization and Operations Research is the Vehicle Routing Problem (VRP),

[1]This work was funded by the Federal Office for Goods Transport (BAG)
[2]Zuse Institute Berlin (ZIB), Takustr. 7, 14195 Berlin-Dahlem, Germany, Email {borndoerfer, sagnol, schlechte, swarat}@zib.de

see [14] for an overview. The main problem is in most cases to determine a set of tours to meet given demands. In this paper we will present a model to set up tours as well, but combined with another optimization problem, the *Crew Rostering*.

The background of the studied problem is that the increase of individual traffic requires from public authorities to spend much investments on extensions or on maintenance of the road network. Therefore in many countries tolls were introduced, especially on motorways, to finance the growing investments. We focus here on the case of Germany where a distance-based toll on motorways and on some federal roads was introduced in 2005. All trucks with more than 12 tonnes vehicle weight have to pay a toll depending on their route distance and their emission class.

The enforcement of the toll is the responsibility of the German Federal Office for Goods Transport (BAG). It is both done by 300 stationary *control gantries* and by tours of about 300 mobile control teams on the entire motorway network. The teams, also called *control groups*, consist mostly of two inspectors, but in some cases of only one. Each team can only control toll roads in its associated *control area*, close to their depot. Germany is subdivided into 21 of those control areas.

In general, the tours should guarantee a network-wide control that takes given spatial and time dependent traffic distributions into account. We will model this problem using a space-time network and formulate a corresponding optimization problem as an Integer Program (IP). To that end we divide the network into *sections*. A section is a subpart of the network with length of approx. 50-70 kilometers, where a team can be assigned to for a control during a certain time interval, e.g., four hours. Furthermore for each inspector the personalized sequence of controls during several weeks, called *roster*, has to fulfill several restrictions. Therefore, our challenge is to integrate and to optimize the vehicle routing and the personalized crew rostering. To the best of our knowledge this is the first optimization approach to toll enforcement where both the tours and the detailed rosters of the crews are computed simultaneously. A typical problem instance is to produce a monthly schedule for one control region with personal data and resources given, e.g., inspectors with working time accounts, feasible routes, vacations, and so on.

The paper is structured as follows: In Section 2, we present our planning problem in general. Section 3 defines a graph formulation for the tour planning and Section 4 defines a rostering model of the crews, respectively. There we will focus on how different legal and organisational requirements are modeled in the rostering part. In Section 5 we will discuss the bicriteria nature of the integrated model in order to analyse the relation between control quality and roster acceptance. Finally, we present results from the real world application at the BAG in Section 6 and provide a conclusion and directions for future research in Section 7.

2 The Toll Enforcement problem

In classical Vehicle Routing problems a set of given demands or tasks has to be met. This is not the case in our approach. Since the number of teams is fixed, a maximization problem with limited resources has to be solved. But there are also problem characteristics that occur in basic VRPs as well. For example, there is a *length restriction* for all tours according to daily working time limitations. We have chosen as our main objective that controls should be planned in order to maximize the number of controlled vehicles. In a companion work we have proposed a game theoretic approach that takes into account the behaviour of drivers and fare evaders to distribute the controls in a more strategical way, see [2] and [1]. We plan to integrate this planning strategy in our optimization model in a follow-up work.

If we assign a profit value to each section, then our problem relates to a *Team Orienteering Problem* or a *Selective Vehicle Routing Problem*, see [5]. In the case of only one vehicle this is known as the *Orienteering Problem*, a variant of the *TSP with profits*. For a recent publication on the Orienteering Problem we refer the reader to [15] and for the more general case of the TSP with profit see Feillet et al. [8] for a literature survey.

Additionally, the profit values vary during different time intervals. Hence, not only the sections of a tour must be set, but also the starting time and the duration of a section control. For example, it might be useful to control one section in the rush hour and another one during the night. An appropriate approach to collect a profit in our setting is then to set the profit to the number of trucks that pass through a motorway section during some time interval. This rewards a control of highly utilized sections more than of sections with low traffic. Another important requirement is that controls should cover the complete network and not concentrate only on the section with the most traffic. This will be both guaranteed by minimum control quota constraints for each network section and by constraints that prohibit parallel controls on the same sections at the same time.

The planning of routes for mobile toll enforcement is not the only problem to solve in our application. In addition, the duties of the inspectors should be scheduled. Therefore, we have a relevant problem extension in comparison to classical VRPs. Most of the solutions of VRPs result in a set of tours. Then drivers are assigned to the tours in a subsequent step. The feasibility of crew assignments is not part of the classical algorithms. The same holds for Team Orienteering problems, where the publications deal not with the case that the problem has to be solved on each day over several weeks with the same crews. But in the toll control setting it is not possible to ignore the availability of crews. There are only a few drivers that can perform a planned tour. Because the sections of the tour can only be controlled by a team that has its home depot not too far away from the control section, since each tour must start and end at the home depot of its associated team. Thus, sequential approaches to plan the tours independently of the crews will fail.

