
Dissertation
zur Erlangung des mathematisch-naturwissenschaftlichen Doktorgrades
„Doctor rerum naturalium“
der Georg-August-Universität Göttingen

im Promotionsprogramm „PhD School of Mathematical Sciences“ (SMS)
der Georg-August University School of Science (GAUSS)

vorgelegt von

Alexander Schiewe
aus Hannover

Göttingen, 2019
Betreuungsausschuss

Prof. Dr. Anita Schöbel, seit 1.1.2019 Fachbereich Mathematik, Technische Universität Kaiserslautern, vorher Institut für Numerische und Angewandte Mathematik, Georg-August-Universität Göttingen

Jun.-Prof. Dr. Anja Fischer, Juniorprofessur Management Science, Technische Universität Dortmund

Mitglieder der Prüfungskommission

Referentin: Prof. Dr. Anita Schöbel, seit 1.1.2019 Fachbereich Mathematik, Technische Universität Kaiserslautern, vorher Institut für Numerische und Angewandte Mathematik, Georg-August-Universität Göttingen

Korreferent: Prof. Dr. Matthias Müller-Hannemann, Institut für Informatik, Martin-Luther-Universität Halle-Wittenberg

Weitere Mitglieder der Prüfungskommission:

Prof. Dr. Jörg Brüdern, Mathematisches Institut, Georg-August-Universität Göttingen

Jun.-Prof. Dr. Anja Fischer, Juniorprofessur Management Science, Technische Universität Dortmund

Prof. Dr. Gerlind Plonka-Hoch, Institut für Numerische und Angewandte Mathematik, Georg-August-Universität Göttingen

Prof. Dr. Anja Sturm, Institut für Mathematische Stochastik, Georg-August-Universität Göttingen

Tag der mündlichen Prüfung: 28.2.2019
Acknowledgements

I want to start by thanking my thesis advisor, supervisor and first reviewer Anita Schöbel. Thank you for always making time for me, despite your high workload and always encouraging me and my work. Additional thanks go to Matthias Müller-Hannemann for agreeing to be the co-reviewer of this thesis and Anja Fischer for her work as my thesis advisor over the last few years.

Especially since this is a cumulative thesis, it would not have been possible without the help of my co-authors. Therefore, I want to thank Anita Schöbel, Julius Pätzold, Marie Schmidt, Markus Friedrich, Matthias Müller-Hannemann, Maximilian Hartl, Philine Schiewe and Ralf Rückert for the very constructive work and feedback on the examined topics. The same holds true for our research unit FOR2083, which funded most of my time as a research assistant, where we always had very constructive discussions and a very good work climate.

This productive working atmosphere extended to the university of Göttingen as well, were I was able to work together with several outstanding researchers. I want to thank all members of the LinTim-team, namely Anita, Benjamin, Felix, Florentin, Jochen, Jonas H., Jonas I., Julius, Kim, Mridul, Philine, Sebastian and Vibhor for the time we worked together and of course my working group consisting of Anita, Anja, Corinna, Fabian, Jonas, Jörn, Julius, Lisa, Marco, Mirko, Philine, Sebastian and Sönke. Thank you for all your encouragement throughout the years and your help in forgetting the working stress when it was appropriate, be it by playing cards, chess or by working on our several programming projects or competitions.

In the end I want to thank my friends and especially my family for always supporting me in these last years and always being there whenever I needed help. I would not have been able to enjoy my work as much as I do without your support. This of course holds especially true for Philine and Emelie, thanks Philine for proofreading this thesis and always being there for me and my problems and thanks Emelie for always reminding me of the important things in life.
## Contents

1. Introduction .................................................. 1

2. Literature Review ............................................. 5  
   2.1. Line Planning ............................................ 6  
   2.2. Timetabling ............................................. 8  
   2.3. Vehicle Scheduling ..................................... 10  
   2.4. Integration ............................................. 12  

3. Paper Summaries ............................................. 17  
   3.1. Public Transport Planning - Manually Generated and Algorithmic Solutions .................................................. 18  
   3.2. System Headways in Line Planning .......................... 22  
   3.3. Integrating Passengers’ Assignment in Cost-Optimal Line Planning .................................................. 25  
   3.4. The Line Planning Routing Game ........................... 29  
   3.5. Look-Ahead Approaches for Integrated Planning in Public Transportation .................................................. 34  
   3.6. An Iterative Approach for Integrated Planning in Public Transportation .................................................. 38  
   3.7. Cost-Minimal Public Transport Planning .................... 42  

4. Discussion .................................................... 47  

5. Outlook ....................................................... 51  

6. Own Contributions ........................................... 53  

Bibliography ..................................................... 55  

Appendix ........................................................ 69  
   A. Public Transport Planning - Manually Generated and Algorithmic Solutions .................................................. 69  
   B. System Headways in Line Planning .......................... 85  
   C. Integrating Passengers’ Assignment in Cost-Optimal Line Planning .................................................. 103  
   D. The Line Planning Routing Game ........................... 121
E. Look-Ahead Approaches for Integrated Planning in Public Transportation ........................................... 137
F. An Iterative Approach for Integrated Planning in Public Transportation .................................................. 155
G. Cost-Minimal Public Transport Planning .......................... 197
1. Introduction

Public transport planning is a research topic of increasing importance in current times. Since the total population in urban areas is rising further, competitive public transport systems are the only possibility to satisfy the transportation needs of the future, while still allowing for protecting the environment and therefore allowing for a future of humanity on earth. Even when neglecting this important argument, individual traffic is not able to compete with public transportation systems in densely populated areas with respect to travel times due to congestions. Therefore, we need to design good public transport systems to be able to satisfy the future transportation demand of humanity.

In mathematical public transport planning, the planning process is traditionally divided into several stages, see Figure 1. The first stage, network design, involves finding good places for stops and deciding which direct connection, e.g. tracks, should be build or used between them. Afterwards, during load generation, a passenger demand is distributed to the edges, resulting in traffic loads which are used in line planning to decide which lines should be served from an existing line pool. When the lines with corresponding frequencies are given, the timetabling stage determines times for all departures and arrivals of the lines at their stops and during vehicle scheduling, the assignment of vehicles to the lines is decided. After that, multiple planning stages, such as crew scheduling or delay management may occur that are not the focus of this thesis.

The overall goal in this cumulative thesis is to design a cost-optimal public transport plan, i.e., to find a line plan, a timetable and a vehicle schedule such that the operational costs of the system are minimized. We do this by focusing on the operational costs throughout the planning process, developing algorithms for single planning stages as well as integrating several stages to achieve better solutions.

The different publications presented in this thesis have different focus points on the planning procedure:

- In [Friedrich et al., 2017a], see Appendix A, the difference between manual and algorithmic planning approaches are examined. A benchmark dataset is created which is small enough to understand different solutions but big enough to already see meaningful differences in the chosen approaches. The proposed dataset is used throughout this thesis for computational evaluations in the various publications.
Figure 1: Overview of the planning process in mathematical public transport planning

- In [Friedrich et al., 2018a], see Appendix B, a single problem stage, namely line planning is examined. Although cost-optimal line planning is a topic of extensive research, important practical requirements, e.g., the concept of a system headway, are neglected in mathematical public transport planning. This concept is introduced here and examined theoretically and in computational experiments.

- The focus of [Friedrich et al., 2017b], see Appendix C, is the integration of cost-optimal load generation into the line planning stage. These often separated stages are integrated and the benefit of integration is examined theoretically. For a computational evaluation, the current state of the art is compared to a heuristic from practical public transport planners and a newly developed heuristic.

- Another approach to integrating load generation into line planning is introduced in [Schiwe et al., 2019], see Appendix D. A game-theoretic model is proposed, interpreting the passengers as players and allowing them to choose their paths selfishly while taking a share of the costs to allow for cost-efficient line plans. The resulting equilibria are examined theoretically and computationally.

- The work [Pätzold et al., 2017] shifts the focus to planning a cost-optimal public transport plan, i.e., not restricted to line planning but finding a good timetable and vehicle schedule as well. The sequential planning process is improved by considering the effects on the vehicle schedule from the start on,
since this is the stage determining the operational costs. Three improvements to the planning process are proposed and evaluated computationally.

- In [Schiewe and Schiewe, 2018], see Appendix F, a re-optimization approach for a public transport plan is proposed, fixing two of the three stages line planning, timetabling and vehicle scheduling while re-optimizing the third one. While ensuring feasibility, this allows especially for the operational costs to be improved significantly without planning the complete system at once.

- [Pätzold et al., 2019], see Appendix G, examines finding a public transport plan with minimal costs. An integrated optimization model is developed to compute such a solution. Since this is computationally challenging, multiple heuristics are proposed, including optimality conditions and easy to compute theoretical bounds on the optimal costs of a public transport plan. Note that [Pätzold et al., 2019] is an extension of the already published [Pätzold et al., 2018].

The remaining thesis is structured as follows: In Chapter 2, a literature overview for public transport planning in general and for the planning stages line planning, timetabling and vehicle scheduling is given as well as an overview of literature on integration in public transport planning. Subsequently, Chapter 3 summarizes the publications of this thesis. The main results are discussed in Chapter 4 while some conclusions and an outlook are stated in Chapter 5. Chapter 6 gives an overview of my contributions to the publications of this thesis.
Public transport planning is a topic that is traditionally divided into separate planning stages. In this thesis, mainly the stages line planning, with some connections to load generation, timetabling and vehicle scheduling are examined, as depicted in Figure 1. There are several publications giving an overview of the general planning process.

In [Bussieck et al., 1997b], an example planning process is described, presenting several models for each step. [Huisman et al., 2005] and [Lu et al., 2018] both provide an overview of all stages as well, where [Lu et al., 2018] additionally provide connections to “smart” public transport topics, e.g. data-driven approaches and shared mobility. [Borndörfer et al., 2010] give an overview of the different stages while referencing several success stories of mathematical optimization in public transport planning, e.g. revenue management or crew scheduling. [Guihaire and Hao, 2008] provide an overview of the stages line pool generation, line planning and timetabling.

The overall goal is to find a good public transport plan \((\mathcal{L}, \pi, \mathcal{V})\), i.e., a line concept \(\mathcal{L}\), a (periodic) timetable \(\pi\) and a vehicle schedule \(\mathcal{V}\). For a more formal definition of the single stages, see Sections 2.1 to 2.3.

There are several in-depth survey papers, concentrating on the single problem stages mentioned above. An overview on network design can be found in [Kepaptsoglou and Karlaftis, 2009]. For line planning, [Schöbel, 2012] provides an overview of different models and current research, more literature is presented in Section 2.1. For timetabling, see [Lusby et al., 2011] for an in-depth review article and Section 2.2. Furthermore, an overview of vehicle scheduling can be found in [Bunte and Kliewer, 2009] as well as in Section 2.3. For an overview on crew scheduling, see [Van den Bergh et al., 2013]. Since integrating multiple planning stages is a topic of ongoing research in the public transport planning community and topic of this thesis, Section 2.4 provides an overview of current literature on integration in public transport planning and beyond.

When evaluating public transport plans, different objectives are considered in the literature and often represent different points of view. For one, the operator of a public transport plan is often working with a fixed budget or is a for-profit organization, emphasizing the importance of considering the operational costs of a public transport plan. While this is the main focus of this thesis, other objectives are important as well, namely the passenger convenience and the robustness of public transport plans. See [Goerigk, 2012, Goerigk et al., 2013, Parbo et al., 2016,
Friedrich et al., 2017c, Friedrich et al., 2018b] for different aspects of these objectives. There are mainly two approaches to public transport planning, manual planning with computer-aided evaluation and mathematical planning that is algorithm-based. For computer-aided evaluation, there are several commercial software vendors, providing complex software systems. An example is the PTV group providing VISUM, see [PTV Group, 2016]. For stages such as vehicle or crew scheduling, mathematical algorithms already found their way into such commercial products, see [Borndörfer et al., 2010]. For the other stages, mathematical optimization tools are more experimental and often not sophisticated enough for real-world examples without further modifications. See [Schiewe et al., 2018a, Schiewe et al., 2018b] for an open-source software library containing multiple packages for every planning stage discussed in this thesis. For a more in-depth analysis of the different approaches, see [Friedrich et al., 2017a] in Appendix A, summarized in Section 3.1. To test and compare algorithms, the availability of datasets is of utmost importance. See [FOR2083, 2018] for a collection of several open-source datasets with various reference solutions.

The rest of this chapter is structured as follows. Sections 2.1, 2.2 and 2.3 provide an overview of the literature and recent advances in line planning, timetabling and vehicle scheduling, respectively, and define the problems formally. At last, Section 2.4 provides an overview of different approaches to integrated planning with a focus on but not limited to public transport planning.

## 2.1. Line Planning

Line planning is a very fundamental problem of public transport planning. The chosen lines play an important role in influencing the quality or even feasibility of the overall public transport plan, see [Goerigk et al., 2013]. An early survey on line planning for bus networks can be found in [Chua, 1984], whereas [Schöbel, 2012] presents more recent models and literature.

To formally define the problem, let an infrastructure network \((V, E)\) with stops \(V\) and direct connections \(E\) be given. We call this a public transport network (PTN). Additionally, most literature assumes a set of lines, a line pool \(L^0\), to be given, where a line is a path in the PTN. A selection of lines \(L\) with frequencies \(f_l, l \in L\), is called a line concept. For the feasibility of a line concept, a common assumption is that lower and upper frequency bounds \(f_e^\text{min}\) and \(f_e^\text{max}\) to be given and to define

\[
f_e^\text{min} \leq \sum_{l | e \in l} f_l \leq f_e^\text{max} \quad e \in E
\]

as the feasibility constraints of a line concept, while both cost-oriented and passenger-oriented approaches are common as an optimization goal in literature. Here, the
lower frequency bounds ensure feasibility for the passengers, i.e., that every feasible line concept contains a path for each passenger, and the upper frequency bounds are e.g. security constraints.

To determine such lower frequency bounds, a problem called load generation is considered. The bounds are often based on traffic loads $w_e$ for each edge $e \in E$ and the vehicle capacity $\text{Cap}$, see e.g. [Claessens et al., 1998]. However, determining such traffic loads is often not considered in mathematical public transport literature and loads are assumed to be based on a shortest path assignment of the passengers, see e.g. [Bussieck et al., 1997a]. [Nachtigall and Jerosch, 2008] present a column generation approach to solve the integrated problem, while [Pfetsch and Borndörfer, 2006] consider different route choice models and compare them computationally for path-based models. For load-based models, i.e., models that use traffic loads on the PTN edges, [Friedrich et al., 2017b], see Appendix C, consider the integration of the load generation stage into line planning model and compare different heuristics for the load generation problem. For a summary, see Section 3.3.

For the line planning problem, basic cost models assign each line a fixed or frequency-based cost term and minimize the total costs, i.e., the sum of the line costs weighted by the respective frequencies. While such a model is introduced in [Claessens et al., 1998], more sophisticated models try to approximate the costs better by already considering possible vehicle schedules and estimating the number of needed vehicles. [Bussieck et al., 2004] and [Goossens et al., 2004] both assume line-pure vehicle schedules, i.e., a line being served by the same vehicle back and forth, to achieve this. In this thesis, [Pätzold et al., 2017], see Appendix E, and [Pätzold et al., 2019], see Appendix G, choose similar approaches. For the corresponding summaries see Section 3.5 and 3.7, respectively.

Another approach is to focus on the quality of the resulting line concept from a passengers’ point of view, often using a budget for limiting the costs. First approaches are optimizing the direct travelers, see [Bussieck et al., 1997a, Bussieck, 1998], maximizing the number of passengers who can travel from their origin to their destination on a preferable path without changing lines. Different approaches to computing such preferable paths include shortest path computations or allowing certain detour factors. [Scholl, 2005, Schöbel and Scholl, 2006] present models measuring the travel time of the passenger, allowing for a more detailed optimization of the passenger convenience. Since the timetable is not known, transfer times are only approximated by fixed values. The resulting models are only solvable for small instances, therefore solution techniques such as Dantzig-Wolfe decompositions are used to improve computability. Recently, [Bull et al., 2016, Bull et al., 2018] developed a similar model, solving the problem by using multi-commodity flows. Transfer times are estimated depending on the frequencies, allowing a more detailed approximation but making the problem even harder to solve.
Several publications not only optimize one of the above objectives, but choose bicriterial approaches. [Borndörfer et al., 2007, Borndörfer et al., 2009] both use a path-based model optimizing a weighted sum of travel time and line costs and solving the problem using column generation. [Borndörfer et al., 2009] are able to compute solutions for a real-world instance and provide a comparison to the currently implemented solution.

Such bicriterial approaches are often accompanied by heuristic approaches that do not assume a given line pool but construct the lines as well. [Silman et al., 1974] present a two-stage model, first determining good lines and afterwards choosing the lines to operate. [Sonntag, 1979] chooses the approach to start with ideal lines for the passenger, i.e., to focus on passenger convenience, and afterwards iteratively adapting the lines until an operational feasible solution is achieved, i.e., costs are only considered in a second step. A similar approach can be found in [Arbex and da Cunha, 2015], where first only shortest paths for the passengers and paths with a small detour are considered in the line pool. Afterwards, for the line planning stage a genetic algorithm with alternating objectives is chosen, allowing for optimizing the costs later on. Likewise, [Viggiano, 2017] bundles passengers on corridors to find passenger-oriented but cost-sensible lines. Recently, [Harbering, 2016, Gattermann et al., 2017] propose a more general approach, i.e., a tree-based heuristic, iteratively building a line pool until a feasible line concept can be found. Here, the objective for determining new lines is variable.

Several additional concepts are also considered in line planning literature. First, there are different procedures from practical public transport planning that are integrated in traditional line planning. [Vuchic, 2017] describes the concept of a pulse or system headway to improve the memorability of a timetable based on the found line concept. Such an approach is modeled in [Friedrich et al., 2018a], see Appendix B, and summarized in Section 3.2. Another important aspect from practice is the ability to plan for varying stopping patterns, i.e., to allow lines to skip single stations during the service. How to include this in line planning models is investigated in [Goossens, 2004, Goossens et al., 2006]. At last, [Borndörfer et al., 2018a] recently considered the addition of the planning of off-peak-hours into the planning process and compared different approaches on a real-world instance.

### 2.2. Timetabling

For a given line concept, *(periodic) timetabling* describes the problem of assigning departure and arrival times for the services of the chosen lines. For a recent survey on timetabling, see [Lusby et al., 2011]. There are some success stories for the practical usage of mathematical timetabling, namely [Kroon et al., 2009] for the computation of the new Dutch timetable in 2006 and [Liebchen, 2008a] for the creation of the
2005 timetable of the Berlin subway.

Formally, periodic timetabling for a period length \( T \) often uses an event-activity network \((E, A)\) with events \( E \) and activities \( A \). For every line \( l \) in a given line concept \( L \), the set of events \( E \) contains an arrival and a departure event at every stop in \( l \). These events are connected with \textit{drive} and \textit{wait} activities. To allow transferring of the passengers, \textit{transfer} activities connect arrival and departure events of different lines at the same stop. Several other activity types, e.g. \textit{sync} or \textit{headway} activities, are possible as well and are introduced later. For each activity \( a \in A \), lower and upper bounds \( L_a \) and \( U_a \) on its duration are given. A timetable \( \pi = (\pi_e)_{e \in E} \) assigns a time to each event \( e \in E \) and is feasible if \( 
\begin{align*}
(\pi_j - \pi_i - L_a) \mod T + L_a & \leq U_a \quad a = (i, j) \in A
\end{align*}
\) is satisfied. To measure the quality of a timetable, passenger weights \((c_a)_{a \in A}\) are given for each activity \( a \in A \), denoting the number of passengers using activity \( a \). With this, an often used goal of timetabling is to minimize the total travel time, i.e.,

\[
\sum_{a \in A} c_a \cdot ((\pi_j - \pi_i - L_a) \mod T + L_a).
\]

To better evaluate the effects on the passengers, the concept of \textit{perceived travel time} is used in most of this thesis, modeling the discomfort of transfers by a penalty term. Formally, the goal is to minimize

\[
g^{\text{time}}(\pi) = \sum_{a \in A} c_a \cdot ((\pi_j - \pi_i - L_a) \mod T + L_a) + \sum_{a \in A_{\text{transfer}}} c_a \cdot \text{pen},
\]

where \( A_{\text{transfer}} \subset A \) is the set of transfer activities and pen is a penalty term for each transfer.

The most common approach to modeling periodic timetabling problems is the formulation as a periodic event scheduling problem (PESP). For the definition of PESP, see [Serafini and Ukovich, 1989]. The periodic timetabling problem can be modeled using PESP constraints, see [Odijk, 1996, Nachtigall, 1998], and the models are improved throughout the years. Extensions range from allowing variable trip times, see [Kroon and Peeters, 2003], to considering multiple frequencies, see [Peeters, 2003], and different constraints that can be modeled using PESP constraints, including fixed events, headway constraints and many more. For an overview, see [Liebchen, 2006, Liebchen and Möhring, 2007].

Since integer programming formulations of PESP are hard to solve, early solution approaches use heuristics such as genetic algorithms, see [Nachtigall and Vogt, 1996]. Later on, a special heuristic for the periodic timetabling problem, the modulo simplex, is introduced in [Nachtigall and Opitz, 2008] and further improved in [Goerigk and Schöbel, 2013]. Lately, [Goerigk and Liebchen, 2017] introduced an iterative
approach, mixing the modulo simplex and an integer programming approach. An experimental comparison of different models is presented in [Siebert and Goerigk, 2013]. Another specialized heuristic is the MATCH approach introduced in [Pätzold and Schöbel, 2016], allowing for a very fast computation of good solutions using line clusters.

To improve the performance of integer programming solvers, [Peeters and Kroon, 2001] introduced a new formulation based on cycle bases which leads to notably shorter runtimes compared to a classical PESP formulation. The advantages of using cycle bases and their properties are further investigated in [Liebchen, 2003, Liebchen and Peeters, 2009, Borndörfer et al., 2016].

Another idea is to model the periodic timetabling problem as a satisfiability problem (SAT problem). For an overview on SAT problems, see [Biere et al., 2009]. A SAT formulation of a PESP model can be found in [Großmann et al., 2012] and [Kümmling et al., 2015] use such a formulation to resolve conflicts in an overly constrained transportation system. Recently, [Matos et al., 2018] present a model to combine a SAT formulation with machine learning approaches.

For practical public transport systems, having a robust timetable, i.e., a timetable that is not easily disturbed by delays is an important property and therefore an extension of periodic timetabling that is often considered. [Parbo et al., 2016] contains a review on the effect of disturbances on the passengers and how they experience delays. Further on, [Galli and Stiller, 2018] discuss modern challenges in timetabling, including a framework for robust timetabling. To handle delays in practice, software frameworks such as PANDA, see [Müller-Hannemann and Rückert, 2017, Rückert et al., 2017], are currently tested in practice.

Since the passenger weights \( (c_a)_{a \in A} \) are fixed before the optimization, special attention needs to be given to the passenger routing step. A first routing is done before the optimization, resulting in the fixed weights used in the optimization process. Afterwards, the passengers are often routed again, since the initial paths do not need to be optimal for the resulting timetable. For information on how to find good passenger paths efficiently, see [Bast et al., 2016]. For literature on integrating the routing decision into the timetabling stage, see Section 2.4.

There are several other problems related to periodic timetabling. See e.g. [Caprara et al., 2002] for timetabling on a single track with capacity constraints, [Kinder, 2008] for a time-expanded model and [Cacchiani et al., 2010] for aperiodic timetabling.

2.3. Vehicle Scheduling

Vehicle scheduling is the problem of assigning vehicles to the different servings of lines throughout a planning horizon. For an overview, see [Daduna and Paixão, 1995, Bunte and Kliewer, 2009]. In this thesis, mostly aperiodic vehicle scheduling
is considered, i.e., some given periodic line concept $\mathcal{L}$ and timetable $\pi$ are *rolled out* for $p_{\text{max}}$ planning periods. This results in a set of *trips* $t \in \mathcal{T}$, one for each serving of a line in $\mathcal{L}$ and while the timetable is periodic, the vehicle schedule can be changed between planning periods. Two such trips are *compatible*, if there is enough time between the end of the first trip $t_1$ and the beginning of the second trip $t_2$, such that a single vehicle can serve both directly after each other, i.e., there is enough time to drive from the last station of $t_1$ to the first station of $t_2$, possibly including additional buffer in form of a minimal turnover time $L_{\text{turn}}$. The corresponding departure and arrival times are determined by the given timetable $\pi$. Compatible trips can then be combined into *vehicle routes* $(t_1, \ldots, t_n)$, where $t_i$ and $t_{i+1}$ need to be compatible for all $i \in \{1, \ldots, n - 1\}$ and a set of vehicle routes is called a *vehicle schedule* $\mathcal{V}$.

A vehicle schedule is called *feasible*, if every trip $t \in \mathcal{T}$ is covered exactly once and it is *line-pure* if every vehicle route alternately serves the backwards and forwards direction of a single line.

The objective of the vehicle scheduling stage is often cost-based since the passenger convenience is already fixed and independent of the vehicle schedule. Thus, a weighted sum of the number of vehicles, the distance driven (including empty connections between trips in a vehicle route) and the time needed (including time for empty connections between two trips in a vehicle route) should be minimized. Additionally, the objective function may contain costs for starting from a depot before each route and ending each route in a depot. We call this cost term the *operational costs* of a vehicle schedule.

For the case without a depot, [Saha, 1970] provides a minimum decomposition formulation but does not allow for empty trips between line servings. This is added in [Orloff, 1976], resulting in a model similar to the definitions mentioned above. Both publications only allow for a single type of vehicle, this is extended e.g. in [Rangaraj et al., 2006].

[Gavish and Shlifer, 1979] include the costs to drive from and to a depot into a single vehicle type context, handling the single depot case. Here, a savings problem is formulated, examining how much costs can be saved by a vehicle schedule compared to the trivial solution of serving each trip directly from the depot. Additionally, a maximal number of vehicles can be enforced. A similar problem is examined in [Paixão and Branco, 1987] and [Silva et al., 1999] where a quasi-assignment model is chosen to solve the problem. Concerning the computational complexity, [Bertossi et al., 1987] show that the single depot case is solvable polynomial time, including the case for a restriction on the number of vehicles if reasonable cost functions are chosen. Furthermore, the single depot case with general cost functions and a vehicle-restrictions as well as the multi depot case are proven to be NP-hard.

For this multi-depot case, [Carpaneto et al., 1989] provide a branch-and-bound approach and [Hadjar et al., 2006] formulate a branch-and-cut algorithm to solve the
problem. Additionally, [Kliewer et al., 2002] extend the problem to multi vehicle types, adding additional complexity to the problem. In this thesis, only a single vehicle type and at most one depot are considered.

For real-world applications, [Maróti, 2006] splits the vehicle scheduling problem into different types, ranging from tactical and maintenance routing to strategic routing and examines the different routing types separately. Realistic instances are also solved by [Reuthler and Schlechte, 2018] using a column-generation approach. Another practical aspect is the difference between periodic and aperiodic vehicle schedules, where [Borndörfer et al., 2018b] show that the problems are equivalent for an sufficiently large rollout period without a depot and when only considering the number of vehicles in the cost function.

Another important aspect is the connection to robustness, where [Borndörfer et al., 2017a] provide a template-based approach to recover from disturbances of the vehicle schedule and [van der Hurk et al., 2018] combine the rescheduling of vehicles with passenger advice, allowing to take new passenger flows into account during the planning process.

2.4. Integration

Of course, the overall goal in practice is to not only find solutions for the single planning stages, but to find a good overall system, i.e., a public transport plan \((\mathcal{L}, \pi, \mathcal{V})\) with a line concept \(\mathcal{L}\), a periodic timetable \(\pi\) and a vehicle schedule \(\mathcal{V}\) such both the passenger convenience in the timetable and the operational costs mainly determined by the vehicle schedule is optimized. Therefore looking into integrated planning is to be preferred over sequential planning.

This intent already proofed useful in other applications. [Lundqvist, 1973] provides early insights into integrating several interdependencies into urban planning. Especially for scheduling, several publications integrate other stages, see [Lenderink and Kals, 1993] and [Tan and Khoshnevis, 2000] for process planning and [Grossmann et al., 2002] for integration of general planning problems. Furthermore, [Barratt and Oliveira, 2001] discuss integration in a supply chain context and [Darvish and Coelho, 2018] compare different sequential and integrated approaches for the same problem. Other applications include the location planning for distribution centers, see [Nozick and Turnquist, 2001], or multi-modal route planning, which in itself is a form of integrated planning, since a route through multiple transportation systems is planned in an integrated fashion instead of sequentially. For an example, see [Dibbelt et al., 2015].

Due to the advances in other topics, integrated planning gained popularity in the public transport research community as well and remains an ongoing problem. For recent overviews see [Borndörfer et al., 2017c] for a collection of several success
stories in practice and the recent special issue presented by [Meng et al., 2018]. Therefore, in the following some possible integration stages are shortly described and some corresponding literature is given.

First, the integration of line planning and timetabling is discussed. [Goerigk et al., 2013] present that the consideration of later planning stages when evaluating a line concept is crucial, since the chosen lines influence the quality of the resulting timetable and may even lead to infeasibility in later stages. While [Schmidt, 2005] combines line sections into lines and sets their times integratedly, [Rittner and Nachtigall, 2009] choose a column generation approach to solve an integrated integer programming model. Other approaches often use heuristics to find solutions for the integrated problem, see e.g. [Kaspi, 2010, Kaspi and Raviv, 2013] for solving line planning with stopping patterns and timetabling using a cross-entropy heuristic or [Torres and Irarragorri, 2014] for the planning of multiple planning periods with possibly different passenger demand with two metaheuristics. More recently, [Burggraeve et al., 2017] presented an iterative approach, focusing on travel time in the line planning stage and robustness in the timetabling stage. Here, the transfer stations are restricted beforehand to reduce problem size.

Since the chosen passenger weights $c_a$ in the timetabling stage greatly influence the quality of the resulting timetable, many researchers investigate the effect of integrating the routing decision into the timetabling model instead of solving it in a preprocessing step separately. [Borndörfer et al., 2017b] show that the theoretical gap between these two approaches is unbounded. [Siebert, 2011] introduces an integrated model to solve both stages at the same time while [Schmidt, 2014] includes the routing decision additionally into other stages such as line planning and provides several NP-hardness results for the resulting problems. To deal with the computational complexity, [Gattermann et al., 2016] integrate the routing stage into the SAT model of [Großmann et al., 2012], since using SAT solvers to find solutions for periodic timetabling models is able to deliver good computational results in practice. As another approach, [Schiewe and Schöbel, 2018] present an integer programming model, including exact preprocessing methods to reduce the problem size. For line planning, [Schmidt and Schöbel, 2015a] show that integrating the routing stage results in an NP-hard problem. This is true for integrating routing into aperiodic timetabling as well, see [Schmidt and Schöbel, 2015b], even though the aperiodic timetabling problem itself is solvable in polynomial time. More recently, [Robenek et al., 2017] present a model integrating the routing into a mostly periodic plan, but with additional trips for peak hours.

One of the problems of solving periodic timetabling and vehicle scheduling sequentially is the underlying conflicts of objective functions. As discussed in Section 2.2 and 2.3, timetabling models often focus on passenger convenience, while most vehicle scheduling models try to optimize the operational costs. Solving both of these
stages independently therefore often leads to undesirable solutions w.r.t. the operational costs, since good solutions for the passengers may not allow any cost-efficient solution. Therefore, there is much research focusing on an integrated approach to solving these two planning stages. One possible approach is to consider the effects on possible vehicle schedules in the timetabling step. [Lindner, 2000] integrates cost approximations into timetabling, allowing for a model for periodic timetabling that optimizes the costs while [Dutta et al., 2017] adds some vehicle scheduling constraints into the timetabling model. A similar approach is chosen in [Pätzold et al., 2017], see Appendix E and the summary in Section 3.5. Another approach is to integrate both problems into a single integer programming model. [Schiewe, 2018] presents such a model which is still able to solve medium-sized instances with commercial solvers to optimality in a reasonable time frame. [Schmid and Ehmke, 2015] present another bi-objective model for a vehicle scheduling problem with time windows. The goal is here to balance the departure times in timetabling and it is achieved using a metaheuristic and a weighted sum approach. For aperiodic timetabling, [Ibarra-Rojas and Rios-Solís, 2011] present an integrated model, but additionally include sync intervals for the timetable, resulting in nearly periodic plans. [Cadarso and Marín, 2012] solve a similar problem with extra shunting constraints. Since solving both problems simultaneously is computationally more challenging, other research focuses on heuristic approaches. [Mandl, 1980] presents a re-optimization of the vehicle schedule afterwards, trying to reduce the passenger travel time after a vehicle schedule is fixed. Similarly, [Petersen et al., 2013] present a model to modify the timetable during the vehicle scheduling stage to reduce the operational costs without decreasing the timetable quality too much. It is solved using a large neighborhood search heuristic. Other literature includes the local optimization of both solutions after they were computed, as is e.g. presented in [van den Heuvel et al., 2008] for periodic and in [Guihaire and Hao, 2010] for aperiodic timetabling. [Yue et al., 2017] present an integrated model for aperiodic timetabling and vehicle scheduling as well, using a simulated annealing method and [Fonseca et al., 2018] present a matheuristic approach for a similar problem, changing some departures and arrivals in each iteration before computing a new vehicle schedule.

There is also some work on integrating vehicle scheduling and crew scheduling, see e.g. [Mesquita and Respício, 2009] for a branch&bound and branch&price approach for the multi-depot case.

There are some first results on integrating all three stages, namely line planning, periodic timetabling and vehicle scheduling, but due to the computational challenging aspects of such big models, only heuristic approaches are able to solve reasonable sized instances. [Lübbecke et al., 2018] present such an integrated model, examining decomposition approaches for solvability of very small instances. [Li et al., 2018] integrate aspects of line planning and vehicle scheduling into timetabling for the special
case of one single track line. Other approaches are iterative, e.g. [Liebchen, 2008b] presents an integrated model for timetabling and vehicle scheduling, which is then iterated with a line planning heuristic to compute public transport plans. [Schöbel, 2017] presents a theoretic meta-model, interpreting models for the sequential problems as nodes in a graph called eigenmodel. These nodes can then be combined in different orderings, providing different heuristics for finding a public transport plan. For more information on this model, see the discussion in Chapter 4. [Michaelis and Schöbel, 2009] present such a possible combination, starting with the vehicle scheduling in the sequential planning process.

There are also some more theoretical works on the benefit of integrating. [Lee et al., 1997] analyze the problem of not integrating in a supply chain context, while [Kidd et al., 2018] provide the value of integration for the same area. More generally, [Schiewe, 2018] defines the price of sequentiality, a measurement of the benefit of integration for general multi-stage problems and presents some theoretical results, e.g. under the assumption of some structures of objective functions and constraints. A related topic to integrated optimization is the consideration of interwoven problems, i.e., multiple optimization problems that are not structured hierarchical as in the cases of integrated optimization considered in this thesis but coequally with a shared set of variables and associated constraints. For a general introduction, see [Klamroth et al., 2017].
3. Paper Summaries

In this chapter, the publications of this thesis are summarized. The following publications are included.

First, [Friedrich et al., 2017a], see Appendix A, is summarized in Section 3.1. Here, a benchmark dataset for comparing and understanding manual and algorithmic solutions is created and analyzed. The dataset is afterwards used for computational experiments in all publications of this thesis.

To optimize the costs of a public transport plan, first the influence on a single problem stage, namely line planning, is examined. Despite being a well researched topic in public transport planning, there are practical requirements on a line concept that were not considered before in the mathematical literature. One such requirement, namely a system headway, is examined in Section 3.2, especially with respect to a cost-oriented model. This section is a summary of [Friedrich et al., 2018a], see Appendix B.

The next two sections extend the focus from line planning to considering the cost-oriented integration of load generation into line planning. First, in Section 3.3, several passenger distribution algorithms, including newly designed algorithms and algorithms from the literature, are compared and analyzed. This section is a summary of [Friedrich et al., 2017b], see Appendix C. Afterwards, Section 3.4 summarizes a game-theoretic model presented in [Schiewe et al., 2019], see Appendix D, interpreting the passengers as players. Here, the operational costs are distributed to the passengers, providing a motivation to find a line concept with low costs. Several theoretical results regarding equilibrium solutions are presented.

Afterwards, the problem is again extended to finding complete cost-oriented public transport plans. For this, two heuristics are presented. Section 3.5 presents a sequential approach introduced in [Pätzold et al., 2017], see Appendix E, where the operational costs are considered in every stage, allowing for more cost-efficient solutions. In Section 3.6, an already existing system is re-optimized, fixing two of the three stages line planning, timetabling and vehicle scheduling in each step and improving the remaining stage. The resulting problems are modelled mathematically, algorithms to solve them are proposed and convergence of a resulting iterative algorithmic scheme is examined theoretically. This is a summary of [Schiewe and Schiewe, 2018], see Appendix F.

In the end, Section 3.7 describes a completely integrated approach presented in [Pätzold et al., 2019], see Appendix G, i.e., a model to compute a cost-minimal
public transport plan from scratch in one model. Since such an approach is computationally not competitive for real-world instances, several smaller models are presented with computable bounds on the solution quality and special cases are identified where the optimal solution can be found by the models which are easier to solve.

3.1. Public Transport Planning - Manually Generated and Algorithmic Solutions\textsuperscript{1}

There exist two different approaches to public transport planning used by practical public transport planners and mathematicians, respectively. On the one hand, practical public transport planners often design solutions manually, using computer-aided analysis techniques to evaluate the solutions found. On the other hand, the more theoretical approach is to use mathematical optimization tools for a systematic search of the solution space.

Despite promising to find optimal solutions, mathematical optimization has only found its way into a few planning stages in the real world, especially vehicle and crew scheduling. Other stages, such as line planning and timetabling, are still mostly done manually in practice.

In [Friedrich et al., 2017a], Appendix A, the authors compare these two approaches and analyze the differences of the methods and solutions. To achieve this, a benchmark dataset is proposed, containing all information and simplifications necessary to allow both mathematicians and practical public transport planners to create solutions. Additionally, the dataset should be small enough to still understand the different solutions but large enough to provide meaningful feedback. The created dataset is used throughout this thesis for evaluation of the developed algorithms.

To define such a dataset, first several input parameters need to be set. A total of 25 stops are created and arranged in a grid-layout, see Figure 2a. To simplify the instance, unified edge lengths and vehicle speeds are proposed. Afterwards, VISUM, see [PTV Group, 2016], is used to create a realistic demand structure for 30,000 commuters, resulting in a total of 2,531 passengers in the considered peak morning hour. The corresponding demand is depicted in Figure 2b. Afterwards, parameters are fixed for the evaluation of the two objective functions considered here, namely the operational costs and the perceived travel time of the passengers, i.e., the travel time with a penalty for each transfer. The resulting dataset is called Grid in the rest of this thesis.

\textsuperscript{1}Original title: Angebotsplanung im öffentlichen Verkehr - Planerische und Algorithmische Lösungen
For creating the comparative solutions, the two different approaches mentioned earlier are used:

**Manual Approach** To create a solution by hand, first the lines need to be designed. Here, an axisymmetric (P_1) and a point-symmetric (P_2) solution are created. A system headway is used to improve clarity for the planner and memorability for the passengers, i.e., a frequency of 3 is used for each line. The resulting lines and frequencies for P_2 are depicted in Figure 3a. Afterwards, the central node is used as a main transfer node. The driving times of the lines are based on line-pure vehicle schedules and the lines are then shifted to allow for good transfers at the central node.

**Algorithmic Approach** To create a solution automatically, several optimization algorithms implemented in the open-source software framework LinTim, see [Schiewe et al., 2018a], are used. First, the lines are generated using an algorithm proposed in [Gattermann et al., 2017]. Afterwards, an integer program for a cost-oriented formulation is used to determine the frequencies of the lines, see [Claessens et al., 1998, Schöbel, 2012]. The resulting lines and frequencies are depicted in Figure 3b. To determine the timetable, a PESP model is solved using a modulo simplex heuris-
tic, see [Serafini and Ukovich, 1989, Goerigk and Schöbel, 2013]. In the end, the vehicle schedules are determined using a model optimizing the operational costs of the vehicle schedule, see [Bunte and Kliewer, 2009, Uffmann, 2010].

For the algorithmic solutions, several different starting solutions are used. All algorithmic solutions use the traffic load provided by the two manual solutions P_1 or P_2.

- A_1_1 and A_2_1 - Manual line concept + algorithms: The lines and frequencies are fixed to the manual solution, other stages are solved with the above algorithms.

- A_1_2 and A_2_2 - Manual lines + algorithms: The lines are fixed to the manual solution, other stages are solved with the above algorithms.

- A_1_3 and A_2_3 - Straight lines + algorithms: The line pool is fixed to ten straight lines, other stages are solved with the above algorithms.

- A_1_4 and A_2_4 - Algorithms from scratch: All stages are solved with the above algorithms.

- A_1_5 and A_2_5 - Algorithms from scratch + manual lines: All stages are solved with the above algorithms but the line pool is extended by the manual lines.

For all the above solutions, the operational costs and the average perceived travel time for the passengers are computed and depicted in Figure 4. Especially for
solutions based on P_2, the travel time can be decreased significantly when planning different stages with algorithms instead of manually, see Figure 4b. This is mainly due to better synchronization of the transfers for the passengers. Reducing the costs is more challenging for the algorithm solution procedure, since the operational costs mainly depend on the vehicle schedules which are not known in the beginning and can only be approximated for the algorithms used here. However, solution A_1_2 is able to decrease the frequency of one line, preserving feasibility and reducing the costs, see Figure 4a. Due to the used system headway, this is not possible for the manually created solutions. Note that the vehicle schedules found with VISUM are always optimal in the solutions discussed here, i.e., they cannot be further improved using the optimization algorithms mentioned above.

Another important aspect is the improvement going from A_1_4 to A_1_5 or from A_2_4 to A_2_5 respectively. Both costs and passenger convenience can be improved by including the manual lines in the automatically generated line pool. The authors therefore conclude that especially for the line generation step, the experience of manual planners is still beneficial to improve the overall solution.

Note that the dataset created by the authors is published as [FOR2083, 2018] and sparked an ongoing competition for creating competitive solutions. Several publications, namely [Friedrich et al., 2017c, Friedrich et al., 2018b, Liebchen, 2018], used the dataset to evaluate and compare their approaches to the currently 73 uploaded solutions. Similarly, all publications summarized in this thesis use dataset Grid for evaluation of the developed algorithms.
3.2. System Headways in Line Planning

As discussed in Section 2.1, line planning is a well researched problem. There are several models in the literature with various objectives, e.g. for optimizing costs, see [Claessens et al., 1998], as well as passenger-oriented models such as direct traveler approaches, see [Bussieck, 1998], or travel time approaches, see [Schöbel and Scholl, 2006, Schmidt, 2014]. But solutions obtained by above models often fall short with respect to objectives that are hard to measure but used in practice, e.g. the memorability of the created system. A common concept to achieve memorability is a system or pulse headway, see [Vuchic, 2017], allowing for regular departures and transfers of the passengers. To incorporate this important practical aspect into mathematical line planning models, especially into cost-oriented ones, is a new approach presented in [Friedrich et al., 2018a], see Appendix B.

The authors define a system headway as a common divisor of the frequencies of all lines, i.e., for a given line concept \( \mathcal{L} \) with frequencies \( f_l \) for line \( l \in \mathcal{L} \) a common divisor \( i \neq 1 \) of all \( f_l \) is called a system headway. With this, the requirement of a system headway can be included in a general line planning model, i.e., extending

\[
\begin{align*}
(P) \quad & \text{min } \text{obj}(f, x) \\
& \text{s.t. } g(f, x) \leq b \\
& \quad f_l \in \mathbb{N}_0 \quad l \in \mathcal{L}^0 \\
& \quad x \in X
\end{align*}
\]

to

\[
\begin{align*}
(P(i)) \quad & \text{min } \text{obj}(f, x) \\
& \text{s.t. } g(f, x) \leq b \\
& \quad f_l = \alpha_l \cdot i \quad l \in \mathcal{L}^0 \\
& \quad \alpha_l \in \mathbb{N}_0 \quad l \in \mathcal{L}^0 \\
& \quad f_l \in \mathbb{N}_0 \quad l \in \mathcal{L}^0 \\
& \quad x \in X
\end{align*}
\]

where \( \text{obj}(f, x) \) is an objective function dependent on the frequencies and some auxiliary variables \( x \) and with general constraints \( g(f, x) \leq b \), some variable domain \( X \), a line pool \( \mathcal{L}^0 \) and a (fixed) system headway of \( i \). The authors first analyze the complexity of the arising formulation and derive the following theorem.

**Theorem 2.1** ([Friedrich et al., 2018a], Theorem 1). Let \( (P) \) be a general line planning problem for a given instance based on a fixed planning period. Then problem \( P(i) \) is equivalent to a line planning problem \( (P') \). The new line planning problem \( (P') \) has the same number of variables and constraints as \( (P) \).
The authors also provide a formulation $(P_{\text{sys-head}})$ to find the best system headway $\alpha$ for a given problem instance, i.e., to find a line concept with a system headway, without fixing it beforehand. Since the provided formulation is a quadratic integer program and therefore not competitive in practice, further analysis of good system headway values is provided. The authors derive the property that for a given number $i$ as a system headway, divisors of $i$ always provide better system headway values, see [Friedrich et al., 2018a], Lemma 1, resulting in the following corollary and limiting the search space for optimal system headways immensely.

**Corollary 2.2** ([Friedrich et al., 2018a], Corollary 1). There always exists an optimal solution $(\alpha, f, x)$ to $(P_{\text{sys-head}})$ in which the optimal system headway $\alpha$ is a prime number.

Apart from divisors, there are no known practical conditions on the relation between different system headway values, e.g. there are cases where a smaller system headway may have worse objective value or may even be infeasible. Examples for both cases are given for common constraint types and objective functions. Additionally the authors provide classes of line planning problems where the feasibility of system headway solutions can be guaranteed, see [Friedrich et al., 2018a], Lemma 3.

Furthermore, it is possible to determine a priori bounds in special cases. For this, the authors consider a cost-oriented model without upper frequency bounds, i.e., the problem

$$
\begin{align*}
\min & \quad \sum_{l \in \mathcal{L}^0} f_l \cdot \text{cost}_l \\
\text{s.t.} & \quad f_{e}^{\min} \leq \sum_{\substack{l \in \mathcal{L}^0: e \in l \cap \mathcal{L}^0}} f_l & e \in E \\
& \quad f_l = \alpha_l \cdot i & l \in \mathcal{L}^0 \\
& \quad f_l, \alpha_l \in \mathbb{N}_0 & l \in \mathcal{L}^0
\end{align*}
$$

for given costs $\text{cost}_l$ for every line $l \in \mathcal{L}^0$.

For this problem, the worst case ratio of the optimal objective values $\text{opt}(i)$ and $\text{opt}(j)$ for system headways $i$ and $j$ can be determined beforehand.

**Theorem 2.3** ([Friedrich et al., 2018a], Theorem 2). Let $i, j \in \mathbb{N}$, $i \leq j$. Then $\text{opt}(j) \leq \frac{1}{i} \text{opt}(i)$.

Luckily, these rather high theoretical bounds are not realized in practice, as can be seen in the experimental evaluations, see e.g. Figure 5a.

Unfortunately, the authors show that it is not possible to determine such bounds for passenger-oriented models. These often work with budget constraints to prevent
a trivial system that is optimal for every passenger but to costly for the operator. But when such constraints are used it is not possible to guarantee feasibility for different system headways or provide bounds on the objective values beforehand.

To check the practical effects of system headways, the authors provide experimental evaluations on three different datasets, the benchmark dataset Grid created in [Friedrich et al., 2017a] and close-to real-world datasets Goettingen and Germany, representing the bus network of Göttingen and the long-distance railway network of Germany, respectively. All experiments are done using the open-source software framework LinTim, see [Schiewe et al., 2018a]. For each dataset, solutions are created for every system headway value from 2 to 10, using a cost-oriented and a direct traveler model with a budget. Additionally, a solution without a system headway is computed as a reference value, marked with 1 in the figures presented here. In Figure 5, solutions are depicted for dataset Grid and dataset Germany. Figure 5a shows that despite increasing the costs for higher system headway values, the theoretical bound is not reached in practice. Additionally, it is not always true that a higher system headway leads to higher costs, see e.g. the different cost values for system headways of 2 and 3 which results from the demand structure of the used dataset. For the direct traveler model, Figure 5b provides the insight that increasing the system headway results in worse objective values due to the inability to fill the budget efficiently. Again, removing the budget would solve this problem but would result in trivial solutions for all system headway values.
As discussed extensively in the literature, see e.g. [Goerigk et al., 2013, Burggraeve et al., 2017, Schöbel, 2017], line planning solutions should not be considered isolated from later planning stages. To check the influence of the computed solutions on the travel time of the passengers, a periodic timetable is computed for each line plan, using the heuristic MATCH approach, see [Pätzold and Schöbel, 2016]. Some of the results are depicted in Figure 6. Overall, a higher system frequency seems to provide a denser system, allowing faster travel and transfer times of the passengers. But again, this is not always the case, sometimes leading to an increase in travel time when the system headway is increased.

3.3. Integrating Passengers’ Assignment in Cost-Optimal Line Planning

Line planning is a well researched topic in public transport planning, see e.g. [Schöbel, 2012]. As for almost all problems in public transport planning, the quality of a line concept depends on the quality of the earlier stages, since traditional approaches are two-stage: First, the passengers are distributed to the infrastructure network before the resulting traffic loads are used as an input for line planning problems, see e.g. [Bussieck et al., 1997a, Claessens et al., 1998].

In [Friedrich et al., 2017b], see Appendix C, the authors present an analysis of the gap resulting from using this two-stage approach in cost-oriented line planning, develop an integrated model to solve both stages simultaneously and compare several
algorithms for passenger distribution. The algorithms are later on evaluated on a benchmark dataset.

Algorithm 3.1 Sequential approach for cost-oriented line planning
1: Input: PTN \((V, E)\), \(W_{uv}\) for all \(u, v \in V\), line pool \(\mathcal{L}^0\) with costs \(c_l\) for all \(l \in \mathcal{L}^0\), vehicle capacity Cap
2: Compute traffic loads \(w_e\) for every edge \(e \in E\) using a passengers’ assignment algorithm (Algorithm 3.2)
3: Solve the line planning problem \(\text{LineP}(w)\) and receive \((\mathcal{L}^0, f_l)\)

First, the authors formally define the traditional sequential approach for cost-oriented line planning, see Algorithm 3.1. Next to the infrastructure network PTN \((V, E)\) and a vehicle capacity Cap, the input contains a passenger demand given as an OD matrix \(W\) with entries \(W_{uv}\) stating the demand from stops \(u\) to \(v\) in the planning period. First, traffic loads are determined using a separate algorithm, transforming the OD matrix into a load \(w = (w_e)_{e \in E}\) on the edges \(e \in E\) of the PTN. Afterwards, lines are chosen from a given line pool \(\mathcal{L}^0\) such that the sum of the given line costs \(\text{cost}_l\) are minimized and the traffic loads are covered for every edge, i.e., the goal is to find a solution for the following line planning problem.

\[
\text{LineP}(w) \quad \min \sum_{l \in \mathcal{L}^0} f_l \cdot \text{cost}_l \\
\text{s.t.} \quad \sum_{l \in \mathcal{L}^0: e \in l} f_l \geq \frac{w_e}{\text{Cap}} \quad e \in E \\
\quad f_l \in \mathbb{N} \quad l \in \mathcal{L}^0
\]

Algorithm 3.2 Passengers’ assignment algorithm
1: Input: PTN \((V, E)\), \(W_{uv}\) for all \(u, v \in V\)
2: for every \(u, v \in V\) with \(W_{uv} > 0\) do
3: \hspace{1em} Compute a set of paths \(P_{uv}\) from \(u\) to \(v\) in the PTN
4: \hspace{1em} Estimate weights for the paths \(w_p \geq 0, p \in P_{uv}\) with \(\sum_{p \in P_{uv}} w_p = W_{uv}\)
5: end for
6: for every \(e \in E\) do
7: \hspace{1em} Set \(w_e := \sum_{u,v \in V} \sum_{p \in P_{uv}} w_p\)
8: end for

Of course, the distribution algorithm used in line 3 of Algorithm 3.1 is crucial. The general procedure can be found in Algorithm 3.2. For every OD pair, a set of
paths and weights is computed. These paths are afterwards accumulated to traffic loads on the edges of the PTN.

A common approach is to use a shortest path algorithm in line 3 of Algorithm 3.2. However, since this may lead to good solutions for the passengers but not for the operational costs, this approach should not be the only considered possibility. But only considering the costs in this step may lead to unintended solutions as well, as can be shown in an example provided by the authors where the travel time of the passengers is unbounded when only the costs are optimized, see [Friedrich et al., 2017b], Example 1.

The integrated model proposed here therefore contains a detour factor, allowing to restrict the maximal lengths of the computed passenger paths w.r.t. the shortest possible path in the network. To analyze the differences between different shortest-path assignments, two examples are given where the sequential solution is worse w.r.t. the line costs than the integrated solution. This is especially the case for specific line pools, where the gap may be unbounded. But even when the complete line pool, i.e., the pool containing all possible paths in the PTN, is considered, a gap between two different shortest path assignments can be observed. However, the authors are able to provide a worst-case bound for this case.

**Lemma 3.1** ([Friedrich et al., 2017b], Lemma 5). Consider two shortest-path-based assignments $w$ and $w'$ for a line planning problem with a complete pool $\mathcal{L}^0$ and without fixed costs. Let $f_l, l \in \mathcal{L}^0$, be the cost optimal line concept for LineP($w$) and $f'_l, l \in \mathcal{L}^0$, be the cost optimal line concept for LineP($w'$). Then

$$\sum_{l \in \mathcal{L}^0} \text{cost}_l f_l \leq |OD| \sum_{l \in \mathcal{L}^0} \text{cost}_l f'_l,$$

where $|OD|$ denotes the number of non-zero entries in the OD matrix $W$.

The authors show that the bound needs to be increased to the number of passengers if the passengers of an OD pair are allowed to choose different paths and that it is equal to 1 if the LP relaxations of LineP is considered.

To examine the effects of load generation in practice, three algorithms are compared:

- A shortest-path approach SP, routing all passengers of an OD pair on the same shortest path

- A reduction algorithm Reduction, originally developed in [Hüttemann, 1979]. This is an iterative approach, where a higher passenger load leads to reduced costs of an edge in the subsequent iteration. In the end, a shortest path routing where all formerly unused edges are forbidden determines the final traffic loads.
• A new algorithm **Reward**, similar to **Reduction**, but rewarding not the pure passenger load on an edge but the number of places left until the next vehicle is needed, i.e., an edge gets lower costs if the vehicles on this edge are used efficiently.

Additionally, all passenger distributions are used in a variant where the routing is computed in the Change&Go-network (CGN), see [Schöbel and Scholl, 2006], a network where passengers can be distributed to different lines, allowing for a more precise approximation of the transfers needed and the vehicle usage. This is therefore deemed to be especially promising for the **Reward** heuristic.

![Performance graphs](a) Performance for a line pool with 33 lines (b) Performance for a line pool with 275 lines

Figure 7: Performance of the distribution algorithms for different line pools on datasets **Grid**

After formally defining the three distribution algorithms, the authors measure their performance using the dataset **Grid** combined with 5 different line pools, ranging from 33 to 275 lines. For every solution, a periodic timetable is computed and evaluated to determine the perceived travel time of the passengers. The performance for two line pools is depicted in Figure 7. As expected, the solution with the lowest perceived travel time is always provided by a **SP** distribution, but especially for the large line pool, the cheapest solution provided by using **Reduction** on a PTN is not much worse w.r.t. perceived travel time but can improve the costs drastically. Especially for small line pools, **Reward** in combination with a CGN can provide very cheap solutions but this effect as well as the benefit of the CGN itself decreases with line pool size.
The authors determine that the last step of Algorithm Reduction is crucial, namely the rerouting on shortest paths when forbidding formerly unused edges. For this, a representation of the iteration steps of the algorithms is given in [Friedrich et al., 2017b], Figure 5a. The final solutions always dominate the last iteration of the algorithm.

To evaluate the real-world competitiveness, the overall cheapest solution found was additionally combined with a vehicle schedule and evaluated using LinTim, see [Schiewe et al., 2018a], and VISUM, see [PTV Group, 2016]. In the ongoing competition, started in [Friedrich et al., 2017a], see Section 3.1 and Appendix A, to provide good solutions for dataset Grid, the solution found here was the cheapest completely automatic solution found to the time of the original publication.

3.4. The Line Planning Routing Game

In [Schiewe et al., 2019], see Appendix D, the authors present a new, game-theoretic approach to line planning. Instead of determining system-optimal solutions, the passengers are interpreted as players and are therefore able to shape the solution themselves. This can be interpreted as integrating load generation into line planning and, since a share of the operational costs is part of the individual cost functions of the players, the overall goal is to create a cost-efficient solution that still benefits the passengers.

First, the authors define the problem they aim to solve. The line planning problem with travel quality and cost objective is defined as follows.

**Definition 4.1** ([Schiewe et al., 2019, Definition 3.1]). Given a PTN $(V, E)$, a line pool $L^0$, a vehicle capacity $\text{Cap}$, a set of passengers $Q$, a parameter set $(\alpha_1/\alpha_2, \beta, \gamma_1/\gamma_2)$, and a period length $T$, the line planning problem with travel quality and cost objective (LPQC) is defined as follows: Find a pair of frequencies $f$ and routes $R$ which fulfills $z_{(e,l)}(R) \leq f_l \cdot \text{Cap}$ and minimizes the objective function

$$H(R, f) := \sum_{q \in Q} \left( \alpha_1 \cdot c_q(R_q) + \alpha_2 \cdot \tau_q(R_q, f) + \beta \cdot \text{transfer}_q(R_q) \right)$$

$$+ \gamma_1 \cdot \sum_{l, f_l > 0} k^1_l + \gamma_2 \cdot \sum_{l, f_l > 0} k^2_l f_l$$

Here, for passenger $q \in Q$, $c_q(R_q)$ is the in-vehicle time, $\tau_q(R_q, f)$ is an approximation of the transfer time, relative to the frequencies $f$, $\text{transfer}_q(R_q)$ measures the number of transfers and $k^1_l, k^2_l$ are cost factors with and without respect to the
frequency of line \( l \). Additionally, \( x_{(e;l)}(\mathcal{R}) \) is the number of passengers using the infrastructure edge \( e \in E \) with line \( l \) in routing \( \mathcal{R} \). This definition allows the authors to use a very detailed evaluation of a line concept.

To find solutions to this problem which are not only good on average but really represent the behavior of the passengers, the authors define the outline of the line planning routing game as follows.

**Definition 4.2** ([Schiewe et al., 2019, Definition 3.3]). In the line planning routing game (LPRG), the passengers \( q \in Q \) act as players. Every passenger (player) chooses among the routes from his origin to his destination (strategies) to minimize his individual objective function \( h_q(R_q, \mathcal{R}^{-q}) \) which depends both on the route \( R_q \) chosen by \( q \) and the routes chosen by the other passengers \( \mathcal{R}^{-q} \).

The objectives of the passengers are set to be a weighted sum of the travel quality \( t_q \) of the passengers introduced in Definition 4.1 and a share of the overall costs. The cost share is set to be line-based, i.e., the passengers share the costs of all lines they use relative to the total number of passengers using each line, or edge-based, where the costs of a line are distributed to its edges and passengers only share costs for edges they actually use.

Afterwards, the authors analyze the relation between the line planning problem LPQC and the line planning routing game LPRG. Since for any routing \( \mathcal{R} = (R_q)_{q \in Q} \) the frequencies of all lines \( l \) in a given line pool \( \mathcal{L}^0 \) can be set by

\[
f_l(\mathcal{R}) := \max_{e \in l} \left[ \frac{x_{(e;l)}(\mathcal{R})}{\text{Cap}} \right],
\]

every equilibrium of LPRG can be interpreted as a solution to LPQC.

On the other hand, the authors show that every optimal solution to LPQC is a system-optimum for LPRG, since the sum of the individual players objective function is the objective function of LPQC. Note that for edge-based costs this is only true under the assumption that there is no unused edge covered by an operated line in the network. With this observation, the first relation between equilibria of LPRG and solution quality of LPQC is stated, namely that if the price of anarchy, i.e., the worst case bound between system-optimal and equilibrium solutions, is bounded by \( \xi \), every equilibrium to LPRG is a \( \xi \)-approximation for LPQC. But even though the objective value of the solutions found by LPRG may be worse than the optimal objective value of LPQC, the equilibrium solutions found by LPRG are more balanced, i.e., the benefit of a single passenger is not sacrificed for the ‘greater good’. This may very well happen in system-optimal solutions, i.e., optimal solutions to LPQC, as is shown in an example by the authors.

To determine equilibria for LPRG, a **best response algorithm**, outlined in Algorithm 4.1, is used. For using such an algorithm efficiently, it is important that the
Algorithm 4.1 [Schiewe et al., 2019, Algorithm 1]

**Input:** PTN, line pool, set of passengers $Q$, individual objective functions $h_q$, maximal number of iterations $m \in \mathbb{N} \cup \infty$

**Output:** A route set $\mathcal{R}$

Start with an empty route set (or with an arbitrary non-empty route set)

while improvements for the passengers possible and $m$ not reached do

for passenger $q \in Q$ do

Calculate optimal passenger route $R_q$ according to $h_q$

end for

end while

Routing step can be solved in polynomial time. The authors therefore identify cases where this is not the case, namely when using a line-based cost-share, see [Schiewe et al., 2019, Theorem 4.2] or frequency-based transfer times, see [Schiewe et al., 2019, Theorem 4.3] and a case where the routing can be done efficiently, namely the case of edge-based costs, see [Schiewe et al., 2019, Lemma 4.4]. In the following, efficient heuristics are discussed to approximate line-based costs and frequency-based transfer times and the convergence of these heuristics to equilibrium solutions is analyzed.

First, the authors show that convergence to an equilibrium is not guaranteed in general, but using the concept of potential functions from game theory literature, they determine criteria for convergence, namely for (individual) objective functions of the players of the form

$$h_q(R_q, \mathcal{R}^{-q}) = \sum_{a \in R_q} \tilde{w}_a(x_a),$$

(3.1)

where the cost of an edge $a$ in the path of a player only depends on the number of passengers $x_a$ using edge $a$ and not on the rest of the network. With this, a convergence guarantee is formulated.

**Lemma 4.3** ([Schiewe et al., 2019, Lemma 4.5]). Let $I$ be an instance of the LPRG with $I := (PTN, \mathcal{L}^0, Q, \{h_q : q \in Q\})$ such that edge weight functions as specified in (3.1) exist. Then

1. $\Phi(\mathcal{R}) := \sum_{a \in \mathcal{A}} \sum_{i=1}^{x_a(\mathcal{R})} \tilde{w}_a(i)$ is a potential function for $I$,

2. there exists an equilibrium to $I$,

3. Algorithm 4.1 converges to an equilibrium in a finite number of steps,

4. each of the steps can be executed in polynomial time.
Note, that the quality of the equilibrium can be bad, i.e., the algorithm is not guaranteed to converge to a good equilibrium and additionally, there may be system-optimal solutions that are no equilibria and therefore cannot be found using Algorithm 4.1. Both cases are shown in examples provided by the authors.

Nevertheless, the quality of the solution is bounded by the price of anarchy and the authors identify cases, where the price of anarchy can be bounded itself. The provided bound is sharp, as is shown in an example.

Lemma 4.4 ([Schiewe et al., 2019, Lemma 4.6]). If there exist non-increasing edge weight functions $w_a$, $a \in A$ with $w_a(1) \leq x \cdot w_a(x)$ for all $x \in \mathbb{N}$, the price of anarchy in LPRG is at most the number of passengers.

This is especially the case for edge-based cost functions that do not depend on frequency-based costs and transfer times. For other cases, the authors develop heuristic objective functions, approximating the cases without convergence-guarantee and satisfying the prerequisites of Lemma 4.4. For this, two heuristics are discussed by the authors:

- **Auxiliary frequencies**: Identify critical lines, i.e., lines that need to increase their frequency if an additional passenger is using them, and assume that all are used by a passenger, i.e., providing a lower bound on transfer time and an upper bound on costs, or none are used by a passenger, providing an upper bound on transfer time and a lower bound on costs. These two approximations can be combined into an overall lower and upper bound on the individual passenger objectives. Both result in a routing step that is solvable in polynomial time, see [Schiewe et al., 2019, Lemma 4.9], but there is no guarantee for equilibrium convergence.

- **Auxiliary arc weights**: Extend the idea for the auxiliary frequencies, but now assume that all passengers use a transfer or line for the one case or only the passengers using the current edge use a transfer or line for the other case. This again provides lower and upper bounds for transfer time and costs and can be combined accordingly to provide a lower and an upper bound on the original passenger objective, as above. This construction satisfies the requirements of Lemmas 4.3 and 4.4 and therefore guarantees convergence to an equilibrium with bounded objective value.

The authors additionally show that every objective function satisfying the requirement of Lemma 4.4 also allows for a bound on the objective function after one iteration of Algorithm 4.1 which is the same as the bound on the price of anarchy, see [Schiewe et al., 2019, Lemma 4.8].

In the end, the authors provide an extensive computational evaluation of the provided objective functions and the game-theoretic approach itself. For this, solutions
on dataset Grid are computed using Algorithm 4.1 with the original objective (BR) and all discussed heuristics (AF ub, AF lb, AW ub, AW lb) as well as an integer programming approach to LPQC. Here, only two observed properties are discussed:

- The solutions found by Algorithm 4.1 are indeed more balanced than the system-optimal solutions found by LPQC, i.e., the path lengths for the passengers in the equilibria found by BR or the heuristics do not differ for any given OD pair. This is not the case for the system-optimal solution. Details on the deviations for LPQC can be found in Table 4.1.

- The equilibria found by the different objective functions are not system-optimal, but Algorithm 4.1 is able to provide solutions much faster than an integer programming approach to LPQC, as detailed in Table 4.2.

<table>
<thead>
<tr>
<th>relative objective</th>
<th>runtime</th>
<th># iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPQC</td>
<td>5:36</td>
<td>-</td>
</tr>
<tr>
<td>BR</td>
<td>0:14</td>
<td>7</td>
</tr>
<tr>
<td>AF ub</td>
<td>0:23</td>
<td>6</td>
</tr>
<tr>
<td>AF lb</td>
<td>0:26</td>
<td>7</td>
</tr>
<tr>
<td>AW ub</td>
<td>0:14</td>
<td>7</td>
</tr>
<tr>
<td>AW lb</td>
<td>0:12</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 4.1.: Comparison of solutions for LPQC and LPRG on dataset Grid

<table>
<thead>
<tr>
<th>relative objective</th>
<th>runtime</th>
<th># iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPQC</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>BR</td>
<td>1.391</td>
<td>0:14</td>
</tr>
<tr>
<td>AF ub</td>
<td>1.357</td>
<td>0:23</td>
</tr>
<tr>
<td>AF lb</td>
<td>1.481</td>
<td>0:26</td>
</tr>
<tr>
<td>AW ub</td>
<td>2.329</td>
<td>0:14</td>
</tr>
<tr>
<td>AW lb</td>
<td>1.391</td>
<td>0:12</td>
</tr>
</tbody>
</table>

Table 4.2.: Comparison of solutions for LPQC, the heuristics and BR on dataset Grid, runtime in min:sec
3.5. Look-Ahead Approaches for Integrated Planning in Public Transportation

Optimizing operational costs in the sequential planning process of public transport planning is very difficult, since the costs can only be correctly evaluated after computing a vehicle schedule, which is done after the line concept and the timetable are already fixed. To reduce the operational costs, in [Pätzold et al., 2017], see Appendix E, the authors propose several approaches to improving the approximation of the costs in the line pool generation, the line planning and the timetabling stage and therefore enhance the optimization of the costs while maintaining the sequential approach.

To evaluate a public transport plan \((L, \pi, V)\), consisting of a line concept \(L\), a timetable \(\pi\) and a vehicle schedule \(V\), costs are computed using weight parameters \((c_1, \ldots, c_5)\), i.e.,

\[
g^\text{cost}(L, \pi, V) := c_1 \cdot \text{dur}^{\text{full}} + c_2 \cdot \text{len}^{\text{full}} + c_3 \cdot \text{veh} + c_4 \cdot \text{dur}^{\text{empty}} + c_5 \cdot \text{len}^{\text{empty}}.
\]

Here, \(\text{dur}^{\text{full}}\) and \(\text{dur}^{\text{empty}}\) are the full and empty duration of the vehicle schedule, i.e., the time spend while serving a line or serving an empty or connecting trip, respectively, \(\text{len}^{\text{full}}\) and \(\text{len}^{\text{empty}}\) are the full and empty distance, i.e., the distance driven to serve a line or a connecting trip, respectively, and veh is the number of vehicles necessary to operate the vehicle schedule. To evaluate the passengers convenience, the perceived travel time \(g^\text{time}(\pi)\) is used.

Therefore, the authors consider the following problem.

**Problem.** Find a feasible public transport plan \((L, \pi, V)\) that minimizes the two objectives \(g^\text{cost}(L, \pi, V)\) and \(g^\text{time}(\pi)\).

Since \(g^\text{cost}(L, \pi, V)\) can only be computed after the vehicle schedule is known, the authors concentrate on approximating this objective already in earlier planning stages, allowing for public transport plans with lower overall costs. Therefore, the following three improvements to the sequential planning process are proposed.

**Improvement 1: New Costs for Line Planning** To approximate the operational costs in the line planning stage, the authors assume line-pure vehicle schedules to be used to provide an upper bound, i.e., every vehicle serves a single line and its
backwards direction. This reduces the empty distance length to zero. The empty duration can be easily computed in such a model with

\[
\text{empty duration after serving line } l = \frac{T}{2} - (\text{dur}_l \mod \frac{T}{2}),
\]

where \(\text{dur}_l\) is the duration of the line, fixed by the lower travel time bounds on the edges in the PTN. Similar, the number of vehicles needed for the operation of a line can be computed by

\[
\#\text{vehicles needed for line } l \text{ and backwards direction} = \left\lceil 2 \cdot (\text{dur}_l + \text{L}_{\text{turn}})/T \right\rceil,
\]

where \(\text{L}_{\text{turn}}\) is the minimal turnover time between the service of two lines.

Using these two formulas, the cost of a line in the line planning stage is replaced by

\[
\text{cost}_l = 2 \cdot c_1 \cdot \text{dur}_l + 2 \cdot c_2 \cdot \text{len}_l + \frac{c_3}{p_{\text{max}}} \cdot \left\lceil 2 \cdot \frac{\text{dur}_l + \text{L}_{\text{turn}}}{T} \right\rceil + 2 \cdot c_4 \cdot \left(\frac{T}{2} - \text{dur}_l \mod \frac{T}{2}\right),
\]

where \(\text{len}_l\) is the length of line \(l\) and \(p_{\text{max}}\) is the number of time periods covered by the vehicle schedule.

**Improvement 2: A New Line Pool** To include more lines which are well suited for a line-pure vehicle schedule into the line pool, the authors adapt the algorithm described in [Gattermann et al., 2017]. Here, the goal is to only construct lines which can be used efficiently in a line-pure vehicle schedule, i.e., without too much buffer time when serving the line and its backward direction consecutively. This is achieved by introducing the inequality

\[
\frac{T}{2} - \text{L}_{\text{turn}} - \alpha \leq \text{dur}_l \mod \frac{T}{2} \leq \frac{T}{2} - \text{L}_{\text{turn}} - \alpha
\]

as a feasibility constraint into the algorithm. Forward and backward direction of a line served consecutively therefore only differ at most \(2\alpha\) from a multiple of the period length, making a line-pure vehicle schedule very efficient, depending on the choice of \(\alpha\). Note that choosing \(\alpha\) too small may lead to infeasible solutions, depending on the problem instance.

**Improvement 3: Vehicle Scheduling Before Timetabling** As a third improvement, the authors introduce turnaround activities into the timetabling model, restricting the possible departure times for lines. The goal is to restrict the time between the end of a line and the beginning of its backward direction such that there is always enough time to serve the lines consecutively by the same vehicle.
\((L_{\text{turn}})\) as well as not too much time to make such a vehicle schedule inefficient \((L_{\text{turn}} + 2\alpha)\). This ensures the feasibility of the line-pure vehicle schedule and can therefore be interpreted as a vehicle scheduling step before timetabling. Hence, we call this improvement \(\text{VS-first}\). After timetabling, the vehicle schedule is optimized again, potentially improving the operational costs even further.

To test the proposed improvements, the authors provide computational experiments on the datasets \text{Grid} and \text{Germany}, where all three improvements are tested separately and in combination.

![Figure 8: Different improvements for dataset Grid](image)

Figure 8 depicts the different objective values for dataset \text{Grid}. Almost all combinations of improvements are able to reduce the costs of the original approach which is depicted by a circle, filled gray in the left half. The overall least costly solution can be found by combining all approaches, i.e., using a combined pool, consisting of new and old lines, the new cost structure for line planning and solving vehicle scheduling before timetabling. In general, solving timetabling first always leads to a faster solution for the passengers but increases the costs compared to solving the vehicle scheduling problem first.

Since the choice of \(\alpha\) is crucial for the quality of the obtained lines, Figure 9 depicts the influence of this parameter on the quality for dataset \text{Grid}. The authors conclude that the choice of a smaller \(\alpha\) most of the times improves the operational
costs, but it may not be chosen too small to still allow for the feasibility of the public transport plan.

In the end, the authors analyze the effects on the bigger dataset Germany, where improvements of more than 40% for the operational costs are possible when considering the new and combined pool. Additionally, solving the vehicle scheduling stage first can save up to 5% of the costs.
3.6. An Iterative Approach for Integrated Planning in Public Transportation

As discussed in Section 2.4, integrated optimization has recently gained in importance in mathematical public transport planning. Since solving integrated optimization problems exactly is often computationally not feasible for real-world instances, heuristic solutions are a topic of ongoing research.

![Diagram]

Figure 10: Overview of the algorithms

The objective of the authors in [Schiewe and Schiewe, 2018], see Appendix F, is to find a solution for the following problem:

**Problem.** Find a public transport plan \((L, \pi, V)\), i.e., a line concept \(L\) with a corresponding timetable \(\pi\) and vehicle schedule \(V\) such that the travel time of the passengers and the operational costs are minimized.

The authors introduce an iterative approach, where in each iteration two of the three planning stages line planning, timetabling and vehicle scheduling are fixed and the remaining stage is re-optimized, while guaranteeing the feasibility of the overall system. Figure 10 shows an overview of the proposed algorithmic scheme, combining the sequential optimization of the single planning stages in an arbitrary order. Since two of the resulting problems are new, completely new optimization problems need to be modeled and evaluated regarding their performance.

**ReVehicleScheduling:** For Algorithm ReVehicleScheduling, known algorithms from the vehicle scheduling literature, e.g. from [Bunte and Kliwer, 2009], can be used, since this is part of the standard sequential optimization problem. Here, the authors chose an aperiodic, cost-oriented model without a depot implemented in the open-source software framework LinTim, see [Schiewe et al., 2018a].
ReTimetabling: To achieve the feasibility of the vehicle schedule when re-optimizing the (periodic) timetable, additional constraints need to be added to the classic PESP IP formulation, see e.g. [Serafini and Ukovich, 1989], commonly used for solving the periodic timetabling problem. For the aperiodic vehicle scheduling problem, the authors assume that each line in the line concept $L$ is covered $p_{\text{max}}$ times, resulting in the set of trips $T = \{(p, l) : p \in \{1, \ldots, p_{\text{max}}\}, l \in L\}$ that each need to be served by a vehicle schedule $V$ exactly once. Two trips $(p_1, l_1), (p_2, l_2)$ are \textit{compatible} if there is sufficient time to get from the last station of line $l_1$ to the first station of line $l_2$. The authors denote this (aperiodic) time as $\text{end}_{p_1, l_1}$ and $\text{start}_{p_2, l_2}$, respectively, and the minimal time between $l_1$ and $l_2$ as $L_{l_1, l_2}$. $L_{l_1, l_2}$ is assumed to be given by a fixed shortest path in the underlying infrastructure network, determined by some lower travel time bounds on the edges. A vehicle schedule contains a set of \textit{vehicle routes} where each vehicle route is a list of compatible trips, i.e., all consecutive trips are compatible. The set of all such connecting trips is denoted as $C$. With this, the constraints

\begin{equation}
L_{l_1, l_2} \leq \text{start}_{p_2, l_2} - \text{end}_{p_1, l_1} \quad ((p_1, l_1), (p_2, l_2)) \in C
\end{equation}

in addition to several auxiliary constraints to ensure the correct values for the $\text{start}$ and $\text{end}$ variables need to be added to the classical PESP IP model. Using this, a new solution still allows the current vehicle schedule to be feasible, since all connecting trips remain compatible due to (3.2).

ReLinePlanning: More work needs to be done to maintain feasibility of the timetable and the vehicle schedule when re-optimizing the line concept, since the lines are such an integral part of both fixed stages. Additionally, allowing aperiodic vehicle schedules makes finding new lines difficult, since lines have to appear periodically. First, the authors define a public transport plan $(L', \pi', V')$ to be \textit{consistent} to another plan $(L, \pi, V)$ if

- the vehicle paths on trips for $V'$ are contained in the physical paths of the vehicle in $V$, including coinciding times and
- the duration for trips of new lines in $L'$ allow for the service of the line.

Note that the last point is necessary, since connecting trips may not be converted to lines if their duration is too short, e.g. if passengers would not have enough time for boarding and alighting the line in each station. Although this definition restricts the possible lines for a new public transport plan, it is e.g. possible to connect different lines served by the same vehicle or split lines up, allowing for new optimization potential in the other stages of the iterative approach.
Algorithm 6.1 ReLinePlanning

1: Define line network:
2: One edge for each (aperiodic) service of an edge in the PTN
3: Label these edges with the vehicle and the starting time
4: Define collapsed line network:
5: Combine parallel edges from the line network with the same periodic
6: starting time
7: Label each of these edges with a tuple of vehicles using it and the
8: periodic starting time
9: Find set of longest paths \( P \), s.t. all edges in a path have identical labels
10: Set the line pool as the set of all subpaths of \( P \)
11: Solve a line planning problem such that
12: all infrastructure edges are covered according to given minimal frequencies,
13: all collapsed edges are covered at most once
14: and the costs are minimized

Algorithm 6.1 provides an overview on re-optimizing the line concept of a given public transport plan, i.e., how to find a new public transport plan that is consistent with the current solution and minimizes the line costs. The authors proof the correctness of the algorithm with Theorem 6.1.

Theorem 6.1 ([Schiewe and Schiewe, 2018], Theorem 11). Let \( (\mathcal{L}, \pi, \mathcal{V}) \) be given. Let the duration of the edges in connecting trips in \( \mathcal{V} \) be uniquely determined and let for each edge \( e \in E \) the aperiodic departure times be unique for all trips \((p, l) \in \mathcal{V}\) and connecting trips \( e \in \mathcal{V} \), i.e., there is at most one departure using a specific edge in the PTN at any point in time. Then Algorithm ReLinePlanning finds a public transport plan \((\mathcal{L}', \pi', \mathcal{V}')\) that is consistent with \((\mathcal{L}, \pi, \mathcal{V})\) such that line concept \( \mathcal{L}' \) is feasible and minimizes the line costs.

Note that the assumption of unique departure times can be easily guaranteed by using headway constraints for the underlying public transport plan and is often satisfied in practice. The assumptions of uniquely determined durations for the connecting trips edges may on the other hand not be satisfied in practice. If this is the case, the algorithm still finds a feasible solution, but the optimality cannot be guaranteed.

After modeling the problems and proposing algorithms, these can now be combined iteratively. Since they can be combined in any order, each specific order provides an algorithm for re-optimizing a public transport plan. The authors investigate several approaches and analyze the convergence behavior.
**Theorem 6.2** ([Schiewe and Schiewe, 2018], Theorem 18). Let $P_0$ be a feasible public transport plan with travel time $t_0$. Let $P_i$, $i \in \mathbb{N}^+$, be a public transport plan derived from $P_{i-1}$ by applying either ReTimetabling or ReVehicleScheduling and let $t_i$ be the travel time of $P_i$. Then the sequence of travel time values $(t_i)_{i \in \mathbb{N}}$ decreases monotonically and converges.

**Theorem 6.3** ([Schiewe and Schiewe, 2018], Theorem 19). Let $P_0$ be a feasible public transport plan with operational costs $c_0$ where duration based costs are neglected. Let $P_i$, $i \in \mathbb{N}^+$, be a public transport plan derived from $P_{i-1}$ by applying either ReLinePlanning, ReTimetabling or ReVehicleScheduling and let $c_i$ be the operational costs of $P_i$. Then the sequence of operational cost values $(c_i)_{i \in \mathbb{N}}$ decreases monotonically and converges.

Note, that for the cases where convergence is not guaranteed, i.e., the possible increase of the travel time in Algorithm ReLinePlanning and the possible increase of the costs when duration based costs are considered, the authors give examples for the non-convergence, see [Schiewe and Schiewe, 2018], Examples 14 and 15, respectively.

The computational experiments cover two different datasets, dataset Grid as a case study and dataset Regional, a close-to real-world representation of the regional train system in southern Lower Saxony, Germany. For dataset Regional, several demand scenarios are created and the average changes in the objectives are discussed. Some results are presented in the following.

Figure 11 depicts a typical behavior of the objective values when re-optimizing a public transport plan. Algorithm ReLinePlanning may increase the travel time and does so in several cases while simultaneously reducing the costs or facilitating
this for the vehicle scheduling step afterwards. Overall, the operational costs can be reduced by around 18% while only increasing the travel time by 8%, yielding a new competitive solution. Other iteration schemes provide different trade-offs, but the overall picture stays the same. Especially the monotonic behavior of the iteration schemes discussed in Theorem 6.2 and 6.3 is shown in Figure 14 of [Schiewe and Schiewe, 2018].

The influence of the re-optimization on the line concept is shown in Figure 12. There are multiple lines staying the same, e.g. the dotted blue line, but other lines are either shortened to reduce costs of unneeded coverage, e.g. the dashed orange line, or new connections are formed, e.g. the dash-dotted cyan line. Here, passengers from $v_6$ are allowed a direct connection to $v_{12}$, $v_{17}$ and $v_{22}$ after the re-optimization where at least one transfer was necessary beforehand.

3.7. Cost-Minimal Public Transport Planning

While the problem of building a passenger-optimal public transport system has a known solution, namely building a direct connection for each passenger, it is not clear how a cost-optimal system has to be build. In [Pätzold et al., 2019], see Appendix G, the authors present three different models with increasing complexity and increasing quality of the bounds to solve such a problem. Here, the third problem is a fully integrated integer program to find a cost-optimal public transport plan. Additionally, optimality conditions and bounds on the optimal objective value of the overall problem are given for the first two models and all models and their
bounds are compared in computational experiments on multiple datasets.

To define the problem, the authors first note that for a cost-minimal system the line concept only needs to adhere to one set of feasibility constraints, namely that every passenger can travel. Thus, for every OD pair \((u, v)\) with weight \(W_{uv}\) a set of paths \(P_{uv}\) and weights \(w_p\) for \(p \in P_{uv}\) have to exist such that \(\sum_{p \in P_{uv}} w_p = W_{uv}\) and

\[
\sum_{p \in \bigcup_{u,v \in V} P_{uv}, e \in p} w_p \leq \text{Cap} \cdot |\{l \in \mathcal{L} : e \in l\}|
\]

is satisfied with \(\text{Cap}\) being the vehicle capacity. Note that these constraints do not require a certain quality for the passenger paths in \(P_{uw}\).

To compute the costs of a cost-minimal system, the authors guarantee in the timetabling stage that for every line the duration of drive and wait activities always corresponds to the lower bounds, assuring the shortest possible duration for lines. Note that this is possible due to the assumption of the authors that it is always less costly to wait at the end of a line than to have some buffer in between or on drive activities. Therefore, the operational costs of a public transport plan \(g^{\text{cost}}(\mathcal{L}, \pi, \mathcal{V})\) do not depend on the timetable but only on the line concept and the vehicle schedule.

**Problem (cost-opt).** Find a feasible public transport plan \((\mathcal{L}, \pi, \mathcal{V})\) with minimal costs \(g^{\text{cost}}(\mathcal{L}, \pi, \mathcal{V})\).

**Model 1:** Load generation

**Model 2:** Integrating up to line planning

**Model 3:** Integrating up to timetabling and vehicle scheduling, i.e., solving it all

Figure 13: Three proposed models for solving (cost-opt)

In the following, the authors define three models, depicted as an overview in Figure 13. First, Model 1 for load generation is presented, allowing the computation of bounds for (cost-opt) and conditions are stated where this model finds an optimal solution for the overall problem. Afterwards, Model 2 extends the load
generation by integrating the line planning stage, allowing for tighter bounds and a relaxed optimality condition. In the end, the integrated Model 3 is presented, solving (cost-opt) exactly.

Model 1: For a known passenger distribution, lower frequency bounds $f_{e}^{\min}$ on all edges can be determined, giving a bound on the number of times each PTN edge needs to be covered per planning period. Model 1 uses these bounds as an optimization goal, approximating the costs of the overall solution. The obtained optimal objective value ($z_{1}^{\text{opt}}$) provides a lower bound on the costs of the optimal public transport plan ($z^{\text{opt}}$).

**Theorem 7.1** ([Pätzold et al., 2019, Theorem 5]). Model 1 is a relaxation of (cost-opt), i.e.,

$$z_{1}^{\text{opt}} \leq z^{\text{opt}}.$$

In order to not only obtain a lower but also an upper bound, Model 1 can be adapted slightly to Model 1*, providing feasible solutions for (cost-opt). The authors are able to provide a theoretical bound on the quality of the obtained solutions of Model 1 and Model 1* and using this, give a condition where Model 1 already finds the optimal objective value for (cost-opt).

**Corollary 7.2** ([Pätzold et al., 2019, Corollary 9]). Let $L^{\text{wait}} = L^{\text{turn}}$. Then the optimal objective of Model 1 and Model 1* is equal to the optimal objective of (cost-opt).