If we assign a crew to each tour, it must fit within a feasible *crew roster*, respecting all legal rules, over a time horizon of several weeks. Minimum rest times, daily working times, vacations or maximal amounts of consecutive working days are examples of important requirements for the planning of rosters. Hence, a personalized *duty roster planning* must be used in our application. Therefore, we developed a novel integrated approach, that leads to a new type of vehicle routing problems.

To the best knowledge of the authors there is no optimization approach to toll enforcement in the literature yet. Related publications deal with scheduling of highway patrols [9], or with the scheduling of security teams in mass transit railway networks [12]. We have called our optimization problem *Toll Enforcement Problem (TEP)* and it is first introduced in [4]. In [3] a case study is presented, that shows the benefit of using the TEP for the planning of toll enforcement. In the following, the previous work is extended into several directions. There is a more extensive view on the modeling power of our graph-theoretic approach for the roster optimization. Moreover, the bicriterion nature of the problem is analysed in more detail with a main focus of the concept of Pareto-optimality. And finally, our algorithm is now used at the BAG in a pilot operation, such that we can present a broader variety of instances with even faster solution times compared to our previous publications. We give in the next two sections a more detailed presentation of the TEP with a main focus on duty sequencing rules for the inspector roster planning.

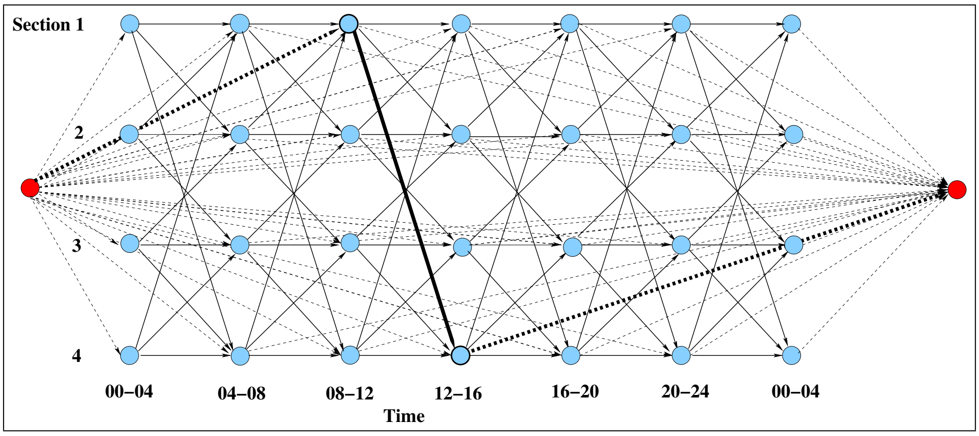

Figure 1: Construction of the tour planning graph.

3 The Tour Planning Problem

To determine the daily routes we use a time-expanded graph model. It is based on a division of the network into sections and on transfer edges between those sections. An edge links two sections that have at least one motorway junction in common. This guarantees that there are no deadhead trips between parts of the control tour. For each time step there exists a copy of each section node and the arcs connect either the same or neighbored sections between two time steps t and $t+1$. The size of the resulting space-time graph D, called *Tour Planning Graph*, mainly depends on the chosen time discretization Δ. Each duty has a duration of eight hours in our simplified model, i.e., by excluding the setup and postprocessing time. If we then consider both the length of the sections and the duration of a duty, only values of two or four hours are appropriate for Δ. In the traditional manual planning approach $\Delta = 4h$ is used. This corresponds to the control of one section then a break and then a second control section. If we add two depot nodes d_1 and d_2, a feasible control tour corresponds to a d_1-d_2-path in D. In Figure 1 a small example of a network with four sections and $\Delta = 4h$ is presented. The thick path shows a exemplary chosen tour.

The *Tour Planning Problem (TPP)* correlates then to a Multi-Commodity Flow Problem in D. It is formulated by an IP based on path variables. Since each control team can only control some local sections, it is still possible to generate all paths by a simple enumeration. As explained in the previous

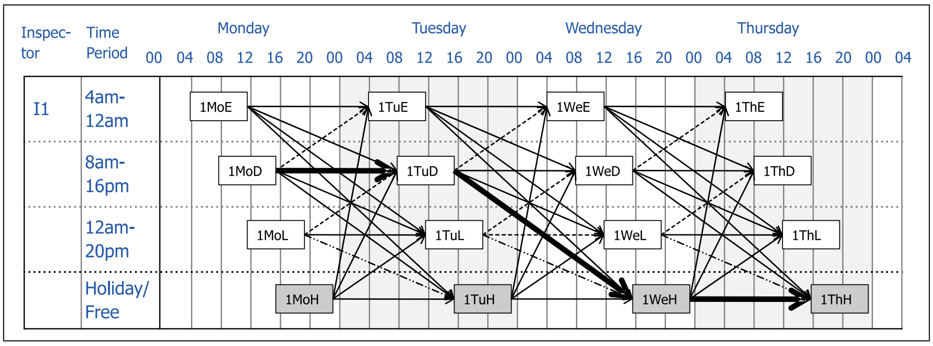

Figure 2: Excerpt of an Inspector Roster Graph with one inspector, three time periods and four days.

section, a profit value is assigned to each node in D that could be covered by a tour. Hence, one can see the TPP as a special team orienteering problem. Beside the network cover constraints (see Section 2) the most important constraints are the natural restriction of at most one tour per day for each team and the interdiction to control with too many teams the same section at the same time. A more detailed description on the graph model and the resulting IP can be found in [4, 3].