Here, $L^{\text{wait}}$ denotes the lower bound on waiting times at stops, which is assumed to be equal for all stops. The assumption $L^{\text{wait}} = L^{\text{turn}}$ of this theorem can be relaxed if non-simple and directed lines are allowed, in which case Model 1 always finds the optimal objective value of (cost-opt) but the obtained solutions may be undesirable in practice, see [Pätzold et al., 2019, Corollary 10] and [Pätzold et al., 2019, Example 11].

Additionally, the authors provide valid inequalities, improving the performance of Model 1 such that in the computational experiments, runtime improvements of up to 50% could be measured on the investigated datasets.

Model 2: To improve the approximation of Model 1, the authors additionally integrate the line planning stage, resulting in Model 2. This allows for a distinction between waiting and turnaround times, allowing for a better lower bound.
Theorem 7.3 ([Pätzold et al., 2019, Theorem 12]). The optimal objective value of Model 2, denoted by \( z_{2}^{\text{opt}} \), is a lower bound on the optimal objective value of (cost-opt) and an upper bound on the optimal objective value of Model 1, i.e.,

\[
z_{1}^{\text{opt}} \leq z_{2}^{\text{opt}} \leq z^{\text{opt}}.
\]

Again, the authors are able to provide an optimality condition for a slightly adapted version Model 2*.

Theorem 7.4 ([Pätzold et al., 2019, Theorem 15]). An optimal solution to Model 2* solves (cost-opt) under the restriction that only line-pure vehicle schedules are allowed.

Again, the authors are able to provide a quality bound between the two models, see [Pätzold et al., 2019, Theorem 17], but in contrast to the bound of Model 1, this bound is computable a priori.

One disadvantage of Model 2 is the presence of a big M constraint on the number of lines the model is allowed to create. Allowing too few lines may result in suboptimal solutions while choosing the number of lines too big increases the computational complexity. To overcome this, the authors propose an iterative approach which starts by solving Model 2 for an arbitrary number of lines. Using the solution found here, an upper bound on the number of lines needed for an overall optimal solution can be computed, using a bound provided in [Pätzold et al., 2019, Theorem 17]. Afterwards, the obtained bound can be used to find an optimal solution to Model 2 in a second computation.

Model 3: By further extending Model 2, i.e., by including vehicle scheduling, the authors are able to model (cost-opt) as an integer program in Model 3 and formally prove its correctness, see [Pätzold et al., 2019, Theorem 22].

Unfortunately, this integrated model is hard to solve but the authors provide several suggestions for improving the runtime, including a preprocessing, where the lines found by Model 2 are used as an input instead of the complete line pool.

To investigate the practical results of the proposed models, the authors present computational experiments on four different datasets, ranging from small example datasets Linear and Toy to close-to real-world datasets Grid and Germany. Additionally, the cases \( L_{\text{turn}} = L_{\text{wait}} \) and \( L_{\text{turn}} > L_{\text{wait}} \) are considered separately, allowing a detailed analysis of the cases where Model 1 finds an optimal objective value. An overview of the other case, namely \( L_{\text{turn}} > L_{\text{wait}} \), is provided in Table 7.1. Here, especially the increasing quality of the bound and the increasing complexity can be observed, with Model 2 being unable to solve dataset Grid to optimality and Model 3 only being able to solve dataset Linear to optimality. Note that the observed bounds are always consistent with the stated results when the models are solved to optimality.
<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model 1</th>
<th>Model 1*</th>
<th>Model 2</th>
<th>Model 2*</th>
<th>lb</th>
<th>ub</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>80</td>
<td>130</td>
<td>130</td>
<td>130</td>
<td>130</td>
<td>130</td>
</tr>
<tr>
<td>Toy</td>
<td>1424</td>
<td>1474</td>
<td>1424</td>
<td>1696</td>
<td>1288°</td>
<td>1539°</td>
</tr>
<tr>
<td>Grid</td>
<td>1034</td>
<td>1134</td>
<td>1030°</td>
<td>1140</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Germany</td>
<td>74462°</td>
<td>85612°</td>
<td>54148°</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Table 7.1: Objective values for the case of $L^{\text{turn}} > L^{\text{wait}}$, solutions marked by ° are not solved to optimality.

The authors give special attention to the solution found by Model 1 on dataset Grid, which is about 23% less costly than the cheapest solutions found to date in the ongoing competition started in [Friedrich et al., 2017a], see Section 3.1 and Appendix A. Different solutions of the competition and the new lower bound on the costs are depicted in Figure 14.

**Figure 14:** Multiple solutions for dataset Grid, submitted to the competition started in [Friedrich et al., 2017a] and published at [FOR2083, 2018], evaluated by their cost per hour and perceived travel time. The costs of the new solution found by Model 1 is depicted by a red line.
4. Discussion

The main contribution of this thesis is the cost-oriented view point for different planning stages in public transport planning. New optimization models, in some cases integrating multiple planning stages, are proposed to optimize the operational costs or practical requirements that were not considered before are added to known optimization models. While the problems in Sections 3.2 and 3.3 are modeled as integer programs, specialized heuristics are developed in the Sections 3.3 and 3.5. Section 3.6 even combines both approaches: While an iterative heuristic scheme is used to find public transport plans, two of the subproblems are also newly developed, one using an integer program and the other one a specialized algorithm. In Section 3.4 a game theoretic approach is used, comparing system-optimal solutions to socially optimal ones. All models are extensively computationally evaluated and compared to current state-of-the-art methods for minimizing the costs of public transport systems.

First, Section 3.1 gives an approach to compare the manual and algorithmic planning procedures, allowing both sides to learn from the other. Especially, this work allowed the authors to develop several ideas used in the other works of this thesis:

- Comparing the solutions showed that algorithmic solutions were competitive with respect to passenger convenience but needed improvements with respect to the operational costs. Especially the manual approach to plan with a line-pure vehicle schedule early on gave first ideas for the look-ahead approaches presented in Section 3.5.

- To the time of publication, the algorithmic solutions all relied on the traffic load computed in the manual solutions, since the shortest path approaches of the mathematical planners were not competitive. This led to the theoretical examination and new load generation methods presented in Section 3.3.

- The importance of bounds on the objective values for public transport plans became clear. For passenger convenience, such a bound could be computed easily, for operational costs the investigations summarized in Section 3.7 were necessary.

- In the end, practical requirements on memorability and the solutions of the practical planners using system headways made it clear that such concepts should be integrated into mathematical planning as well, see Section 3.2.
Figure 15: Depiction of the eigenmodel described in [Schöbel, 2017]. The traditional sequential approach is dashed, the approach of Section 3.5 is marked in red and dotted and the approach of Section 3.6 is marked in green and dash-dotted. Other algorithms are marked in gray.

Additionally, the positive reaction of other researchers on the open-source publication of the developed dataset clarified the need for such benchmark datasets for mathematical public transport planning. The implicitly formulated challenge motivated several researchers to use the dataset or submit solutions for the dataset, see [Friedrich et al., 2017c, Friedrich et al., 2018b, Liebchen, 2018], whereby an ongoing competition was created. For the current status, see [FOR2083, 2018].

Two different approaches to the integration of load generation and line planning are presented in Section 3.3 and 3.4. While Section 3.3 considers system optimal solutions to determine cost-efficient line plans, transfers work from practical public transport planners to mathematical planning and compares different heuristics, Section 3.4 examines the effect of letting passengers choose their own paths. This leads to equilibrium solutions that are maintainable in practice, without totally sacrificing cost-efficiency due to sharing the costs between all passengers. The difference to the user-optimal route choice proposed in [Goerigk and Schmidt, 2017] is that while the
solutions proposed there are equilibria for strictly travel time oriented objectives, allowing the passengers to choose longer paths while saving costs may lead to a more cost-efficient solution.

Section 3.5 and 3.6 both present heuristic solutions for finding cost-efficient public transport plans. Note that while the sequential approach described in Section 3.5 is able to find a solution “from scratch”, this is not possible for the re-optimization approach of Section 3.6. Here, a public transport plan has to be given which can then be improved, allowing it to build upon an arbitrary algorithm for computing a public transport plan, e.g. the approach described in Section 3.5. However, both approaches can be interpreted as a sequential approach in the algorithmic scheme called eigenmodel, described in [Schöbel, 2017].

In [Schöbel, 2017], a new iterative meta-algorithm for computing public transport plans is described, as depicted in Figure 15. Every node in the graph represents a stage in finding a public transport system, e.g. finding an initial line concept or finding a vehicle schedule for a given line concept and timetable. Every directed edge represents a possible concatenation of these stages, i.e., every path through the network ending in one of the three center nodes is a planning process for finding a public transport plan. This eigenmodel includes unorthodox planning procedures as well as the traditional one, e.g. starting with a vehicle schedule, afterwards computing a line concept and ending with the computation of a timetable for the given line concept and vehicle schedule. Many of these procedures were not considered in the public transport literature before and may be worth investigating, while some heuristic approaches found in the literature fit into this framework. One example is [Michaelis and Schöbel, 2009], where an algorithmic procedure starting with the vehicle schedules instead of a line concept is proposed.

The edges in the “inner circle”, depicted in green and dash-dotted, and the corresponding nodes represent the re-optimization procedure described in Section 3.6, starting with a public transport plan and re-optimizing one of its components. For the approach of Section 3.5, two different interpretations are possible. Both are depicted as red dotted edges in Figure 15. On one hand, the proposed procedure to choose a line-pure vehicle schedule before timetabling can be seen as solving the planning stages line planning, vehicle scheduling and timetabling in this order, i.e., starting from the left and going down in Figure 15. On the other hand, only including lines that will allow for a good vehicle schedule, i.e., the proposed adaptations on line costs and line pool generation, can be seen as a separate vehicle scheduling stage before line planning, resulting in the path starting at the vehicle scheduling stage in the bottom of Figure 15 and traversing the graph to the left. Note, that other models developed in this thesis can be used in the eigenmodel as well, both as a replacement for the line planning stage in the left (Section 3.2 to 3.4) and as a starting point for the inner circle (Section 3.7).
Other models discussed in this thesis can be compared as well. The integrated model presented in Section 3.3 for integrating load generation and line planning and the second model in Section 3.7 are both models for solving a cost-oriented line planning problem while integrating the passenger distribution. But there are some key differences. First, the model presented in Section 3.3 does not allow arbitrary paths for the passengers as in Section 3.7, but only allow detours up until a given factor, therefore still guaranteeing a viable solution for the passengers. Additionally, the operational costs are only approximated using a line-based cost function while the model presented in Section 3.7 builds upon the insights of Section 3.5 for better approximating the costs of a line by assuming line-pure vehicle schedules, implicitly planning those as well. As discussed in Section 3.7, this even allows for the computation of cost-optimal solution under certain assumptions.

The models presented in Section 3.6 and the fully integrated Model 3 in Section 3.7 have similarities as well, both being algorithms for computing public transport plans. However, the iterative model presented in Section 3.6 takes a complete public transport plan as an input, whereas the model in Section 3.7 is an integrated model, computing all stages simultaneously and providing provable cost-optimal solutions. Of course this capability results in a higher complexity which is why only the iterative approach of Section 3.6 is able to compute solutions for close-to real-world instances.
5. Outlook

This thesis shows how to design a cost-optimal public transport plan, for single problem stages as well as for integrating multiple stages. This could be further extended, integrating more planning stages such as network design or crew scheduling into the models presented here. Crew schedules are especially interesting as they are an important part of the operational costs that are omitted here.

The comparison of manual and algorithmic planning procedures in Section 3.1 allows for several starting points of further research. While the line planning model presented in Section 3.2 is a first step to improving the passenger convenience, further understanding of concepts such as the memorability of the timetable would hopefully enable optimization models to optimize this as an additional criterion, allowing for more passenger-friendly public transport plans. Another starting point would be the problem of finding solutions that are not overcrowded when implemented in practice. While the integrated approaches of Sections 3.3 and 3.4 allow for this in the line planning stage, overcrowding may occur after implementing the timetable. The evaluation of the solutions for the competition started for dataset Grid shows that algorithmic solutions often do not respect vehicle capacities when passengers are allowed to choose a shortest path with respect to the implemented timetable. Although there is already research into integrating passenger routing decisions into timetable models, see e.g. [Schiewe, 2018], doing so while respecting vehicle capacities is not sufficiently researched yet.

Another topic of further research would be to extend the models presented in this thesis to better accommodate the passengers. While the focus of this thesis is to optimize the costs, a good public transport plan should be competitive compared to individual traffic for the passengers to generate sufficient demand. Several models presented in this thesis could be extended in this direction: The system headway approach presented in Section 3.2 could be extended to several other line planning models, since implementation into a general line planning model is already described. This includes passenger-oriented models as well. The effect on the quality for the passengers in these new models could be interesting to investigate, even when theoretical bounds on the quality are not achievable as they are for the cost model. Additionally, the heuristics presented in Sections 3.5 and 3.6 and, in extension, the eigenmodel approach depicted in Figure 15 in Chapter 4 could be extended to be more passenger-friendly. Several paths in the eigenmodel are not yet explored and could yield competitive algorithmic procedures for finding public
transport plans. But even the nodes already explored in this thesis could be replaced by more passenger-friendly models, shifting the focus from operational costs to passenger convenience. Having multiple different algorithms per node would allow the operator to choose a fitting algorithm for the specific use case or even allow meta-algorithms such as machine learning approaches to generate better public transport plans.

In addition to passenger-oriented models, focusing more on the robustness of the resulting solutions would be interesting in the future. Especially cost-oriented planning tends to produce fragile solutions if delays in the resulting systems are considered, e.g. including buffer times or examining more complex delay scenarios would improve the competitiveness for real-world usage. First approaches on how to measure robustness can be found in [Friedrich et al., 2017c, Friedrich et al., 2018b], introducing a third robustness dimension to the two objective functions of operational costs and passenger convenience examined in this thesis. Using these concepts to develop models optimizing the costs while maintaining a basic level of robustness would improve the usability of the computed solutions.

In the end, the response of the research community on publishing dataset Grid under an open-source license shows that publishing more datasets should be a goal of the mathematical public transport community. Several additional datasets are already published at [FOR2083, 2018], namely a ring network and an already known benchmark dataset from practical public transport planning, including reasonable input data for both the practical and the mathematical public transport planning community. The open-source software library LinTim, see [Schiewe et al., 2018a], contains several other datasets as well, but those are more oriented for usage in mathematical public transport planning and lack the details often used by practical planners. Extending this collection further would enable an extended comparison of algorithms on extensively researched benchmark datasets of different sizes, allowing for a better comprehensible and reproducible evaluation of new and already existing algorithms.
6. Own Contributions

In this chapter, I summarize my contributions to the publications presented in this thesis.

[Friedrich et al., 2017a] is joint work with Markus Friedrich, Maximilian Hartl and Anita Schöbel. All implementation in the open-source software library LinTim ([Schiewe et al., 2018a]) and the description of the algorithmic solution procedure were done by myself. The rest of the work was done jointly with all co-authors. Overall, I judge my contributions to be around 35%.

[Friedrich et al., 2018a] is joint work with Markus Friedrich, Maximilian Hartl and Anita Schöbel. All implementations and most of the writing and the proofs were done by myself. The rest of the work was done jointly with all co-authors. Overall, I judge my contributions to be around 70%.

[Friedrich et al., 2017b] is joint work with Markus Friedrich, Maximilian Hartl and Anita Schöbel. All implementations and most of the writing and the proofs were done by myself. Overall, I judge my contributions to be around 60%.

[Schiewe et al., 2019] is joint work with Marie Schmidt and Philine Schiewe. The work was done jointly with all co-authors. Overall, I judge my contributions to be around 40%.

[Pätzold et al., 2017] is joint work with Julius Pätzold, Philine Schiewe and Anita Schöbel. Note that [Pätzold et al., 2017] received the ATMOS Best Paper Award in 2017. All propositions regarding the vehicle scheduling step were implemented and written by myself. The rest of the work was done jointly with all co-authors. Overall, I judge my contributions to be around 15%.

[Schiewe and Schiewe, 2018] is joint work with Philine Schiewe. The work was done jointly with the co-author. Overall, I judge my contributions to be around 50%.

[Pätzold et al., 2019] is joint work with Julius Pätzold and Anita Schöbel. Development, implementation and proof of correctness of the integrated model were done by myself, as were the proofs for the gaps between the first two models and the integrated model. Development and implementation of the first two models were done by Julius Pätzold. The rest of the work was done jointly with all co-authors. Overall, I judge my contributions to be around 40%.
Bibliography


Appendix

A. Public Transport Planning - Manually Generated and Algorithmic Solutions

M. Friedrich, M. Hartl, A. Schiewe, A. Schöbel
Angebotsplanung im öffentlichen Verkehr - Planerische und Algorithmische Lösungen
[Friedrich et al., 2017a]
Angebotsplanung im öffentlichen Verkehr -
Planerische und algorithmische Lösungen

Prof. Dr. Markus Friedrich
Lehrstuhl für Verkehrsplanung und Verkehrsleittechnik, Universität Stuttgart, Pfaffenwaldring 7,
D-70569 Stuttgart, Tel. +49-711-68582480, Fax. +49-711-68572484,
E-Mail: markus.friedrich@isv.uni-stuttgart.de

M. Sc. Maximilian Hartl
Lehrstuhl für Verkehrsplanung und Verkehrsleittechnik, Universität Stuttgart, Pfaffenwaldring 7,
D-70569 Stuttgart, Tel. +49-711-68584414, Fax. +49-711-68574414,
E-Mail: maximilian.hartl@isv.uni-stuttgart.de

M. Sc. Alexander Schiewe
Universität Göttingen, Institut für Numerische und Angewandte Mathematik, Lotzestraße 16-18,
D-37083 Göttingen, Tel. +49-551-397872, Fax +49-551-393944, E-Mail: a.schiewe@math.uni-goettingen.de

Prof. Dr. Anita Schöbel
Universität Göttingen, Institut für Numerische und Angewandte Mathematik, Lotzestraße 16-18,
D-37083 Göttingen, Tel. +49-551-3912237, Fax +49-551-393944, E-Mail: schoebel@math.uni-goettingen.de

Kurzfassung

Obgleich Optimierungsverfahren für den Entwurf des ÖV-Angebots seit mehr als 40 Jahren ent-
wickelt werden, haben bisher nur Verfahren der Umlauf- und Dienstplanung den Weg in die Pra-
xis der Angebotsplanung gefunden. Dagegen sind bei der Erstellung von Linien und Fahrplä-
nen rechnergestützte Entwurfsverfahren weiterhin die Standardmethode in der ÖV-Angebots-
planung. Um die Anforderungen der Planungspraxis besser erfüllen zu können, werden in die-
sicem Beitrag planerische und algorithmische Lösungen für eine Testinstanz erzeugt und mitein-
ander verglichen. Der Vergleich soll dann in nachfolgenden Schritten genutzt werden, um die
Optimierungsverfahren weiter zu verbessern.
1 Einleitung

Das Verkehrsangebot im öffentlichen Verkehr (ÖV) hat die primäre Aufgabe, Fahrgäste zu befördern. Der ÖV soll darüber hinaus eine Alternative zum Pkw anbieten, da er verglichen mit dem Pkw auf einem Fahrweg gleicher Breite deutlich mehr Menschen als der Pkw befördern kann und ab einem durchschnittlichen Auslastungsgrad der Sitzplätze von rund 40% einen niedrigeren spezifischen Energieverbrauch pro Personenkilometer aufweist. Diese positiven Eigenschaften des ÖV dienen als eine Rechtfertigung öffentlicher Zuschüsse für den ÖV. Da bei einem Ausbau des ÖV die Kosten in der Regel stärker steigen als die Erlöse, müssen bei einer integrierten Planung im öffentlichen Verkehr die Wirkungen auf die Fahrgäste und die Wirkungen auf die Betreiber gleichermaßen berücksichtigt werden. Daraus ergibt sich die übergeordnete Fragestellung, die den vorliegenden Beitrag motiviert: Wie entwirft man ein möglichst gutes Angebot im öffentlichen Verkehr?

Zur Lösung dieser Fragestellung verfolgen Vertreter der Verkehrsplanung und der angewandten Mathematik bzw. des Operations Research unterschiedliche Ansätze:


Obgleich Optimierungsverfahren für den Entwurf des ÖV-Angebots seit mehr als 40 Jahren entwickelt werden, haben bisher nur Verfahren der Umlauf- und Dienstplanung den Weg in die Praxis der Angebotsplanung gefunden. Beim Entwurf von Linien und Fahrplänen sind rechnergestützte Entwurfsverfahren weiterhin die Standardmethode bei der ÖV-Angebotsplanung. In der DFG-Forschergruppe “Integrierte Planung im öffentlichen Verkehr” haben sich Vertreter der Mathematik, der Informatik und des Verkehrswesens mit dem Ziel zusammengefunden,
Tabelle 1: Kenngrößen des Verkehrsangebots

Methoden der mathematischen Optimierung für die Zwecke der ÖV-Angebotsplanung so zu erweitern, dass die Anforderungen der Planungspraxis besser erfüllt werden können. Dieser Beitrag berichtet über ein Teilprojekt, das auf die Schritte Liniennetzplanung, Fahrplanung und Umlaufplanung fokussiert ist. Ein wesentlicher Forschungsansatz in diesem Teilprojekt besteht darin, dass für eine gegebene Aufgabenstellung, d.h. für eine gegebene Siedlungsstruktur und ein gegebenes Verkehrswegezustand, planerische und algorithmische Lösungen erzeugt und miteinander verglichen werden. Der Vergleich soll dann in nachfolgenden Schritten genutzt werden, um die Optimierungsverfahren zu verbessern.

2 Kenngrößen, Parameter und Variablen eines Verkehrsangebots


- Bevölkerungs- und Siedlungsstruktur
- Verkehrsangebot der anderen Verkehrsmodi
- Präferenzen der Verkehrsteilnehmer (Mobilitätsverhaltensparameter)
- Verkehrswegezustand, das von ÖV-Fahrzeugen genutzt werden kann
- Eigenschaften der Fahrzeuge (Kapazität, Verbrauch)
Die Variablen eines Verkehrsangebots umfassen die Größen, die im Rahmen der Planung festgelegt werden. Wesentliche Variablen eines ÖV-Angebots sind:

- Anzahl und Lage der Haltestellen
- Anzahl der Linien und Verlauf der Linienwege
- Fahrzeiten zwischen Haltestellen und Haltezeiten
- Abfahrtszeiten an den Haltestellen
- Fahrzeugfolgezeit bzw. Takt
- Fahrzeugtyp bzw. Fahrzeuggröße


3 Beschreibung Testinstanz

In diesem Beitrag sollen planerische und algorithmische Lösungen beispielhaft für eine Testinstanz verglichen werden. Dabei wird von folgenden Annahmen ausgegangen:

**Verkehrswege netz:** Gegeben sei das in Abbildung 1a dargestellte Rasternetz mit 25 vorgegebenen Haltestellen. Die Strecken des Netzes haben alle eine einheitliche Länge von 2 km. Für die Fahrzeit der Busse zwischen den Haltestellen wird eine mittlere Fahrge schwindigkeit inkl. Haltestellenaufenthaltszeit von 20 km/h angenommen, so dass die Fahrzeit zwischen zwei Haltestellen 6 Minuten beträgt.

**Verkehrsnachfrage:** Die Verkehrsnachfrage wurde mit einem Verkehrsnachfragemodell er mittelt, dass zwei Modi (Pkw und ÖV) und vier Aktivitätenpaare (Wohnen-Arbeit, Arbeit Wohnen, Wohnen-Sonstiges, Sonstiges-Wohnen) unterscheidet. Es werden die Wege von 30.000 Erwerbstätigen modelliert, die in 25 Verkehrszellen wohnen. Jede Verkehrs zelle ist genau einer Haltestelle zugeordnet, so dass die Verkehrsteilnehmer keine Hal testellenwahlentscheidungen treffen können. Die Nachfrage wird für jede Stunde des Tages berechnet, Grundlage der Linienplanung ist die ÖV-Verkehrsnachfrage in der morgendlichen Hauptverkehrszeit. Sie umfasst insgesamt 2.531 Fahrten. Bild 1b zeigt den Quell- und Zielverkehr für jede Verkehrs zelle und die Streckenbelastungen, die sich bei einer


4 Vorgehensweise bei der Angebotserstellung

4.1 Planerische Vorgehensweise

Für die Testinstanz werden zwei planerische Lösungen P_1 und P_2 entwickelt, die in Abbildung 2a und 3a dargestellt sind. Die Vorgehensweise bei der Erstellung der beiden planerischen Lösung lässt sich vereinfacht wie folgt beschreiben:

1. Festlegung eines Systemtakts: Um einen merkbaren Fahrplan und regelmäßige Anschlüsse zwischen den Linien anbieten zu können, wird aus der Nachfrage ein Systemtakt (Grundtakt) abgeleitet. Für die Testinstanz wird ein 20-Minutenakt (Frequenz = 3) gewählt. Bei diesem Systemtakt gewährleisten Linienlängen von 6 und 8 Strecken eine geringe Standzeit bei einer linienreinen Umlaufbildung.

2. Festlegung eines Linienplans: Beiden planerischen Lösungen liegt die Idee eines zentralen Umsteigeknotens im Zentrum (Knoten 303) zugrunde. Die Lösung P_1 baut auf einer
Stammachse auf, was zu einer achsensymmetrischen Lösung führt. Bei Lösung P_2 sind vier Linien punktsymmetrisch. Aus Kapazitätsgründen ist in beiden Lösungen eine Verstärkerlinie (B6 bzw. B3) erforderlich.


4.2 Algorithmische Vorgehensweise

Für die Erstellung der algorithmischen Lösungen werden die in der Software-Bibliothek LinTim ([9], [17]) gesammelten Optimierungsroutinen zur Planung des öffentlichen Verkehrs genutzt. Dabei ergibt sich folgender Ablauf:


5 Ergebnisse

5.1 Beschreibung der Lösungen

Für die in Abschnitt 3 beschriebene Testinstanz werden die beiden planerischen Lösungen $P_1$ und $P_2$ mit folgenden automatisiert erstellten Lösungen verglichen. Diese sind nach dem in Abschnitt 4.2 beschriebenen Vorgehen erstellt und unterscheiden sich durch den jeweils zugrunde gelegten Linienpool und den Grad der Automatisierung.

- $P_1$ und $P_2$ - Planerische Lösungen: Dargestellt in Abbildung 2a bzw. 3a.

- $A_{1.1}$ und $A_{2.1}$ - Planerisches Linienkonzept + LinTim: Hier werden der Linienplan und die Frequenzen der entsprechenden planerischen Lösung übernommen, der Fahrplan und der Umlaufplan werden mit LinTim erstellt.

- $A_{1.2}$ und $A_{2.2}$ - Planerischer Pool + LinTim: Hier werden nur die Linien der jeweiligen planerischen Lösung übernommen und als Linienpool verwendet; die Frequenzen, der Fahrplan und der Umlaufplan werden durch LinTim erstellt. Die Ergebnisse sind in Abbildung 2b bzw 3b dargestellt.


5.2 Bewertung der Lösungen

Die sich ergebenden Lösungen werden untereinander und mit der grundlegenden planerischen Lösung anhand ihrer Kenngrößen Kosten und empfundene Reisezeit (siehe Abschnitt
Abbildung 2: Linienwege der Lösungen basierend auf P_1

(a) Linienwege der Lösungen P_1 und A_1_1
(b) Linienwege der Lösung A_1_2
(c) Linienwege der Lösung A_1_3
(d) Linienwege der Lösung A_1_4
(e) Linienwege der Lösung A_1_5
Abbildung 3: Linienwege der Lösungen basierend auf P_2

(a) Linienwege der Lösungen P_2 und A_2_1
(b) Linienwege der Lösung A_2_2
(c) Linienwege der Lösung A_2_3
(d) Linienwege der Lösung A_2_4
(e) Linienwege der Lösung A_2_5
2) bewertet und in den Abbildungen 4a und 4b dargestellt. Die mittlere empfundene Reisezeit der dargestellten Lösungen liegt zwischen 18 und 26 Minuten. Bei einer für den Fahrgast optimalen Lösung, in der jeder Fahrgast auf direkten Weg ohne Umstieg befördert wird, würde die mittlere empfundene Reisezeit 15,4 Minuten betragen. Die Kosten für solch ein Angebot würden bei 24.600 Euro/h liegen, also eine ungefährige Steigerung um den Faktor 12 im Vergleich zu den besten hier vorgestellten Lösungen. Dafür müssten über 300 Fahrzeuge vorgehalten werden, die auf den meisten Relationen lediglich ein Fahrt pro Stunde anbieten.

Betrachtet man die Auswirkung der Optimierungsalgorithmen auf die Fahr- und Umlaufplanung, so lässt sich zunächst feststellen, dass der planerisch gefundene Umlaufplan für den Fahrplan bereits optimal ist. Hier konnte in keiner planerischen Lösung eine Verbesserung erreicht werden. Hält man von P_2 nur die Linien und ihre Frequenzen fest, kann die Fahrplanerstellung mit LinTim durch eine zusätzliche Synchronisierung der Taktung eine Verbesserung der empfundenen Reisezeit erreichen (A_2_1). Ein darauf aufbauender Umlaufplan hat allerdings deutlich...
höhere Kosten, da die Umläufe nun schlechter synchronisiert sind als in der planerischen Lösung.
Für P_1 ist eine solche Verbesserung nicht so einfach möglich, eine Verbesserung der empfundenen Reisezeit tritt hier erst nach zusätzlicher Anpassung der Frequenzen auf. Dies führt dann aber auch zu einem Umlaufplan mit geringeren Kosten als bei P_1. Der Grund liegt darin, dass die in der Optimierung gefundene Lösung niedrigere Frequenzen wählt als in der planerischen Lösung (siehe auch Abbildung 2b), die gerade noch für den Transport der Fahrgäste ausreichen und durch Synchronisierung der Umstiege eine bessere empfundene Reisezeit für die Passagiere im Vergleich zur planerischen Lösung erreicht.

Die weiteren Lösungen zeigen, dass der Einfluss des zugrunde gelegten Linienpools auf die Erstellung des Linienplans, des Fahrplans und des Umlaufplans signifikant ist. Wählt man als Linienpool ausschließlich die Menge der geraden Linien, so ergeben sich die in den Abbildungen 2c/3c dargestellten Lösungen A_1_3/A_2_3, die entweder in der empfundenen Reisezeit oder in den Kosten schlechter abschneiden als alle anderen Lösungen. Der Grund ist die beschränkte Auswahl an potentiellen Linien. Ob diese für die Passagiere gut geeignet sind, hängt von den Startbelastungen der Kanten ab. Wenn P_1 als Ausgangspunkt genutzt wird, sind die Belastungen so verteilt, dass 6 Linien zum Erreichen der Kapazitätsziele ausreichen. Dies führt daraufhin zu einer kostengünstigen Lösung, die allerdings keine guten Passagierkennzahlen aufweist. Das umgekehrte Bild tritt bei P_2 als Ausgangspunkt auf, da hier alle 10 Linien eingerichtet werden müssen. Dies führt zu einer besseren Abdeckung für die Passagiere, die aber sehr teuer ist. Es muss also bereits bei der Erstellung des Linienpools im ersten Schritt, des in Abschnitt 4 beschriebenen Prozesses, Aufmerksamkeit geschenkt werden.


Ergänzt man den LinTim Pool noch um die Linien aus der planerischen Lösung, so erhält man in beiden Fällen Lösungen (A_1_5 und A_2_5), welche die Lösung des Vorschrittes (A_1_4 und A_2_4) hinsichtlich beider Zielfunktionen verbessern kann. Die größere Auswahl an Linien erlaubt einen Linienplan, der sowohl einen guten Fahrplan als auch einen guten Umlaufplan ermöglicht. Diese Lösung weist in beiden Szenarios die niedrigste empfundene Reisezeit und geringe Kosten auf. Sie liegt für beide Ausgangssituationen auf der Pareto-Front der Lösungen, d.h. es gibt keine Lösung, die in beiden Zielfunktionen besser ist. Allerdings treten in dieser Lösung sehr unterschiedliche Frequenzen der Linien auf, sie ist also weit von einem in der Verkehrsplanung üblichen Systemtakt entfernt.

Sowohl bei der Erstellung der planerischen als auch der algorithmischen Lösung findet nach der Angebotserstellung eine erneute Umlegung statt, so dass Änderungen des Angebots auf die Routenwahl wirken. In der planerischen Lösung werden diese Änderungen in der Angebotserstellung berücksichtigt und die Lösung entsprechend angepasst. In dem fertig erstellten Verkehrsangebot treten also weder auf Strecken- noch auf Fahrplanebene Überlastungen auf.

6 Fazit und Ausblick

In der vorliegenden Untersuchung wurde anhand einer einfachen Testinstanz gezeigt,

a) dass eine planerische Lösung durch Optimierungs Routinen verbessert werden kann, die eine bessere Synchronisation erreichen und sparsamere Frequenzen nutzen,

b) dass es möglich ist, Lösungen mit vergleichbarer Qualität auch vollautomatisch zu erzeugen,

c) dass algorithmische Lösungen einer Rückkopplung mit der Umlegung bedürfen, um Kapazitätsüberschreitungen auszuschließen, und

d) dass die besten Ergebnisse durch ein Zusammenspiel von Planung und Optimierung erzielt werden, nämlich wenn man die Linien aus der planerischen Lösung mit automatisch erzeugten Linien zusammenlegt, um den Linienpool für die Optimierung zu bilden.

Es ist zu bemerken, dass die betrachteten Kennzahlen nicht die einzigen Kriterien zur Beurteilung der Qualität einer Lösung sein können. So zeichnet sich die planerische Lösung gegenüber der algorithmischen Lösung durch ein symmetrisches Liniennetz mit weniger Linien aus, was die Übersichtlichkeit verbessert. Außerdem erleichtert die Anwendung eines Systemtakts die Merkbarkeit des Fahrplans.

7 Literatur

7.1. Bücher


7.2. Zeitschriftenartikel


7.3. Beiträge aus Tagungsbänden


7.4. Schriftenreihen


7.5. Sonstiges


B. System Headways in Line Planning

M. Friedrich, M. Hartl, A. Schiewe, A. Schöbel
System Headways in Line Planning
[Friedrich et al., 2018a]
System Headways in Line Planning

Markus Friedrich · Maximilian Hartl · Alexander Schiewe · Anita Schöbel

Abstract Line Planning is an important stage in public transport planning. This stage determines which lines should be operated with which frequencies. Several integer programming models provide solutions for the line planning problem. However, when solving real-world instances, integer optimization often falls short since it neglects objectives that are hard to measure, e.g., memorability of the system. Adoptions to known line planning models are hence necessary.

We analyze one such adaption, namely that the frequencies of all lines should be multiples of a fixed system headway. This is common in practice and improves memorability and practicality of the designed line plan. We model the requirement of such a common system headway as an integer program and compare line plans with and without this new requirement theoretically by investigating worst case bounds, as well as experimentally on artificial and close to real-world instances.

Keywords Public Transport Planning · Line Planning · Integer Optimization
1 Introduction

Line planning in public transport is a well researched problem. Its goal is to choose the number and the shape of the lines to be operated and to determine their frequencies, i.e., how often services should be offered along every line within the planning period $T$. The lines together with their frequencies are called a line concept. Existing models optimize the costs, e.g., (Claessens et al, 1998b), (Goossens et al, 2006), the number of direct travelers, e.g., (Dienst, 1978),(Bussieck, 1998), or the approximated passengers’ traveling times, e.g., (Schmidt, 2014), (Schöbel and Scholl, 2006) of the line concept. Overviews on different models can be found in (Schöbel, 2012) and (Kepaptsoglou and Karlaftis, 2009).

Recent developments include different planning stages into the line planning problems, i.e., they consider integrated planning in public transport. Examples are to integrate the timetabling step (Burggraeve et al, 2017), the demand (Viggiano, 2017) or treating several planning stages in an integrated way (Schöbel, 2017; Huang et al, 2018). Other work examine the effect of time dependent demand (Borndörfer et al, 2018) or the differences of route choice and assignment (Goerigk and Schmidt, 2017).

Nevertheless, solutions to the line planning problem often fall short in important criteria that are not easily measurable in integer optimization problems. One important criterion is the memorability of the resulting timetable. Ideally, public transport passengers need to memorize only one specific minute and a headway for a particular stop, e.g., minute 01 every 10 minutes. To achieve such properties, transport planners use specific concepts when designing line plans. One common concept is a system or pulse headway describing a minimum headway, which must be achieved by all lines, see (Vuchic et al, 1981) and (Vuchic, 2017). The application of a system headway leads not only to regular departure times but also to regular connections when passengers have to transfer.

More precisely, let a line concept consisting of a set of lines $\mathcal{L}$ and their frequencies $f_l$ for all $l \in \mathcal{L}$ be given. If there exists a natural number $i \neq 1$ which is a common divisor of all frequencies $f_l$ we say that the line concept has a system headway.

In this paper, we want to model the concept of a system headway mathematically. In particular, we show how the requirement for a system headway can be added to existing integer optimization models, and we derive properties for general line planning models and for a cost-based formulation.

2 Modeling system headways

Before we introduce our adaptions to the integer programming models, we define formally what the line planning problem is. Let a public transport network $\text{PTN}=(V,E)$ be given, with nodes $V$ as stations and undirected edges $E$ between them. A line $l$ is a path in the PTN. In this paper we assume that a
line pool $\mathcal{L}$ is given. It contains a (large) set of potential lines from which we want to choose the ones to establish. A line concept $(\mathcal{L}, f)$ assigns a frequency $f_l \in \mathbb{N}_0$ to every line $l$ in the given line pool $\mathcal{L}$. (Lines which are not chosen from the pool receive a frequency of zero).

There exist many different models for line planning. The frequencies $f_l$ for all $l \in \mathcal{L}$ are the variables to be determined in all line planning models. Sometimes, additional variables $x \in X \subseteq \mathbb{R}^n$ are also present which might for example be used for modeling the paths of the passengers.

The general line planning model can hence be written as

$$(P) \quad \min \, \text{obj}(f, x)$$

s.t. $g(f, x) \leq b$

\begin{align*}
  f_l & \in \mathbb{N}_0 \quad \text{for all } l \in \mathcal{L} \\
  x & \in X,
\end{align*}

where $g : \mathcal{L} \times X \rightarrow \mathbb{R}^m$ is a linear function containing $m$ constraints and $b \in \mathbb{R}^m$. Common choices for the linear objective function $\text{obj} : \mathcal{L} \times X \rightarrow \mathbb{R}$ are to minimize the costs or the traveling time of the passengers, or to maximize the number of direct travelers. The constraints are written in the general form $g(f, x) \leq b$, but as noted in (Schöbel, 2012) most line planning models contain constraints of the type

$$\sum_{l \in \mathcal{L} : e \in l} f_l \geq f_{e\min}^l \quad \forall e \in E,$$  \hspace{1cm} (LEF)

and of the type

$$\sum_{l \in \mathcal{L} : e \in l} f_l \leq f_{e\max}^l \quad \forall e \in E$$  \hspace{1cm} (UEF)

for given lower and upper edge frequency bounds $f_{e\min}^l \leq f_{e\max}^l$ for every edge $e \in E$. The constraints (LEF) are called lower edge frequency constraints and are used to ensure that all passengers can be transported while the upper edge frequency constraints (UEF) are needed due to the limited capacity of tracks, or due to noise restrictions. They also bound the costs of the line concept. Allowing to set $f_{e\min}^l = 0$ and $f_{e\max}^l = \infty$ we can without loss of generality assume that constraints of type (LEF) and (UEF) always are present in the general line planning model.

Typically, cost-oriented models minimize the costs of a line concept and contain (LEF) while passenger-oriented models optimize the traveling time or the number of transfers passengers have. To prevent the model to establish all lines with high frequencies, constraints of type (UEF) may be used or a budget constraint (BUD) (see Section 5).

The main definition for this work is the following.
Definition 1 A system headway (also called system frequency) is defined as a common divisor of all frequencies \( f_l, l \in \mathcal{L} \), i.e., \( i \in \mathbb{N} \) is a system headway for \((\mathcal{L}, f)\) if and only if \( i \geq 2 \) and \( i f_l \) for all \( l \in \mathcal{L} \).

In the following we look for line concepts which have a system headway. Note that we only consider system headways greater than one, as choosing \( i = 1 \) as a system headway poses no restriction on the model and is therefore considered as having no system headway at all.

Including the system headway requirement into the general line planning model (P) is possible with only small adaptions. Let us first consider a given and fixed system headway \( i \in \mathbb{N} \). Since the frequencies \( f_l \) are integer variables we can include a system headway by adding only the constraints (1) and (2):

\[
\begin{align*}
(P(i)) \quad & \min \ obj(f, x) \\
& \text{s.t.} \quad g(f, x) \leq b \\
& \quad \quad \quad f_l = \alpha_l \cdot i \quad \forall l \in \mathcal{L}, \quad (1) \\
& \quad \quad \quad \alpha_l \in \mathbb{N}_0 \quad \forall l \in \mathcal{L} \quad (2)
\end{align*}
\]

\( f_l \in \mathbb{N}_0 \) for all \( l \in \mathcal{L} \) \( x \in X \).

By \( \text{opt}(i) \) we denote the optimal objective function value of \( P(i) \). At first it is unclear, whether (1) and (2) add to the difficulty of the model. In fact, they do not do this, as the following theorem shows.

**Theorem 1** Let \( (P) \) be a general line planning problem for a given instance based on the period \( T \). Then problem \( P(i) \) is equivalent to a line planning problem \( (P') \). The new line planning problem \( (P') \) has the same number of variables and constraints as \( (P) \).

**Proof** We introduce new variables \( f'_l := \frac{f_l}{i} \) for all \( l \in \mathcal{L} \). Substituting \( f_l \) by these new variables in \( P(i) \) and using the linearity of \( obj \) and of \( g \), we receive

\[
\begin{align*}
(P'(i)) \quad & \min \ i \cdot obj(f', x) \\
& \text{s.t.} \quad i \cdot g(f', x) \leq b \\
& \quad \quad \quad i \cdot f'_l = \alpha_l \cdot i \quad \forall l \in \mathcal{L} \\
& \quad \quad \quad \alpha_l \in \mathbb{N}_0 \quad \forall l \in \mathcal{L} \\
& \quad \quad \quad f_l \in \mathbb{N}_0 \quad \text{for all } l \in \mathcal{L} \quad (1) \\
& \quad \quad \quad x \in X
\end{align*}
\]

From \( i \cdot f'_l = i\alpha_l \) we conclude that \( f_l = \alpha_l \) for all \( l \in \mathcal{L} \) and the variables \( \alpha_l \) are not needed any more. \( P'(i) \) hence simplifies to

\[
\begin{align*}
(P'(i)) \quad & \min \ obj(f', x) \\
& \text{s.t.} \quad g(f', x) \leq \frac{b}{i} \\
& \quad \quad \quad f_l \in \mathbb{N}_0 \quad \text{for all } l \in \mathcal{L} \\
& \quad \quad \quad x \in X
\end{align*}
\]
which is a line planning problem with the same number of variables and constraints, but a right hand side \( \frac{b}{T} \).

Note that the new line planning problem can be interpreted as using the period \( T' := \frac{T}{t} \) instead of \( T \). This can be seen by looking at (LEF) and (UEF) which in (P') now read as

\[
\frac{f_{\min}}{t} \leq \sum_{l \in L; e \in l} f_l \leq \frac{f_{\max}}{t} \quad \forall e \in E,
\]

i.e., we restrict how many vehicles are allowed to pass an edge in the new period \( T' := \frac{T}{t} \).

**Example 1** We are interested in a solution with system headway \( i = 4 \). Then instead of using lower and upper edge frequency bounds of 3 and 6, respectively, we can bound the number of vehicles running along this edge within 15 minutes to be between \( \frac{3}{4} \) and \( \frac{6}{4} \). Since

\[
\sum_{l \in L; e \in l} f_l \in \mathbb{N}
\]

we can furthermore use integer rounding and obtain the only feasible solution of four vehicles per hour running along this particular edge.

It might also be interesting to determine the line concept with a best possible system headway, i.e., we have no particular number \( i \) for a system headway given but we wish to find a line concept which satisfies the system headway requirement for some natural number \( i \geq 2 \). A naive approach is to solve \( P(i) \) for all \( i \) smaller than the period length \( T \) and choose the solution with best objective value \( \text{opt}(i) \). However, choosing the best possible system headway can also be formulated as an integer quadratic program by adding the constraints (3) and (4) to \( P(i) \) and hence leaving \( \alpha = i \) as variable:

\[
(P_{\text{sys-head}}) \quad \min \ obj(f, x) \\
\text{s.t.} \quad g(f, x) \leq b \\
\quad f_l = \alpha_l \cdot \alpha \quad \forall l \in L, \\
\quad \alpha_l \in \mathbb{N}_0 \quad \forall l \in L \\
\quad \alpha \geq 2 \\
\quad f_l \in \mathbb{N}_0 \quad \text{for all } l \in L \\
\quad x \in X \\
\quad \alpha \in \mathbb{N}. \tag{3}
\]

In the following we analyze which system headways are reasonable and how much one loses in quality or costs of a line plan when (the best) system headway is chosen. We first have a look at the general line planning problem and then discuss the classic cost-oriented model and the direct travelers approach.
3 The size of a system headway in the general line planning problem

In this section we investigate which numbers \( i \) are suitable as system headways and how we can find a best solution among all possible system headways.

In the following we compare the result of \( P(i) \) for different values of \( i \). Our first result states that a divisor \( i \) of a given system headway \( j \) always yields a better solution than using \( j \) itself. This holds for all general line planning problems.

**Lemma 1** Let \( i, j \in \mathbb{Z} \) and \( i \mid j \). Then \( \text{opt}(i) \leq \text{opt}(j) \).

*Proof* Let \((f(i), x(i))\) denote a feasible solution to \( P(i) \), and \((f(j), x(j))\) denote a feasible solution to \( P(j) \). This means \( j \mid f(j) \). Together with the assumption \( i \mid j \) we obtain that \( i \mid f(j) \), hence \( f(j) \) satisfies (1) and (2) also in \( P(i) \). The other constraints \( g(x, f) \leq b \) of \( P(i) \) are also constraints of \( P(j) \), hence every feasible solution for \( P(j) \) is also feasible for \( P(i) \) and their objective functions coincide. Therefore, \( P(i) \) is a relaxation of \( P(j) \) and \( \text{opt}(i) \leq \text{opt}(j) \). \( \square \)

The previous lemma shows that searching for the best solution using a system headway can be done more efficiently: Instead of testing every possible value, it is enough to restrict ourselves to prime numbers.

**Corollary 1** There always exists an optimal solution \((\alpha, f, x)\) to \((P_{\text{sys-head}})\) in which the optimal system headway \( \alpha \) is a prime number.

Unfortunately, it cannot be seen beforehand which prime number results in the best solution. In practice, choosing a smaller system headway is often better (as can be seen in Section 6). However, depending on the constraints \( g(f, x) \leq b \), there are counterexamples where a smaller system headway is not even feasible. This is even true if \( g(f, x) \leq 0 \) only consists of lower and upper edge frequency constraints (LEF) and (UEF) as the following example shows.

**Example 2** Consider a simple PTN with only two stations and a connecting edge, as depicted in Fig. 1. Let the lower and upper edge frequencies of this edge be both set to three. Then there is a feasible solution for a system headway of \( i = 3 \) but not for \( i = 2 \).

Such examples raise the question in which cases \((P_{\text{sys-head}})\) has a feasible solution. Clearly, if the original line planning problem \((P)\) is infeasible then certainly also all \( P(i) \) and \((P_{\text{sys-head}})\) are. As Example 2 shows, (LEF)
Lemma 2 Let \((P)\) be a general line planning problem containing constraints of type (LEF) and of type (UEF). \((P_{\text{sys-head}})\) is infeasible if there exists an edge \(e\) with \(f_{e}^{\min} = f_{e}^{\max} = 1\).

Proof Edge \(e\) needs to be covered by exactly one line \(l\) with frequency \(f_l = 1\) which then is not an integer multiple of any \(i \geq 2\).

On the other hand, in case the only constraints contained in \(g(l, x) \leq b\) are constraints of type (LEF), then we have a positive result.

Lemma 3 Let \((P)\) be a feasible line planning problem in which only has constraints of type (LEF) or constraints which depend on \(x\), but not on \(f\). Then \(P(i)\) is feasible for all possible system headways \(i \geq 2\).

Proof Take a solution \((f, x)\) for \((P)\). For all \(l \in L\) define

\[
f'_l := \min\{k : i|k\ and\ k \geq f_l\}.
\]

Then \(f'_l\) satisfies (1) and (2). Furthermore, since \(f'_l \geq f_l\) also (LEF) are satisfied, and satisfaction of constraints which just depend on \(x\) is not changed when replacing \(f\) by \(f'\). Hence, \((f', x)\) is a feasible solution to \(P(i)\).

Note, that even if the conditions of Lemma 3 are met, a smaller system headway does not need to be better, as can be seen in Example 3.

4 Bounds for a cost model in line planning

We now turn our attention to a particular model in line planning, namely the basic cost model. It has been extracted from the cost model in Claessens et al. (1998a) and stated in Schöbel (2012). The model allows to study how much we lose when requiring a system headway compared to the original model without the system headway requirement.

Since we know from Lemma 2 that (UEF) may destroy feasibility of line planning problems we only consider problems without upper edge frequency bounds for the rest of this section, i.e.,

\[
f_e^{\max} = \infty \quad \forall e \in E.
\]

The cost model we study here is the following: Passengers are first routed along shortest paths in the PTN. The number of passengers which travel along edge \(e\) in these shortest paths is then counted and divided by the (common)
capacity of the vehicles. This gives the minimal number of vehicles $f^\text{min}_e$ needed to cover edge $e$. The costs of a line concept are approximated as

$$\text{cost}(\mathcal{L}, f) = \sum_{l \in \mathcal{L}} f_l \cdot \text{cost}_l,$$

where $\text{cost}_l$ is a given cost per line $l \in \mathcal{L}$. This often includes time- and distance-based costs of a line. In this work, we pose no assumptions on the structure of the costs $\text{cost}_l$, i.e., they can be chosen arbitrarily for each line. Including the system headway requirement results in model $P(i)$:

$$\min \sum_{l \in \mathcal{L}} f_l \cdot \text{cost}_l$$

$$\text{s.t. } f^\text{min}_e \leq \sum_{l \in \mathcal{L} : e \in l} f_l \quad \forall e \in E$$

$$f^\text{max}_e \geq \sum_{l \in \mathcal{L} : e \in l} f_l \quad \forall e \in E$$

$$f_l = \alpha_l \cdot i \quad \forall l \in \mathcal{L}$$

$$f_l, \alpha_l \in \mathbb{N}_0 \quad \forall l \in \mathcal{L}$$

As before, $\text{opt}(i)$ denotes the optimal cost value for $P(i)$.

First note, that even in this simple model, $\text{opt}(i) \leq \text{opt}(j)$ for $i \leq j$ need not hold as the next example shows.

**Example 3** Consider again the simple PTN of Fig. 1. Let the lower edge frequency of this edge be three as before, while the upper edge frequency is now deleted (or set to $f^\text{max}_e = \infty$). Let only one line $l$ serve edge $e$. Then the optimal solution for a system headway of $i = 3$ is $f_l = 3$ which leads to an objective function value $\text{opt}(3) = 3 \cdot \text{cost}_l$. Now, taking a smaller system headway of $i = 2$ requires a frequency of $f_l = 4$ for line $l$ in order to serve edge $e$. This means we obtain

$$\text{opt}(2) = 4 \cdot \text{cost}_l > 3 \cdot \text{cost}_l = \text{opt}(3).$$

Nevertheless, even if monotonicity does not hold, the structure of the cost model allows to prove the following result.

**Theorem 2** Let $i, j \in \mathbb{Z}$, $i \leq j$. Then $\text{opt}(j) \leq \frac{j}{i} \text{opt}(i)$.

**Proof** Let $f^i$ be an optimal solution to $P(i)$. Then $f' = \frac{j}{i} f^i$ is a feasible solution for $P(j)$, since $j f'$ and the lower edge frequency requirements (LEF) are still satisfied:

$$\sum_{l \in \mathcal{L} : e \in l} f'_l = \sum_{l \in \mathcal{L} : e \in l} \frac{j}{i} f^i_l \geq \sum_{l \in \mathcal{L} : e \in l} f^i_l \geq f^\text{min}_e \quad \forall e \in E.$$
Therefore, the optimal objective value of $P(j)$ can be bounded by the objective value of $f'$:

$$\text{opt}(j) \leq \sum_{l \in L} f'_{l} \cdot \text{cost}_l = \sum_{i \in L} \frac{j}{i} f'_{i} \cdot \text{cost}_l = \frac{j}{i} \text{opt}(i).$$

\[\square\]

Note that this lemma also holds for $i = 1$, i.e., the case for no system headway. This yields the following corollary.

The result also allows to compare the costs of an optimal solution for the original problem $(P)$ to the costs of an optimal solution for problem $P(i)$ with a system headway of $i$.

**Corollary 2** Let $\text{opt}$ be the optimal objective value of the cost model. Then the optimal costs $\text{opt}(i)$ of a system headway $i$ compared to the model without the requirement of a system headway are bounded by

$$\text{opt}(i) \leq i \cdot \text{opt}^*.$$ 

Therefore requiring a system headway of, e.g., $i = 2$ can in the worst case double the costs.

Although this factor is often not attained in practice (see Section 6), the bound is sharp.

**Example 4** Consider again the simple PTN of Fig. 1 but now with a lower edge frequency of one, i.e., the edge must be covered and only one line $l$ serving edge $e$. Then the optimal solutions for a system headway of 2 and 3 fulfill:

$$\text{opt}(2) = 2 \cdot \text{cost}_l = \frac{2}{3} \cdot 3 \cdot \text{cost}_l = \frac{2}{3} \text{opt}(3)$$

### 5 Passenger-oriented models

There are several passenger oriented models known in literature. We mainly consider the direct traveler model introduced in (Bussieck, 1998). For this problem, the number of direct travelers, i.e., the number of passengers that can travel from their origins to their destinations without changing lines, should be maximized. Other models try to minimize the approximated travel time of the passengers, e.g., (Schöbel and Scholl, 2006; Borndörfer et al, 2007).

Passenger oriented models need other types of constraints than those in the cost model of Section 4. Including (LEF) may not be necessary any more since the passengers are treated in the objective function. Including (LEF) is one way to restrict the costs of the line plan (and used, e.g., in Bussieck (1998)). There may also be a budget constraint in the form of

$$\sum_{l \in L} \text{cost}_l \cdot f_l \leq B,$$ 

(BUD)
where $\text{cost}_l$ are given cost coefficients for every line $l \in \mathcal{L}$ which may include time- and distance-based costs of a line. In this work, we pose no assumptions on the structure of the costs $\text{cost}_l$, i.e., they can be chosen arbitrarily for each line.

When we remove such a constraint from a passenger oriented model, the problem often becomes trivial, since it might be an optimal solution to establish all lines with high frequencies (which can then be chosen as multiples of the given system headway $i$). Hence, a constraint of the type of (BUD) is necessary. However, with a budget constraint, we obtain similar problems to Lemma 2, as can be seen in the following example.

**Example 5** We again consider the PTN given in Fig. 1. When we now assume that we have a budget constraint restricting the costs of the solution to a single line with frequency 1, there is no feasible solution for any system headway.

Similarly, we can construct examples equivalent to Example 2 and Example 3.

The conclusion is the following: It can always happen that the original line planning model (P) is feasible while the corresponding problem $\text{P}(i)$ with a fixed system headway $i$ or even $(\text{P}_{\text{sys-head}})$ become infeasible. This means that a result such as Theorem 2 for the cost model is not possible for (reasonable) passenger-oriented models and that the relative difference between the objective of a system headway and the objective without this requirement may be arbitrarily large.

### 6 Experiments

For the practical experiments, we consider three instances with different characteristics:

**Grid**: A small example first presented in (Friedrich et al, 2017). It is designed to be small enough to understand effects of decisions but still contains a realistic demand structure. It has 25 stops, 40 edges and 2546 passengers. For a representation of the infrastructure, see Fig. 2a. The instance has been tackled by several researchers and can be downloaded at (Grid, 2018).

**Goettingen**: An instance based on the bus network in Göttingen, a small city in the geographical center of Germany. It contains 257 stops, 548 edges and 406146 passengers. For a representation of the infrastructure, see Fig. 2b.

**Germany**: An instance based on the long-distance rail system in Germany. It contains 250 stops, 326 edges and 3147382 passengers. For a representation of the infrastructure, see Fig. 2c.

All experiments are done using the LinTim-software framework (Goerigk et al, 2013; Schiewe et al, 2018). We computed a line concept without system headway as well as for every system headway from 2 to 10 while optimizing the given line planning problem.
First, we consider solving the cost model discussed in Section 4. An evaluation containing the costs of the different solutions and the worst case costs of Lemma 2 can be found in Fig. 3.

There are mainly two things to observe here: First of all, the assumption that higher system headways lead to higher costs is often, but not always true. In all but one case, the costs are strictly increasing for increasing system headways.

But, as was seen in Section 3, this does not always have to be the case. This can be observed in Fig. 3a where the solution for a system headway of \( i = 3 \) has lower costs than the solution for a system headway of \( i = 2 \). This occurs in cases where the demand on most edges can be met by lines with a frequencies of three. Then a system headway of \( i = 2 \) leads either to more lines or to line frequencies of four.

Additionally, note that the worst case factor for using a system headway from Lemma 2 is not obtained in practice but the difference to the theoretical bound decreases with increasing instance size.
Fig. 3: Solutions for the Cost Model

Fig. 4: Solutions for the Direct Travelers Model
Next, we consider the case of a passenger-oriented line planning model. We chose the direct travelers model of (Bussieck, 1998), see also Section 5. For this, we set a budget to examine the effect of the system headway on a restricted problem.

In Fig. 4 we can clearly see the effects of the system headway. In the instance Goettingen (Fig. 4a), we again observe that the quality of the line plan decreases most of the times with increasing system headway but there may be cases where a bigger system headway can use the given budget a little bit better, resulting in a better plan for the passenger. Hence, monotonicity of the objective function is also here likely, but not guaranteed.

In the instance Germany (Fig. 4b), we see the effect of a late drop-off of the quality, resulting from a budget that is big enough to not be restrictive for the first few cases.

It has been recognized in several publications (Burggraewe et al, 2017; Schöbel, 2017; Huang et al, 2018) that line planning should not be treated isolated from other planning stages, but an integrated approach is needed. We are hence interested not only in the effects a system headway has on line plans, but also consider if there are effects on the resulting timetable. Note that the line plan influences the resulting passengers’ travel time obtained by the timetable significantly Friedrich et al (2017); Goerigk et al (2013).

To consider the results of system headways on the timetable, we compute a periodic timetable for each of the line plans and compare their qualities, evaluating the perceived travel time of the passengers in the timetable, i.e., the travel time including a small penalty for every transfer. For the computation of the timetable, we use the fast MATCH approach introduced in (Pätzold and Schöbel, 2016). The results are depicted in Fig. 5.

Again, we see the anticipated results: A higher system headway results in a public transport supply with shorter headways. This leads in many cases to shorter transfer waiting times and reductions in the perceived travel time,
indicating a higher quality for the passengers. However, also here, this inter-
relation does not apply without exception as Fig. 5 shows.

7 Outlook

We added the system headway constraint to line planning models, derived the-
etorical bounds on their effects and examined the results on practical instances
for a cost model and a passenger-oriented model. It would be interesting to see
the proposed system headway adjustments implemented into even more line
planning models to further extend the comparison and examine the effects on
public transport systems.

Another interesting topic is the evaluation of the impact of a system head-
way on passengers. Important metrics, such as the memorability of a timetable,
can only be measured inadequately using the state-of-the-art mathematical
evaluation systems and can therefore not be compared conclusively. One way
of evaluating the impacts is to estimate the changes in public transport travel
demand. This requires a mode choice model, which captures not only travel
time and number of transfers as indicators for service quality, but also the
service frequency and the regularity. This can be achieved by an indicator
adaption time, which quantifies the time difference between the desired depar-
ture time of a traveler and the provided departure time of the public transport
supply. In car transport the adaption time is always zero. A public transport
supply with regular and short headways reduces adaption time and thus makes
public transport more competitive. Experiments with the grid instance indi-
cate that especially in networks with low demand the additional costs of a
system headway can partially be compensated by a shift from car to public
transport. In networks where high demand leads to solutions with headways
below 10 minutes, the impact of a system headway on additional cost and
demand is smaller. Here the modal share primarily depends on differences in
in travel time and travel costs. Future work is necessary to better understand
the impact of regularity and adaption time on passengers travel behavior.

References

to line planning in public transport. Transportation Science 41:123–132
T (2018) Line planning on path networks with application to the istanbul
timetabling in line plan optimization for railway systems. Transportation
Research C 77:134–160
nische Universität Braunschweig


Schöbel A (2017) An eigenmodel for iterative line planning, timetabling and
vehicle scheduling in public transportation. Transportation Research C
Schöbel A, Scholl S (2006) Line planning with minimal transfers. In: 5th Work-
shop on Algorithmic Methods and Models for Optimization of Railways, no.
060901 in Dagstuhl Seminar Proceedings
Viggiano C (2017) Bus network sketch planning with origin-destination travel
data. PhD thesis, Massachusetts Institute of Technology
Wiley & Sons
Vuchic VR, Clarke R, Molinero A (1981) Timed transfer system planning,
design and operation. Departmental Papers (ESE)
C. Integrating Passengers’ Assignment in Cost-Optimal Line Planning

M. Friedrich, M. Hartl, A. Schiewe, A. Schöbel
Integrating Passengers’ Assignment in Cost-Optimal Line Planning
[Friedrich et al., 2017b]
Integrating Passengers' Assignment in Cost-Optimal Line Planning

Markus Friedrich¹, Maximilian Hartl², Alexander Schiewe³, and Anita Schöbel⁴

¹ Lehrstuhl für Verkehrsplanung und Verkehrsleittechnik, Universität Stuttgart, Stuttgart, Germany
markus.friedrich@isv.uni-stuttgart.de

² Lehrstuhl für Verkehrsplanung und Verkehrsleittechnik, Universität Stuttgart, Stuttgart, Germany
maximilian.hartl@isv.uni-stuttgart.de

³ Institut für Numerische und Angewandte Mathematik, Universität Göttingen, Göttingen, Germany
a.schiewe@math.uni-goettingen.de

⁴ Institut für Numerische und Angewandte Mathematik, Universität Göttingen, Göttingen, Germany
schoebel@math.uni-goettingen.de

Abstract
Finding a line plan with corresponding frequencies is an important stage of planning a public transport system. A line plan should permit all passengers to travel with an appropriate quality at appropriate costs for the public transport operator. Traditional line planning procedures proceed sequentially: In a first step a traffic assignment allocates passengers to routes in the network, often by means of a shortest path assignment. The resulting traffic loads are used in a second step to determine a cost-optimal line concept. It is well known that travel time of the resulting line concept depends on the traffic assignment. In this paper we investigate the impact of the assignment on the operating costs of the line concept.

We show that the traffic assignment has significant influence on the costs even if all passengers are routed on shortest paths. We formulate an integrated model and analyze the error we can make by using the traditional approach and solve it sequentially. We give bounds on the error in special cases. We furthermore investigate and enhance three heuristics for finding an initial passengers’ assignment and compare the resulting line concepts in terms of operating costs and passengers’ travel time. It turns out that the costs of a line concept can be reduced significantly if passengers are not necessarily routed on shortest paths and that it is beneficial for the travel time and the costs to include knowledge on the line pool already in the assignment step.

1998 ACM Subject Classification G.1.6 Optimization, G.2.2 Graph Theory, G.2.3 Applications

Keywords and phrases Line Planning, Integrated Public Transport Planning, Integer Programming, Passengers’ Routes

Digital Object Identifier 10.4230/OASIcs.ATMOS.2017.5

Introduction
Line planning is a fundamental step when designing a public transport supply, and many papers address this topic. An overview is given in [18]. The goals of line planning can roughly...
be distinguished into passenger-oriented and cost-oriented goals. In this paper we investigate cost-oriented models, but we evaluate the resulting solutions not only with respect to their costs but also with respect to the approximated travel times of the passengers.

In most line planning models, a line pool containing potential lines is given. The cost model chooses lines from the given pool with the goal of minimizing the costs of the line concept. It has been introduced in [5, 26, 25, 6, 12] and later on research provided extensions and algorithms.

Traditional approaches are two-stage: In a first step, the passengers are routed along shortest paths in the public transport network, still without having lines. This shortest path traffic assignment determines a specific traffic load describing the expected number of travelers for each edge of the network. The traffic loads and a given vehicle capacity are then used to compute the minimal frequencies needed to ensure that all passengers can be transported. These minimal frequencies serve as constraints in the line planning procedure. We call these constraints lower edge frequency constraints. Lower edge frequency constraints have first been introduced in [24]. They are used in the cost models mentioned above, but also in other models, e.g., in the direct travelers approach ([7, 4, 3]), or in game-oriented models ([15, 14, 20, 21]).

If passengers are routed along shortest paths, the lower edge frequency constraints ensure that in the resulting line concept all passengers can be transported along shortest paths. Although the travel time for the passengers includes a penalty for every transfer, routing them along shortest paths in the public transport network (PTN) guarantees a sufficiently short travel time. However, routing passengers along shortest paths may require many lines and hence may lead to high costs for the resulting line plan. An option is to bundle the passengers on common edges. To this end, [13] proposes an iterative approach for the passengers’ assignment in which edges with a higher traffic load are preferred against edges with a lower traffic load in each assignment step. Other papers suggest heuristics which construct the line concept and the passengers’ assignment alternately: after inserting a new line, a traffic assignment determines the impacts on the traffic loads ([23, 22, 17]).

Our contribution: We present a model in which passengers’ assignment is integrated into cost-optimal line planning. We show that the integrated problem is NP-hard.

We analyze the error of the sequential approach compared to the integrated approach: If passengers’ are assigned along shortest paths, and if a complete line pool is allowed, we show that the relative error made by the assignment is bounded by the number of OD-pairs. We also show that the passengers’ assignment has no influence in the relaxation of the problem. If passengers can be routed on any path, the error may be arbitrarily large.

We experimentally compare three procedures for passengers’ assignment: routing along shortest paths, the algorithm of [13] and a reward heuristic. We show that they can be enhanced if the line pool is already respected during the routing phase.

2 Sequential approach for cost-oriented line planning

We first introduce some notation. The public transport network $\text{PTN}=(V,E)$ is an undirected graph with a set of stops (or stations) $V$ and direct connections $E$ between them. A line is a path through the PTN, traversing each edge at most once. A line concept is a set of lines $L$ together with their frequencies $f_l$ for all $l \in L$. For the line planning problem, a set of potential lines, the so-called line pool $L^0$ is given. Without loss of generality we may assume that every edge is contained in at least one line from the line pool (otherwise reduce the set
Algorithm 1: Sequential approach for cost-oriented line planning.

**Input:** PTN = (V, E), Wuv for all u, v ∈ V, line pool L0 with costs cl for all l ∈ L0

1. Compute traffic loads we for every edge e ∈ E using a passengers’ assignment algorithm (Algorithm 2)
2. For every edge e ∈ E compute the lower edge frequency \( f_{\text{min}}^e := \lceil \frac{w_e}{\text{Cap}} \rceil \)
3. Solve the line planning problem LineP(\( f_{\text{min}} \)) and receive (L, fl)

Algorithm 2: Passengers’ assignment algorithm.

**Input:** PTN = (V, E), Wuv for all u, v ∈ V

for every u, v ∈ V with Wuv > 0 do
    Compute a set of paths \( P_{1,uv}^{uv}, \ldots, P_{N,uv}^{uv} \) from u to v in the PTN
    Estimate weights for the paths \( \alpha_{1,uv}^{uv}, \ldots, \alpha_{N,uv}^{uv} \geq 0 \) with \( \sum_{i=1}^{N,uv} \alpha_{i,uv}^{uv} = 1 \)
end for every e ∈ E do
    Set \( w_e := \sum_{u,v \in V} \sum_{i=1}^{N,uv} \alpha_{i,uv}^{uv} W_{uv} \)
end

of edges E). If the line pool contains all possible paths as potential lines we call it a complete pool. For every line \( l \in L^0 \) in the pool its costs are

\[
\text{cost}_l = c_{\text{km}} \sum_{e \in l} d_e + c_{\text{fix}},
\]

i.e., proportional to its length plus some fixed costs, where \( d_e \) denotes the length of an edge. Without loss of generality we assume that \( c_{\text{km}} = 1 \).

The demand is usually given in form of an OD-matrix \( W \in \mathbb{R}^{|V| \times |V|} \), where Wuv is the number of passengers who wish to travel between the stops u, v ∈ V. We denote the number of passengers as |W| and the number of different OD pairs as |OD|.

The traditional approaches for cost-oriented line planning work sequentially. In a first step, for each pair of stations (u, v) with Wuv > 0 the passenger-demand is assigned to possible paths in the PTN. Using these paths, for every edge e ∈ E the traffic loads are computed. Given the capacity Cap of a vehicle, one can determine \( f_{\text{min}}^e := \lceil \frac{w_e}{\text{Cap}} \rceil \), i.e., how many vehicle trips are needed along edge e to satisfy the given demand. These values \( f_{\text{min}}^e \) are called lower edge frequencies. They are finally used as input for determining the lines and their frequencies, Algorithm 1.

The problem LineP(\( f_{\text{min}} \)) is the basic cost model for line planning:

\[
\min \left\{ \sum_{l \in L^0} f_l \cdot \text{cost}_l : \sum_{l \in L^0} \sum_{e \in l} f_l \geq f_{\text{min}}^e \text{ for all } e \in E, f_l \in \mathbb{N} \text{ for all } l \in L^0 \right\}.
\]

Cost models (and extensions of them) have been extensively studied as noted in the introduction.

Step 1 in Algorithm 1 is called passengers’ assignment. The basic procedure is described in Algorithm 2.

There are many different possibilities how to compute a set of paths and corresponding weights \( \alpha_{1,uv}^{uv} \); we discuss some in Section 5. In cost-oriented models, often shortest paths through the PTN are used. I.e., \( N_{uv} = 1 \) for all OD-pairs \( \{u, v\} \) and \( P_{1,uv}^{uv} = P_{uv} \) is an
Algorithm 3: Sequential approach for cost-oriented line planning.

**Input:** PTN = \((V, E), W_{uv}\) for all \(u, v \in V\), line pool \(\mathcal{L}^0\) with costs \(c_l\) for all \(l \in \mathcal{L}^0\)

1. Compute traffic loads \(w_e\) for every edge \(e \in E\) using a passengers’ assignment algorithm (Algorithm 2).
2. Solve the line planning problem \(\text{LineP}(w)\) and receive \((\mathcal{L}, f_l)\) (arbitrarily chosen) shortest path from \(u\) to \(v\) in the PTN. We call the resulting traffic loads shortest-path based. Furthermore, let \(\text{SP}_{uv} := \sum_{e \in P_{uv}} d_e\) denote the length of a shortest path between \(u\) and \(v\).

In order to analyze the impacts of the traffic loads \(w_e\) on the costs, note that for integer values of \(f_l\) we have for every \(e \in E\):

\[
\sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq \left\lfloor \frac{w_e}{\text{Cap}} \right\rfloor \iff \text{Cap} \sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq w_e,
\]

hence we can rewrite (2) and receive the equivalent model \(\text{LineP}(w)\) which directly depends on the traffic loads:

\[
\text{LineP}(w) \quad \min g^{\text{cost}}(w) := \sum_{l \in \mathcal{L}^0} f_l \text{cost}_l \\
\text{s.t.} \quad \text{Cap} \sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq w_e \text{ for all } e \in E \quad (3) \\
\quad f_l \in \mathbb{N} \text{ for all } l \in \mathcal{L}^0
\]

We can hence formulate Algorithm 1 a bit shorter as Algorithm 3.

Note that the paths determined in Algorithm 3 will most likely not be the paths the passengers really take after (3) is solved and the line concept is known. This is known and has been investigated in case that the travel time of the passengers is the objective function: Travel time models such as [19] intend to find passengers’ paths and a line concept simultaneously. The same dependency holds if the cost of the line concept is the objective function, but a model determining the line plan and the passengers’ routes under a cost-oriented function simultaneously has to the best of our knowledge not been analyzed in the literature so far.

### 3 Integrating passengers’ assignment into cost-oriented line planning

In this section we formulate a model in which Steps 1 and 2 of Algorithm 3 can be optimized simultaneously. Our first example shows that it might be rather bad for the passengers if we optimize the costs of the line concept and have no restriction on the lengths of the paths in the passengers’ assignment.

**Example 1.** Consider Figure 1a with edge lengths \(d_{AD} = d_{BC} = 1, d_{AB} = d_{DC} = M\), a line pool of two lines \(\mathcal{L}^0 := \{l_1 = ABCD, l_2 = AD\}\) and two OD-pairs \(W_{AD} = \text{Cap} - 1\) and \(W_{BC} = 1\).

For a cost-minimal assignment we choose \(P_{AD} = (ABCD), P_{BC} = (BC)\) and receive an optimal solution \(f_{l_1} = 1, f_{l_2} = 0\) with costs of \(g^{\text{cost}} = \text{cap} + 2M + 1\). The sum of travel times for the passengers in this solution is \(g^{\text{time}} = (\text{Cap} - 1) \times (2M + 1) + 1\).
For the assignment $P_{AD} = (AD)$, $P_{BC} = (BC)$ we receive as optimal solution $f_{l_1} = 1$, $f_{l_2} = 1$ with only slightly higher costs of $g^{\text{cost}} = 2c_{\text{fix}} + 2M + 2$, but much smaller sum of travel times for the passengers $g^{\text{time}} = (\text{Cap} - 1) \times 1 + 1 = \text{Cap}$. 

From this example we learn that we have to look at both objective functions: costs and traveling times for the passengers, in particular when we allow non-shortest paths in Algorithm 2. When integrating the assignment procedure in the line planning model we hence require for every OD-pair that its average path length does not increase by more than $\beta$ percent compared to the length of its shortest path $SP_{uv}$. The integrated problem can be modeled as integer program (LineA) 

$$(\text{LineA}) \quad \min g^{\text{cost}} := \sum_{l \in \mathcal{L}^0} f_l \left( \sum_{e \in E} d_e + c_{\text{fix}} \right)$$

s.t. $\text{Cap} \sum_{l \in \mathcal{L}^0} f_l \geq \sum_{u,v \in V} x_{uv}^e$ for all $e \in E$

$\Theta x_{uv}^e = b_{uv}^e$ for all $u,v \in V$

$\Theta x_{uv}^e \leq \beta SP_{uv} W_{uv}$

$f_l \in \mathbb{N}$ for all $l \in \mathcal{L}^0$

$x_{uv}^e \in \mathbb{N}$ for all $l \in \mathcal{L}^0$

where

- $x_{uv}^e$ is the number of passengers of OD-pair $(u,v)$ traveling along edge $e$
- $\Theta$ is node-arc incidence matrix of PTN, i.e., $\Theta \in \mathbb{R}^{|V| \times |E|}$ and

$$\Theta(v,e) = \begin{cases} 
1 & \text{, if } e = (v,u) \text{ for some } u \in V, \\
-1 & \text{, if } e = (u,v) \text{ for some } u \in V, \\
0 & \text{, otherwise}
\end{cases}$$

- $b_{uv}^e \in \mathbb{R}^{|V|}$ which contains $W_{uv}$ in its $u$th component and $-W_{uv}$ in its $v$th component.

Note that $\beta = 1$ represents the case of shortest paths to be discussed in Section 4. For $\beta$ large enough an optimal solution to (LineA) minimizes the costs of the line concept. Formulations including passengers’ routing have been proven to be difficult to solve (see [19, 2]). Also (LineA) is NP-hard.
Theorem 2. \((\text{LineA})\) is NP-hard, even for \(\beta = 1\) (i.e. if all passengers are routed along shortest paths).

Proof. See [9].

The sequential approach can be considered as heuristic solution to (LineA). Different ways of passengers' assignment in Step 1 of Algorithm 3 are discussed in Section 5.

4 Gap analysis for shortest-path based traffic loads

In this section we analyze the error we make if we restrict ourselves to shortest-path based assignments in the sequential approach (Algorithm 3) and in the integrated model (LineA).

More precisely, we use only one shortest path \(P_{uv}\) for routing OD-pair \((u,v)\) in Algorithm 2 and we set \(\beta = 1\) in (LineA). The traffic loads in Step 2 of Algorithm 2 are then computed as

\[
w_u := \sum_{v \in V : e \in P_{uv}} W_{uv}.
\]

Assigning passengers to shortest paths in the PTN is a passenger-friendly approach since we can expect that traveling on a shorter path in the PTN is less time consuming in the final line network than traveling on a longer path (even if there might be transfers). It also minimizes the vehicle kilometers required for passenger transport. Hence, shortest-path based traffic loads can also be regarded as cost-friendly. Nevertheless, if we do not have a complete line pool or we have fixed costs for lines, it is still important to which shortest path we assign the passengers as the following two examples demonstrate.

Example 3 (Fixed costs zero). Consider the small network with stations A,B,C,D, and E depicted in Figure 1b. Assume that all edge lengths are one. There is one passenger from B to E.

Let us assume a line pool with two lines \(L_0 = \{l_1 = ABCE, l_2 = BDE\}\). Since the lines have different lengths their costs differ: \(\text{cost}_{l_1} = 3\) and \(\text{cost}_{l_2} = 2\) (for \(c_{\text{fix}} = 0\)).

For the passenger from B to E, both possible paths (B-C-E) and (B-D-E) have the same length, hence there exist two solutions for a shortest-path based assignments:

- If the passenger uses the path B-C-E, we have to establish line \(l_1\) (\(f_{l_1} := 1, f_{l_2} := 0\)) and receive costs of 3.
- If the passenger uses B-D-E, we establish line \(l_2\) (\(f_{l_1} := 0, f_{l_2} := 1\)) with costs of 2.

Since in this example \(l_1\) could be arbitrarily long, this may lead to an arbitrarily bad solution.

This example is based on the specific structure of the line pool. But even for the complete pool the path choice of the passengers matters as the next example demonstrates.

Example 4 (Complete Pool). Consider the network depicted in Figure 1b. Assume, that the edges BC, CE, BD and DE have the same length 1 and the edge AB has length \(\epsilon\). We consider a complete pool and two passengers, one from A to E and another one from B to E.

The vehicle capacity should be at least 2. If both passengers travel via C, the cost-optimal line concept is to established the dashed line \(l_1\) with costs \(c_{\text{fix}} + 2 + \epsilon\). For one passenger traveling via C and the other one via D, two lines are needed and we get costs of \(2c_{\text{fix}} + 4 + \epsilon\).

For \(\epsilon \to 0\) the factor between the two solutions hence goes to \(\frac{2c_{\text{fix}} + 4 + \epsilon}{c_{\text{fix}} + 2 + \epsilon} \to 2\) which equals the number of OD pairs in the example.

The next lemma shows that this is, in fact, the worst case that may happen.
Algorithm 4: Passengers’ Assignment: Shortest Paths.

\[\text{Input: } \text{PTN} = (V, E), W_{uv} \text{ for all } u, v \in V\]

\[
\text{for every } u, v \in V \text{ with } W_{uv} > 0 \text{ do } \\
\text{Compute a shortest path } P_{uv} \text{ from } u \text{ to } v \text{ in the PTN, w.r.t edge lengths } d \\
\text{end}\]

\[
\text{for every } e \in E \text{ do } \\
\text{Set } w_e := \sum_{u,v \in V \in P_{uv}} W_{uv} \\
\text{end}\]

Lemma 5. Consider two shortest-path based assignments \(w\) and \(w'\) for a line planning problem with a complete pool \(L^0\) and without fixed costs \(c_{fix} = 0\). Let \(f_l, l \in L\), be the cost optimal line concept for \(\text{LineP}(w)\) and \(f'_l, l \in L'\), be the cost optimal line concept for \(\text{LineP}(w')\). Then \(g^{\text{cost}}(w) \leq |OD|g^{\text{cost}}(w')\).

Proof. See [9].


5 Passengers’ assignment algorithms

We consider three passengers’ assignment algorithms. Each of these is a specification of Step 1 in Algorithm 2. Each algorithm will be introduced in one of the following subsections. They differ in the objective function used in the routing step, i.e., whether we need to iterate our process or not.

5.1 Routing on shortest paths

Algorithm 4 computes one shortest paths for every OD pair, i.e., all passengers of the same OD pair use the same shortest path.

5.2 Reduction algorithm of [13]

Algorithm 5 uses the idea of [13]. It is a cost-oriented iterative approach. The idea is to concentrate passengers on only a selection of all possible edges. To achieve this, edges are made more attractive (short) in the routing step if they are already used by passengers.

The length of an edge in iteration \(i\) is dependent on the load on this edge in iteration \(i - 1\), higher load results in lower costs in the next iteration step. This is iterated until no further changes in the passenger loads occur or a maximal iteration counter \(\text{max}_it\) is reached. When this is achieved, the network is reduced, i.e., every edge that is not used by any passenger is deleted. In the resulting smaller network, the passengers are routed with respect to the original edge lengths.
Algorithm 5: Passengers’ Assignment: Reduction.

Input: PTN = (V,E), W_{uv} for all u, v ∈ V

i := 0
w_i^0 := 0 ∀ e ∈ E
repeat
  for every u, v ∈ V with W_{uv} > 0 do
    Compute a shortest path \( P_{uv}^i \) from u to v in the PTN, w.r.t.
    \[
    \text{cost}_i(e) = d_e + \gamma \cdot \frac{d_e}{\max\{w_i^{e-1}, 1\}}
    \]
  end
  for every \( e ∈ E \) do
    Set \( w_i^e := \sum_{u,v \in V} w_{uv} \in P_{uv}^i \)
  end
  i := i + 1
until \( \sum_{e \in E} (w_i^e - w_i^e)^2 < \epsilon \) or \( i > \text{max_it} \)

Compute a shortest path \( P_{uv} \) from u to v in the PTN, w.r.t.
\[
\text{cost}(e) = \begin{cases} 
  d_e, & w_i^e > 0 \\
  \infty, & \text{otherwise}
\end{cases}
\]
Set \( w_e := \sum_{e \in P_{uv}} W_{uv} \)

5.3 Using a grouping reward

Algorithm 6 uses a reward term if the passengers can be transported without the need of a new vehicle. Again, we want to achieve higher costs for less used edges. We reward edges, that are already used by other passengers. In order to fill up an already existing vehicle instead of adding a new vehicle to the line plan we reward an edge more, if there is less space until the next multiple of Cap. To achieve a good performance, we update the edge weights after the routing of each OD pair and not only after a whole iteration over all passengers.

5.4 Routing in the CGN

For line planning, usually a line pool is given. In particular, if the line pool is small, it has a significant impact on possible routes for the passengers, since some routes require (many) transfers and are hence not likely to be chosen. Moreover, assigning passengers not only to edges but to lines has a better grouping effect. We therefore propose to enhance the three heuristics by routing the passengers not in the PTN but in the co-called Change&Go-Network (CGN), first introduced in [19]. Given a PTN and a line pool \( L^0 \), CGN = (V, \tilde{E}) is a graph in which every node is a pair (v, l) of a station v ∈ V and a line l ∈ L^0 such that v is contained in l. An edge in the CGN can either be a driving edge \( \tilde{e} = ((u,l),(v,l)) \) between two consecutive stations (u, v) ∈ E of the same line l or a transfer edge \( \tilde{e} = ((u,l_1),(u,l_2)) \) between two different lines l_1, l_2 passing through the same station u. In the former case we say that \( \tilde{e} \in \tilde{E} \) corresponds to \( e \in E \). We now show how to adjust the algorithms of the previous section to route the passengers in the CGN in order to obtain a traffic assignment in the PTN. For this we rewrite Algorithm 4 and receive Algorithm 7.

We proceed the same way to rewrite the routing step in the repeat-loop of Algorithm 5,
Algorithm 6: Passengers’ Assignment: Reward.

Input: PTN = (V, E), W_{uv} for all u, v ∈ V

\[ i := 0 \]

repeat
  \[ i = i + 1 \]
  \[ w_i^e := w_i^{e-1} \quad \forall e \in E \]
  for every u, v ∈ V with W_{uv} > 0 do
    Compute a shortest path \( P_{uv} \) from u to v in the PTN, w.r.t.
    \[ \text{cost}_i(e) = \max\{d_e \cdot (1 - \gamma \cdot (w_i^{e-1} \mod \text{Cap})/(\text{Cap}) \}, 0\} \]
    for every \( e \in P_{uv}^{-1} \) do
      Set \( w_i^e := w_i^e - W_{uv} \)
    end
    for every \( e \in P_{uv} \) do
      Set \( w_i^e := w_i^e + W_{uv} \)
    end
  end
until \( \sum_{e \in E}(w_i^{e-1} - w_i^e)^2 < \epsilon \) or \( i > \text{max}_i \);

Algorithm 7: CGN routing for Algorithm 4.

\[ \text{for every u, v ∈ V with W}_{uv} > 0 \text{ do} \]
Compute a shortest path \( \tilde{P}_{uv} \) from u to v in the CGN, w.r.t.
\[ \text{cost}(\tilde{e}) = \begin{cases} 
  d_e & \text{if } \tilde{e} \text{ is a driving edge which corresponds to } e \\
  \text{pen} & \text{if } \tilde{e} \text{ is a transfer edge, where pen is a transfer penalty}
\end{cases} \]
\[ \text{end} \]
\[ \text{for every } e \in E \text{ do} \]
\[ \text{Set } w_e := \sum_{\tilde{e} \text{ corr. to } e} \sum_{u,v \in V} \sum_{\tilde{e} \in \tilde{P}_{uv}} W_{uv} \]
\[ \text{end} \]

where we use
\[ \text{cost}(\bar{e}) = \begin{cases} 
  \text{cost}_i(e) & \text{if } \bar{e} \text{ is a driving edge which corresponds to } e \\
  \text{pen} & \text{if } \bar{e} \text{ is a transfer edge, where pen is a transfer penalty}
\end{cases} \]
as costs in the CGN. We still compare the weights \( w_i^e \) and \( w_i^{e-1} \) in the PTN for ending the repeat loop, also the reduction step, i.e., the routing after the iteration in Algorithm 5 remains untouched. For the detailed version see Algorithm 8 in Appendix A.