4 Duty Roster Planning for Inspectors

The second task in the TEP is the planning of the rosters, called the *Inspector Rostering Problem (IRP)*. There, the objective is to penalize some feasible but inappropriate sequences of duties. We formulate the IRP again as a Multi-Commodity flow problem in a directed graph $\tilde{D} = (\tilde{V} = (\hat{V} \cup \{s,t\}), \tilde{A})$ with two artificial start and end nodes s, t. The nodes $\hat{v} \in \hat{V}$ represent duties as a pair of day and time period. The arcs $(u, \tilde{v}) \in \tilde{A} \subseteq \tilde{V} \times \tilde{V}$ model a feasible sequence of two duties according to legal rules. Therefore, we call the arcs in the model *Duty Sequence Arcs*. By $\tilde{A}_m$ we name all arcs representing duty arcs of inspector $m \in M$. We call this graph *Inspector Roster Graph*. Therefore for each inspector its roster corresponds to exactly one s-t path, called *roster path*. Figure 2 shows a small part of an Inspector Roster Graph. There duty sequence arcs of one inspector between four days are shown. Three different time periods (E[arly], D[ay] and L[ate]) for the duties can be chosen and the other nodes indicate days-off or holidays. The thick path exemplary shows a potential roster path for this inspector.

This problem is again modeled by an Integer Program, but based on arc variables, as we will discuss later in this section. Our model is similar to the approach in [6] and also related to set partitioning approaches like in [11]. The main difference in our approach is that no duties or activities are given. In the TEP the control tours correspond to the duties in the classical models, but they are calculated in the same step when the rosters are generated by connecting the two IPs via coupling constraints. In the following it will be shown how some examplary chosen requirements on crew rostering can be modeled.

The most important requirement for a duty roster is to respect minimum rest times between two subsequent duties. The German *Working Hours Act (Arbeitszeitgesetz)* lays down 11 hours as minimum rest times. This is a local decision in our model since there is a conflict between two duties if the precendent duty ends less than 11 hours earlier than the seconds starts. A graph based approach, as it is used here, is very suitable to resolve this conflict. It can simply be modeled there by only setting arcs between two duty nodes, if the head node duty starts at the earliest 11 hours after the tail node duty has ended. This has the advantage that we do not need any rest time constraints in our model. It is the main algorithmic aspect and contribution in this paper that we try to model as many constraints as possible as local decisions in our graph model. This is a key issue to reduce the high complexity of the integrated problem to get a final optimization problem to solve that is as small as possible. According to this modeling of rest times it is a simple observation that $\tilde{D}$ is acyclic and almost no arcs exist between duty nodes belonging to the same day.

A similar regulation is valid for the rest time when there are days off in between. In case of a time-off there should be at least two days off to ensure a sufficient rest time for the inspectors. This is also modeled by duty sequence arcs. Therefore, a one-day-duty-off would be represented by sequence arcs between nodes on day j and nodes on day $j + 2$. Then a two-days-off is accordingly modeled by arcs between nodes of day j and day $j + 3$. To prevent "short weekends", i.e., only one day-off, no arcs between days j and $j + 2$ are generated. As a consequence if an inspector has a duty on day j, then his next duty cannot be on day $j + 2$.

Another important issue of the model is to take annual leave days or weeks into account. Suppose an inspector m has a leave day on a certain day, e.g., on a Wednesday. At this day inspector m must not be assigned to a duty. To this purpose for the respective day a "vacation duty"-node is generated. All arcs of $\tilde{A}_m$, which start or end at a node belonging to this Wednesday, are incident to this node. Hence, it is not possible to define a path in $\tilde{D}$ with any other than the vacation node on this Wednesday.

An additional relevant consideration for the inspectors is that duties must not start at arbitrary points of time during the day. Therefore, we define a *Duty Type* by its start and end time and only duties corresponding to one of the pre-defined duty types may be scheduled. This can easily be modeled since each potential duty corresponds to a node in the roster graph. We explained above that a node is defined by a specific day and a time period. Hence, only nodes whose time period corresponds to feasible duty types are generated. In addition, the duty types also define the duration of a duty and hence guarantee that daily working time limitations are respected.

Naturally, there are also requirements that can not only be respected by the construction of the graph. To that end, we briefly present some parts from the Integer Programming formulation for the rostering problem to discuss further requirements that are modeled by the IP. For a more detailed presentation of the IP model we refer the reader to our previous publications. There it is also described how both the tour planning model and the rostering are connected by *coupling constraints* to an integrated formulation. Since a flow based formulation is used for the IRP, a variable $x^m_{u,v}$ is introduced for each arc $(u, v) \in \tilde{A}$ and inspector m. This leads to the following IP formulation:

$$\min \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{u,v} x^m_{u,v} \tag{1}$$

$$\sum_v x^m_{s,v} = 1, \qquad \forall m \in M, \tag{2}$$

$$\sum_k x^m_{v,k} - \sum_u x^m_{u,v} = 0, \qquad \forall v \in \hat{V}, m \in M, \tag{3}$$

$$\sum_{(u,v) \in \tilde{A}_m} \omega_u x^m_{u,v} + a_m \leq u_m, \qquad \forall m \in M, \tag{4}$$

$$\sum_{(u,v) \in \tilde{A}_m} \omega_u x^m_{u,v} + a_m \geq \ell_m, \qquad \forall m \in M, \tag{5}$$

$$x^m_{u,v} \in \{0,1\}, \quad \forall (u,v) \in \tilde{A}, m \in M. \tag{6}$$

The objective function (1) minimizes the costs of the rosters. The costs are represented by penalties on the duty sequence arcs. We will discuss later what kind of penalties are present in the model. The next constraints (2) and (3) represent the flow value and the flow conservation for the inspectors. Constraints (4) and (5) involve the working time of the inspectors. They will be discussed in the next paragraph. And last but not least the integrality constraints for the flow variables (6) are given.

In the following, the observance of working hours of the inspectors is discussed. It is very important that their average working time is approximately kept. In the IP model this is done by constraints (4) and (5). At the beginning of the planning horizon the current account value a_m of each inspector $m \in M$ is given. At the end of the planning horizon a feasible interval for the working time account is given with bounds ℓ_m and u_m. Each duty u on the roster path of m consumes some working time ω_u and therefore the constraints bound the length of the roster paths. Kohl and Karisch [11] call this kind of rule a *horizontal rule* according to the fact, that in many rotas the roster of an inspector is displayed in a single row and only one roster is involved by this rule. In contrast to that they name constraints as *vertical rules* if they affect several rosters. An example for a vertical rule are so-called *Duty Mix*-Constraints. They guarantee a pre-defined mix of the duty types. An important purpose of them is to give upper bounds on certain duty types in percentage of all generated duties, e.g., for duty types corresponding to overnight duties.

The objective is to minimize a cost factor that belongs to the duty sequence arcs. But in contrast to classical duty rostering approaches the objective is not to minimize the cost of the duties itself but to penalize unfavorable duty sequences. To that end, let us look at a typical week of one inspector, having duties from Monday to Friday, each starting at 8am. Next, we assume a second week again starting at 8am from Monday to Wednesday, but on Thursday and Friday at 10am. In this case we have a change in the duty starting time between Wednesday and Thursday.

A change of the duty starting time on two subsequent days is called a *rotation*. Since in our example the duty on Thursday starts later than on Wednesday, this is an example of a *forward rotation*. According to this definition the case that a duty on a subsequent day starts earlier than on the day before is called *backward rotation*. See our previous example and assume the case that the duty on Thursday starts at 6am. Then we have a backward rotation. In the case that the previous duty does not end after 7pm on the day before, this backward rotation is feasible with respect to the miminum rest time. Backward rotations can only occur, if they do not violate the minimum rest time between the corresponding duties. Even though rotations are legally feasible they should be avoided. It is particularly known for backward rotations that they alter the human biorythms and affect the sleep [10].

5 Bicriteria Optimization

After the presentation of both subproblems with a detailed view on the modeling of duty sequencing rules it can be concluded that the model has two objectives: on the one hand the rewards of the sections to control and on the other hand the costs for the rotations. This poses the interesting question about the relationship of the two objectives or more general, how to come to a decision if there are several objectives. This is also an important question that arises in real-world planning of toll enforcements and in a lot of other planning problems. The planners at the BAG must balance between severals objectives. On the one side an overall and efficient toll control must be set up but on the other side it is essential to take the matters of the employees into account. This leads us to the important field of *Multicriteria Optimization*, see [7] for an overview.

But the strength of an algorithmic planning method, as it is presented here, lays exactly in the case when there are several criterias to decide. Namely, it is possible to use different parameter settings and focuses on objectives to execute several optimization runs for a planning scenario. This results in control plans that have different focuses on objectives. Then the planner can compare the different solutions and to choose one as the best. In manual planning methods a comparison of alternatives is not possible since it takes a long time to generate even one plan. For the TEP the following questions are relevant: Do the objectives coincide, do they contradict and what is the overall optimum of the integrated model? We analyse those questions with the well-known concept of *Pareto optimality* [13].

Definition 1 (Pareto optimality)**.** *Let* $\max_{x \in X}(f_1(x), f_2(x))$ *be a bi-objective problem. A feasible solution* $\hat{x}$ *is called* Pareto optimal*, if there is no other solution* x *such that* $f_i(x) \geq f_i(\hat{x})$ *for* $i = 1, 2$ *and* $f_i(x) > f_i(\hat{x})$ *for at least one* $i \in \{1, 2\}$*.*

A feasible solution $\hat{x}$ *is called* weakly Pareto optimal*, if for all other solutions* $x \in X$ *holds, that* $f_1(\hat{x}) \geq f_1(x)$ *or* $f_2(\hat{x}) \geq f_2(x)$*.*