Finally, we consider Algorithm 6. Here routing in the CGN is in particular promising since a line-specific load is more suitable to improve the occupancy rates of the vehicles. In the routing version of 6 we construct the CGN already in the very first step in the same way as in Algorithm 7. We then perform the whole algorithm in the CGN, but compute the traffic loads \( w_i^e \) in the PTN at the end of every iteration in order to compare the weights \( w_i^e \) and \( w_i^{e-1} \) in the PTN for deciding if we end or repeat the loop. For the detailed version see Algorithm 9 in Appendix A.
6 Experiments

For the experiments, we applied the models introduced in Section 5 on the data-set from [8], a small but real world inspired instance. It consists of 25 stops, 40 edges and 2546 passengers, grouped in 567 OD pairs. We started with five different line pools of different sizes, ranging from 33 to 275 lines, using [10] and lines based on k-shortest path algorithms. We use a maximum of 15 iterations for every iterating algorithm. For an overview on runtime, see [9].

6.1 Evaluation of costs and perceived travel time of the line plan

We first evaluate a line plan by approximating its cost and its travel times. Both evaluation parameters can only be estimated after the line planning phase since the real costs would require a vehicle- and a crew schedule while the real travel times need a timetable. We use the common approximations:

\[ g_{\text{cost}} = \sum_{l \in L} f_l \cdot cost_l, \text{i.e., the objective function of (LineP(w)) and (LineA)} \]

that we used before, and

\[ g_{\text{time}} = \sum_{u,v \in V} SP_{uv} + \text{pen} \cdot \#\text{transfers}, \text{describing the sum of travel times of all OD-pairs where we assume that the driving times are proportional to the lengths of the paths and we add a penalty for every transfer.} \]

Comparison of the three assignment procedures

We first compare the three assignment procedures. Figure 2a and 2b show the impact of the assignment procedure for a small line pool (33 lines) and for a large line pool (275 lines). For both line pools we computed the traffic assignment for Shortest Paths, Reduction, and Reward, both in the PTN and in the CGN. This gives us six different solutions, for each of them we evaluated their costs \( g_{\text{cost}} \) and their travel times \( g_{\text{time}} \).

Figure 2a shows the typical behaviour for a small line pool: We see that Shortest Path leads to the best results in travel time, i.e., the most passenger friendly solution. Routing in the CGN is better for the passengers than routing in the PTN, the PTN solutions are dominated. Reward, on the other hand, gives the solutions with lowest costs. Also here, the costs are better when we route in the CGN instead of the PTN. Note that the travel time of the Reward solution in the CGN is almost as good as the Shortest Path solution.
Figure 2b shows the behaviour for a larger line pool. Still, the solution with lowest travel time is received by Shortest Path, and it is still better in the CGN than in the PTN but the difference is less significant compared to the small line pool. The lowest cost for larger line pools are received by Reduction. Note that both Reduction solutions have lower cost than the Reward solution. This effect increases with increasing line pool.

Dependence on the size of the line pool

We have already seen that for larger line pools, cost optimal solutions are obtained by Reduction and for smaller line pools by Reward. Figures 3 and 4 now study further the dependence of the line pool.

In all our experiments, the best travel time was achieved by Shortest Paths. In Figure 3 we see that the travel time is lower if we route in the CGN compared to routing in the PTN for all instances we computed. The difference gets smaller with an increasing size of the line pool; for the complete line pool routing in the CGN and in the PTN would coincide.

For Reward and Reduction we see two effects: First we see a decrease in the costs when we have more lines in the line pool. This is to be expected, since the line concept algorithm used profits from a bigger line pool. Furthermore, we see the for Reduction there are cases, where the cost optimal solution can be found with the PTN routing.

Figure 3 Travel time and cost of Shortest Path solutions for increasing line pool size.

(a) Cost of Reduction.  
(b) Cost of Reward.

Figure 4 Cost of Reward and Reduction solutions for increasing line pool size.
Tracking the iterative solutions in Reduction and Reward

Reduction and Reward are iterative algorithms. They require an assignment in each iteration. For each of these assignments we can compute a line concept and evaluate it. Such an evaluation is shown in Figure 5a where we depict the line concepts computed for the passengers’ assignments in each iteration for Reduction. For Reward, see [9]. For Reduction we see that the rerouting in the reduced network in the end is crucial. In most of our experiments the resulting routing dominates all assignments in intermediate steps with respect to costs and travel time of the resulting line concepts. For Reward we observe no convergence. It may even happen that some of the intermediate assignments lead to non-dominated line concepts.

6.2 Using the line plan as basis for timetabling and vehicle scheduling

In this section we exemplarily evaluate the line concept obtained by Reduction with routing in the PTN for a large line pool of 275 lines in more detail. The line plan is depicted in Figure 5b. For its evaluation we used LinTim [1, 11] to compute a periodic timetable and a vehicle schedule. The resulting public transport supply was evaluated by VISUM ([16]). More precisely, we computed

- the cost for operating the schedule given by the number of vehicles, the distances driven and the time needed to operate the lines, and
- the perceived travel time of the passengers (travel time plus a penalty of five minutes for every transfer) when they choose the best possible routes with respect to the line plan and the timetable.

The resulting costs are 1830 which leads to be best completely automatically generated solution obtained so far for this example (for other solutions, see [8]) and shows that the low costs in line planning lead to a low-cost solution when a timetable and vehicle schedule is added. As expected, the travel time for the passengers increased (by 18%).

7 Conclusion and Outlook

We showed the importance of the traffic assignment for the resulting line concepts, regarding the costs as well as the passengers’ travel time. We analyzed the effect of different assignments theoretically as well as examined three assignment algorithms numerically. As further steps
we plan to analyze the impact of the passengers’ assignment together with the generation of the line pool. We also plan to develop algorithms for solving (LineA) exactly with the goal of finding the cost-optimal assignment in the line planning stage, and finally a lower bound on the costs necessary to transport all passengers in the grid graph example. Furthermore, more optimization in the implementation is necessary to solve the discussed models on instances of a more realistic size.

References

Integrating Passengers’ Assignment in Cost-Optimal Line Planning


A Algorithms

Algorithm 8: CGN routing version of Algorithm 5.

Input: PTN $=(V,E), W_{uv}$ for all $u,v \in V$

Construct the CGN $(\tilde{V}, \tilde{E})$ with

$$d_{\tilde{e}} = \begin{cases} d_e, & \text{for drive edges } \tilde{e}, \text{ where } e \text{ is the corr. PTN edge} \\ \text{pen,} & \text{for transfer edges } \tilde{e}, \text{ where pen is a transfer penalty} \end{cases}$$

$i := 0$

$w_i^0 := 0 \forall e \in E$

repeat

$i = i + 1$

for every $u,v \in V$ with $W_{uv} > 0$
do

Compute a shortest path $\tilde{P}_{uv}$ from $u$ to $v$ in the CGN, w.r.t.

$$\text{cost}_i(\tilde{e}) = d_{\tilde{e}} + \gamma \cdot \frac{d_{\tilde{e}}}{\max\{w_i^{e^-1}, 1\}},$$

where $e$ is the PTN edge corresponding to $\tilde{e}$.

end

for every $e \in E$
do

Set $w_i^e := \sum_{u,v \in V} \sum_{\tilde{e} \in \tilde{E}_{uv}} W_{uv}$

end

until $\sum_{e \in E} (w_i^{e^-1} - w_i^e)^2 < \epsilon$ or $i > \max_{\text{it}}$

for every $u,v \in V$ with $W_{uv} > 0$
do

Compute a shortest path $P_{uv}$ from $u$ to $v$ in the PTN, w.r.t.

$$\text{cost}(e) = \begin{cases} d_e, & w_i^e > 0 \\ \infty, & \text{otherwise} \end{cases}$$

end

for every $e \in E$
do

Set $w_e := \sum_{u,v \in V} W_{uv}$

end

**Input:** PTN = (V,E), W_{uv} for all u, v ∈ V

Construct the CGN (\tilde{V}, \tilde{E}) with

\[ d_{\tilde{e}} = \begin{cases} d_e, & \text{for drive edges } \tilde{e}, \text{ where } e \text{ is the corr. PTN edge} \\ \text{pen}, & \text{for transfer edges } \tilde{e}, \text{ where } \text{pen is a transfer penalty} \end{cases} \]

\[ i := 0 \]
\[ w^0_{\tilde{e}} := 0 \forall \tilde{e} \in \tilde{E} \]

repeat

\[ i = i + 1 \]
\[ w^i_{\tilde{e}} := w^{i-1}_{\tilde{e}} \forall \tilde{e} \in \tilde{E} \]

for every u, v ∈ V with W_{uv} > 0 do

Compute a shortest path \( \tilde{P}_{u,v} \) from u to v in the CGN, w.r.t.

\[ \text{cost}_i(\tilde{e}) = \max\{d_{\tilde{e}} \cdot \left(1 - \gamma \cdot \frac{w^{i-1}_{\tilde{e}} \text{ mod Cap}}{\text{Cap}}\right), 0\} \]

for every \( \tilde{e} \in \tilde{P}_{u,v} \) do

Set \( w^i_{\tilde{e}} := w^i_{\tilde{e}} - W_{uv} \)

end

for every \( \tilde{e} \in \tilde{P}_{u,v} \) do

Set \( w^i_{\tilde{e}} := w^i_{\tilde{e}} + W_{uv} \)

end

for every e ∈ E do

Set \( w_e := \sum_{\tilde{e} \in \tilde{E}, \tilde{e} \text{ corr. to } e} \sum_{u,v \in V} W_{uv} \)

end

until \( \sum_{e \in E}(w^i_e - w^0_e)^2 < \epsilon \) or \( i > \text{max_it} \).
D. The Line Planning Routing Game

A. Schiewe, P. Schiewe, M. Schmidt
The Line Planning Routing Game
[Schiewe et al., 2019].
The line planning routing game

Alexander Schiewe, Philine Schiewe, Marie Schmidt

1. Introduction

Due to the high complexity of public transportation planning, the planning process is normally subdivided in subsequent steps, such as network design, line planning, timetabling, vehicle scheduling, etc. The line planning problem aims at determining the routes, called lines, which are served regularly by a vehicle and the frequencies of these services. When evaluating such a set of lines both the emerging costs and the quality from the passengers’ perspective are taken into account. Various variants of line planning have been formulated and solved as optimization problems. We take a new perspective on line planning: we propose to model line planning as a routing game where passengers choose routes based on travel quality and a cost share, which depends on the amount of passengers who share (parts of) the route. In this paper we address the question on how to find equilibria of this so-defined line planning routing game (LPRG) and compare them to line planning solutions found by optimization approaches.

The remainder of this paper is structured as follows. In Section 1.1 we review literature on line planning before we detail our contribution in Section 1.2. We then briefly introduce some concepts from game theory in Section 2. In Section 3 we introduce the line planning problem we study, both in its centralized version (Section 3.1) and as line planning routing game (Section 3.2) and discuss the relations between the two problems (Section 3.3).

In Section 4 we investigate properties of the line planning routing game. We sketch the best-response algorithm used to find equilibria to LPRG, and in Section 4.1 we investigate under which conditions the line planning model a passenger’s best-response can be calculated efficiently. The existence of equilibria and the convergence of the best-response algorithm are investigated in Section 4.2. Section 4.3 evaluates the solutions found by the best-response algorithm with respect to solutions found with a centralized approach. Finally, in Section 5 we illustrate and compare the different models on some small line planning instances.

1.1. Related literature

Line planning is an important step in the public transportation planning process. There are many line planning models which differ with respect to the decisions covered by the term line planning, the level of detail with which real-world constraints are included in the model, and the way of measuring the travel quality of a line plan. In this paper, we give a brief overview on the line planning models and solution methods which are most relevant for this paper. See, e.g., Schöbel (2011) and Schmidt (2014) for more extensive overviews on line planning.

Line planning aims at finding a line concept (that means: line routes and frequencies) which is good from an operational point of view and offers good travel quality for the passengers. Cost-oriented line planning models focus on minimizing the operational costs subject to the constraint that passenger demand has to be satisfied (see, e.g., Borndörfer, Hoppmann, Karststein et al., 2013; Bussieck, 1998; Claessens, van Dijk, & Zwanenberg, 1998; Goossens, van Hoesel, & Kroon, 2006).
Possible ways to measure the quality of a line concept from the point of view of a passenger are the (generalized) travel time and the number of transfers on the route that a passenger would choose.

A few passenger-oriented line planning models aim at minimizing the overall travel time while keeping the costs below a predefined threshold (Schmidt, 2014; Schöbel & Scholl, 2006). There are also passenger-oriented models which measure quality by the number of direct travelers (Bussieck, 1998; Bussieck, Kreuzer, & Zimmermann, 1997; Dienst, 1978). Several models combine quality and cost into one objective (Borndörfer, Grötschel, & Pfetsch, 2008; Guan, Yang, & Wirasinghe, 2006; Pfetsch & Borndörfer, 2006).

Line planning problems are often modeled and solved as integer programs. Solution approaches for cost-oriented models often assign the demand to the network edges in a preprocessing step and formulate covering or packing models. Solution techniques include branch-and-bound (Bussieck, 1998; Claessens et al., 1998), branch-and-cut (Goossens, van Hoesel, & Kroon, 2004), and variable fixing heuristics (Bussieck, Lindner, & Lübbeke, 2004).

Passenger-oriented line planning assumes that passengers choose the “best” route with respect to the chosen line concept (where “best” is often understood as travel-time minimal). For this purpose, passengers’ routes cannot be determined in a preprocessing step but have to be determined together with the line concept. Schöbel and Scholl (2006) model passengers as flows in a change-and-go network, which allows to include transfer times in the travel time, and solve the LP-relaxation using Dantzig-Wolfe decomposition. However, this leads to very large IP models and relatively long solution times. Borndörfer, Grötschel, and Pfetsch (2007); Borndörfer and Karsten (2012); Borndörfer and Neumann (2010) use column generation to generate passengers’ routes. In Borndörfer and Neumann (2010) it is shown that this can lead to a significant speed-up with respect to flow formulations in change-and-go networks. However, in order to achieve problem formulations which can be solved for practical instances, these models use several simplifications. Often, transfer times are assumed to be independent of line frequencies (see, e.g., Borndörfer & Karsten, 2012; Borndörfer & Neumann, 2010; Schmidt, 2014; Schöbel & Scholl, 2006) or not taken into account at all (Borndörfer et al., 2007). Goossens et al. (2004, 2006) use a model that allows to adjust transfer times to frequencies, but make a different restriction: for each passenger, the path in the network on which he travels is fixed beforehand (even if the exact connection, i.e., the sequence of lines used on this path, is not).

A further drawback of the described passenger-oriented models is that they determine a system-optimum with respect to the cumulated objective functions of all passengers. In order to achieve a system-optimal solution, single passengers may be assigned to routes which are significantly worse than their individually optimal route. Goerigk and Schmidt (2017); Schmidt (2014) introduce a model where only line concepts which allow all passengers to travel on shortest paths (with respect to the line concept) are considered feasible and propose an IP formulation as well as a genetic algorithm.

Solution approaches to line planning which are not IP-based, often concentrate on the line routes only and postpone frequency setting to a later step. They use greedy strategies (Ceder & Wilson, 1991; Pape, Reinecke, & Reinecke, 1995; Quak, 2003) to construct lines or successively remove lines from a big line pool (Patz, 1925; Sonntag, 1979). Furthermore, metaheuristics like genetic algorithms (Fan & Machemehl, 2006b; Fusco, Gorl, & Pettei, 2002; Goerigk & Schmidt, 2017; Szeto & Wu, 2011), neighborhood search (Canca, De Los Santos, Laporte, & Mesa, 2017; Szeto & Wu, 2011), and simulated annealing (Fan & Machemehl, 2006a) are used. Jancseková, Blátha, and Trčkman (2010); Mandl (1980), and Schmidt (2014) describe iterative approaches, where line planning/frequency setting and route assignment steps are iterated.

Furthermore, the trend in research goes towards the integration of different planning steps in public transportation, like line planning and rolling stock planning (Canca, De Los Santos, Laporte, & Mesa, 2016), line planning and timetabling (Burggraaf, Bull, Vansteenwegen, & Lusby, 2017) or even all three problems (Patzold, Schiewe, Schiewe, & Schöbel, 2017; Schöbel, 2017).

There are also game-theoretic approaches to line planning which model line operators as players who compete for a good utilization of the lines they offer (Bessaas, Kontogiannis, & Zarolias, 2009; 2011; Neumann, 2014; Schöbel & Schwarz, 2006; Schöbel & Schwarz, 2013; Schwarz, 2009). In Laporte, Mesa, and Perea (2010), the problem of finding a line concept which is robust against link failures is modeled as a game between the network provider and an adversary. However, to the extent of our knowledge, so far no attempt has been made to model line planning as a game with passengers as players.

In the field of transit assignment, models from game theory are used to model passenger flows on networks (see, e.g., Constantin & Florian, 1995; De Coa & Fernández, 1993; Friedrich, Hartl, Schiewe, & Schöbel, 2017b; Nguyen & Pallottino, 1988; Schmickler, Fonzone, Shimamoto, Kurauchi, & Bell, 2011; Sheffi, 1985; Spiess & Florian, 1989; Szeto, Solaymani-Damashi, & Wong, 2011). These models take into account different modeling requirements from practice, like e.g., limited seat capacity or uncertain information about the next arriving vehicles. Equilibria are often found by mathematical programming.

Routing games on networks are also studied from a more theoretical perspective in the area of algorithmic game theory. A good overview of this line of research, both for atomic and non-atomic flow, is given, e.g., in Roughgarden (2007). Questions of interest cover the existence and quality of equilibria and algorithmic approaches to identify equilibria (see, e.g., Anshelevich, Dasgupta, Kleinberg, Tardos, Wexler, & Roughgarden, 2004; Awerbuch, Azar, & Epstein, 2005; Rosenthal, 1973; Roughgarden, 2005; Roughgarden, 2007; Tardos & Wexler, 2007)

1.2. Contribution of this paper

In this paper, we propose a new perspective on line planning problems with cost and travel quality objective, which motivates a novel algorithmic approach to solve line planning problems. Instead of integrating planning and routing steps or iterating between both as done in the approaches described above, we regard only the routing step and include all planning decisions in this step. To this end, we define an individual objective function for each passenger which is composed of travel time, transfer penalties, and a share of the overall cost of the solution. This way, the line planning problem can be interpreted as a game in which the passengers are the players who aim at minimizing their objective functions.

To find equilibria we propose a best-response algorithm. We investigate the algorithmic viability of this approach, that is, under which conditions on the line planning model a passenger’s best-response can be calculated efficiently and which properties are needed to guarantee convergence of the best-response algorithm. For cases where we do not have these properties, we propose heuristics which simplify the routing step.

We compare the solutions found by our algorithm to solutions found by centralized approaches, both theoretically, by investigating the price of anarchy, and experimentally. Furthermore, we show that the solutions found by our approach are more balanced in the sense that passengers with the same origin and destination are assigned to paths with the same general costs.
2. Basics from game theory

In this section we describe some basic concepts from game theory which are used in the remainder of this paper. See, e.g., Nisan, Roughgarden, Tardos, and Vazirani (2007) for a more comprehensive introduction to game theory.

Game theory studies the dynamics of situations where players try to minimize individual, conflicting objective functions. In a game \((Q, \text{Strat}, h)\) each player \(q \in Q\) has a set of strategies \(\text{Strat}_q\) among which he can choose. The individual objective function \(h_q(S) = h_q(S_q, S' - q)\) of player \(q\) depends on his chosen strategy \(S_q\), but also on the strategies \(S' - q = (S_1, S_2, \ldots, S_{q-1}, S_{q+1}, \ldots, S_{|Q|})\) chosen by the other players.

A central concept of game theory is the concept of equilibria. A set of strategies \((S_1, \ldots, S_Q)\) is called (Nash) equilibrium if none of the players can improve his individual objective function by changing his strategy given that all other players do not change their strategies. i.e., \(S = (S_1, \ldots, S_Q)\) is an equilibrium if for all \(q \in Q\) it holds that

\[
h_q(S_q, S' - q) = h_q(S_q, S'' - q) \quad \forall S' \in \text{Strat}_q
\]

Not all games have equilibrium, and even if equilibria exist, they can be hard to find and they do not need to be unique.

A special class of games with good properties is the class of potential games. We call a function \(\Phi: \text{Strat} = \text{Strat}_1 \times \text{Strat}_2 \times \ldots \times \text{Strat}_Q \rightarrow \mathbb{R}\) potential function, if it satisfies the relation

\[
\Phi(S) - \Phi(S') = h_q(S_q, S' - q) - h_q(S_q, S'' - q)
\]

for all solutions \(S = (S_1, \ldots, S_Q)\) \(\in\) Strat, all players \(q \in Q\) and all solutions \(S' = (S_1, \ldots, S_{q-1}, S'_q, S_{q+1}, \ldots, S_{|Q|})\) \(\in\) Strat which can be obtained from \(S\) by exchanging the strategy of player \(q\). A game with potential function is called potential game. The existence of a potential function allows us to interpret the problem of finding an equilibrium in \((Q, \text{Strat}, h)\) as a minimization problem. As we can easily verify in (1), an optimal solution to \(\Phi\) is an equilibrium for the considered game (although there may be equilibria which are not optimal for \(\Phi\)).

Furthermore, the relation (1) implies that every time a player changes his strategy to improve his personal objective (while the other players’ strategies remain unchanged), the solution becomes better with respect to \(\Phi\) and, in this sense, closer to an equilibrium. This motivates the approach of using best-response algorithms to find equilibria: in every step, one of the players changes his strategy to the best response with respect to the other players’ strategies, i.e., he picks a solution of the optimization problem min\(_{S \in \text{Strat}_q} h_q(S_q, S' - q)\) as a new strategy. If there is only a finite number of strategies, this procedure converges to an optimum of \(\Phi\), and hence to an equilibrium of the game in a finite number of steps.

A centralized way to evaluate a solution \(S = (S_1, \ldots, S_Q)\) is to sum up the individual objective functions to a centralized objective function \(H(S) = \sum_{q=1}^Q h_q(S_q, S' - q)\). We call \(S\) \(\in\) Strat system-optimal if it minimizes \(H\).

There exist different concepts to measure the inefficiency of equilibria with respect to the centralized objective. The price of anarchy is defined as

\[
\max_{\text{S is an equilibrium}} \frac{H(S)}{\min_{\text{S is an equilibrium}} H(S)}
\]

Assuming that over time, selfish behavior will converge to equilibrium solutions, the price of anarchy gives a worst-case bound on the quality of such a convergence process.

The price of stability is

\[
\min_{\text{S is an equilibrium}} \frac{H(S)}{\min_{\text{S is an equilibrium}} H(S)}
\]

in contrast, quantifies how far the best equilibrium (i.e., the best solution that would be accepted by the players) is away from system optimality.

3. Line planning with travel quality and cost objective

3.1. The centralized approach

Line planning aims at determining routes and frequencies of vehicles like trains, metros, or buses. As a basis, we consider the underlying public transportation network \((\text{PTN}) G = (V, E)\). The nodes \(V\) of this network represent stations. Two stations are connected by an edge \(e \in E\) if there is a direct track connection between the corresponding stations. In this paper, we consider a line pool \(L\) of possible lines, which are simple paths in the network, as input to the problem. The main task of line planning is to find a line concept, i.e., to assign a frequency \(f_q\) to every line \(l\) in the line pool \(L\). In many line planning models from the literature, constraints on the number of lines which can pass an edge \(e\) are imposed. While this is certainly important in practice, in order to keep our line planning model as simple as possible, we do not consider this constraint in this paper.

We denote the costs of a line, depending on its frequency, as \(cost(f)\). We model \(cost(f)\) as composed of a frequency-independent cost \(k_1\), which represents, e.g., administration costs, and a frequency-based cost \(k_2\), e.g., fuel or labor costs. We obtain \(cost(f) = \gamma_1 k_1^f + \gamma_2 k_2^f\), if \(f > 0\) and 0 otherwise, where \(\gamma_1\) and \(\gamma_2\) are non-negative constants. The cost of a line concept represented by frequencies \(f\) is thus given as \(cost(f) := \sum_{l \in L} c_l^f (\gamma_1 k_1^f + \gamma_2 k_2^f)\).

We consider passenger demand per period given in form of origin-destination (OD)-pairs \((u_q, v_q)\), specifying origin \(u_q\) and destination \(v_q\) of passenger \(q\) from the set of passengers \(Q\). To be able to evaluate the quality of the line plan from the passengers’ perspective, together with the line concept we determine a set of passenger routes \(R := (R_q : q \in Q)\). A route \(R_q\) for passenger \(q\) specifies a path \(P_q = (e_1, \ldots, e_{l_q})\) from \(u_q\) to \(v_q\) and for every edge \(e_1, \ldots, e_{l_q}\) a line \(l_q\) which is used while traveling on \(e_i\), i.e., \(R_q\) can be written as a sequence \(R_q = ((e_1, l_1), (e_2, l_2), \ldots, (e_{l_q}, l_{l_q}))\). For a given set of routes \(R\) we denote the number of passengers who use line \(l\) on edge \(e\) by \(x_{e,l}^R\) such that \(\sum_{e \in E} |x_{e,l}^R| = |Q|\).

We call a pair of frequencies \(f\) and passenger route set \(R\) feasible, if the number of passengers does not exceed the vehicle capacity in any run of any line on any edge, under the assumption that passengers spread evenly over all vehicles runs of one line. That is, if for every line \(l\) and every \(e \in l\) it holds that \(x_{e,l}^R \leq f(l) \cdot B\), where \(B\) denotes the capacity of a single vehicle.

To evaluate a line concept, we use a weighted sum of costs, travel time, and transfers. Here, travel time consists of in-vehicle time and transfer time, that is, we do not take waiting times at the origin station into account. The in-vehicle time on route \(R_q\) depends only on the chosen route in the PTN. It is given as \(cV_q(R_q) := \sum_{e \in E} c_e x_{e,l}^R\), where \(c_e\) is the in-vehicle time for an edge \(e \in G\). The transfer time \(cT_q(R_q, f)\) is estimated based on the frequencies of the lines involved in the transfers on the route. In this paper, for a transfer from line \(l_1\) to line \(l_2\) we assume a transfer time of \(\sum_{l_1,l_2 \in L} cT_{l_1,l_2}\), where \(T\) is the period length (often one hour). This models the expected transfer time under the assumption that passengers choose their route based on a periodic timetable. The overall transfer time of passenger \(q\) on route \(R_q\) is \(cT_q(R_q, f) := \sum_{l_1,l_2 \in L} cT_{l_1,l_2}\), where \((l_1, l_2, \ldots, l_k)\) is the sequence of lines used on \(R_q\). Furthermore, we include the number of transfers \(\sum_{l_1,l_2 \in L} cT_{l_1,l_2}\) into the evaluation of each route. This models the inconvenience arising for the passenger from ea ch transfer.
Definition 3.1. Given a PTN $G$, a line pool $L$, a capacity bound $B$, a set of passengers $Q$, a parameter set $(\alpha_1, \alpha_2, \beta, \gamma_1, \gamma_2)$, and a period length $T$, the line planning with travel quality and cost objective (LPQC) is defined as follows: find a pair of frequencies $f$ and routes $R$ which fulfills $x_{u,j}(R) \leq f_{u} \cdot B$ and minimizes the objective function

$$H(R, f) := \sum_{R_q \in Q} \left( \alpha_1 \cdot c_q(R_q) + \alpha_2 \cdot \tau_q(R_q, f) + \beta \cdot transfer_q(R_q) \right)$$

\[= \text{travel}(R,f) + \gamma_1 \cdot \sum_{l \in R_q} k_l^1 + \gamma_2 \cdot \sum_{l \in R_q} k_l^2 f_l. \tag{2} \]

(LPQC) takes a centralized perspective on line planning: we aim to minimize the sum of costs and total travel time (summed up over all passengers). This does not necessarily mean that the travel time for each individual passenger is short. In fact, particular passengers may be forced to take detours for the 'greater good' of allowing short routes for others. See Section 3.3 for an example.

The following observation from Schmidt (2014) will be useful in the remainder of this paper:

Observation 3.2. Given a route set $R$, we can easily determine a corresponding line concept $f(R) = f_{\text{opt}}(R_{\text{opt}})$ by setting

$$f_{\text{opt}}(R) := \max \left[ \frac{x_{u,j}(R)}{B} \right]$$

Observation 3.2 allows us to omit the line concept as argument in the function $H$, thus in the following we use the notation $H(R) := H(R, f)$ when convenient. The same holds for the functions $\tau_q$, where we write $\tau_q(R)$ or $\tau_q(R, R^-)$ instead of $\tau_q(R, f(R))$.

3.2. The line planning routing game

In this paper, we interpret line planning as a routing game. The passengers $Q$ are the players. The strategies of a passenger $q$ are the routes $R_q$ from his origin $u_q$ to his destination $v_q$. Based on a set of routes chosen by the passengers $R$, we determine the line concept as $f(R)$ as described in Observation 3.2. Each passenger has an individual objective function $h_q(R_q, R^-)$ on which he bases the route choice. It depends on his chosen route $R_q$ and the routes chosen by the other passengers $R^-$. We call this game line planning routing game (LPRG) and interpret equilibria $R^*$ of this game as solutions $(R^*, f(R^*))$ of the line planning problem. The choice of the individual objective functions $h_q$ is of course crucial for the quality of the obtained solutions. We want the individual objective functions to

- account for individual travel quality as well as costs in order to find a solution which is balanced between the two partly contradicting objectives of minimizing costs while maximizing quality, and
- model passengers' behavior as realistically as possible.

We propose the following general model. The passengers' individual objective functions are composed of the travel quality of the solution travel$_q := \alpha_1 \cdot c_q(R_q) + \alpha_2 \cdot \tau_q(R_q) + \beta \cdot transfer_q(R_q)$ and a share of the overall costs, cost$_q(R_q, R^-)$, that is, we have $h_q(R_q, R^-) := \text{travel}_q(R_q, R^-) + \text{cost}_q(R_q, R^-)$.

To share the costs among the passengers, we propose two models:

1. equally divide the cost of all lines among all passengers that are choosing this line as part of their route

   $$\text{cost}_q(R) := \frac{\sum_{l \in R_q} \text{cost}_l(f(R))}{|R_q \in Q : l \in R_q|}$$

   (called line-based cost model in the following), or

2. split the line costs of line $l$ among the edges $e \in l$ as edge costs

   $$\text{cost}_{e,l}(R) := \text{cost}_l(f(R)) / \sum_{e \in l} \text{cost}_e$$

   and compute the cost for passenger $q$ as

   $$\text{cost}_q(R) := \sum_{e \in l} \text{cost}_{e,l}(f(R)) \cdot \gamma(e).$$

In this paper, we assume that the edge costs are proportional to the edge lengths $\gamma(e)$, i.e.,

$$\text{cost}_{e,l}(f(R)) := (\gamma_1 k_l^1 + \gamma_2 k_l^2 f_l) \cdot \gamma(e).$$

In Definition 3.3 we summarize the definition of the LPRG:

Definition 3.3. In the line planning routing game (LPRG), the passengers $q \in Q$ act as players. Every passenger (player) chooses among the routes from his origin $u_q$ to his destination $v_q$ (strategies) to minimize his individual objective function $h_q(R_q, R^-)$ which depends both on the route $R_q$ chosen by $q$ and the routes chosen by the other passengers $R^-$. If

Note that in the definition of the quality functions in Section 3.1 and the individual objective functions in the section, we implicitly assumed that all passengers have the same perception of quality of a travel route since we assume the weighting factors $\alpha_1, \alpha_2, \beta, \gamma_1$, and $\gamma_2$ to be the same for each passenger. It would be possible to replace these common weighting factors by a set of individual weighting factors for each passenger. However, for the sake of simplicity, in this paper we only consider the case of common weighting factors for all passengers.

3.3. Relation between LPQC and LPRG

In this section we discuss the relation between the objective function $H$ of the line planning problem with travel quality and cost objective (LPQC) and the individual objective functions $h_q$ of the line planning routing game (LPRG).

By definition $\sum_{q \in Q} \text{travel}_q(R_q, R^-) = \text{travel}(R, f(R))$. Furthermore, in the line-based cost model, we have $\sum_{q \in Q} \text{cost}_q(R_q, R^-) = \text{cost}(f(R))$. This is also true in the edge-based cost model, as long as it is ensured that a line does not contain an edge which no passenger is using on this particular line, which we will assume in the following. We conclude that

$$\sum_{q \in Q} h_q(R_q, R^-) = H(R, f(R)).$$

That is, a system-optimal route set for LPRG corresponds to an optimal solution of LPQC. Hence, if the price of anarchy in the LPRG is small, an equilibrium $R^*$ of the game provides us with a good approximation $(R^*, f(R^*))$ for LPQC.

Lemma 3.4. Denote by $l$ an instance of the LPQC. Assume that the price of anarchy for the corresponding instance $I_{\text{LPQC}}$ of LPRG is bounded by $\xi$. Then any equilibrium $R^*$ of $I_{\text{LPQC}}$ is an $\xi$-approximation $(R^*, f(R^*))$ for $l$.

So, on the one hand, finding an equilibrium to LPRG may be regarded as a new, decentralized, way of solving LPQC. On the other hand, one may argue that in some cases, optimal solutions to LPQC are not desirable in practice. Indeed, it may happen that the route set $R$ in a solution $(R,f)$ to LPRG allows very long routes to some
passengers for the ‘greater good’ of a solution which is optimal with respect to the centralized objective function $H$.

We discuss an example for the latter in the remainder of this section. Consider the situation shown in Fig. 1. There are seven (railway) stations and two lines (depicted by gray arrows) from station $v_1$ to station $v_2$. One is a fast line which stops only at one intermediate station, the other one is a regional line which serves a geographically different route and visits many small stations in between. Assume that the transportation capacity of each line is $B = 100$. The demand situation is as follows: 100 passengers want to travel from $v_1$ to $v_2$. 50 want to travel from $v_1$ to $v_3$, and some smaller amounts of passengers are traveling to and from the regional stations. Hence, both lines have to be established. Now, if the cost parameters $y_1$ and $y_2$ in the centralized objective function $H$ are comparatively large, both lines will be established with frequency 1 in an optimal solution ($\mathcal{R}, \mathcal{f})$ to LPQC. This means that 50 of the 100 passengers from $v_1$ to $v_2$ will be sent via the regional train route in an optimal solution.

However, if this solution was implemented in real life, at station $v_1$, when the passengers from $v_1$ to $v_2$ have to make a decision which train to board, the fast train is still empty. To implement the solution ($\mathcal{R}, \mathcal{f})$ into practice, somebody would have to convince these 50 passengers to use a slower connection to reserve the seats in the fast train for the passengers from $v_1$ to $v_2$ boarding later. It is not hard to imagine, that the passengers from $v_1$ to $v_3$ would board the train anyway so that the ones starting in $v_2$ could not board or the train would be overcrowded.

This would not happen in the solution ($\mathcal{R}^*, f(\mathcal{R}^*))$ provided by an equilibrium $\mathcal{R}^*$ of the corresponding routing game LPRG. In this solution, all passengers from $v_1$ to $v_2$ would choose the fast train and the planner would be forced to provide enough frequency here to avoid overcrowding - unless taking the slow line would be cheap enough to be a favorable option for the passengers. Hence, if we assume that $c_{R_0}(R_0)$ is an estimate of the real costs that a passenger pays on a route $R_0$, in this example the solution ($\mathcal{R}^*, f(\mathcal{R}^*))$ defined by an equilibrium $\mathcal{R}^*$ of LPRG models passenger behavior in a better way, provides better estimates of actual solution quality and helps to avoid overcrowding and is therefore, from this perspective, preferable to the solution ($\bar{\mathcal{R}}, \bar{\mathcal{f}}$) found by the centralized perspective taken in LPQC.

4. Finding equilibria to LPRG

To find equilibria to the LPRG, we use a best-response algorithm which is outlined below.

In the remainder of this paper we discuss under which assumptions we can find routes for passengers in the routing step of Algorithm 1 in polynomial time (Section 4.1), for which instances of the LPRG Algorithm 1 converges to an equilibrium (Section 4.2), and the quality of the equilibria (Section 4.3). We conclude the section in 4.4 with the description of heuristic modifications of the individual objective functions which guarantee polynomial solvability of the routing step and convergence.

Algorithm 1 Best response algorithm.

Require: $\mathcal{R}$, line pool $\mathcal{L}$, origin $v_o$, destination $v_d$ and individual objective functions $h_u$, maximal number of iterations $m \in \mathbb{N} \cup \{\infty\}$

Ensure: A route set $\mathcal{R}$

Start with an empty route set (or with an arbitrary non-empty route set).

while improvements for the passengers possible and $m$ not reached do

for Passenger $q \in \mathcal{Q}$ do

Calculate optimal passenger route $R_q$ according to $h_q$.

end for

end while

4.1. The routing problem

In every step of Algorithm 1 we have to solve the following routing problem for passenger $q$:

Definition 4.1. Given $\mathcal{R}$, $\mathcal{L}$, origin $v_o$, destination $v_d$, and individual objective functions $h_u$ for passenger $q$, the problem is defined by parameter set $(a_1/a_2, \beta, y_1/y_2)$ and period length $T_1$ and routes $R_q$ for all passengers $q' \in \mathcal{Q} \setminus \{q\}$, the routing problem for passenger $q$ is to find a route $R_q$ from $v_o$ to $v_d$ such that $h_q(R_q, R^{-\{q\}})$ is minimized.

Unfortunately, the routing problem which has to be solved in each iteration of Algorithm 1 is NP-hard in general. We see in Section 4.1.1 that there are two components which make the problem hard: (1) line-based costs (Theorem 4.2), and (2) frequency-based transfer times (Theorem 4.3). However, if costs are assumed to be edge-based with $\gamma_2 = 0$ and transfer times are neglected, the problem becomes much better tractable, as we are going to discuss in Section 4.2.2. Heuristics to incorporate frequency-based transfer times are discussed in Section 4.4.

4.1.1. NP-hardness of the routing problem

For determining the complexity of our problems we use reductions from the set cover problem (SCP). An instance of SCP is given by a set of elements $\mathcal{M} = \{m_1, \ldots, m_\ell\}$, a set of sets $\mathcal{C}$ with $C \subseteq \mathcal{M}$ for every $C \in \mathcal{C}$ and an integer $K \in \mathbb{N}$. The problem is to determine whether there exists a subset $C' \subseteq C$ such that $\cup_{C \in C'} C = \mathcal{M}$ and $|C'| \leq K$.

We first show that the assumption of line-based costs leads to an NP-hard routing problem.

Theorem 4.2. The routing problem (as in Definition 4.1) with line-based costs is NP-hard, even if there is only one passenger and neither transfer times nor transfer penalties nor frequency-based costs are taken into account, i.e. if $\alpha_2 = \beta = \gamma_2 = 0$.

Proof. We show that SCP given by $(\mathcal{M}, C, K)$ can be reduced to the decision version of the routing problem with line-based costs. Given an instance $(\mathcal{M}, C, K)$ of SCP we construct an instance of the decision version of the routing problem as follows.

We create a station $v_0$ and for each $m_i \in \mathcal{M}$, $i = 1, \ldots, n$ a station $v_i$ and an edge $e_i = (v_{i-1}, v_i)$. For all $C \in \mathcal{C}$ we create a line $L_0 \subseteq \mathcal{C}$ containing all edges $e_i$ such that $m_i \in C$ and additional edges to ensure that the lines are connected paths in the PTN. We set edge lengths to $c(e_i) := 0$ for all edges related to $m_i \in \mathcal{M}$ and to $c(e_i) := K + 1$ for all additional edges. We consider a passenger $q$ who wants to travel from $v_0$ to $v_n$. Furthermore we assume line costs of $c_0 = 1$ for all lines $L$. The parameters of the objective function are $\alpha_1 = \gamma_1 = 1$ and $\alpha_2 = \beta = \gamma_2 = 0$. $C$ can be set to an arbitrary value since $\alpha_0 = 0$. An example for the construction is given below.
Now there is a solution to the routing problem with objective value less or equal to K if and only if there is a solution to SCP with objective value less or equal to K.

Let $C'$ be a solution to SCP. Then the set of lines $L' := \{ (e, l) : e \in C' \}$ has costs less or equal to K and allows q to travel from origin to destination with zero travel time. On the other hand, in every solution to the constructed instance of the routing problem with travel time less or equal to K, $q$ uses the edge sequence $e_1, \ldots, e_4$, because otherwise his travel time would be greater than K. Hence, $C' = \{ C : q \text{ uses } e_1 \}$ is a solution to SCP.

The following example illustrates the construction of an instance of the routing problem from an instance of SCP. Consider the instance of SCP given by $M = \{ 1, 2, 3, 4 \}$, $C = \{ v_1, v_2 \}$, $C_0 = \{ 1, 2 \}$, and $K = 2$. The number of passengers is $P = 2$. The PTN shown in Fig. 2 where $e_1, \ldots, e_4$ correspond to $M$ and have length $c(e_i) = 0$ for $i = 1, \ldots, 4$ and $e_5$ is an auxiliary edge for $C_2$ with $c(e_5) = K + 1 = 3$.

The line pool is $L = \{ (l_1 = (e_1, e_2), l_2 = (e_1, e_3, e_4), l_3 = (e_4, e_3) \}$. It is easy to see that any path from $v_0$ to $v_2$ with zero travel time must contain all edges $e_i, i = 1, \ldots, 4$, and hence for each of these edges a line needs to be included. Note that analogously, we can show that the routing problem is NP-hard even for one passenger for frequency-independent costs $\gamma_1 = 0$ (and $\gamma_2 = 0$), by interchanging the roles of frequency-based cost and frequency-independent costs in the construction made in the proof of Theorem 4.2.

Due to the result of Theorem 4.2, in the remainder of this paper we restrict ourselves to edge-based cost functions. However, even without considering costs, the routing problem with frequency-based transfer times is NP-hard.

**Theorem 4.3.** The routing problem as in Definition 4.1 is NP-hard, even if transfer penalties and operational costs are not taken into account, i.e., $\beta = 0$ and $\gamma_1 = \gamma_2 = 0$.

See the appendix for a proof of this result.

**4.1.2. Cases with polynomially solvable routing problem**

A convenient way to represent route choice in line planning problems is the change-and-go network (CGN) $\mathcal{G} = (V, A)$, which was first introduced in Schöbel and Scholl (2006). The set of nodes of the CGN consists of station nodes $V_{stat} := \{ (v, board) : v \in V \} \cup \{ (v, alight) : v \in V \}$ and travel nodes $V_{trans} := \{ (l, v) : l \in L, v \in V \}$. The set of arcs is $A := A_{line} \cup A_{trans} \cup A_{board}$ with

- line arcs $A_{line} := \{ (e, l) : l \in L, e \in l \}$ for each edge $e$ covered by a line $l$,
- transfer arcs $A_{trans} := \{ ((v, l_1), (v, l_2)) : v \in V, l_1 \neq v, l_2 \neq v \}$, and
- arcs for boarding and alighting $A_{board} := \{ ((v, board), (v, l)) : l \in L, v \in V \}$

or $\{ ((v, l_1), (v, l_2)) : l \in L, v \in V \}$.