Since our problem is a bi-objective problem we restrict our definition of Pareto-optimality to this case. In the following we want to find (weak) Pareto-optimal solutions for the TEP and investigate some properties of these solutions. The first step is the analysis of the Pareto front of one exemplary chosen problem instance to get an indicator for the Pareto front of TEP instances. To this purpose the bi-objective problem is transformed into a single objective optimization problem by the weighted sum approach [13]. The first part of the weighted objective relates to the Tour Planning model, that was briefly described in Section 3 and in more detail in [4]. By P we denote the set of paths in the IP model. For all $p \in P$, $w_p \geq 0$ gives the profit of p. This relates to the sum of the time-dependent profits of all controlled sections, that belong to the tour that corresponds to the path p in the graph. The binary variable z_p indicates if p is chosen in a solution or not. Therefore, we introduce a parameter $\lambda \in [0, 1]$ and change the objective function to:

$$\max\,(1-\lambda)\sum_{p \in P} w_p z_p - \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} x^m_{(uv)} \tag{7}$$

The second part was presented as part of the duty rostering model (1).

For our test we choose an instance r1-may (corresponding to a German control area called r_1 for the month of may). This instance involves 21

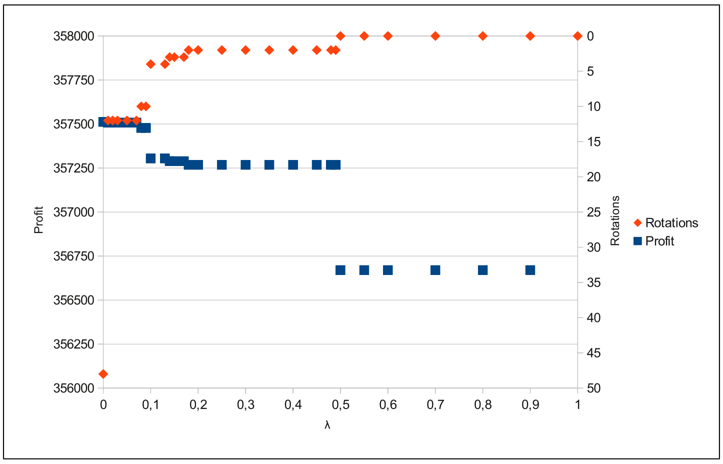

Figure 3: Pareto frontier of instance r1-may, on the x-axis weight factors λ and on the y-axis the control profit on the left and the number of rotation on the right in reverse direction.

inspectors, 17 control sections and 6 duty types. The resulting MIP (before presolve) has then 7738 constraints, 96526 variables and 1233369 nonzeros. We varied the weight parameter λ for several values including the extremal ones. The tests were done on a Dell Power Edge M620 computer with an 8-core Intel Xeon CPU of 2,70 GHz using Cplex 12.5 with 8 threads and default parameter settings. Since this instance was quite easy, each run with a different value of λ was solved to optimality within two hours.

In Figure 3 one can see the results of our computations. The x-axis corresponds to different values of λ, while at the left y-axis the control profit and on the right axis the number of rotations in reverse direction is shown. In the case of no rotation penalties, i.e., $\lambda = 0$, there are quite a lot rotations. But even for a very small value of λ, e.g., 0.02, the number of rotations is reduced rapidly by approximately 75%, while the profit value remains almost the same. An explanation for this observation is that there is some symmetry in the problem. Since each section can be controlled by more than one team, in some cases a simple permutation of the tours of two teams can resolve a rotation without changing the profit value. As an example see two teams t_1 and t_2 that both can control on sections s_1 and s_2. We are given a plan with two rotations, for t_1 on Monday at 7am on section s_1 and on Tuesday at 10am again s_1. For t_2 on Monday at 10am and on Tuesday at 7am on s_2

on both days. The schedule on Tuesday is now changed as follows: team t_1 controls from 7am s_2 and team t_2 from 10am s_1. Then the profit remains the same, but two rotations are resolved.

A value $\lambda > 0.1$ again reduces the number of rotations until $\lambda = 0.5$, where the number of rotations equals zero. Indeed the profit value is decreasing when the weight factor for the rotation penalties increases but the loss is very small. It is approximately as high as the profit of a tour where sections are controlled during a time horizon when the traffic volume is rather low. Another interesting case is $\lambda = 1$, where no profit value is considered for the toll enforcement. The resulting profit value in the solution, i.e., the value that is assigned to nodes on solution paths in the tour planning graph, can be arbitrary bad, since there is no incentive for the model to control sections with high traffic more than those with low traffic. In this case it was only 238963.

The benefit of using the weighted-sum approach is stated by the following Lemma. Although the general result is well-known [13], we give a short proof applied to the TEP for the sake of completeness.

Lemma 1 (Weak Pareto-optimal solutions)**.** *We are given a TEP instance in the integrated IP formulation, but with the weighted objective (7) depending on the parameter* λ*. We call this problem TEP(*λ*). Then* $\forall \lambda \in [0, 1]$ *an optimal solution* (x^*, z^*) *of the TEP(*λ*) is weak Pareto-optimal.*

Proof. Let (x^*, z^*) be an optimal solution of the TEP(λ). Assume this solution is not weak Pareto-optimal. Hence, there exists another solution $(\tilde{x}, \tilde{z})$ with both

$$\sum_{p \in P} w_p \tilde{z}_p > \sum_{p \in P} w_p z_p^*$$

and

$$\sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} \tilde{x}^m_{(uv)} > \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)}.$$

There, we replaced the subtraction in the objective by a sum by setting $c_{(u,v)} < 0$, $\forall (u, v) \in \tilde{A}$. Then it follows that

$$\begin{aligned}(1 - \lambda) \sum_{p \in P} w_p z_p^* + \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)} \quad < \\ (1 - \lambda) \sum_{p \in P} w_p \tilde{z}_p + \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} \tilde{x}^m_{(uv)}.\end{aligned}$$

But this is a contradiction to the condition that (x^*, z^*) is an optimal solution of the TEP(λ) $\square$

The proof shows also, why it is important to consider only the weak Pareto-optimality condition. Namely, in the case of $\lambda = 0$ we discussed that a simple permutation of duties between two teams can reduce the number of rotations without changing the control profit. If we denote our optimal solution again by x^* and the permutated by $\tilde{x}$, it holds that $f_1(x^*) = \sum_{p \in P} w_p z_p^* = \sum_{p \in P} w_p \tilde{x}_p = f_1(\tilde{x})$ but $f_2(x^*) < f_2(\tilde{x})$. The same holds for the case of $\lambda = 1$ by exchanging f_1 and f_2 in the equations of the previous sentence. This violates the condition of (non-weak) Pareto-optimality. Hence, one can state the following Proposition:

Lemma 2 (Pareto-optimal solutions)**.** *We are given a TEP(λ) instance. Then $\forall \lambda \in (0, 1)$ an optimal solution (x^*, z^*) of the TEP(λ) is Pareto-optimal.*

Proof. Let again (x^*, z^*) be an optimal solution of the TEP(λ). Assume this solution is not Pareto-optimal. Hence, there exists another solution $(\tilde{x}, \tilde{z})$ with w.l.o.g.

$$\sum_{p \in P} w_p \tilde{z}_p > \sum_{p \in P} w_p z_p^*$$

and

$$\sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} \tilde{x}^m_{(uv)} = \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)}.$$

Since $(1 - \lambda) > 0$ it follows that

$$\begin{aligned}
(1-\lambda) \sum_{p \in P} w_p z_p^* + \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)} \quad &< \\
(1-\lambda) \sum_{p \in P} w_p \tilde{z}_p + \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)} \quad &= \\
(1-\lambda) \sum_{p \in P} w_p \tilde{z}_p + \lambda \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} \tilde{x}^m_{(uv)} &
\end{aligned}$$

But this is a contradiction to the condition that (x^*, z^*) is an optimal solution of the TEP(λ). The case, that $\sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} \tilde{x}^m_{(uv)} > \sum_{m \in M} \sum_{(u,v) \in \tilde{A}} c_{(u,v)} (x^*)^m_{(uv)}$ can be shown analogously. $\square$

There remains an interesting question if a rotation free plan can be achieved without a big loss of the profit. The results of our test above indicate this. We tried to verify that this result was not an exceptional case. For this purpose, four other real world instances from control area r_1 were tested with regard to the influence of rotation penalties on the control reward. The control profit value of two parameter settings was compared. First, a setting without rotation penalties was used, i.e., $\lambda = 0$. In this setting any number of rotations can be part of the solution. The only focus lies on the control rewards. Second, we used a setting where the penalties lead to an optimal solution without any rotations, e.g., $\lambda = 0.6$ in our previous example. The solutions in the second setting with no rotations never had a loss of more than 0.8% of the control profit achieved by the first setting. That means introducing strong penalty factors for forward and backward rotations causes only a small decrease in the control profit. Hence, we can conclude that computing employee-friendly plans without rotations does not lead to a significantly worse control.

6 Results from the Pilot and Roll-Out in Germany

We implemented our model and algorithm in a computer program, called TC-OPT, which has an interface to the commercial planning suite IVU.plan since release 11.2 from the IVU Traffic Technologies AG. In 2012 we started a pilot operation for two control areas in Germany where the planners at the BAG used our algorithm and tool to compute control plans. From those successfull real world tests we choose six representative instances. In the following we analyse the performance of TC-OPT for those instances. The two control areas, called regions in the following, that are denoted by r_1 and r_2. All instances are based on a time discretization of $\Delta = 4h$.

Table 1 includes the description of all six test instances with respect to their input size and the size of the integrated IP formulations. We used a set of standard (legal) rules, like minimum rest times or working time regulations. The reason for the different number of sections in region r_1 is that in 2012 the toll was extended to several main roads. To control those new network parts new sections were generated. Furthermore, the instances differ partly according to the number of duty types. Even if some input

Instance	Inspectors	Duty Types	Sections	IP Rows	IP Columns
r1-may	21	6	17	7738	96526
r1-july	21	6	17	9650	152541
r1-dec	22	4	22	8010	101791
r1-june	21	6	17	10498	230502
r2-nov	23	6	24	13252	314042
r2-sept	23	8	24	15417	402285

Table 1: Testset: Description of input data and problem sizes.

parameter - like the number of inspectors or the number of sections - seem to be very similar, there is a broad variation in the number of columns. That means, the number of tours and duty sequence arcs differs a lot among the instances.