For an example of a CGN, see Fig. 3.

Now every route $R_q$ for a passenger $q$ can be uniquely represented in $\mathcal{G}$ as a path $P_q$ from $(v_0, board)$ to $(v_q, alight)$ in $\mathcal{G}$.

For $a \in A$ we denote by $x_a(R_q)$ the number of passengers, using arc $a$ of the CGN, i.e., $x_a(R_q) := |\{ q \in Q : P_q = a \}|$ where $P_q$ is the path in the CGN corresponding to $R_q$. To abbreviate, we sometimes omit the route set and use the notation $x_a := x_a(R_q)$.

Let now assume that, given $R_q - \gamma_l$, we can express the objective value of a route $R_q$ as the sum of edge weights over all edges contained in the corresponding path $P_q$, i.e., that there are arc weights $w_a(R_q - \gamma_l) \geq 0 \forall a \in A$ such that

$$h_q(R_q, R_q - \gamma_l) = \sum_{a \in A} w_a(R_q - \gamma_l).$$

This is the case if costs are edge-based with $\gamma_2 = 0$ and $\gamma_1 = 0$. Indeed, since in this case the edge cost function $\alpha_{\text{edge}} := \text{cost}_{\text{edge}}(R_q) := \alpha_{\text{edge}}(R_q) = \gamma_1 \alpha_{\text{edge}}$ is independent of the routing of the current passenger, it is easy to check that the weights $w_a(R_q - \gamma_l) := \alpha_{\text{edge}}(R_q) + \beta \alpha_{\text{transfer}}(R_q)$ if $a \in A_{\text{line}}$

satisfy (3). In Section 4.4, different approaches to define arc weights are studied.

If edge weights of the form (3) can be found, we obtain the following lemma:

**Lemma 4.4.** Consider an instance 1 of the routing problem (Definition 4.1). If there are arc weights $w_a(R_q - \gamma_l)$ as defined in (3), $(R_q)$ can be solved in polynomial time.

**Proof.** In this case, any shortest path from $(u_0, \text{board})$ to $(v_q, \text{alight})$ with respect to the edge weights $w_a(R_q - \gamma_l)$ is an optimal solution to $l$. Hence, we can find a solution using, e.g., Dijkstra's algorithm.

Hence, in this case, we can use Algorithm 1 with, e.g., Dijkstra's algorithm in the routing step to search for an equilibrium of the LPRG.

4.2. Existence of equilibria and convergence of the best-response algorithm

In this section we study under which assumptions equilibria to the LPRG exist and can be found by Algorithm 1. We start with an example which shows that in the general case the existence of an equilibrium is not guaranteed.

4.2.1. Non-existence of equilibria

In this section, we give an intuition for why some instances of LPRG do not have equilibria. A more detailed description of the example and proof of non-existence of equilibria for this example can be found in the appendix.

We regard the PTN from Fig. 4 and assume that every edge is served by one directed line (which contains only this edge). Because of this one-to-one correspondence of lines and edges, in this example we use 'edges' as a synonym for 'lines'. We set the vehicle capacity to $B = 1$, so that the frequency of an edge is given by the number of passengers on it. We consider three main passengers $q_3$ from $u_1$ to $v_1$, $q_2$ from $u_2$ to $v_2$, and $q_1$ from $u_3$ to $v_3$. For each of these passengers, there exist two routes from origin to destination, we denote the route starting with edge $(u_i, v_i)$ as $R_i$ and the route starting with edge $(u_i, v_i)$ as $R_i$. Note that each of these routes consists of a sequence of dotted edge, two thick edges, and a dashed edge.

For the sake of simplicity, in our objective function we take only the transfer time into account, i.e.,

$$\langle \alpha_{\text{edge}}, \beta, \gamma_1 / \gamma_2 \rangle = (0, 1, 0, 0/0).$$

We assume that the line frequency on the dashed edges in the PTN is already very high (which we ensure by adding auxiliary OD-pairs which have to use these edges). The dotted edges, which originate in the nodes $u_l$, will have a frequency of 1 if the passenger $q_l$ travels on them, or 0 otherwise. Consequently, the transfer time of a passenger only depends on whether he shares the thick edges with other passengers or not. Furthermore, transfer time towards the
dashed edges is small anyway, due to their high frequency. Hence, the first two transfers on a passengers’ route make up for most part of the objective function.

Now we show that in this example there is no equilibrium in which passenger \( q_1 \) travels on route \( R_1^0 \) by contradiction. Assume that \( R \) is an equilibrium of the described line planning routing game where \( q_1 \) travels on \( R_1^0 \). We can conclude that \( q_2 \) travels on route \( R_1^2 \) because no matter which route \( q_1 \) chooses, the transfer time on \( R_1 \) will be lower than on \( R_2^1 \) (see the appendix for details). Given the routes \( R_1^1 \) and \( R_1^3 \) for \( q_1 \) and \( q_2 \), it is easy to see that for \( q_3 \) the transfer times are lowest on \( R_1^1 \).

However, if \( q_2 \) travels on \( R_1^1 \) and \( q_3 \) travels on \( R_1^3 \), for \( q_3 \) transfer times would be lower on \( R_1^3 \), which contradicts the assumption that \( R \) is an equilibrium.

Analogously, we can show that there is no equilibrium in which \( q_1 \) travels on \( R_1^3 \). Hence, there is no equilibrium in this example.

4.2.2. Line planning routing games with potential functions

In contrast to the example in Section 4.2.1 we show in Lemma 4.5 that existence of equilibria and convergence can be guaranteed if for every \( a \in A \) there is an arc weight function \( \bar{w}_a : N \times N \rightarrow \mathbb{R} \) such that

\[
\bar{w}_a(R_k, R^{−1}) = \sum_{x_k \in R_k} \bar{w}_a(x_k)
\]

for every route \( R_k \) from \( u_0 \) to \( u_k \) and its corresponding path \( P_k \) in the CGN.

In case of edge-based costs with \( \alpha_2 = 0 \) (in this case, again, we can write \( \text{cost}_{u_0, u_1} \) instead of \( \text{cost}_{u_0, u_1}(f([R])) \)) and \( \alpha_2 = 0 \), such arc weight functions are given by

\[
\bar{w}_a(x) := \begin{cases} 
\alpha_1 c(e) + \frac{\text{on}_a(x)}{\beta} & \text{if } a = (e, l) \in A_{\text{trans}} \\
0 & \text{if } a = (e, l) \in A_{\text{line}}.
\end{cases}
\]

Lemma 4.5. Let \( I := (G, \mathcal{L}, \{h_q : q \in \mathcal{Q}\}) \) be an instance of the LPRG such that arc weight functions as specified in (4) exist. Then

1. \( \Phi(R) := \sum_{x \in R} \sum_{a \in \mathcal{A}} \bar{w}_a(x) \) is a potential function for \( I \),
2. there exists an equilibrium to \( I \),
3. Algorithm 1 converges to an equilibrium in a finite number of steps,
4. each of the steps can be executed in polynomial time.

The proof follows standard arguments for convergence of atomic routing games, and can be found in the appendix.

We conclude that in particular for all line planning routing games with \( \gamma_2 = 0 \) and \( \alpha_2 = 0 \) and edge-based costs, Algorithm 1 finds an equilibrium after a finite number of steps.

4.3. Quality of equilibria

4.3.1. Two examples for ‘bad’ equilibria

We start with an example which illustrates that the LPRG can have different equilibria and that Algorithm 1 does not necessarily find a good one, even when convergence to some equilibrium is guaranteed because the conditions of Lemma 4.5 are fulfilled.

We consider a PTN consisting of four nodes \( v_1, v_2, v_3, v_4 \), edges \( \{v_1, v_2\} \) and \( \{v_2, v_3\} \), \( \{v_2, v_4\} \), and \( \{v_3, v_4\} \) with length 0. Our line pool consists of five lines, the corresponding CGN is shown in Fig. 4. Note that for the sake of a more compact representation, we contracted boarding and alighting node for each station \( v_i \) to a node \( (v_i, 0) \).

We consider two passengers: \( q_1 \) wants to travel from \( v_1 \) to \( v_2 \) and \( q_2 \) wants to travel from \( v_1 \) to \( v_4 \). The parameters of the individual objective functions are \( \alpha_1 = \gamma_1 = 1 \) and \( \alpha_2 = \beta = 2 = \gamma_2 = 0 \). That is, we only take in-vehicle time and frequency-independent costs into account.

Line \( L_1 \) has costs 100, while all other line costs are 0.

For the reader’s convenience, we specify the arc-weight functions as a sum of in-vehicle travel time and costs for the line arcs next to the corresponding arcs in Fig. 4. All other arc weight functions are 0 in this example. There are two equilibria:

1. \( R^1 \): \( q_1 \) uses line 1 and \( q_2 \) uses line 2. For both passengers, the individual objective values are \( h_{q_1} = 99 \).
2. \( R^2 \): \( q_1 \) uses line 2 and \( q_2 \) uses line 3. For both passengers, the individual objective values are \( h_{q_2} = 50 \).

Clearly, the second equilibrium is preferable to the first one, since for both passengers the individual objective functions are almost twice as high in the first one. However, e.g., when starting with an empty solution, Algorithm 1 will find the first equilibrium.

It can be easily seen that in this example, the second and ‘better’ equilibrium is also a system-optimium, that is, it optimizes \( H = h_{q_1} + h_{q_2} \), the objective function of LPQC. Hence, in this example the price of anarchy is \( \frac{2}{1} \) but the price of stability is 1.
However, system-optima to LPRC (that is: optimal solutions to LPQC) are not necessarily equilibria. To illustrate this, we use a slightly modified version of the previous example:

We consider a PTN consisting of four nodes $v_1$, $v_2$, $v_3$, $v_4$, edges $[v_1, v_2]$ with length 32, $[v_1, v_3]$ with length 49, and edges $[v_2, v_1]$, $[v_2, v_3]$, and $[v_1, v_4]$ with length 0.

Again, our line pool consists of five lines, of which line 12 has frequency-independent costs 100 and the other lines have costs 0. The corresponding CGN is shown in Fig. 5, where, again, we design a graph and a node for each station to a line edge (0, 0). This time we consider three passengers: $q_1$ wants to travel from $v_1$ to $v_2$, $q_2$ wants to travel from $v_1$ to $v_3$, $q_3$ wants to travel from $v_2$ to $v_3$. As in the previous example, for the (individual) objective function(s) we use the parameters $a_1 = y = 1$, $a_2 = a_3 = y = 0$. Again, the conditions of Lemma 4.5 are met and we specify the arc-weight functions for the line arcs next to the corresponding arcs in Fig. 4, all other weight functions are 0.

In this case, there is only one equilibrium $R^*$: $q_1$ uses line 1, $q_2$ uses line 3, $q_3$ uses line 2; with $H(R^*) = 32 + 49 + 100 = 181$. The system-optimal solution (and optimal solution to LPQC) is defined by the route set $R$: $q_1$ uses line 2 and 4, $q_2$ uses line 2 and 5, $q_3$ uses line 2, with overall objective value $H(R) = 49 + 100 + 100 = 249$. So for this example, both price of anarchy and price of stability equal 1.33.

By extending the example given in Fig. 5 in a straight-forward way, we see that for instances with an unbounded number of passengers, the price of stability is not bounded for the considered case for $n$ passengers we can construct an instance with price of stability (and price of anarchy) close to $H_0 = \sum_{a=1}^{n} \frac{1}{i^2}$.

4.3.2. Bounding the price of anarchy

However, we can bound the price of anarchy by the number of passengers if the arc weight functions (4) fulfill the property described in Lemma 4.6.

**Lemma 4.6.** If there exist non-increasing arc weight functions $\psi_a$ with $\psi_a(1) \leq x \cdot \psi_a(x)$ for all $x \in \mathbb{N}$, the price of anarchy in the LPRC is at most the number of passengers.

**Proof.** Let the route set $R := \{R_1, \ldots, R_\ell\}$ represent an equilibrium. Assume that $H(R) > |Q| H(X)$. Then there is at least one passenger $q$ with $h_q(R) > |Q| h_q(X)$. For this passenger $q$ it follows that

$$h_q(R, R^*) = \sum_{q \in Q} \hat{h}_q(x_q) \leq \sum_{q \in Q} \hat{h}_q(1) = \sum_{q \in Q} \psi_q(x_q) \leq \sum_{q \in Q} |Q| \hat{h}_q(\bar{x}_q) < h_q(R, R^*).$$

where $\bar{x}_q := x_q(R, R^*)$ denotes the number of passengers on arc $q$ when passengers follow route $x_q(R, R^*)$ and $x_q := x_q(R)$ the number of passengers on arc $q$ when passengers follow routing $R$. This is a contradiction to the assumption that $R$ is an equilibrium. □

**Corollary 4.7.** If edge-based cost functions with $\gamma_2 = 0$ are considered and $\alpha_2 = 0$, the price of anarchy is bounded by the number of passengers.

**Proof.** The functions given in (5) are non-increasing. Furthermore, for $x \geq 1$, we have for $a \in A_{line}$ $x \cdot \psi_a(x) = x \cdot \psi_a(x)$ and for $a \in A_{route}$ $x \cdot \psi_a(x) = x \cdot \psi_a(x)$. □

To see that there are indeed instances with a price of anarchy that exceeds $|Q|$ consider the example given in Fig. 4. If we set the travel time on $(v_1, v_2)$ and $(v_1, v_3)$ to 100, $R'$ and $R^*$ are still both equilibria and the price of anarchy is 2. We can easily extend this construction to an arbitrary number of passengers.

4.3.3. Algorithm 1 as a heuristic for LPQC

**Corollary 4.8.** If there exist non-increasing arc weight functions $\psi_a$ with $\psi_a(1) \leq x \cdot \psi_a(x)$ for all $x \in \mathbb{N}$, given an empty state of the game, calculating the best response equilibria can be achieved in polynomial time.

To see that Algorithm 1 is a system-optimal solution.

**Proof.** Let $Q = \{1, \ldots, n\}$ be the set of all passengers and $S^q$ for $q = 1, \ldots, n$ the route combination after choosing the best response $R^*$. If passenger $q$, i.e., $S^q = (R_1, R_2, \ldots, R_\ell, \emptyset, \ldots, \emptyset)$ we have

$$h_q(S^q) = \sum_{F \in \mathcal{F}} \hat{h}_q(x_q(S^q)) = \sum_{F \in \mathcal{F}} \hat{h}_q(x_q(S^q) + 1) \leq \sum_{F \in \mathcal{F}} \hat{h}_q(x_q(S^q) + 1)$$

where $P_q$ denotes the path in the CGN corresponding to $R^*$. With this, the following holds:

$$H(S^q) = \sum_{F \in \mathcal{F}} h_q(S^q) \leq \sum_{F \in \mathcal{F}} h_q(S^q) \leq \sum_{F \in \mathcal{F}} \hat{h}_q(x_q(S^q) + 1) \quad \text{due to (6)}$$

129
\[ \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \tilde{u}_i(1) \text{ since } \tilde{u}_i \text{ non-increasing} \]
\[ \leq \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} x_i(j) \tilde{u}_i(x_i(j)) \text{ since } \tilde{u}_i(1) \leq x_i' \tilde{u}_i(x_i) \]
\[ \leq |Q| \sum_{i=0}^{\infty} \tilde{u}_i(x_i(1)) \]
\[ = |Q| \sum_{i=0}^{\infty} h_i(x'_i(\alpha)) = |Q| \cdot H(\alpha). \]

That means that for instances of the LQPC/ILPRF for which there exist non-increasing arc weight functions \( \tilde{u}_i \) with \( \tilde{u}_i(1) \leq x \cdot \tilde{u}_i(x) \) for all \( x \in \mathbb{N} \), that is, in particular if \( \alpha_2 = \gamma_2 = 0 \), a solution \((R, f)\) to the line planning problem with approximation ratio \( |Q| \) can be found in polynomial time.

As described in the previous section, we can show that this bound is tight, i.e., there are instances where \text{Algorithm 1} can get stuck in an equilibrium whose objective value is \( |Q| \)-times the optimal solution value.

### 4.4. Heuristic approaches to the routing problem

In the preceding Sections 4.1–4.3 we have seen that in order to achieve polynomial running time of \text{Algorithm 1}, to be able to prove convergence to an equilibrium, and give bounds on the quality of an equilibrium, strong restrictions on the parameters of the objective function have to be imposed.

In this section we investigate heuristic approaches to the routing problem with general individual objective functions \( h_i(q, R) \rightarrow \tau(q, R) - \text{travel}(q, R, \alpha) \) using edge-based costs \( \text{cost}_q(R) = (\gamma_i^1 + \gamma_i^2) \sum_{(e,i) \in R} \sum_{j=0}^{\infty} (\tilde{u}_j(1))^j \).

In this general case, the routing problem is NP-hard (Theorem 4.3) and \text{Algorithm 1} does not necessarily converge (see Section 4.2.1). To overcome these difficulties in a heuristic way, we simplify the transfer time function \( \tau_q \) and the edge-based cost function \( \text{cost}_q \) in this section.

#### 4.4.1. Auxiliary frequencies

In our first approach, we replace the frequencies \( f(R) \) by auxiliary frequencies \( f(R) \) when determining a route for passenger \( q \). This small trick allows us to define arc weights in accordance to \text{Lemma 4.4} and hence, to solve the routing problem using Dijkstra’s algorithm in the CGN.

Let \( Q \) be a set of passengers and let \( R = \{ R_q \} : q \in Q \) be a set of strategies represented by paths in the CGN. We call an edge \((e,i) \in E\) \text{critical for } \alpha if one additional passenger on the edge would increase the frequency, i.e., \( x_{i,e}(R) = 0 \mod 2 \). A line \( i \in E \) is critical for \( \alpha \) if it contains an edge which is critical for \( \alpha \). In order to find a route, given the routes for all other passengers \( R^- \), we define the auxiliary frequencies
\[ \hat{f}_i(R^-) = \begin{cases} f_i(R^-) + 1 & \text{if } i \text{ is critical for } R^- \\ f_i(R^-) & \text{otherwise} \end{cases} \]

We observe that for every line \( i \) and every passenger \( q \in Q \), \( f_j(R^-) \geq f_j(R) \geq f_j(R^-) \). For all non-critical lines we even have equality. Plugging in the auxiliary frequencies into \( \tau_q \) we obtain an auxiliary transfer time function
\[ \hat{\tau}_q(R) := \sum_{i=0}^{\infty} \frac{T}{f_i(R^-) + f_i(R)} + \text{transfe}_q(R) \]

(where \( l_1, \ldots, l_n \) are the lines used in \( R_q \)) which underestimates the transfer times \( \tau(R) \) in a route set \( R \). To find an overestimating heuristic measure for transfer times, we consider
\[ \hat{\tau}_q(R) := \sum_{i=0}^{\infty} \frac{T}{f_i(R^-) + f_i(R)} + \text{transfe}_q(R) \]

Using the same approach, we can define overestimating auxiliary edge-based cost functions as
\[ \hat{\text{cost}}_{q}(R) := \sum_{(e,i) \in R} \frac{\text{cost}_q(f(R^-))}{x_{e,i}(R, R^-)} \geq \text{cost}_q(R) \]

and underestimating auxiliary edge-based cost functions
\[ \check{\text{cost}}_{q}(R) := \sum_{(e,i) \in R} \frac{\text{cost}_q(f(R^-))}{x_{e,i}(R, R^-)} \leq \text{cost}_q(R) \]

We define over- and underestimated versions of the individual objective functions
\[ \hat{h}_{q}^{a}(R_q, R^-) := \alpha_1 \cdot c(R) + \alpha_2 \cdot \hat{\tau}_q(R) + \beta \cdot \text{transfe}_q(R) \]
\[ \hat{h}_{q}^{b}(R_q, R^-) := \alpha_1 \cdot c(R) + \alpha_2 \cdot \hat{\tau}_q(R) + \beta \cdot \text{transfe}_q(R) \]

and obtain
\[ \hat{h}_{q}^{a}(R_q, R^-) \leq \hat{h}_{q}^{b}(R_q, R^-) \leq \hat{h}_{q}^{b}(R_q, R^-) \]

Given a passenger \( q \) and a set of strategies \( R^- \) for the remaining passengers, the auxiliary frequencies allow us to define weights for the arcs in the CGN which depend only on the strategy choices of the remaining passengers \( R^- \). This observation is summarized in the following lemma.

#### Lemma 4.9. For arc weights
\[ \check{\text{cost}}_{q}(R) := \sum_{(e,i) \in R} \frac{\text{cost}_q(f(R^-))}{x_{e,i}(R, R^-)} \]

or \[ \hat{\text{cost}}_{q}(R) := \sum_{(e,i) \in R} \frac{\text{cost}_q(f(R^-))}{x_{e,i}(R, R^-)} \]

we have
\[ \hat{\text{cost}}_{q}(R_q, R^-) = \sum_{a \in R_q} \hat{\text{cost}}_{q}(R^-) \text{ and } \check{\text{cost}}_{q}(R_q, R^-) = \sum_{a \in R_q} \check{\text{cost}}_{q}(R^-) \]

where \( P_R \) denotes the path in the CGN corresponding to \( R_q \) and the routing problem can be solved in polynomial time.

Here, the last statement follows from \text{Lemma 4.4}.

Note that the use of the auxiliary objective functions \( \hat{h}_q \) does not guarantee the existence of an equilibrium: In fact, in the counterexample shown in Section 4.2.1 we have \( f_i(R) > f_i(R^-) \) for all choices of \( q \) and \( R \). Hence, this example also proves the possibility that no equilibrium for objective functions \( \hat{h}_q \) exists.

#### 4.4.2. Auxiliary arc weights

Since the heuristic from Section 4.4.1 does not always lead to an equilibrium, we consider a further heuristic simplification which guarantees the existence of an equilibrium and the convergence of the best-response-algorithm.

Consider a set of passenger routes \( R \) and a transfer edge \( a = ((e,l),(e,l')) \). Then the frequency of \( l \) and \( l' \), respectively, is at least \( \left\lceil \frac{x_i(R)}{B} \right\rceil \), since at least all passengers transferring from \( l \) to \( l' \) have to use \( l \), respectively. Additionally, all frequencies are at most \( \frac{T}{B} \) since no more than all passengers can use
any given line. This leads to the following approximate arc weight functions:

\[ \tilde{w}_a(x) := \begin{cases} \frac{\alpha_1c(e) + y\bar{x}_a\|x\|}{2\|x\|} + \beta & \text{if } a = (e, l) \in A_{\text{line}} \\ \frac{\alpha_1c(e) + y\bar{x}_a\|x\|}{2\|x\|} + \beta + \sum_{\text{clos}(e)} \frac{c(e)}{\sum_{\text{out}(e)}} & \text{if } a \in A_{\text{trans}} \end{cases} \] (7)

and

\[ \tilde{w}_a^h(x) := \frac{\alpha_1c(e) + y\bar{x}_a\|x\|}{2\|x\|} + \beta + \sum_{\text{clos}(e)} \frac{c(e)}{\sum_{\text{out}(e)}} \] (8)

where \( \tilde{w}_a \) is defined as in (5).

With \( \tilde{h}_a^h(R_q, \mathcal{R}^{-}) := \sum_{a \in A_{R_q}} \tilde{w}_a^h(x) \) and

\( \tilde{h}_q^h(R_q, \mathcal{R}^{-}) := \sum_{a \in A_q} \tilde{w}_a^h(x) \)

(where \( P_q \) is the path in the CGN corresponding to \( R_q \)), we obtain:

**Lemma 4.10.** For every passenger \( p \in Q \) with route \( R_q \) and \( \mathcal{R}^{-} = (R_q, \mathcal{R}^{-}) \) we have

\[ \tilde{h}_p^h(R_q, \mathcal{R}^{-}) = \sum_{a \in A_{R_q}} \tilde{w}_a^h(x) \leq \tilde{h}_q^h(R_q, \mathcal{R}^{-}) = \sum_{a \in A_q} \tilde{w}_a^h(x) = \tilde{h}_p^h(R_q, \mathcal{R}^{-}). \]

From Lemmas 4.4, Lemma 4.5, Lemma 4.6, and Lemma 4.8 we conclude:

**Corollary 4.11.** For individual objective functions \( \tilde{h}_p^h \) and \( \tilde{h}_q^h \), the routing step of Algorithm 1 can be executed in polynomial time using arc weights \( \tilde{w}_a^h(x) \) or \( \tilde{w}_a^h(x) \), respectively, in the CGN.

With respect to these objective functions equilibria exist and Algorithm 1 converges towards an equilibrium. The price of anarchy is at most \( |Q| \), and when starting with an empty state, this quality is already reached after computing the best response once for every passenger.

### 5. Experiments

In this section, we describe a first experimental evaluation of our routing game approach. We tested the best response strategy
with the five different variants for solving the routing problem described in this paper: solving the routing problem exactly (abbreviated as BR), using the auxiliary frequency (AP) heuristic with overestimated (ub) underestimated (lb) transfer times, and using the auxiliary arc weight (AW) heuristic with overestimated (ub) underestimated (lb) transfer times. We furthermore compare it to the exact solution of the non-linear integer program (LPQC) which we solved as a semidefinite quadratic problem with Gurobi 7 (Gurobi Optimizer, 2016) (abbreviated as MP). Note that this is only possible for $\alpha_2 = 0$ (because otherwise it is a non-semidefinite quadratic program).

We tested the different approaches on two different instances. The first instance GRID is based on a 5 x 5 grid instance which was introduced in Friedrich, Hartl, Schieve, and Scholbel (2017a) with a modified line pool. The PTN is depicted in Fig. 5a. It consists of 25 stations and 40 edges, the line pool has 13 lines and there are 1927 passengers in 567 OD pairs. The second instance, GOE, is taken from the LinTim toolchain, see Schieve, Albert, Pätzold, Schieve, and Scholbel (2018a); Schieve, Albert, Pätzold, Schieve, and Scholbel (2018b). The PTN, shown in Fig. 5b, is derived from the bus-network in Göttingen, Germany. The instance consists of 257 stations, 548 edges, 6114 OD pairs and 6522 passengers. A line pool consisting of 44 lines was generated for these experiments. All experiments were done on a CPU of 16 cores with 2,4GHz and 132GB of RAM. The standard parameter set $P_1 = (1/1.10.3/3)$ was chosen to represent a realistic assessment of the generalized costs, provided by practical public transport planners. The parameter sets $P_2$ and $P_3$ are simplifications for the presented algorithms which are chosen to approximate $P_1$.

Table 1 shows the objective values with respect to the (LPQC), running times, and number of iterations for running a best-response strategy, compared to the mathematical program MP, on the instance GRID with parameter set $P_1 = (1/0.20.3/3)$. Note that BR is computed according to $P_2 = (1/0.20.6/0)$ in order to be able to solve the routing problem exactly but it is evaluated according to $P_1$. Objective values are reported relative to the optimal solution/best solution found. We see that BR and our heuristics converge to equilibrium after 6 or 7 iterations, but that these equilibria are not identical to the system optimum, i.e., the solution found by the (LPQC). We also observe that the running times of the best response strategies are only 3.6% to 7.7% of the running time of MP where BR and the simpler heuristics AW ub/lb are faster than AF ub/lb. In turn, the more complicated heuristics AF ub/lb yield on average better solutions than AW lb/ub. Note that the heuristics cannot utilize their full potential in this experiment, since transfer times are neglected here.

In Section 3.3 we describe how in an extreme case, (LPQC) can find a solution which has a better centralized objective value, but is unrealistic in the sense that some passengers have to choose much longer routes than others. Table 2 shows that, to a lesser degree, this is also the case for the experiment presented here.

Here, we compute a more balanced solution using BR instead of MP. Using MP, passengers for the same OD pair are assigned paths of different quality. While the number of transfers does not deviate within an OD pair, the average standard deviation over the number of passengers of the drive time of passengers belonging to the same OD pair is 0.002 with a maximum of 0.227 and the average standard deviation of the transfer time is 0.067 with a maximum of 7.071. Such a system optimal solution may not be possible to implement in reality, similarly as described in the example from Section 3.3. This problem does not occur when applying BR, where all passengers can choose a path of identical quality.

In Table 3 we see a comparison of the different variants of the best-response strategy with respect to the objective value of the (LPQC), running times, and number of iterations for the parameter set $P_1 = (1/1.10.3/3)$ on instance GRID. For MP the solution is computed with parameter set $P_1 = (1/0.20.3/3)$ and for BR with parameter set $P_2 = (1/0.20.6/0)$ (compare Table 1), since we can only apply these methods for $\alpha_2 = 0$ and $\gamma_2 = 0$ in case of BR. Preliminary experiments have indicated that among the parameter sets with $\alpha_2 = 0$, $P_3$ approximates $P_2$ best and among those with $\alpha_2 = \gamma_2 = 0$, $P_1$ approximates $P_2$ best.

We see that in all versions of the best-response strategy, convergence to the equilibrium is reached after 5 to 7 iterations. When comparing the solutions based on the objective value of the (LPQC) we see that the MP, executed with the parameter set $P_1$, still outperforms the best-response heuristics, although this parameter set neglects the transfer times. However, among the best response strategies, we see that the inclusion of transfer times seems to yield a benefit, since multiple heuristics find better solution than BR w.r.t. $P_2$.

To further investigate the different heuristics when transfer times are taken into account, Table 4 shows a comparison for different parameter sets on the instance GRID. We see that the more complex heuristics AF ub/lb always find the best solutions and often both outperform AW ub/lb. The simpler algorithms BR, AW ub/lb are faster than the more complex ones AF ub/lb which in turn are much faster than the optimization model MP.

Additionally to instance GRID, we tested our algorithms on the larger instance GOE as shown in Table 5. Here, the solution found by BR is only 8.3% worse than the one found by MP and the solution quality of most heuristics is similarly good. The runtime of BR and the heuristics range between 12.6% and 28.6% of the runtime of MP, again showing that BR and the simpler heuristics AW ub/lb are significantly faster than AF ub/lb while the more complex heuristics perform better.
6. Conclusions and further research

We presented a new idea to approach line planning by solving a routing game where the passengers are the players who aim at minimizing a weighted sum of their travel time, transfer penalties, and a cost share. Under strong assumptions on the objective function (transfer time is not taken into account and line costs can be assigned to edges and are independent of frequencies) equilibria of this game can be found using the described best-response algorithm. In case that the objective function does not fulfill these properties, applicability and convergence of the best-response approach can be achieved by a slight modification of the individual objective functions.

A logical next step will be to evaluate whether the line planning routing game, besides being an interesting object of study in itself, does indeed lead to a good heuristic for line planning.

First, more experiments of the type presented in Section 5 on instances of realistic size (in particular also with respect to passenger numbers) may lead to more insights on the performance of the different approaches presented in Section 4. A positive effect of increasing passenger numbers is that the approximate frequencies \( f_t(R^*_1) \) and \( f_t(R^*_2) \) become better estimates of actual frequencies \( f_t(\overline{R}) \). However, in the current version of the best-response strategy, in each iteration a shortest path for each passenger has to be found, hence running time increases with increasing number of passengers. For large passenger numbers it may thus make sense to use flow equilibration techniques in the inner loop instead of shortest path computations for each individual passenger.

Second, line planning solutions obtained with the routing game approach should be compared to state-of-the-art exact and heuristic solution methods for line planning with respect to objective function, running time, and practicability of the found solution (in the sense of Section 3.3).

While the terms for travel time and transfers are quite intuitive, many different choices are possible for the cost-sharing among passengers. It remains an interesting question how to divide operational costs among passengers such that, on the one hand, the algorithmic approach is still viable, and on the other hand, cost shares are comparable to real-world travel costs. Furthermore, it would be interesting to investigate whether the routing game approach can also be applied to line planning with additional constraints and other planning problems which can be considered integrated network design and routing problem like, e.g., timetabling or delay management with integrated routing.

Appendix

NP-hardness of the routing problem with transfer times

**Theorem 4.3.** The routing problem as in Definition 4.1 is NP-hard, even if transfer penalties and operational costs are not taken into account, i.e., \( \beta = 0 \) and \( \gamma_1 = \gamma_2 = 0 \).

**Proof.** Similarly to the proof of Theorem 4.2 we prove this theorem by reduction from SCP. Let \((M, C, K)\) denote an instance of SCP and denote \( n := |M| \). Our PTN consists of two parts: The first part is used to ensure that at most \( K \) sets are chosen from \( C \). The second part is similar to the construction in the proof of Theorem 4.2 and is used to determine whether the chosen sets cover \( M \).

The first part of the PTN consists of vertices \( v_i \) for \( i = 1 \ldots 2K + 1 \) and edges \( e_i = (v_i, v_{i+1}) \) for \( i = 1 \ldots 2K \) with \( c(e_i) = 0 \). For every edge \( e_{2k+1} \) with an odd index we introduce a line \( l_{2k+1} \) which consists of this edge only.

The second part of the PTN consists of vertices \( w_i \) for \( i = 1 \ldots 2n + 1 \) and edges \( e_i = (w_i, w_{i+1}) \) for \( i = 1 \ldots 2n \) with \( c(e_i) = 0 \). Furthermore, we add edges \( q_i \) which connect all pairs of vertices \( w_i \) and \( w_j \) with \( i < j \) and whose length is \( c(q_j) := K + 1 \), where \( K := \lfloor 2K + 2n \rfloor \). For each \( i = 1, \ldots, n \) we introduce a line \( l_{2k+1} \) which covers the edge \( q_{2k+1} \).

For every \( C \subseteq C \) we create a line \( l_C \in C \) containing all edges \( q_1, \ldots, q_{2n+1} \) from the first part of the PTN, the transition edge \( e_1 \) and the edges \( e_2 \) with \( m_i \in C \) from the second part of the PTN. We add additional edges with lengths \( K + 1 \) wherever needed to ensure that the lines are connected paths in the PTN.

In contrast to the proof of Theorem 4.2, in this proof we have \(|C| = 1 \) edges. Each passenger \( q_C \) with \( C \subseteq C \) has origin \( v_1 \) and destination \( v_{2n+1} \) and his route \( R_0 \) is identical to line \( l_C \) from \( v_1 \) to \( v_{2n+1} \). The passenger \( q_C \) for which we have to solve the routing problem has origin \( v_1 \) and destination \( v_{2n+1} \). We set the capacity in each vehicle to \( B = 1 \). For the objective function we use the parameters \( \alpha_1 = 1, \beta = 1, \gamma_1 = \gamma_2 = 0 \) and \( T = 1 \). Note that line costs can be set to arbitrary values, since \( \gamma_1 = \gamma_2 = 0 \).

We now show that there is a solution to the considered instance of SCP if and only if there is a solution \( R_0 \) to the routing problem (RP1) with individual objective value \( h(R_0, R^*_1) < K \).

First note that any such route \( R_0 \) in the first part of the PTN will use the lines \( l_C \) on edges with an odd index and some lines \( l_C \) on the ones with an even index, because otherwise \( h(R_0; R^*_1) > K \). Note that whenever the passenger uses a line \( l_C \), the frequency of this line is set to \( f_C = \infty \). Consequently, for all of these paths the contribution from the first part of the PTN to the transfer time component \( t^* \) in the individual objective function is \( 2K \), since the length of every used edge is 0 and on each such path there is a transfer at each station between a line \( l_C \) with frequency 1 (used only by passenger \( q_C \)) and a line \( l_C \) with frequency 0. In the second part of the PTN, only edges \( e_i \) can be used in such a route \( R_0 \), because otherwise \( h(R_0, R^*_1) > K \). Hence \( \sum q_C(R_0) = 0 \). Now consider the contribution to \( t^* \) of route \( R_0 \) in the second part of the PTN. At each node in the second part of the PTN a transfer has to take place, between a line \( l_{2k+1} \) and a line \( l_{2k} \). Hence, transfer time is \( \frac{1}{2} \) if passenger \( q_C \) did not use line \( l_{2k} \) in the first part of the PTN, \( \frac{1}{2} \) if he used it. Since there are \( 2n \) such transfers, any path with individual objective value less or equal to \( K \) uses on edge \( e_2 \) a line that was already used in the first part of the PTN (because otherwise \( h(R_0; R^*_1) > K \)).

Due to the construction of the lines \( l_C \), this means that if there is a route \( R_0 \) with \( h(R_0; R^*_1) < K \), for each element \( m_i \in M \) at least one line \( l_C \) with \( C \subseteq C \) is used in the first part of the PTN. Since no more than \( K \) such lines can be used in \( R_0 \), there must be a solution to the considered instance of SCP.

On the other hand, if there is a solution \( C^* \equiv \{C_1, \ldots, C_k\} \) with \( k \leq K \) to the considered instance of SCP, using line \( l_C \) on edge \( e_2 \) for \( i = 1, \ldots, K \) (and arbitrary lines on \( e_2 \) for \( i = k + 1, \ldots, K \)) allows the passenger to choose a path with transfer time \( \frac{1}{2} \) in the second part of the PTN and thus yields an individual objective value of at most \( K \).

Non-existence of equilibria

We now describe the example for non-existence of equilibria from Section 4.2.1 more formally and prove that no equilibrium exists.

We consider the PTN from Fig. 3 with 12 nodes and 18 edges. Every edge is served by one directed line which contains only this edge, so that we have a one-to-one correspondence between edges and lines. The capacity of a vehicle is \( B = 1 \). There are three main passengers: \( q_1 \) from \( v_1 \) to \( v_2 \), \( q_2 \) from \( v_2 \) to \( v_3 \), and \( q_3 \) from \( v_3 \) to \( v_4 \) and six sets of auxiliary passengers: \( q_i^* \) for \( i = 1, \ldots, 3 \) and \( j = 1, 2 \) contains \( M \) passengers from \( v_j^* \) to \( v_j \) (where \( M \) is a sufficiently large
number, e.g., M > 12. We denote by H the union of the auxiliary passengers.

In our objective function we take only the transfer time into account, i.e., \( \alpha_1 = \beta = \gamma_1 = \gamma_2 = 0 \) and \( h_q(R) := t_q(R, R^{-}) \). We set \( T = 1 \).

Note that for the auxiliary passengers there is only one route from origin to destination, hence, each of them only has one strategy. Let \( R^i \) denote the set of these strategies. Each of the main passengers \( q \) has two different strategies: to take the route \( R^1 \) starting with edge \( (u, v^1) \) or to take the route \( R^2 \) starting with edge \( (u, v^2) \).

We now show that there does not exist an equilibrium in the described situation. Assume that \( R \) is an equilibrium of the described line planning routing game. Denote by \( \Pi_q \) the strategy chosen by \( q \). Without loss of generality, assume that \( J_1 = 1 \). Then

\[
g_1(R^1, R^2, R^1, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

if \( j_3 = 1 \).

and

\[
g_2(R^1, R^2, R^2, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

if \( j_3 = 2 \).

Since \( R \) is an equilibrium, we conclude that \( J_2 = 1 \), i.e., \( R^2 = R^1 \).

Now

\[
g_1(R^1, R^2, R^1, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

and

\[
g_2(R^1, R^2, R^2, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

Since \( R \) is an equilibrium, we conclude that \( J_3 = 1 \), i.e., \( R^3 = R^1 \).

Now we take a look at the strategies for \( q \):

\[
g_1(R^1, R^2, R^1, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

and

\[
g_2(R^1, R^2, R^2, R^{-}) = 1 + \frac{1}{4} \left( \frac{1}{2} + \frac{1}{2} \right) + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + \frac{1}{6} + 1 + \frac{1}{6} = 7 + \frac{7}{2} + \frac{7}{2}
\]

Thus, \( g_i(R^1, R^2, R^1, R^{-}) = g_i(R^1, R^2, R^2, R^{-}) \). This is a contradiction to \( R^2 \) being part of an equilibrium.

Due to the symmetry of the construction of the instance, the assumption that \( R^2 \) is part of an equilibrium leads to a contradiction in the same way.

**Proof of existence of potential functions for games with arc weight functions**

**Lemma 4.5.** Let \( I := (G, L, Q, \{ h_q \mid q \in Q \}) \) be an instance of the LPRG such that \( \Pi(R) := \sum_{i \in R} \sum_{j \in R^i} \psi_q(i) \) is a potential function for \( I \),

1. there exists an equilibrium for \( I \),
2. Algorithm 1 converges to an equilibrium in a finite number of steps.

4. each of the steps can be executed in polynomial time.

**Proof.** This proof follows standard arguments for convergence of atomic routing games, compare, e.g., Roughgarden (2007).

Let \( R \) and \( R^{-} \) be two route sets. We denote with \( P_1 \) and \( P_2 \) the corresponding paths for passenger \( q \) in the CGN and with \( x_q := x_q(R) \) and \( x_q^{-} := x_q(R^{-}) \) the corresponding flows on edge \( e \) of the CGN. We first observe that

\[
\Phi(R, R^{-}) - \Phi(R^{-}, R^{-}) = \sum_{a \in R} \psi_q(a_1) - \sum_{a \in R^{-}} \psi_q(a) = \Psi_q(R) - \Psi_q(R^{-})
\]

hence \( \Phi \) indeed is a potential function by (1).

2. Hence, every optimum of \( \Phi \) is an equilibrium of the game. Since the number of solutions is finite, there exists at least one optimum of \( \Phi \) of equilibrium of \( I \).

3. Since in each step of Algorithm 1 there is a non-zero improvement in the individual objective function and thus also in the potential function, and the number of solutions is bounded, Algorithm 1 converges to an optimum of \( \Phi \) which is an equilibrium.

4. We set \( w_a(R^{-}) := w_a(a) + 1 \).

\[
\Psi_q(R) := \sum_{a \in R} \psi_q(a_1) + 1
\]

and

\[
\Psi_q(R^{-}) := \sum_{a \in R} \psi_q(a_1) + 1
\]

The proposition follows from Lemma 4.4.

**References**


E. Look-Ahead Approaches for Integrated Planning in Public Transportation

M. Pätzold, A. Schiewe, P. Schiewe, A. Schöbel
Look-Ahead Approaches for Integrated Planning in Public Transportation
[Pätzold et al., 2017]
Look-Ahead Approaches for Integrated Planning in Public Transportation

Julius Pätzold, Alexander Schiewe, Philine Schiewe, and Anita Schöbel

Abstract

In this paper we deal with three consecutive planning stages in public transportation: Line planning (including line pool generation), timetabling, and vehicle scheduling. These three steps are traditionally performed one after another in a sequential way often leading to high costs in the (last) vehicle scheduling stage. In this paper we propose three different ways to “look ahead”, i.e., to include aspects of vehicle scheduling already earlier in the sequential process: an adapted line pool generation algorithm, a new cost structure for line planning, and a reordering of the sequential planning stages. We analyze these enhancements experimentally and show that they can be used to decrease the costs significantly.

1998 ACM Subject Classification G.1.6 Optimization, G.2.2 Graph Theory, G.2.3 Applications

Keywords and phrases line pool generation, line planning, vehicle scheduling, integrated planning, public transport

Digital Object Identifier 10.4230/OASIcs.ATMOS.2017.17

1 Sequential versus integrated planning

Planning a public transport supply can have many goals. Two major goals are usually minimizing the perceived travel times of passengers as well as the costs that incur to the public transportation company. Motivated by this we consider a bi-objective model for railway or bus planning with these two objectives.