An important reason for this observation is the very different number of absences of inspectors in the planning horizon due to holidays or diseases among the instances. Another reason is that on some days duties must not be scheduled for a couple of inspectors since there are pre-assigned duties like staff briefings or stationary controls. Therefore, a higher number of absences and duties descreases the number of variables in the IP model and the degree of freedom for the planning of tours is lower. In constrast to that an increased number of duty types, as for r2-sept, leads to a higher degree of freedom. The instance r1-may was used for the analysis of the Pareto frontier in Section 5. All computations were done on a Dell Power Edge M620 computer with an 8-core Intel Xeon CPU with 2,70 GHz and SUSE Linux 12.4 as operating system. The memory limit for the solution tree was 40 GB. Furthermore, there was a time limit of 6 hours (= 21600 seconds) for each instance. As an IP Solver CPLEX 12.5 by IBM with the default parameter setting was applied by using up to eight threads.

Table 2 presents a solution analysis. The main result is that we were able to compute high quality solutions for all instances which can be implemented in real world. The 2nd column displays the solution time for the root LP relaxation. There is a broad variability from 6 to 100 seconds. The ordering of the solution time corresponds in most cases to the ordering of the number of columns, so it can be concluded that an increasing number of columns very often leads to an increasing root relaxation solution time. In column three and four the time and the resulting gap of the first primal solutions

Instance	time(lp) [sec.]	time(1st) [sec.]	gap (1st) (%)	gap (%)	total time [sec.]
r1-may	6.76	80	2.13	-	918.19
r1-july	17.48	75	inf.	-	4041.95
r1-dec	7.56	80	inf.	-	12978.98
r1-june	32.43	500	3.63	0.34	21600.00
r2-nov	99.95	8800	5.28	2.86	21600.00
r2-sept	98.86	12500	5.61	1.31	21600.00

Table 2: IP-Solution analysis for all instances: The solution time of the root LP is denoted by time(lp). The column "time(1st)" indicates when the first primal solution was found while "total time" gives the overall solution time. The column "gap" shows the final solution gap and "gap(1st)" the gap when the first primal bound was found.

are given. The instances r1-july and r1-dec have the value infinity for the first integrality gap, since the objective value was negative in both cases. Although the quality of the first solutions was very poor, for both instances it took less than eight minutes of the total solution time until a solution was found where the resulting integrality gap was less than 9%. The easiest instance r1-may has found a feasible solution when the most difficult instance r2-sept was still at the stage of solving the root LP. Again there is a broad range of values between 75 seconds and three and a half hours. For several instances the quality of the first solution is quite good since the maximum integrality gap is below 6%. And for those whose first solution is not so good, we explained above that a solution of similar quality can be found very quickly. After 10 minutes at least one integer feasible solution was found for all instances from region r_1.

The columns five and six present information obtained at the end of the algorithm, the final gap and the overall solution time. We point out that three instances, r1-may, r1-july and r1-dec, can be solved to optimality. The first two even very fast, namely after 16 minutes and after 68 minutes. In many IPs the final optimal solution is found quickly and most of the time is used to close the final gap. But in our tests, this is not the case. Here, for r1-july and r1-dec it was about one third of the total time to proove the final solution as optimal. All other instances reached the time limit of 6 hours. The fourth instance from region r_1 achieved a final gap of less than 1% and therefore it is almost solved to optimality. The gap of the instances from region r_2 is significantly higher but the solutions still have a very good quality. It is a general observation that the r_2 instances are more difficult to solve. Partially this could be explained by the different number of duty types, inspectors and sections. But maybe there are also characteristics in the assignment of inspectors to sections, in the traffic distribution of r_2 or in the temporal distribution of duty types such that it is more difficult to find feasible solutions and to close the gap by the LP bound.

Therefore, after the succesfull “reality-test” in 2012, the use of our algorithm was extended in 2013 to additional regions as part of an extensive Roll-Out project. The goal of the BAG and the Roll-Out project is that in 2014 all control regions in Germany will be planned by using mathematical optimization. Hence, we can proudly conclude that our mathematical optimization approach gained acceptance in practice. This is the first time that toll enforcement tours and duties are planned by a integrated optimization approach. Our approach supports the planners to achieve a better quality of control plans by using state of the art mathematical optimization techniques. This makes their daily work easier, more transparent, more objective, and most of all more efficient.

7 Conclusion and Future Research

In this paper we presented a special type of an integrated Vehicle Routing Problem, the Toll Enforcement Problem, that is used to compute optimal tours and inspector rosters for the truck toll on German motorways. We shortly presented graph models for the tour planning and the rostering of the crews. They are sufficiently general that they can be used to deal with other inspection problems, like police, security or ticket inspections. One main focus was the description of typical requirements and legal rules for the crew rostering. In many cases these conditions can be satisfied according to an appropriate local modeling in the roster graph. Therefore no constraints were needed in the IP formulation to guarantee compliance of those rules. This results in a strong graph formulation, reducing the IP complexity and leading to a compact integrated problem formulation. That is one important reason for the good performance of our approach.