Traditionally, public transportation planning is done in sequential stages. The first stage after the design of a network, that is spanned by stops (or stations) and their direct connections (edges or tracks), is line planning. In this stage, first a set of possible lines, the line pool, has to be generated on the network. Research towards the effect of line pool
Look-Ahead Approaches for Integrated Planning in Public Transportation

generation, and an algorithm to find suitable line pools is presented in [7]. In the line planning problem one then chooses a feasible subset of lines from the line pool, i.e., a set of lines such that all passengers can be transported. See [21] for an overview. With a given line plan one can create an event-activity network which constitutes the input for the timetabling stage. Periodic timetabling consists of deciding when and how fast vehicles (trains or buses) should drive along the edges and how long they should wait at stops (or stations). The problem is modeled as a periodic event scheduling problem (PESP), see [23]. Other timetabling models can be found in [10]. After a timetable is chosen, vehicle schedules are planned, determining which vehicle should drive which route such that all lines are operated according to their timetables. A survey on vehicle scheduling is given in [4]. Finally, crew scheduling and rostering are planning stages to be performed after the vehicle schedules are found.

Obviously, proceeding sequentially does not need to lead to an optimal solution as there are dependencies between the different subproblems. It would hence be beneficial to solve the entire problem in an integrated system. Since this is computationally too complex, heuristic approaches have been proposed as in [22].

Our contribution. We consider line planning, timetabling and vehicle scheduling in conjunction with each other. To this end we formally define what an integrated transport supply (LTS-plan), consisting of a line plan, a timetable, and a vehicle schedule, is and how it can be evaluated. We propose three enhancements of the traditional approach which consider the vehicle scheduling costs already in the line planning stage. Finally, we evaluate them experimentally and show that our proposed enhancements lead to LTS-plans with significantly smaller costs than the traditional sequential approach.

2 A bi-objective model for integrated planning in public transportation

In this section we formally describe what a feasible transport supply (LTS-plan), consisting of a line plan (L), a timetable (T), and a vehicle schedule (S), is and how its quality can be evaluated. Note that for the single stages, i.e., for a line plan, for a timetable, and for a vehicle schedule, this has been extensively discussed in the literature. However, it is in the literature usually assumed that an event-activity network is already known for timetabling and a set of trips is already given for vehicle scheduling. Since we plan from scratch, we also have to describe the intermediate steps, i.e., how to build the event-activity network and how to build the set of trips. In order to keep the timetabling step tractable, we restrict ourselves in this paper to periodic LTS-plans for which all lines are operated with the same frequency.

As input for the bi-objective model we are given:

- A public transport network PTN = (V,E) consisting of a set of stops V and direct connections E between them.
- For every node v ∈ V:
  - lower and upper bounds \( l_{\text{wait}}^v \leq u_{\text{wait}}^v \) for the time vehicles wait at stop v,
  - lower and upper bounds \( l_{\text{trans}}^v \leq u_{\text{trans}}^v \) for the time passengers need to transfer between two vehicles at the same stop v.
  - We furthermore need for every pair v, u ∈ V the time(v,u) a vehicle needs if it drives directly from stop v to stop u.
- For every edge e = (v_1, v_2) ∈ E:
  - a length (in kilometers) \( l_e \),
  - lower and upper edge frequency bounds \( f_{\text{min}}^e \leq f_{\text{max}}^e \),
  - lower and upper bounds on the travel times along the edge, i.e., \( l_{\text{drive}}^e \leq u_{\text{drive}}^e \).
An OD-matrix $W$ with entries $W_{uv}$ for each pair of stops $u, v \in V$. The OD-matrix is assumed to be consistent with the lower edge frequencies, i.e., there exist $P_{uv}$ for every OD-pair $(u, v)$ through the PTN such that for every edge $e$ we have:

$$\sum_{u,v \in V : e \in P_{uv}} W_{uv} \leq Cap \cdot f_{e}^{\min}$$

for $Cap$ being the capacity of the (identical) vehicles, i.e., each passenger can be transported,

- a period length $T$,
- the number of periods $p$ to be considered for planning
- a minimal turnaround time for vehicles $L_{\text{min}}$,
- cost parameters
  - $c_1$ costs per minute for a vehicle driving with passengers,
  - $c_2$ costs per kilometer for a vehicle driving with passengers,
  - $c_3$ costs per vehicle for the whole planning horizon ($p$ periods),
  - $c_4$ costs per minute for a vehicle driving empty (i.e., without passengers),
  - $c_5$ costs per kilometer for a vehicle driving empty (i.e., without passengers).

We then look for an LTS-plan, which consists of a line plan (L), a periodic timetable (T) and a vehicle schedule (S) which are together feasible. These objects are defined as follows:

**Line plan L**

A line is a path through the PTN. A line plan is a set of lines $\mathcal{L}$, which is feasible if $f_{e}^{\min} \leq |\{l \in \mathcal{L} : e \in l\}| \leq f_{e}^{\max}$, i.e., if each edge of the PTN is covered by the required number of lines. We assume that lines are symmetric, i.e., they are operated in both directions. In our setting all lines are operated with a frequency of 1.

**Timetable T**

Given a set of lines, a timetable assigns a time to every departure and arrival of every line at its stops. These times are then repeated periodically. In order to model a timetable usually event-activity networks $\mathcal{N} = (\mathcal{E}, \mathcal{A})$ are used (see, e.g., [11, 12, 14, 17, 18]). The set of events $\mathcal{E}$ consists of all departures and all arrivals of all lines at all stops, and the set $\mathcal{A}$ connects these events by driving, waiting and transfer activities. For each activity, the number of passengers using this activity is usually given as input for timetabling. (It is subject of ongoing research how this can be relaxed, see [3, 6, 19, 20]). The lower and upper bounds $L_a$ and $U_a$ are set as

- $L_{\text{drive}}$ and $U_{\text{drive}}$ if $a$ is a driving activity on edge $e \in \mathcal{E}$,
- $L_{\text{wait}}$ and $U_{\text{wait}}$ if $a$ is a waiting activity in stop $v \in V$, and as
- $L_{\text{trans}}$ and $U_{\text{trans}}$ if $a$ is a transfer activity in stop $v \in V$.

A timetable $\pi$ is an assignment of times $\pi_{j} \in \mathbb{Z}$ to every event $j \in \mathcal{E}$. It is feasible if it respects the lower and upper bounds for all its activities, i.e., if

$$\langle \pi_{j} - \pi_{i} - L_a \rangle \mod T \in [0, U_a - L_a] \text{ for all } a = (i,j) \in \mathcal{A}.$$  

The objective function in timetabling minimizes the total slack times. If all passengers use the paths they have been assigned to in the event-activity network this is equivalent to minimizing the sum of passengers’ travel times.
17:4 Look-Ahead Approaches for Integrated Planning in Public Transportation

Vehicle schedule $S$

Given a set of lines and a timetable, a *vehicle schedule* determines the number of vehicles and the exact routes of the vehicles for operating the timetable. To this end, we use the line plan and the timetable to construct a set of *trips* $T$ where each trip $t = (l_t, v_{t,\text{start}}, v_{t,\text{end}}, \bar{\pi}_{t,\text{start}}, \bar{\pi}_{t,\text{end}}) \in T$ is specified by a line $l_t$ together with its first and last stop $v_{t,\text{start}}$ and $v_{t,\text{end}}$ and its corresponding *start time* $\bar{\pi}_{t,\text{start}}$ and *end time* $\bar{\pi}_{t,\text{end}}$. These times can be taken from the periodic timetable, but we have to consider the real time (e.g. in minutes after midnight) by adding the correct multiple of the period length. The end time $\bar{\pi}_{t,\text{end}}$ of a line at its final stop is the arrival time at this stop plus some minutes allowing passengers to deboard. Analogously, the start time $\bar{\pi}_{t,\text{start}}$ of a line at a stop is the time when it arrives at this stop, i.e., a bit earlier than its departure time there. For every line $l_t$ we receive two trips starting per period, namely one forward and one backward trip. A route of a vehicle is given by its sequence of trips $r = (t_1, \ldots, t_k)$ such that

$$(\bar{\pi}_{t_{i+1},\text{start}} - \bar{\pi}_{t_i,\text{end}}) \geq \text{time}(v_{t_{i+1},\text{start}}, v_{t_i,\text{end}}) \text{ for all } i = 1, \ldots, k-1.$$ 

A set of vehicle routes $R$ is feasible if all its routes are feasible and if each trip is contained in exactly one route.

Evaluating an LTS-plan

An LTS-plan is specified by a line plan, a corresponding timetable and a corresponding vehicle schedule, i.e., it is specified by the tuple $(L, \pi, R)$. Given a feasible LTS-plan we use the two most common evaluation criteria: the sum of passengers’ travel times (including a penalty for every transfer) and the costs. These objectives are formally defined below:

**Costs.** The costs of an LTS-plan depend mainly on the costs of the corresponding vehicle schedule and thus on the distance which is driven, the total duration of driving and the number of required vehicles. For the distance and the duration of the trips we distinguish if the vehicle drives on a trip which can be used by passengers (here called *full ride*) or if the vehicle drives empty between two consecutive trips $t_i, t_{i+1}$ in the same vehicle route (here called an *empty ride*) as the costs can be different for full and empty rides.

As the vehicle schedule in general is aperiodic, we consider the costs for a whole planning horizon (e.g. a day) instead of a planning period by rolling out the periodic line plan and timetable for a fixed time span which is given by the number of periods $p$ it covers. Note that we have to take special care at the beginning and the end of the roll-out period, regarding lines traversing the period boundaries. For simplicity reasons we do not go into detail here how this is handled explicitly.

Before defining the costs, we introduce the duration and the length of a line and an empty ride. Let a line be defined as a sequence of nodes and edges.

The duration of a line can be determined after the timetable is known. We get

$$\text{dur}_l = \sum_{a=(i,j) \in A_{\text{drive}}, \text{a belongs to } v \in l} (L^\text{drive}_a + (\pi_j - \pi_i - L^\text{drive}_a \mod T)) + \sum_{a=(i,j) \in A_{\text{wait}}, \text{a belongs to } v \notin l} (L^\text{wait}_a + (\pi_j - \pi_i - L^\text{wait}_a \mod T)).$$
i.e., all driving times along edges and waiting times at stops are added. When a heuristic approach to timetabling is used where the duration of all driving and waiting activities is set to their respective lower bounds, as done here, the duration of a line simplifies to

$$\text{dur}_l = \sum_{e \in l} L^\text{drive}_e + \sum_{e \in l} L^\text{wait}_e.$$  \hfill (3)

The length of a line is computed as sum over all edge lengths

$$\text{length}_l = \sum_{e \in l} \text{length}_e$$

and is independent from the timetable. The duration of an empty ride between two trips

$$t_1 = (l_{t_1}, v_{\text{start}t_1}, v_{\text{end}t_1}, \tilde{\pi}_{\text{start}t_1}, \tilde{\pi}_{\text{end}t_1})$$

and

$$t_2 = (l_{t_2}, v_{\text{start}t_2}, v_{\text{end}t_2}, \tilde{\pi}_{\text{start}t_2}, \tilde{\pi}_{\text{end}t_2})$$

can be computed as

$$\text{dur}_{t_1,t_2} = \tilde{\pi}_{\text{end}t_2} - \tilde{\pi}_{\text{start}t_1},$$

i.e., the time between the end of $t_1$ and the start of $t_2$.

The length of the empty ride is defined as

$$\text{length}_{t_1,t_2} = SP(v_{\text{end}t_1}, v_{\text{start}t_2}),$$

i.e., we assume that a vehicle takes the shortest path from the last station $v_{\text{end}t_1}$ of trip $t_1$ to the first station $v_{\text{start}t_2}$ of trip $t_2$.

Now we can define the following cost components. Note that we have to count the full duration and length of each line twice as two trips belong to every line (one in forward and one in backward direction).

- **full duration**, i.e., time it takes to cover all trips (full rides):

$$\text{dur}_\text{full} = \sum_{l \in \mathcal{L}} 2 \cdot \text{dur}_l \cdot p_i,$$

- **full distance**, i.e., distance driven along lines:

$$\text{length}_\text{full} = \sum_{l \in \mathcal{L}} 2 \cdot \text{length}_l \cdot p_i,$$

- **number of vehicles**: $\text{veh} = |\mathcal{R}|$.

- **empty duration**, i.e., time of empty rides between trips:

$$\text{dur}_\text{empty} = \sum_{r = (t_1, \ldots, t_k) \in \mathcal{R}} \sum_{i = 1}^{k-1} \text{dur}_{t_i,t_{i+1}},$$

- **empty distance**, i.e., distance of empty rides between trips:

$$\text{length}_\text{empty} = \sum_{r = (t_1, \ldots, t_k) \in \mathcal{R}} \sum_{i = 1}^{k-1} \text{length}_{t_i,t_{i+1}}.$$

In total we get

$$g^\text{cost}(\mathcal{L}, \pi, \mathcal{R}) := c_1 \cdot \text{dur}_\text{full} + c_2 \cdot \text{length}_\text{full} + c_3 \cdot \text{veh} + c_4 \cdot \text{dur}_\text{empty} + c_5 \cdot \text{length}_\text{empty}. \hfill (4)$$
Travel times. For determining the travel time we follow the traditional approach of fixing the passengers’ routes when constructing the event-activity network, assuming that the passengers use these assigned paths. In the event-activity network, passengers are routed on a shortest path according to the lower bounds on the activities and assigned as weights $c_a$ to the activities $a \in \mathcal{A}$. Additionally to the travel time, we consider a penalty $\text{pen}$ for every transfer. The total perceived travel time on these fixed paths can then be determined as

$$g^\text{time}(L, \pi, R) = \sum_{a=(i,j) \in \mathcal{A}} c_a \cdot (L_a + (\pi_j - \pi_i - L_a \mod T)) + \sum_{a \in \mathcal{A}_{tr}} c_a \cdot \text{pen}.$$  

(5)

Note that the travel time does not depend on the vehicle schedule.

The two objective functions we have sketched here are common in the literature when broken down to one single planning stage:

Nearly all papers dealing with vehicle scheduling minimize a combination of empty kilometers and number of vehicles needed, i.e., $\text{veh} + a \cdot \text{length}_{\text{empty}}$. This is equivalent to $g^\text{cost}$ if the duration of full and empty rides are weighted equally and $a$ is chosen as $a = \frac{1}{2}$ since the duration and the length of the lines are all known due to the timetable being fixed.

In timetabling, the goal is usually to minimize the sum of (perceived) travel times for the passengers. Since it is computationally very difficult, most papers make the simplifying assumption that the number of travelers on every activity in the event-activity network is known and fixed, as it is done here.

Pareto optimal LTS-plans. We call a feasible LTS-plan $(L, \pi, R)$ Pareto optimal if there does not exist another LTS-plan $(L', \pi', R')$ which satisfies

$$g^\text{cost}(L', \pi', R') \leq g^\text{cost}(L, \pi, R), \quad g^\text{time}(L', \pi', R') \leq g^\text{time}(L, \pi, R)$$

with one of the two inequalities being strict.

3 Traditional sequential approach

The traditional approach is a combination of algorithms which have been described in the literature. It goes through line planning, timetabling, and vehicle scheduling sequentially and finds (close to) optimal solutions in each of the steps.

Step L: Line planning. There exists a variety of algorithms for line planning, see [21]. Some of them assume a line pool to be given, others determine the lines during their execution ([2]). If a line pool is required, a line pool generation procedure can be used (see [7] and references therein).

In our experiments: We use the cost model for a fixed line pool which is either given (dataset Bahn) or generated by [7] (dataset Grid).

Step T: Timetabling. Solving the integer programming formulations is too time-consuming for most instances, hence often heuristics ([9, 15, 16]) are used.

In our experiments: We use the fast MATCH heuristic [16].

Step S: Vehicle scheduling. There exists a variety of algorithms, see [4].

In our experiments: We use the flow-based model of [4].

We remark that even if all three steps are solved optimally, the resulting LTS-plan need not be Pareto optimal. This is due to the sequential approach: the line plan is the basis for the timetable and the vehicle schedule, but optimal lines cannot be determined without knowing the optimal timetable and the optimal vehicle schedule.
4 Look-ahead enhancements

As already mentioned, the vehicle schedules have a large impact on the costs of an LTS-plan. Since the vehicle schedules are determined only in the last of the three considered planning stages, the costs of an LTS-plan determined by the sequential approach are usually not minimal. We propose three enhancements in order to receive LTS-plans with better costs than in the sequential approach. We nevertheless also evaluate the perceived travel times for the passengers.

4.1 Using new costs in the line planning step

When evaluating the costs of an LTS-plan, (4) shows that the costs are determined to a large amount by the number of vehicles needed. Even if as few lines as possible are established it is not clear how many vehicles are needed in the end and how many empty kilometers are necessary.

In the traditional approach the costs of a line are usually assumed to be proportional to its length with some fixed costs to be added, i.e.,

$$\text{cost}_l = \text{cost}_{fix} + c \cdot \text{length}_l$$

where $\text{cost}_{fix}$ $\in$ $\mathbb{R}^+$ and $c$ $\in$ $\mathbb{R}^+$ is a scaling factor.

Here, we now try to compute the costs of a line as closely as possible to the costs it may have later in the evaluation of the LTS-plan. The idea is to approximate the costs per line by distributing the costs specified in (4) to the lines and computing the costs per period, i.e., we want to get

$$g_{cost} \approx \sum_{l \in L} \text{cost}_l \cdot p.$$

For full duration and distance this can be done straightforwardly, as we only need to know the number of planning periods which are considered in total as the length and duration of a line does not change between periods. Under our assumptions, we know the duration of a line beforehand by (3). The number of vehicles needed, the empty distance and the empty duration are in general more difficult to approximate as they can differ between the planning periods due to an aperiodic vehicle schedule. As upper bound we use a very simple vehicle schedule where all vehicles periodically cover only one line and its backwards direction. This gives us that the empty distance is always zero and can be neglected. The empty duration of a line can be computed as

$$\text{empty duration after driving on line } l = \frac{T}{2} - (\text{dur}_l \mod \frac{T}{2}),$$

and for a given minimal turnaround time $L_{min}$ of a vehicle, the number of vehicles needed to serve a line and its backwards direction can be approximated by

$$\#\text{vehicles needed for line } l \text{ and backwards direction } = \lfloor 2 \cdot (\text{dur}_l + L_{min})/T \rfloor.$$

Summarizing, we can approximate the line costs as:

$$\text{cost}_l = 2 \cdot c_1 \cdot \text{dur}_l + 2 \cdot c_2 \cdot \text{length}_l + c_3 \cdot \left( \frac{2 \cdot \text{dur}_l + L_{min}}{T} \right) + 2 \cdot c_4 \cdot \left( \frac{T}{2} - \text{dur}_l \mod \frac{T}{2} \right).$$

(7)
4.2 Line pool generation with look-ahead

The next idea is to take account of good vehicle schedules already in the very first step: we construct the lines in the line pool in a way such that no empty kilometers are needed and that the resulting lines are likely to be operated with a small number of vehicles.

To create a line pool which already considers the vehicle routing aspect, we modified the line pool generation algorithm described in [7]. For a given minimal turnaround time $L_{\text{min}}$ of a vehicle and a maximal allowed buffer time $\alpha$ we ensure that the duration $\text{dur}_l$ as defined in (3) of a line $l$ satisfies

$$\frac{T}{2} - L_{\text{min}} - \alpha \leq \text{dur}_l \mod \frac{T}{2} \leq \frac{T}{2} - L_{\text{min}}.$$  \hspace{1cm} (8)

Here, the duration of a line is computed according to the minimal driving time on edges and the minimal waiting time in stops. Equation (8) ensures that at the end of a trip, i.e., the driving of a line, the vehicle has enough time to start the trip belonging to the backwards direction of the same line and has to wait no more than $\alpha$ minutes to do so. Thus, we get that the round-trip of forward and backward direction together differs from an integer multiple of the period length by at most $2 \cdot \alpha$.

4.3 Vehicle scheduling first

In our last suggestion we propose to switch Step T and Step S in the sequential approach, i.e., to find (preliminary) vehicle schedules directly after the line planning phase. This is particularly interesting if the line plan contains lines which can be operated efficiently by one vehicle, i.e., lines with small $\alpha$, since it ensures that the timetable will not destroy this property. This is done as follows:

**Step L:** This step is done as in the traditional approach.

**S-first:** For every line $l$ we introduce turnaround activities in the periodic event-activity network between the last arrival event of the line in forward direction and the first departure event of the line in backward direction, and vice versa. The lower bound for these activities is set to $L_{\text{min}}$ and the upper bound to $L_{\text{min}} + 2 \cdot \alpha$. These activities ensure that the timetable to be constructed in the next step allows the vehicle schedule we want, namely that only one vehicle operates the line.

**Step T:** We then proceed with timetabling as in the traditional approach but respecting the turnaround activities such that the resulting timetable does not destroy the desired vehicle schedule.

**Step S:** After timetabling we perform an additional vehicle scheduling step as in the classic approach: We delete the turnaround activities and proceed with vehicle scheduling as usual. Nevertheless, it is likely, that many of the vehicle routes already determined in S-first will be found again.

Note that S-first can be performed very efficiently in the number of lines in the line concept. We furthermore remark that for a line plan in which all lines have a buffer time $\alpha = 0$, the Step S can be omitted since having line-pure vehicle schedules is an optimal solution in such a case. Even if not all lines have zero buffer times, fixing a timetable in Step T with respecting the turnaround activities often already determines the optimal vehicle schedule. This means that vehicle scheduling in Step S is often redundant, which was not only observable in most cases of our experiments, but is also illustrated more precisely in Example 1 of the appendix.
5 Experiments

We compared the traditional approach for finding an LTS-plan against the enhancements proposed using LinTim, a software framework for public transport optimization [1, 8]. We use the following parameters to describe the different combinations of our enhancements.

1. Using the new costs (7) in line planning (Step L) as proposed in Section 4.1 is denoted by new cost, whereas traditional costs are denoted as normal cost.
2. The second option, described in Section 4.2 is to construct a new pool (new pool), whereas normal pool uses some given (standard) pool for line planning (Step L). Combining both pools has been done in a third option (combined pool).
3. The decision of computing the timetable or the vehicle schedules first (so using Step S-first from Section 4.3), is denoted by TT first and VS first respectively.

As test instances we used two significantly different datasets.

Dataset Grid: A grid graph of 5 by 5 nodes and 40 edges, which is a model for a bus network constructed in [5]. In this example, we have $T = 20$ and we used $p = 24$ periods. The normal pool for this instance has been calculated with the tree based heuristic from [7].

Dataset Bahn: This is a close-to-real world instance which consists of 250 stations and 326 edges describing the German ICE network. The period length is $T = 60$, we computed for $p = 32$ periods in order to achieve a reasonable time horizon for vehicle scheduling. Note that $p$ is even larger in practical railway applications. As normal pool we used a pool of Deutsche Bahn. For the computations we used a standard notebook with i3-2350M processor and 4 GB of RAM. The computation time for one data point of the Grid dataset did not exceed 3 min, while computing a solution for the Bahn dataset took up to 30 minutes.

5.1 Dataset Grid

Figure 1 shows 12 solutions, one for every combination of our parameters. These are graphed according to travel times (x-axis) and their costs (y-axis). We computed the costs and the travel times of the LTS-plans as described in (4) and in (5). We observe the following:

- The solution of the traditional approach (circle with grey marker, left side filled) is
dominated by the solution obtained when replacing normal pool by combined pool.
- Using new cost (black markers) instead of normal cost (grey markers) always decreases the costs.
- Using combined pool always has better costs than using new pool or normal pool. The travel times sometimes decrease and sometimes increase.
- The option TT first yields better travel times compared to VS first while VS first always has lower costs than TT first.
- There are five non-dominated solutions, four of them computed by using new cost. Whenever new pool or combined pool was used together with new cost the resulting solution was non-dominated.

The new pool to be generated depends on the parameter $\alpha$. In Figure 1, $\alpha = 3$ was used. We also tested the parameters $\alpha = 2, 3, \ldots, 10$ for all combinations. The result is depicted in Figure 2. Note that $\alpha \geq 10$ implies no restrictions on the line lengths.

The basic findings described for $\alpha = 3$ remain valid also for other line pools generated: Solutions generated with new cost have lower costs while solutions generated with normal cost have smaller travel times. The leftmost solutions correspond to TT first and bottom-most solutions correspond to VS first. In fact, for every single LTS-plan that has been
Figure 1 Different combinations of look-ahead steps.

Figure 2 Different combinations of look-ahead steps and different choices for $\alpha$. 
computed, VS first yielded a cheaper solution than TT first while the latter resulted in a solution with smaller travel time than VS first. Finally, none of the solutions computed by using normal pool is non-dominated; the Pareto front (i.e., the non-dominated solutions) consists mostly of squares, i.e., solutions generated with combined pool. Nevertheless, we see that the quality of the solution obtained depends significantly on the choice of the parameter $\alpha$. This is investigated in Figure 3.

First of all, we again see that for every fixed $\alpha$ new cost yields better solutions than normal cost and that the combined pool always yields lower costs than new pool. If all three look-ahead enhancements new cost, combined pool and VS first are applied, there is a trend of increasing costs once $\alpha$ increases, corresponding to the conjecture that cheap LTS-plans can be found by a small choice of $\alpha$. For $\alpha = 0$ and $\alpha = 1$ the restrictions on the line length implied by equation 8 is in this example of a grid graph so strict that no feasible solution is possible.

5.2 Dataset Bahn

Applying the implemented enhancements to Bahn with the parameter choice $\alpha = 10$ (Note that $\alpha = 3$ for $T = 20$ in dataset Grid is similar to $\alpha = 10$ for $T = 60$ in dataset Bahn.) yields the results depicted in Figure 4.

The remarkable thing observable in this scenario is that new and combined pool lead to drastically vehicle cost reductions of more than 40%, whereas the travel time increases by up to 20%. Next to the fact of combined pool leading to better costs also the behaviour of TT first against VS first remains similar to the Grid instance. One can see that VS first saves costs between 1 and 5% and TT first decreases the travel time by 1 to 3 %. Since the size of the generated line pool had to be chosen small in comparison to the instance size (because of runtime and memory limitations), also the number of feasible line concepts is comparable small. Therefore, this example did not show any impact of using normal or new cost to the vehicle scheduling costs.

6 Relation to the Eigenmodel

In [22], it is proposed to use different paths through the Eigenmodel (depicted in Figure 5 in the appendix) when optimizing an LTS-plan. In this model, the traditional approach (normal cost, normal pool, TT first) has been depicted as the blue path starting with line planning, then finding a timetable and finally a vehicle schedule. In this paper we compared this traditional approach to two other paths:

- The approach (normal cost, normal pool, VS first) corresponds to the red path in which first a line planning step is performed, then vehicle schedules are determined and finally a timetable. We have seen that this approach leads to significantly better costs but to a higher travel time.

- The approach (new cost, new pool, VS first) can be interpreted as the green path in which we start with vehicle scheduling (by generating a line pool with small $\alpha$ only containing lines with low vehicle scheduling costs), choose a line plan out of this pool and finally determine a timetable which respects the preferred vehicle schedules. In Figure 1 we see that this approach generated the solution with lowest costs. Neglecting the tiny difference between normal and new cost this also holds for the Bahn instance.
Look-Ahead Approaches for Integrated Planning in Public Transportation

Figure 3 Impact of choice for $\alpha$.

Figure 4 Different combinations of look-ahead steps.
7 Outlook and further research

Summarizing our experiments, all three look-ahead enhancements lead in the majority of cases to a cheaper LTS-plan. Even choosing only one of the approaches will most likely lead to this goal. It is remarkable that the implementation of the proposed algorithmic ideas even performs very well on the Bahn dataset, that has the size and structure of a real world instance. Since exact approaches are far away from solving data sets of this size, the look-ahead heuristic proves itself useful for revealing the strength of considering integrated public transportation optimization.

The presented look-ahead approaches are designed to find a cost-optimized LTS-plan. One could also try to find heuristic approaches focussing on finding a passenger-convenient LTS-plan. A possible step towards this direction would be to choose a different line planning procedure, in order to optimize not with respect to the costs, but for example with respect to the number of direct travelers in the network.

Further research could also be carried out regarding exact approaches of integrated public transportation planning. It would be interesting to investigate different ways of decomposing the integrated problem, in particular, if also routing decisions are included. First results are under research, see [13].

References

Appendix

The following example shows that it is unlikely to find a better vehicle schedule in Step S.

Example 1. Consider two lines \( l_1 \) and \( l_2 \) such that line \( l_1 \) ends at the station that \( l_2 \) starts at as shown in Figure 6.

Let the duration of the lines be \( \text{dur}_{l_1} = \frac{T}{2} + \epsilon \) and \( \text{dur}_{l_2} = \frac{T}{2} - \epsilon \) such that \( \text{dur}_{l_1} + \text{dur}_{l_2} = T \).

Then using S-first with \( L_{\text{min}} = 0 \) we will need two vehicles to serve line \( l_1 \) and an additional vehicle to serve line \( l_2 \), as the following computation shows. The corresponding vehicle schedule can be seen in Figure 7.

\[
\left\lfloor \frac{2 \cdot (\frac{T}{2} + \epsilon)}{T} \right\rfloor = \left\lfloor \frac{T + 2 \cdot \epsilon}{T} \right\rfloor = 2
\]
\[
\left\lfloor \frac{2 \cdot (\frac{T}{2} - \epsilon)}{T} \right\rfloor = \left\lfloor \frac{T - 2 \cdot \epsilon}{T} \right\rfloor = 1
\]

17:14 Look-Ahead Approaches for Integrated Planning in Public Transportation

Figure 5 The paths investigated in the Eigenmodel.

Figure 6 Lines overlapping at station $u$.

Figure 7 Vehicle schedule derived by S-first.
However, both lines could also be served consecutively by the same vehicle, leading to a total of two instead of three vehicles as can be seen in Figure 8.

\[
\lceil \frac{2 \cdot \left( \frac{T}{T} + \epsilon + \frac{T}{T} - \epsilon \right)}{T} \rceil = \lceil \frac{2 \cdot T}{T} \rceil = 2.
\]

Nevertheless, it is very unlikely that this vehicle schedule is possible after the timetabling stage \(T\). Consider an OD-pair from \(v\) to \(w\). These passengers have to transfer at station \(u\) with a minimal transfer time of \(\epsilon' > 0\). Then, during the timetabling stage (Step \(T\)), the lines will be synchronized such that the passengers can transfer at station \(u\). Therefore, the vehicle schedule shown in Figure 8 will also need three vehicles:

\[
\lceil \frac{2 \cdot \left( \frac{T}{T} + \epsilon + \frac{T}{T} - \epsilon + \epsilon' \right)}{T} \rceil = \lceil \frac{2 \cdot T + 2 \cdot \epsilon'}{T} \rceil = 3.
\]

This shows that the vehicle schedule computed in Step \(S\)-first is already optimal as the vehicle schedule shown in Figure 7 is still feasible.
F. An Iterative Approach for Integrated Planning in Public Transportation

A. Schiewe, P. Schiewe
An Iterative Approach for Integrated Planning in Public Transportation
Submitted to Transportation Science, 2018.
[Schiewe and Schiewe, 2018]
An Iterative Approach for Integrated Planning in Public Transportation

Alexander Schiewe\textsuperscript{1} and Philine Schiewe\textsuperscript{1}

\textsuperscript{1}University of Göttingen, Lotzestr. 16-18, 37083 Göttingen, Germany, \{a,p\}.schiewe@math.uni-goettingen.de

Abstract

Optimization in public transport planning is an important topic of ongoing research. Traditionally, the planning process is separated hierarchically into several stages, e.g. line planning, timetabling and vehicle scheduling. Recently, integrated public transport planning, i.e., optimizing several of the planning stages simultaneously, has gained in importance as this can improve the solution quality immensely. However, since the resulting integrated problems are computationally challenging for close-to real-world instances, heuristic solutions are commonly used. We here introduce a new iterative approach for re-optimizing an existing public transport system. For this, two of the three planning stages line planning, timetabling and vehicle scheduling are fixed while the remaining one is re-optimized. To model the re-optimization, traditional approaches do not suffice and therefore new optimization problems need to be defined. We model these problems and propose solution algorithms for each stage which are theoretically analyzed. Additionally, convergence of the proposed iterative approach is discussed theoretically and computationally tested on a benchmark case study and a close-to real-world data set.

Keywords: Public Transport Planning, Line Planning, Timetabling, Vehicle Scheduling, Iterative Heuristic, Integrated Planning

1 Introduction

With rising population numbers in urban areas the need for transportation rises as well. As public transportation is a very efficient, and - compared to individually traveling by car -
environmentally friendly mode of transport, its importance is increasing. However, the supply of public transport will only increase if its quality - both from an operator’s and a passenger’s perspective - is sufficiently high. Mathematical public transport planning aims to ensure this quality at various stages of the planning process. Here, we consider three of the most important and well researched problems of public transport planning: line planning, timetabling and vehicle scheduling.

All three problems are well researched on their own. For an overview on line planning, see [Sch12], literature on timetabling can be found in [LLER11] and [BK09] contains an overview of vehicle scheduling models.

Traditionally, these problems are solved sequentially, as depicted in Figure 1.

![Sequential Approach Diagram]

figure 1: Sequential approach.

However, these problems highly depend on each other as the output of one stage is the input for the next stage. Additionally, we are interested in the overall outcome, i.e., the line plan with corresponding timetable and vehicle schedule which we call a public transport plan. Thus, our goal is to solve the following integrated problem:

**Problem 1** (Public Transport Plan). Find a line plan with a corresponding timetable and vehicle schedule such that the travel time of the passengers and the operational costs are minimized.

Recently, the focus of research concerning public transportation planning has shifted to integrated planning to harvest the benefits of integration.

An important focus is the integration of passenger routing into the single stages, see e.g. [Sch14]. This can be included in the line planning problem ([PB06, SS06, SS15a]) or the timetabling
stage ([Sie11, SS15b, GGNS16, BHK17, SS18]). The differences between route assignment which focuses on a system-optimal solution and route choice which models the passengers’ behavior more naturally are considered in [GS17].

Another topic of research is the integration of multiple of the three separate stages, i.e., line planning and timetabling, see e.g. [RN09], or timetabling and vehicle scheduling, see e.g. [Lie08, CM12], or even combining all three steps, see e.g. [LPSS18, Sch18].

But as the problems drastically increase in size and thus become even more computationally challenging, heuristic approaches to the integrated problems are more promising. Of course, the traditional sequential approach shown in Figure 1 is such a heuristic but other, more specialized heuristics often perform better.

[BBVL17] developed an iterative approach to line planning and timetabling, solving both steps sequentially. Another approach to the integration of these two problems is the usage of metaheuristics, as done in [TI14]. Both approaches are also applied to the integration of timetabling and vehicle scheduling, see [SE15, FvdHRL18] for a metaheuristic and [GH10, PLM13] for iterative approaches. Finally, there are also iterative approaches for the integration of all three problems in [MS09] and [PSS17].

**Our Contribution**

Here, we present a novel iterative heuristic for the integrated line planning, timetabling and vehicle scheduling problem, attending to the main issue with the sequential approach, i.e., the interdependence of the problems. If a line plan is fixed first and only afterwards a timetable and a vehicle schedule are constructed, this may lead to bad, or even infeasible, solutions, see [GSS13].

Therefore, we develop an iterative approach to re-optimize a given public transport plan where in each step one of the stages is re-optimized and the other ones are regarded as fixed such that a feasible solution is guaranteed, as depicted in Figure 2. For this, two completely new public transportation problems are identified and modeled. An overview can be found in Figure 2. This iterative approach specifies the three steps in the inner circle of the algorithmic scheme called eigenmodel which is introduced in [Sch17].

Of the three algorithms shown in Figure 2, only ReVehicleScheduling has been studied before, while ReLinePlanning and ReTimetabling are newly defined and discussed in Section 3.
Overview of the paper

The remainder of this paper is structured as follows: In Section 2 we formally define a public transport plan by using the classical problems line planning, timetabling and vehicle scheduling. In Section 3 we introduce the models and algorithms for the re-optimization problems where always one of the three stages is re-optimized while the other two stages are fixed. The iterative approach and some theoretical implications are presented in Section 4 while computational experiments on a benchmark data set and close-to real-world data is presented in Section 5.

2 Definition of a Public Transport Plan

In this section, we formally define the parts of a public transport plan, namely line plans, timetables and vehicle schedule, and how to measure its quality.

Note that we consider binary line frequencies in the following which is a common assumption for timetabling, see e.g. [SU89].

We assume the following data to be given. Let PTN=(V,E) be an infrastructure network or public transport network with stops or stations V and direct connections E between them. The lower and upper bounds on the wait times at stops are given as \( L_{\text{wait}} \) and \( U_{\text{wait}} \) while the lower and upper bounds on the transfer times at stops are given as \( L_{\text{trans}} \) and \( U_{\text{trans}} \). We assume that transfers are always possible, i.e.,

\[
U_{\text{trans}} = L_{\text{trans}} + T - 1.
\]

For each edge \( e \in E \) consider the length \( \text{len}_e \) and a lower and upper bound \( L_e \) and \( U_e \) on the drive time on this edge. The passenger demand is given as an OD matrix \( C = (C_{u,v})_{u,v \in V} \) where \( C_{u,v} \) represents the number of passengers traveling from \( u \) to \( v \) in the planning period.
The line plan and the timetable are periodic and the length of the planning period is $T$.

### 2.1 Line Planning

In the line planning stage, the goal is to cover the edges of the PTN by lines chosen from a line pool $\mathcal{L}^0$. A line is a path in the PTN which has to be covered by a vehicle end-to-end while a line pool is a set of lines. The length $\text{len}_l$ of a line $l$ is given by the lengths of its edges, i.e.,

$$
\text{len}_l = \sum_{e \in l} \text{len}_e.
$$

In order to facilitate reasonable travel times for the passengers, lower frequency bounds $f_{e}^{\text{min}}$ have to be satisfied for all edges $e \in E$.

Finding a line plan $\mathcal{L}$ amounts to assigning a frequency $f_l \in \{0, 1\}$ to each line $l \in \mathcal{L}^0$. We say a line $l$ is part of line plan $\mathcal{L}$ or $l \in \mathcal{L}$ if $f_l = 1$. A line plan is feasible if the following condition is satisfied for all edges $e \in E$:

$$
\sum_{l \in \mathcal{L}, \ e \in l} f_l \geq f_e^{\text{min}}.
$$

We assume that the lower frequency bounds $f_{e}^{\text{min}}$, $e \in E$, are given such that the vehicle capacity suffices for routing all passengers in every feasible line plan.

### 2.2 Timetabling

As we consider periodic timetabling that can be represented by the periodic event scheduling problem (PESP) defined in [SU89], we need an event-activity network (EAN) $\mathcal{N} = (\mathcal{E}, \mathcal{A})$. For a given line plan $\mathcal{L}$, the EAN consists of a set of events $\mathcal{E}$ which represent the arrival and departure of lines at stops and a set of activities $\mathcal{A}$ representing driving of vehicles on lines,
vehicles waiting at stops or passengers transferring at stops.

\[ E = E_{\text{arr}} \cup E_{\text{dep}} \]

\[ E_{\text{arr}} = \{(v, l, \text{arr}) : v \in l \cap V, l \in \mathcal{L}\} \]

\[ E_{\text{dep}} = \{(v, l, \text{dep}) : v \in l \cap V, l \in \mathcal{L}\} \]

\[ A = A_{\text{drive}} \cup A_{\text{wait}} \cup A_{\text{trans}} \]

\[ A_{\text{drive}} = \{(v_1, l, \text{dep}), (v_2, l, \text{arr}) : (v_1, v_2) \in E, l \in \mathcal{L}\} \]

\[ A_{\text{wait}} = \{(v, l, \text{arr}), (v, l, \text{dep}) : v \in l \cap V, l \in \mathcal{L}\} \]

\[ A_{\text{trans}} = \{(v, l_1, \text{arr}), (v, l_2, \text{dep}) : v \in l_1 \cap l_2 \cap V, l_1, l_2 \in \mathcal{L}\}. \]

Each activity \( a \in A \) has a lower and an upper bound \( L_a \) and \( U_a \), respectively. Here, the bounds on waiting or transferring at stops are derived from the corresponding PTN bounds \( L_{\text{wait}}, U_{\text{wait}} \) and \( L_{\text{trans}}, U_{\text{trans}} \), respectively, while the bounds on driving activities \( ((v_1, l, \text{dep}), (v_2, l, \text{arr})) \) are derived from the corresponding edge \( e = \{v_1, v_2\} \in E \) with bounds \( L_e, U_e \).

To find a timetable \( \pi \), a time point \( \pi_i \in \{0, \ldots, T - 1\} \) is assigned to each event \( i \in E \). The duration \( d(a) \) of activity \( a = (i, j) \in A \), is defined as

\[ d(a) = (\pi_j - \pi_i - L_a) \mod T + L_a. \]

A timetable \( \pi \) is feasible if the duration of each activities lies within its lower and upper bounds, i.e., if

\[ L_a \leq d(a) \leq U_a. \]

To evaluate the quality of a timetable, we assume that the passenger paths are fixed and for each activity \( a \in A \) the number of passengers using it is given as \( w_a \). We call \( w = (w_a)_{a \in A} \) passenger weights. These weights are determined by a routing step ahead of the optimization, e.g. by assigning to each OD pair a shortest path according to the lower bounds on the activities.

The goal of the optimization is to minimize the travel time of the passengers, i.e.,

\[ R_{\text{fix}}(\pi, w) = \sum_{a=(i,j) \in A} w_a \cdot d(a). \] (1)
2.3 Vehicle Scheduling

Vehicle scheduling for a fixed line plan and a fixed timetable is a well researched problem, see e.g. [BK09]. There exist many different variants, with or without one or multiple depots, with or without a maximal number of vehicles which can be used and with different objectives.

We here consider a model with an unlimited number of vehicles, without a depot and minimize a weighted sum of the number of vehicles, the time needed and the distance covered.

In contrast to line planning and timetabling where a plan is computed for a relatively short time span and then repeated, vehicle schedules are computed for longer time spans. For example, a timetable might repeat every hour, while the vehicle schedule is computed for the whole day and only repeated the next day. We therefore consider an aperiodic problem where each line $l$ of the line plan is to be covered $p_{\text{max}}$ times by a vehicle and the $p$-th covering of line $l$ is called trip $(p,l)$. A trip $(p,l)$ is determined by the line $l$ it covers, the period repetition $p$ it starts in, its start time $\text{start}_{p,l}$ and its end time $\text{end}_{p,l}$. The duration of trip $(p,l)$, $\text{duration}_{p,l}$, is the time between $\text{start}_{p,l}$ and $\text{end}_{p,l}$. The length of a trip $(p,l)$, $\text{len}_{p,l}$, is the length of the corresponding line $l$, i.e., $\text{len}_{p,l} = \text{len}_l$. A vehicle route is a list of compatible trips where two trips $(p_{1},l_{1})$, $(p_{2},l_{2})$ are compatible if there is sufficient time to get from the last station of line $l_{1}$ to the first station of line $l_{2}$ on a fixed shortest path $P$, i.e., if

$$\text{start}_{p_{2},l_{2}} - \text{end}_{p_{1},l_{1}} \geq L_{l_{1},l_{2}},$$

where $L_{l_{1},l_{2}}$ is the needed time to directly drive from the last stop of $l_{1}$ to the first stop of $l_{2}$ and $D_{l_{1},l_{2}}$ as the corresponding distance. We assume that

$$L_{l_{1},l_{2}} = \sum_{e \in P} L_{e}$$

$$D_{l_{1},l_{2}} = \sum_{e \in P} \text{len}_{e}$$

is satisfied. A vehicle schedule is a set of vehicle routes such that all trips in $T = \{(p,l) : p \in \{1, \ldots, p_{\text{max}}\}, l \in L\}$ are covered exactly once.