Another focus was the analysis of the bicriterion character of our problem since the integration of rostering leads to two different objective functions. But fortunately our tests showed no conflict between an efficient control and an employee-friendly crew rostering. On the contrary, we were able to provide solutions without rotation costs at the expense of a hardly smaller control quality. Finally computational results from real world instances were presented. They show that our graph-theoretic model and algorithmic approach is indeed able to solve the instances quite fast and achieves a very good solution quality. Our approach enables us to use the rostering part in an integrated model, which is still computational tractable. Therefore, we can conclude that with our model and in particular with the approach of omitting infeasible duty sequences during the construction of the roster graph it is possible to tackle this challenging real-world problem.

An outline for future research could be three main aspects: First, additional regulations can be added, e.g., introducing individual upper bounds on late, night or weekend duties. Another issue is to increase the simultaneous number of duty types used during one computation. Second, it is an important research direction to decrease the problem complexity, e.g., by a dynamic generation of control tours or duty rosters. In addition, advanced problem specific algorithmic approaches like heuristics, multi-level algorithms or branch and price should be developed. The third aspect is the integration of our game-theoretic approach [1] in the TEP and of course the transfer of our model to other inspection applications.

Acknowledgments

We thank Doris Ludwig-Schreiber and Christian Hoffmann from the German Federal Agency for Goods Transport (BAG) for initiating the project on optimal toll enforcement with the Zuse-Institute Berlin. Furthermore, we thank the technical project manager Eduardo Pinto for organising the project, and in addition, for giving a lot of technical support according to the installation and operation of our tool at the BAG. Our sincere thanks go also to Daniel Schneider and Ralf Haas from the planning department of the BAG, who provided us with many test instances, gave us excellent feedback, and helped us to understand the manifold requirements from real operations. Furthermore, we thank Hans-Stefan Madlung and his colleagues from IVU Traffic Technologies AG for the joint development of a small and flexible interface to exchange the data between TC-OPT und the commercial planning tool "IVU.Plan".

References

[1] Ralf Borndörfer, Julia Buwaya, Guillaume Sagnol, and Elmar Swarat. Optimizing toll enforcement in transportation networks: a game-theoretic approach. *Electronic Notes in Discrete Mathematics*, 41(0):253–260, 2013.

[2] Ralf Borndörfer, Bertrand Omont, Guillaume Sagnol, and Elmar Swarat. A stackelberg game to optimize the distribution of controls in transportation networks. In Vikram Krishnamurthy, Qing Zhao, Minyi Huang, and Yonggang Wen, editors, *Game Theory for Networks*, volume 105 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 24–35. Springer Berlin Heidelberg, 2012.

[3] Ralf Borndörfer, Guillaume Sagnol, and Elmar Swarat. A Case Study on Optimizing Toll Enforcements on Motorways. In Stefan Ravizza and Penny Holborn, editors, *3rd Student Conference on Operational Research*, volume 22 of *OpenAccess Series in Informatics (OASIcs)*, pages 1–10, Dagstuhl, Germany, 2012. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[4] Ralf Borndörfer, Guillaume Sagnol, and Elmar Swarat. An IP Approach to Toll Enforcement Optimization on German Motorways. In Diethard Klatte, Hans-Jakob Lüthi, and Karl Schmedders, editors, *Operations Research Proceedings 2011*, Operations Research Proceedings, pages 317–322. Springer Berlin Heidelberg, 2012.

[5] Sylvain Boussier, Dominique Feillet, and Michel Gendreau. An exact algorithm for team orienteering problems. *4OR*, 5(3):211–230, 2007.

[6] P. Cappanera and G. Gallo. A multicommodity flow approach to the crew rostering problem. *Operations Research*, 52(4):583–596, 2004.

[7] Matthias Ehrgott. *Multicriteria optimization*. Springer, 2005.

[8] Dominique Feillet, Pierre Dejax, and Michel Gendreau. Traveling salesman problems with profits. *Transportation Science*, 39(2):188–205, 2005.

[9] Wu Jiann-Sheng and Lou Tze-Chiang. Highway patrol officer scheduling using an optimization-based scheduling model. In *Advanced Computer*

Theory and Engineering (ICACTE), 2010 3rd International Conference on, volume 2, pages V2–552–V2–557, 2010.

[10] Peter Knauth and Sonia Hornberger. Preventive and compensatory measures for shift workers. *Occupational Medicine*, 53(2):109–116, 2003.

[11] N. Kohl and S.E. Karisch. Airline crew rostering: Problem types, modeling, and optimization. *Annals of Operations Research*, 127(1-4):223–257, 2004.

[12] Hoong Chuin Lau and Aldy Gunawan. The patrol scheduling problem. In D. Kjenstad et al., editor, *PATAT 2012 - 9th International Conference on the Practice and Theory of Automated Timetabling (PATAT)*, pages 175–192, 2012.

[13] Ted K. Ralphs, Matthew J. Saltzman, and Margaret M. Wiecek. An improved algorithm for solving biobjective integer programs. *Annals of Operations Research*, 147(1):43–70, 2006.

[14] P. Toth and D. Vigo. *The Vehicle Routing Problem*. Society for Industrial and Applied Mathematics, Philadelphia, USA, 2002.

[15] Pieter Vansteenwegen, Wouter Souffriau, and Dirk Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1–10, 2011.