2.4 Objectives

To evaluate the quality of a public transport plan, we consider two objectives, namely the operational costs and the travel time for the passengers.
The travel time of the passengers is measured on shortest paths according to the timetable. Note that this may not be the same as the objective function of the timetabling problem as passengers can choose a new, possibly shorter, path. For this, let \( P_{u,v}(\pi) \) be a shortest path from any departure event at stop \( u \) to any arrival event at stop \( v \) w.r.t the timetable \( \pi \). We therefore measure the rerouted travel time

\[
RSP(\pi) = \sum_{(u,v) \in C} C_{u,v} \cdot \sum_{a \in P_{u,v}(\pi)} d(a).
\]

It is also possible to instead measure the perceived travel time where transfers are penalized by a fixed penalty term. This can easily be added to the models presented here by modifying the duration of transfer activities. For easier notation, we consider travel time in the remainder of this paper.

The operational costs are determined by the vehicle schedule and include duration based costs \( \text{cost}_{\text{time}} \), distance based costs \( \text{cost}_{\text{len}} \) and costs per vehicle \( \text{cost}_{\text{veh}} \). In addition to the distance and time needed to cover the trips, vehicles also have to relocate between trips. In order to compute the costs of this relocation, we define connecting trips. Let \( r = ((p_1,l_1),(p_2,l_2),\ldots,(p_n,l_n)) \) be a vehicle route. Then for each \( i \in \{1,\ldots,n-1\} \) the tuple \( ((p_i,l_i),(p_{i+1},l_{i+1})) \) is called a connecting trip. The duration of connecting trip \( c_i = ((p_i,l_i),(p_{i+1},l_{i+1})) \), \( \text{duration}_{c_i} \), is the time between the end of trip \( (p_1,l_1) \) and the start of trip \( (p_2,l_2) \) and its length, \( \text{len}_{c_i} \), is \( D_{l_i,l_{i+1}} \), i.e., the distance to cover when driving from \( l_i \) to \( l_{i+1} \).

Let \( V = \{r_1,\ldots,r_n\} \) be a vehicle schedule with vehicle routes \( r_i \). Then the operational costs of \( V \) are
\[
\text{cost}(V) = \sum_{r \in V} \left( \sum_{\text{trip } t = (p, l) \in r} \text{cost}_\text{len} \cdot \text{len}_t + \text{cost}_\text{time} \cdot \text{duration}_t \right) \\
+ \sum_{\text{connecting trip } c = ((p_1, l_1), (p_2, l_2)) \in r} \text{cost}_\text{len} \cdot \text{len}_c + \text{cost}_\text{time} \cdot \text{duration}_c \\
+ \text{cost}_\text{veh} \cdot |V| \\
= \sum_{r \in V} \left( \sum_{\text{trip } (p, l) \in r} \text{cost}_\text{len} \cdot \text{len}_t + \text{cost}_\text{time} \cdot (\text{end}_{p, l} - \text{start}_{p, l}) \right) \\
+ \sum_{\text{connecting trip } ((p_1, l_1), (p_2, l_2)) \in r} \left( \text{cost}_\text{len} \cdot D_{l_1, l_2} \\
+ \text{cost}_\text{time} \cdot (\text{start}_{p_2, l_2} - \text{end}_{p_1, l_1}) \right) \\
+ \text{cost}_\text{veh} \cdot |V|. 
\]

3 Modelling the Re-Optimization Problems

In this section, we define the re-optimization problems ReVehicleScheduling, ReTimetabling, and ReLinePlanning that we need for the iterative approach. For a given public transport plan, our goal is to always fix the solutions of two of the three stages line planning, timetabling, and vehicle scheduling while re-optimizing the third stage.

3.1 Re-Optimizing the Vehicle Schedule

As mentioned in Section 2.3, vehicle scheduling for a fixed line plan and a fixed timetable is part of the classical sequential planning process and a well researched problem. Therefore, we can use a standard vehicle scheduling model for ReVehicleScheduling. Here, we use a vehicle scheduling model without depot and we minimize the operational costs as defined in Section 2.4. The algorithm used for the experimental evaluation is implemented in the open source software tool LinTim, see [SAP+18].
Problem 2 (ReVehicleScheduling). Given a public transport plan \((\mathcal{L}, \pi, V)\) with line plan \(\mathcal{L}\), periodic timetable \(\pi\) and vehicle schedule \(V\) covering \(p_{\max}\) period repetitions. Let \(L_{l_1,l_2}, l_1,l_2 \in \mathcal{L}\), be the minimal durations of the potential connecting trips and \(D_{l_1,l_2}, l_1,l_2 \in \mathcal{L}\), the lengths of the potential connecting trips. Let \((\text{cost}_{\text{time}}, \text{cost}_{\text{len}}, \text{cost}_{\text{veh}})\) be given cost parameters.

Find a new feasible vehicle schedule \(V'\) for timetable \(\pi\), minimal durations of connecting trips \(L_{l_1,l_2}, l_1,l_2 \in \mathcal{L}\), and trips \(\mathcal{T} = \{(p,l): p \in \{1,\ldots,p_{\max}\}, l \in \mathcal{L}\}\) such that the operational costs \(\text{cost}(V')\) are minimized.

3.2 Re-Optimizing the Timetable

So far, we only described the standard timetabling problem. As mention in Section 2.2, a timetable which is feasible already adheres to the line plan, as it is part of the input and the structure of the EAN. To achieve that also a given vehicle schedule \(V\) stays feasible after a new timetable is found, we need to add further constraints.

Therefore, we consider the set \(\mathcal{C}\) of all connecting trips of vehicle routes in \(V\). Remember that connecting trip \(c = ((p_1,l_1),(p_2,l_2)) \in \mathcal{C}\) means that trip \((p_2,l_2)\) is operated directly after trip \((p_1,l_1)\) by the same vehicle. In order to check that the vehicle schedule remains feasible, we need to ensure that the minimal time \(L_{l_1,l_2}\) between trips on lines \(l_1\) and \(l_2\) is complied with for all connecting trips \(c = ((p_1,l_1),(p_2,l_2)) \in \mathcal{C}\).

An important factor is the distribution of passengers to activities of the event-activity network, especially when the event-activity network is modified during the iteration scheme. Thus the passenger weights \(w = (w_a)_{a \in A}\) have to be determined before applying Algorithm ReTimetabling by a passenger routing. We choose to route the OD pairs on shortest paths in the EAN according to the previous timetable which allows for a convergence result later on.
Problem 3 (ReTimetabling). Given a public transport plan \((\mathcal{L}, \pi, \mathcal{V})\) with line plan \(\mathcal{L}\), periodic timetable \(\pi\) for period length \(T\) and bounds \(L_a, U_a\) on the activities \(a \in A\) of the corresponding EAN \(\mathcal{N} = (\mathcal{E}, A)\) and vehicle schedule \(\mathcal{V}\). Let \(L_{l_1,l_2}, (\(p_1,l_1\),\(p_2,l_2\)) \in r, r \in \mathcal{V}\), be the minimal durations of the connecting trips. Let \(w = (w_a)_{a \in A}\) be passenger weights corresponding to a passenger routing on shortest paths according to timetable \(\pi\).

Find a new periodic timetable \(\pi'\) that is feasible corresponding to the minimal and maximal bounds on the activities as well as the minimal times for the connecting trips and minimizes the travel time of the passengers for fixed weights \(w = (w_a)_{a \in A}\).

**IP Formulation** To give an integer program for the problem ReTimetabling we adapt the classical PESP formulation and use the following variables. Let \(\pi_i \in \{0, \ldots, T - 1\}\) be the scheduled periodic time of event \(i \in \mathcal{E}\), \(z_a \in \mathbb{Z}\) the modulo parameter of activity \(a \in A\) and \(\text{duration}_l \in \mathbb{N}\) the time it takes in the timetable to get from \(\text{first}(l)\) to \(\text{last}(l)\). Here, \(\text{first}(l)\) is the first event in line \(l\) while \(\text{last}(l)\) is the last event in line \(l\). For easier notation we define variables \(\text{start}_{p,l} \in \mathbb{N}\) for the start time of trip \((p, l)\) and \(\text{end}_{p,l} \in \mathbb{N}\) for its end time. Let \(A(l)\) be the activities belonging to line \(l\), i.e., all activities \(a = (i, j)\) where both events \(i\) and \(j\) are departure or arrival events of line \(l\). Then we get the following IP formulation.
(ReTimetabling) \[
\text{min} \sum_{a=(i,j) \in A} w_a \cdot (\pi_j - \pi_i + z_a \cdot T)
\]
s.t. \[
\pi_j - \pi_i + z_a \cdot T \leq U_a \quad a = (i,j) \in A
\]
\[
\pi_j - \pi_i + z_a \cdot T \geq L_a \quad a = (i,j) \in A
\]
\[
\text{duration}_l = \sum_{a=(i,j) \in A(l)} (\pi_j - \pi_i + z_a \cdot T) \quad l \in \mathcal{L}
\]
\[
\text{start}_{p,l} = p \cdot T + \pi_{\text{first}(l)} \quad (p,l) : (\bullet,(p,l)) \in \mathcal{C}
\]
\[
\text{end}_{p,l} = p \cdot T + \pi_{\text{first}(l)} + \text{duration}_l \quad (p,l) : ((p,l),\bullet) \in \mathcal{C}
\]
\[
L_{l_1,l_2} \leq \text{start}_{p_2,l_2} - \text{end}_{p_1,l_1} \quad ((p_1,l_1),(p_2,l_2)) \in \mathcal{C}
\]
\[
\pi_i \in \{0, \ldots, T - 1\} \quad i \in \mathcal{E}
\]
\[
z_a \in \mathbb{Z} \quad a \in \mathcal{A}
\]
\[
\text{duration}_l \in \mathbb{N} \quad l \in \mathcal{L}
\]
\[
\text{start}_{p,l} \in \mathbb{N} \quad (p,l) : (\bullet,(p,l)) \in \mathcal{C}
\]
\[
\text{end}_{p,l} \in \mathbb{N} \quad (p,l) : ((p,l),\bullet) \in \mathcal{C}
\]

Constraints (2) and (3) are the standard timetabling constraints while equation (4) determines the time it takes to traverse line \(l \in \mathcal{L}\). Equations (5) and (6) determine the actual start and end times of trip \((p,l) \in \mathcal{L}, r \in \mathcal{V}\), respectively. Note that to determine \(\text{end}_{p,l}\) it is not sufficient to use the time of \(\text{last}(l)\) for period repetition \(p\) as the duration of the traversal of \(l\) can be longer than the period length \(T\), see Example 5. Constraint (7) makes sure that the minimal time for connecting trips is complied with.

Remark 4. The given IP formulation can easily be extended to the integrated timetabling and vehicle scheduling problem, by making the vehicle connecting trips variable and adding corresponding flow constraints which makes the problem substantially larger. For details, see [LPSS18, Sch18].

Example 5 ([Sch18]). Consider two lines \(l_1, l_2\) with \(L_{l_1,l_2} = L_{l_2,l_1} = 5\). Let the trip length of \(l_1\) which is determined by the bound of the activities belonging to \(l_1\) be in \([60, 120]\) and the trip length of \(l_2\) be fixed to 50 with a planning period of length 60. A possible timetable is given in
Depending on the actual duration of line $l_1$ which might be 60 or 120, we need to implement two different vehicle schedules. If the duration is 60, we can find a vehicle schedule with two vehicles. Vehicle $V_1$ operates trips $(1, l_1), (2, l_2), (3, l_1)$ etc. and Vehicle $V_2$ operates trips $(1, l_2), (2, l_1), (2, l_2)$ etc. But if the duration is 120, the vehicle operating $(1, l_1)$ cannot operate $(2, l_2)$ and we need a third vehicle to cover all trips although the periodic difference between $\text{last}(l_1)$ and $\text{first}(l_2)$ is large enough to accommodate a connecting trip.

![Figure 3](image-url)

**Figure 3:** A possible timetable for Example 5.

### 3.3 Re-Optimizing the Line Plan

For defining the problem $\text{ReLinePlanning}$, we first need to understand how to generate new lines that are consistent with the timetable and the vehicle schedule which are already in place. As lines define a physical path that has to be covered by one vehicle end-to-end, they are an integral part of both the vehicle schedule and the timetable. As lines have to appear periodically, we have to make sure that a path can only be a line if it is covered by one vehicle end-to-end in each planning period at the same periodic time. This is especially difficult as we consider the general case of aperiodic vehicle schedules instead of periodic ones as it is done, e.g. in [DBRB+17, BKLL18].

For formally defining when lines are consistent with a given timetable and vehicle schedule, let $r = ((p_1, l_1), \ldots, (p_n, l_n))$ be a vehicle route. As every connecting trip between two trips $(p_i, l_i), (p_{i+1}, l_{i+1})$ is operated on a fixed shortest path, we can determine the physical path of the vehicle, i.e., the path the vehicle takes in the PTN, which we call $P(r)$. For an edge $e \in (p, l)$ with $l = (l', e, l'')$ we determine the aperiodic departure time as

$$
\tau_{(e,p,l)} = p \cdot T + \sum_{v \in P \cap V} \text{duration}((v, \text{arr}, l), (v, \text{dep}, l)) + \sum_{(u,v) \in P \cap E} \text{duration}((u, \text{dep}, l), (v, \text{arr}, l)).
$$
Note that due to Example 5 we cannot simply compute the aperiodic departure time of $e$ by adding $p \cdot T$ to the periodic departure time of $e$.

Let $c = ((p_1, l_1), (p_2, l_2))$ be a connecting trip with path $(e_1, \ldots, e_k)$. Note that due to our assumptions this path is a fixed shortest path from the last station of line $l_1$ to the first station of line $l_2$. For an edge $e_j \in (e_1, \ldots, e_k)$, we define the departure time as

$$
\tau_{(e_j, c)} = p \cdot T + \text{duration} + \sum_{i=1}^{j-1} \text{duration}(e_i, c).
$$

Here, $\text{duration}(e_i, c)$ is the duration of the edge in the connecting trip, i.e., the time the vehicle takes to cover $e_i$. These durations have to satisfy

$$
\text{duration}(e_i, c) \geq L_{e_i}, \quad i \in \{1, \ldots, k\} \quad (8)
$$

$$
\sum_{i=1}^{k} \text{duration}(e_i, c) = \text{duration}_c. \quad (9)
$$

As changing lines influences the basic level of the corresponding timetable and vehicle schedule, lines cannot even change names without formally changing the timetable and vehicle schedule as lines are used for encoding events and trips. Therefore, we slightly adapt the timetable and the vehicle schedule for a new line plan without changing the physical routes of vehicles during the operation of trips and without changing the times of events that are covered by the new line plan. We thus define consistency of transport plans which are derived from one another by changing the line plan.

**Definition 6.** Let $(L, \pi, V)$ be a public transport plan that is feasible according to upper and lower activity bounds derived from the corresponding PTN bounds $L_e, U_e, e \in E, L_{\text{wait}}, U_{\text{wait}}, L_{\text{trans}}, U_{\text{trans}}$. Let $L_{l_1, l_2}, l_1, l_2 \in L$ be the minimal durations of the potential connecting trips. A public transport plan $(L', \pi', V')$ is consistent with $(L, \pi, V)$, if the following conditions are satisfied.

- $L'$ is a set of lines with corresponding timetable $\pi'$ and vehicle schedule $V'$ which are feasible according to upper and lower activity bounds derived from the corresponding PTN bounds and the minimal times for connecting trips.
- There exists a bijection $b: V \to V'$.
- For all vehicle routes $r \in V$ the paths of all trips in $b(r)$ are contained in the path $P(r)$,
i.e., the new vehicle routes cover the same paths as the old vehicle routes when operating trips but might deviate from them for connecting trips. For an edge \( e \) contained in trip \((p, l) \in r \) and in a trip \((p', l') \in b(r) \) at the same part of the vehicle route, we denote \((p', l')\) as \( b'(e, p, l) \). Analogously, for an edge \( e \) contained in connecting trip \( c \in r \) and in a trip \((p', l') \in b(r) \) at the same part of the vehicle route, we denote \((e, c)\) as \( b(e, p', l'). \)

- For all edges \( e \) contained in a trip \((p, l) \) in vehicle route \( r \) and in a trip \( b'(e, p, l) = (p', l') \) in vehicle route \( b(r) \) the aperiodic departure times coincide, i.e., \( \tau(e, l, p) = \tau(e, l', p) \).

- There have to be durations \( \text{duration}(e, c), e \in c, c \in r, r \in V, \) according to (8) and (9) such that the following condition is satisfied: Let \((e_1, \ldots, e_k) \subset l'\) be the largest subpath of \((p', l') \) in vehicle route \( b(r) \) that is completely contained in \( c \). Then the aperiodic departure times \( \tau(e_i, g, p') \) satisfy

\[
\tau(e_k, g', p') - \tau(e_1, g', p') = \sum_{i=1}^{k} \text{duration}(\tilde{b}(e_i, p', l')).
\]

i.e., the duration of connecting trip \( c \) allows for the operation of line \( l' \).

With this definition, we call a line \( l \) consistent with a public transport plan \((L, \pi, V)\) if there exists a public transport plan \((\{l\}, \pi', V')\) that is consistent with \((L, \pi, V)\). If a line \( l \) is consistent to \((L, \pi, V)\), the following requirements have to be satisfied as direct implications of Definition 6.

- Line \( l \) is operated periodically and all corresponding activity durations are feasible as \( \pi' \) is a feasible periodic timetable.

- Line \( l \) is covered by one vehicle end-to-end in each planning period as \( V' \) is a feasible vehicle schedule.

- For each trip \((p, l), p \in \{1, \ldots, p_{\text{max}}\}\), the path of line \( l \) is part of an old vehicle route due to bijection \( b \).

- The departures times at stations that have formerly also been part of a line are the same as before due to the constraints on the aperiodic departure times.

- The duration of the parts of the line that have formerly been connecting trips fit to the duration of the connecting trip.
To ensure a certain service level for the passengers when minimizing the costs of the new line concept, we use the standard line planning constraints, i.e., we consider fixed minimal frequencies on all PTN edges as described in Section 2.1.

As the operational costs do not only depend on the line plan, we approximate them by using costs per line as it is commonly done in line planning, see e.g. [CvDZ98]. We determine the line costs by using a fixed cost part, a part depending on the length of the edges and one depending on the number of edges, as done e.g. in [GHS17]. The costs of the line plan are therefore

\[ \text{cost}(L) = \sum_{l \in L} \text{cost}_l. \]  

(10)

The problem \texttt{ReLinePlanning} can now be stated as follows.

**Problem 7 (ReLinePlanning).** Given a public transport plan \((L, \pi, V)\) for PTN \((V, E)\) with line plan \(L\) with minimal edge frequencies \(f_e^{\min}, e \in E\), duration bounds \(L_e, U_e, e \in E, L_{\text{wait}}, U_{\text{wait}}, L_{\text{trans}}, U_{\text{trans}}\), periodic timetable \(\pi\) for period length \(T\) and vehicle schedule \(V\) for \(p_{\text{max}}\) period repetitions. Let \(L_{l_1, l_2}, l_1, l_2 \in L\), be the minimal durations of the potential connecting trips.

Find a new public transport plan \((L', \pi', V')\) that is consistent with \((L, \pi, V)\) and minimizes the line costs \(\text{cost}(L')\).

In order to find a new line plan, we first need to create a line pool consisting of lines that are consistent with the original public transport plan. In a second step, we chose a line plan from this pool that can be extended to a public transport plan consistent with the original one. Both steps are described in Algorithm 1.
Algorithm 1 ReLinePlanning

1: **Input:** PTN=(V,E), lower frequency bounds $f_{e_{min}}$, $e \in E$, lower and upper duration bounds $L_e, U_e$, $e \in E$, $L_{\text{wait}}, U_{\text{wait}}, L_{\text{trans}}, U_{\text{trans}}$, period length $T$, number of period repetitions $p_{\text{max}}$, minimal times for potential empty trips $L_{l_1}, l_2$, $l_1, l_2 \in \mathcal{L}$, public transport plan $(\mathcal{L}, \pi, \mathcal{V})$ with $V = \{r_1, \ldots, r_n\}$ and vehicle $V_i$ operating route $r_i$.

2: **Output:** A public transport plan $(\mathcal{L}', \pi', \mathcal{V}')$ consistent to $(\mathcal{L}, \pi, \mathcal{V})$.

3: \(\triangleright\) Define line network.

4: Initialize line network $L = (V_L, E_L)$ with $V_L = V$, $E_L = \emptyset$.

5: for route $r_i \in V$ do

6: for trip edges $e \in (p, l)$, $(p, l) \in r_i$ do

7: \(\triangleright\) Add edge $e$ labeled by aperiodic departure time and vehicle.

8: $E_L = E_L \cup \{(e, \tau(e, p, l), V_i)\}$

9: end for

10: Fix durations $\text{duration}(e, c)$, $e \in c$, $c \in r_i$ satisfying (8) and (9).

11: for connecting trip edges $e_j \in c$, $e \in r_i$ with $c = (e_1, \ldots, e_k)$, $e_j = (u, v)$ do

12: \(\triangleright\) Add edge $e_j$ labeled by aperiodic departure time and vehicle id if it can be used by passengers.

13: if $\tau(e_{j+1}, c) - \tau(e_j, c) \in [L_{e_j} + L_{\text{wait}}, U_{e_j} + U_{\text{wait}}]$ then

14: $E_L = E_L \cup \{(e_j, \tau(e_j, c), V_i)\}$

15: end if

16: end for

17: end for

18: end for
19. Define collapsed line network
20. Initialize collapsed line network \( C = (V_C, E_C) \) with \( V_C = V \), \( E_C = \emptyset \).
21. for \((e, \tau, V_i) \in E_L\) with \( \tau \in \{T, \ldots, 2 \cdot T - 1\} \) do
22. \( \triangleright \) Combine parallel edges from the line network
23. \( \triangleright \) with the same periodic departure time.
24. \( E_L = E_L \setminus \{(e, \tau, V_i)\}, \ \text{VehList=[V]}_i\), \( E_{\text{temp}} = \emptyset \).
25. for \( p = 1, \ldots, p_{\text{max}} - 1 \) do
26. if \( \exists (e, \tau + p \cdot T, V_k) \in E_L \) then
27. \( \text{VehList=[VehList, V]}_k\), \( E_{\text{temp}} = E_{\text{temp}} \cup \{(e, \tau + p \cdot T, V_k)\} \)
28. else
29. Start next iteration in line 21.
30. end if
31. end for
32. \( E_L = E_L \setminus E_{\text{temp}}, E_C = E_C \cup \{(e, \tau \mod T, \text{VehList})\} \)
33. end for
34. \( \triangleright \) Construct line pool.
35. Find set of longest paths \( P \) in collapsed line network \( C \), s.t. all edges in a path have identical labels \( \text{VehList} \) and the departure times of two consecutive edges \((e_1 = (u, v), \pi_1, \text{VehList}), (e_2 = (v, w), \pi_2, \text{VehList})\) satisfy
\[
(\pi_2 - \pi_1 - L_{e_1} - L_{\text{wait}}) \mod T + L_{e_1} + L_{\text{wait}} \in [L_{e_1} + L_{\text{wait}}, U_{e_1} + U_{\text{wait}}].
\]
36. Set the line pool \( \mathcal{L}^0 \) as the set of all subpaths of \( \mathcal{P} \).
37. Find a line plan \( \mathcal{L}' \) by solving a line planning problem for pool \( \mathcal{L}^0 \) such that
38. all PTN edges are covered according to the lower frequency bounds \( f_{e_{\text{min}}} \),
39. all edges \( e \in E_C \) are part of at most one line in \( \mathcal{L}' \)
40. and the line costs are minimized.
41. \( \triangleright \) Find the corresponding timetable and vehicle schedule.
42. Construct timetable \( \pi' \) and vehicle schedule \( \mathcal{V}' \) by using the periodic times from the collapsed line network for the departure times, adding the corresponding arrival times and updating the vehicle routes according to the new lines.
The functionality of Algorithm 1 is demonstrated in the following Example 8.

**Example 8.** We consider the PTN shown in Figure 4, consisting of five nodes and six edges. There are three lines with their corresponding periodic timetable given. The first number stands for the arrival time of the line in the specified station, the second one for the departure time.

Lines

\[
\begin{align*}
& l_1 = (n_1[00’, 05’], n_2[15’, 20’], n_3[25’, 30’]) \\
& l_2 = (n_3[30’, 35’], n_5[40’, 45’], n_4[55’, 00’]) \\
& l_3 = (n_1[00’, 05’], n_4[20’, 25’], n_3[35’, 40’])
\end{align*}
\]

Figure 4: PTN and line plan.

The next figure, Figure 5, shows the vehicle schedule which consists of two vehicle routes. The first vehicle \( V_1 \) operates line \( l_1 \) and line \( l_2 \) alternately while the second vehicle \( V_2 \) operates only line \( l_3 \).

Vehicle \( V_1 \):

\[
\begin{align*}
& l_1 [01:00, 01:30], l_2 [01:30, 02:00] \\
& l_1 [02:00, 02:30], l_2 [02:30, 03:00] \\
& l_1 [03:00, 03:30], l_2 [03:30, 04:00]
\end{align*}
\]

Vehicle \( V_2 \):

\[
\begin{align*}
& l_3 [01:00, 01:40], \emptyset [01:40, 02:00] \\
& l_3 [02:00, 02:40], \emptyset [02:40, 03:00] \\
& l_3 [03:00, 03:40], \emptyset [03:40, 04:00]
\end{align*}
\]

Figure 5: Vehicle schedule.

From this information we now create the line network shown in Figure 6a. Here, we see each driving of a PTN edge marked by the vehicle id and the starting time for the three period repetitions we are looking at where the period length is 60 minutes.

The collapsed line network is shown in Figure 6b. Here, the periodic drivings are shown, marked by the periodic departure time and the corresponding list of vehicles. Note that a vehicle list does not have to consist of only one vehicle, as is the case in this simple example, but could also consist of different vehicles.
Figure 6: Line networks for Example 8.

The last figure, Figure 7, shows which edges of the collapsed line network can be joined to a new line. We get the old line \( l_3 \) as \( l_1^1 \) and all its subpaths as well as a new line \( l_1^2 \) with its subpaths in which the old lines \( l_1 \) and \( l_2 \) are contained.

Figure 7: Coinciding labels.

The line pool generation is now complete and it remains to find a cost-minimal line concept based on this new line pool.

In the following theorem we show that Algorithm 1 finds a public transport plan that is consistent with the public transport plan \((L, \pi, V)\) used as input.

**Theorem 9.** The public transport plan \((L', \pi', V')\) constructed by Algorithm 1 is consistent
with the public transport plan \((L, \pi, V)\) used as input and line plan \(L'\) is feasible w.r.t the lower frequency bounds.

**Proof.** The construction of the line network in lines 4 to 18 assigns an aperiodic departure time for each PTN edge \(e \in P(r)\) covered by vehicle route \(r \in V\) that can be part of a trip according to the lower and upper bounds. In the collapsed line network constructed in line 20 to 33 these aperiodic coverings of edges are accumulated to a periodic one if the edge is covered in each period repetition at the same periodic time point. These collapsed edges are labeled by the list of vehicles which cover them in each period repetition. The construction of the paths in line 35 guarantees that each line is covered by one vehicle end-to-end in each planning period and that the corresponding timetable is feasible as transfers pose no restriction due to Section 2. Additionally, line concept \(L'\) is feasible as the minimal frequencies are respected due to line 38. It remains to show that the new vehicle schedule \(V'\) is feasible, that there exists a bijection \(b: V \to V'\) of the vehicle routes and that the trips of \(b(r)\) are part of the path \(P(r)\) fitting to the duration of the connecting trips if applicable. As bijection \(b\) we map route \(r_i\) of vehicle \(V_i\) to the new route of vehicle \(V_i\). Here, the new route of \(V_i\) consists of trips \((p, l)\) where line \(l\) corresponds to a path in the collapsed line network with label VehList where vehicle \(V_i\) starts in period repetition \(p\). This correspondence is unique as each edge \((e, \pi_i, \text{VehList})\) of the collapsed line network can only be part of one line, see line 39, and the covering of a PTN edge by Vehicle \(V_i\) in period repetition \(p\), represented by line network edge \((e, \pi_i + p \cdot T, V_i)\), can only be part of one edge \((e, \pi_i, \text{VehList})\) of the collapsed line network, see line 32.

The construction of the collapsed line network also guarantees that all trips \((p, l)\) in vehicle route \(b(r)\) are part of \(P(r)\) and that the corresponding aperiodic times coincide. The duration of trips that are part of an old connecting trip is fitting to the durations fixed in line 10 and therefore satisfies (8) and (9). The duration of connecting trips \(((p_1, l_1), (p_2, l_2)) \in b(r), r \in V\) is feasible as well: Let \(v_1\) be the last station of line \(l_1\) and \(v_2\) the first station in line \(l_2\). Then there is a \(v_1 - v_2\) path \(P_{v_1, v_2}\) which is part of \(P(r)\). Covering \(P_{v_1, v_2}\) in vehicle route \(r\) takes at least as long as \(L_{l_1, l_2}\) which is defined as the length of the shortest \(v_1 - v_2\) paths in the PTN according to the lower bounds on the drive times. Therefore, the trips \((p_1, l_1)\) and \((p_2, l_2)\) are compatible and the vehicle schedule \(V'\) is feasible as well.

To prove that this line concept is also cost-minimal under a technical assumption, we start by showing that the line pool constructed in Algorithm 1 contains all consistent lines.
Lemma 10. Let the duration of the edges in connecting trips in $V$ be uniquely determined by (8) and (9) and let for each edge $e \in E$ the aperiodic departure times $\tau_{(e,p,l)}$, $\tau_{(e,c)}$ be unique for all trips $(p,l) \in V$ with $e \in (p,l)$ and connecting trips $c \in V$ with $e \in c$, i.e., there is a most one departure using edge $e$ at any point in time. Then all lines that are consistent with the public transport plan $(L, \pi, V)$ used as input are in the line pool $L^0$ constructed in Algorithm 1.

Proof. Note that due to the fixed duration of edges in connecting trips, the aperiodic departure times of edges in connecting trips can be uniquely determined. Due to the uniqueness of the departure times, the collapsed line network constructed in lines 20 to 33 is unique as well and thus especially the labels VehList.

Let $l$ be a line that is not in $L^0$, i.e., that is not constructed in line 36. We show that this line $l$ is not consistent with $(L, \pi, V)$.

At first we consider the case where each edge $e_i \in l$ corresponds to an edge $(e_i, \pi_i, \text{VehList}_i)$ in $E_C$. As $l \notin L^0$ there either is no common label VehList for all edges $e_i \in l$ or the periodic departure times of two consecutive edges do not fit to the lower and upper bounds. As the aperiodic departure times of all edges are unique, the list of vehicles operating this edge in each planning period is unique and found by Algorithm 1. Therefore, differing labels for different edges show that line $l$ is not covered by one vehicle end-to-end in each period repetition, i.e., the line is not consistent with $(L, \pi, V)$. If the periodic departure times do not fit to the lower and upper bounds, the corresponding timetable $\pi'$ is not feasible, i.e., line $l$ is not consistent with $(L, \pi, V)$.

We therefore only have to consider the case where at least one edge $e \in l$ has no corresponding edge in $E_C$. Due to the uniqueness of the aperiodic departure times, this means that for edge $e$ there is no departure in each period repetition at the same periodic time. Thus, edge $e$ cannot be part of a line consistent with public transport plan $(L, \pi, V)$. □

Using Theorem 9 and Lemma 10, we show that the line plan constructed by Algorithm 1 is cost-minimal.

Theorem 11. Let the duration of the edges in connecting trips in $V$ be uniquely determined by (8) and (9) and let for each edge $e \in E$ the aperiodic departure times $\tau_{(e,p,l)}$, $\tau_{(e,c)}$ be unique for all trips $(p,l) \in V$ with $e \in (p,l)$ and connecting trips $c \in V$ with $e \in c$, i.e., there is a most one departure using edge $e$ at any point in time. Then Algorithm 1 finds a public transport plan $(L', \pi', V')$ that is consistent with the public transport plan $(L, \pi, V)$ used as input such that line
plan \( \mathcal{L}' \) is feasible w.r.t the lower frequency bounds and minimizes the line costs (10).

Proof. Due to Theorem 9, the public transport plan \((\mathcal{L}', \pi', V')\) found by Algorithm 1 is consistent with \((\mathcal{L}, \pi, V)\) and line plan \( \mathcal{L}' \) is feasible according to the lower frequency bounds. The line pool which is used for the optimization problem contains all consistent lines according to Lemma 10. Therefore, it only remains to show that the constraints of the optimization problem posed in lines 38 to 39 of Algorithm 1 are necessary.

The constraints posed in line 38 are necessary to ensure that \( \mathcal{L}' \) is feasible w.r.t the lower frequency bounds. The constraints posed in line 39 are needed to ensure a bijection between the old and the new vehicle routes, i.e., they are necessary to guarantee a consistent line plan. Thus, the line plan constructed by Algorithms 1 is cost-optimal for all feasible line plans that can be extended to a consistent public transport plan.

To show the optimality of the line plan constructed in Algorithm 1 we need two technical assumptions, namely that the duration of edges in connecting trips is unique and that for any edge there is at most one departure at any given point in time. The second assumption is easy to ensure by headway activities and is satisfied for realistic instances due to security concerns. On the other hand, the first assumption is unlikely to be satisfied for realistic instances as it allows for no buffer times in connecting trips. If it is not satisfied, the solution quality of Algorithm 1 depends on the durations fixed in line 10.

4 Iteration Scheme

As described in [Sch17], the re-optimization problems defined in Section 3 can be used in an iterative scheme to modify an existing public transport plan. In theory, the three algorithms \texttt{ReLinePlanning}, \texttt{ReTimetabling} and \texttt{ReVehicleScheduling} can be used in any order. However, not all concatenations of algorithms lead to improvements. In this section, we investigate the influence of different iteration schemes on both the passenger-oriented and the cost-oriented objective of the resulting public transport plan as described in Section 2.4. Remember that the passenger-oriented objective is to minimize the travel time of all passengers on shortest paths according to the timetable while the costs-oriented objective is to minimize the operational costs of the corresponding vehicle schedule.

At first, we consider the influence of the individual algorithms on the travel time and the operational costs. The influence of Algorithm \texttt{ReVehicleScheduling} can be determined most
Lemma 12. Let \((L, \pi, V)\) be a public transport plan and \((L', \pi', V')\) the public transport plan after applying Algorithm \texttt{ReVehicleScheduling} to \((L, \pi, V)\). Then the operational costs do not increase and the travel time is unchanged, i.e.,

\[
\text{cost}(V') \leq \text{cost}(V)
\]

\[
\mathcal{R}_{\text{SP}}(\pi') = \mathcal{R}_{\text{SP}}(\pi).
\]

Proof. Note that \texttt{ReVehicleScheduling} does not change the line plan or the timetable, i.e., \(L' = L\) and \(\pi' = \pi\). Therefore, we get \(\mathcal{R}_{\text{SP}}(\pi') = \mathcal{R}_{\text{SP}}(\pi)\). Additionally, \texttt{ReVehicleScheduling} minimizes the operational costs and as \(V\) is a feasible solution of \texttt{ReVehicleScheduling} we get \(\text{cost}(V') \leq \text{cost}(V)\).

Algorithm \texttt{ReTimetabling} has a clear effect on the travel time while its effect on the operational costs depends on their composition.

Lemma 13. Let \((L, \pi, V)\) be a public transport plan and \((L', \pi', V')\) the public transport plan after applying Algorithm \texttt{ReTimetabling} to \((L, \pi, V)\). Then the travel time does not increase, i.e.,

\[
\mathcal{R}_{\text{SP}}(\pi') \leq \mathcal{R}_{\text{SP}}(\pi).
\]

If the duration based costs are neglected, i.e., for \(\text{cost}_{\text{time}} = 0\), the operational costs are not changed, i.e.,

\[
\text{cost}(V') = \text{cost}(V).
\]

Proof. Note that Algorithm \texttt{ReTimetabling} does not change the line plan, i.e., \(L' = L\) and the composition of the vehicle routes in \(V'\) is the same as in \(V\). However, the start and end times of trips and connecting trips may change. \(\mathcal{R}_{\text{SP}}(\pi)\) evaluates the travel time of all passengers on shortest path w.r.t timetable \(\pi\) and Algorithm \texttt{ReTimetabling} sets the passenger weights \(w\) according to the same paths. As Algorithm \texttt{ReTimetabling} optimizes the travel time of the passengers on these fixed paths, i.e.,
$R_{fix}(\pi', w)$, and $\pi$ is a feasible solution, we get

$$R_{SP}(\pi) \geq R_{fix}(\pi', w).$$

By rerouting the passenger on optimal routes according to timetable $\pi'$ we get

$$R_{SP}(\pi) \geq R_{fix}(\pi', w) \geq R_{SP}(\pi').$$

When evaluating the costs of a public transport plan without regarding the duration-based costs and without depots, we get

$$\text{cost}(\mathcal{V}) = \sum_{r \in \mathcal{V}} \text{cost}_{\text{len}} \cdot \left( \sum_{\text{trip} \in r} \text{len}_t + \sum_{\text{connecting trip} \in r} \text{len}_c \right) + \text{cost}_{\text{veh}} \cdot |\mathcal{V}|.$$ 

As the composition of the vehicle routes in $\mathcal{V}$ and $\mathcal{V}'$ are the same, i.e., they contain the same trips and the same connecting trips, we get

$$\text{cost}(\mathcal{V}) = \sum_{r \in \mathcal{V}} \text{cost}_{\text{len}} \cdot \left( \sum_{\text{trip} \in r} \text{len}_t + \sum_{\text{connecting trip} \in r} \text{len}_c \right) + \text{cost}_{\text{veh}} \cdot |\mathcal{V}| = \sum_{r \in \mathcal{V}'} \text{cost}_{\text{len}} \cdot \left( \sum_{\text{trip} \in r} \text{len}_t + \sum_{\text{connecting trip} \in r} \text{len}_c \right) + \text{cost}_{\text{veh}} \cdot |\mathcal{V}'| = \text{cost}(\mathcal{V}).$$

Example 14 shows that for positive duration based costs, i.e., for $\text{cost}_{\text{time}} > 0$, the operational costs can be increased by Algorithm ReTimetabling.

**Example 14.** Consider an event-activity network as given in Figure 8. Suppose there are $W$ passengers transferring at station $n_1$ from line $l_2$ to line $l_1$ and $W$ passengers transferring from line $l_1$ to line $l_2$ at station $n_2$. Suppose that in the original timetable the departure of line $l_2$ at station $n_2$ is scheduled shortly before the arrival of line $l_1$ at the same station such that the transfer takes almost a full planning period. Then by delaying the departure of line $l_2$ at station $n_2$, the transfer time gets shorter improving the travel time of the passengers but the duration of line $l_2$ increases, leading to higher operational costs.
The effects of Algorithm ReLinePlanning are the most difficult to determine. First note that the travel time can be increased as shown in Example 15.

**Example 15.** Consider the PTN and line plan given in Figure 9a. After applying Algorithm ReLinePlanning we can get the situation depicted in Figure 9b, if the minimal frequency of edge \((n_2, n_3)\) is 1 and the fixed costs of a line are relatively low.

This means that passengers driving from \(n_1\) to \(n_4\) have to transfer at station \(n_2\) and station \(n_3\) and therefore might have significantly higher travel times.

It remains to examine the influence of Algorithm ReLinePlanning on the operational costs.

**Lemma 16.** Let \((\mathcal{L}, \pi, \mathcal{V})\) be a public transport plan and \((\mathcal{L}', \pi', \mathcal{V}')\) the public transport plan after applying Algorithm ReLinePlanning to \((\mathcal{L}, \pi, \mathcal{V})\). Then the operational costs do not increase, i.e.,

\[
\text{cost}(\mathcal{V}') \leq \text{cost}(\mathcal{V}).
\]
Proof. We analyze the operational costs of \((L', \pi', V')\) by looking at the different parts of the operational costs separately. We write

\[
\text{cost}(V) = \text{cost}_{\text{time}} \cdot \sum_{r \in V} \text{duration}(r) + \text{cost}_{\text{len}} \cdot \sum_{r \in V} \text{len}(r) + \text{cost}_{\text{veh}} \cdot |V|
\]

where \(\text{duration}(r)\) describes the duration of vehicle route \(r\) and \(\text{len}(r)\) its length.

From Definition 6 we get bijection \(b\) of the vehicle routes. Thus we get

\[
|V| = |V'|. \quad (11)
\]

The duration of a vehicle route \(r = ((p_1, l_1), \ldots, (p_n, l_n))\), is defined by the duration of its trips and connecting trips, i.e.,

\[
\text{duration}(r) = \sum_{i=1}^{n} (\text{end}_{p_i, l_i} - \text{start}_{p_i, l_i}) + \sum_{i=1}^{n-1} (\text{start}_{p_{i+1}, l_{i+1}} - \text{end}_{p_i, l_i}) = \text{end}_{p_n, l_n} - \text{start}_{p_1, l_1}
\]

Route \(r\) and route \(b(r)\) differ from one another as not all edges in \(r\) have to be covered by \(b(r)\). Especially, the route might start later or end earlier. Thus we get

\[
\text{duration}(r) \geq \text{duration}(b(r)). \quad (12)
\]

The length of a vehicle route \(r = ((p_1, l_1), \ldots, (p_n, l_n))\), is defined by the length of its trips and connecting trips.

\[
\text{len}(r) = \sum_{i=1}^{n} \text{len}_{l_i} + \sum_{i=1}^{n-1} D_{l_i, l_{i+1}}
\]

With \(D_{l_i, l_{i+1}}\) being the length of a shortest path and the definition of \(P(r)\) in the beginning of Section 3.3 we get

\[
\text{len}(r) = \left( \sum_{i=1}^{n} \text{len}_{l_i} + \sum_{i=1}^{n-1} D_{l_i, l_{i+1}} \right) = \sum_{e \in P(r)} \text{len}_e.
\]

From Definition 6 we get that the paths of all trips of \(b(r)\) are contained in the path \(P(r)\) but connecting trips of \(b(r)\) use a shortest path. With the triangle inequality we get

\[
\text{len}(r) \geq \text{len}(b(r)). \quad (13)
\]
Combining equations (11), (12) and (13) we get

\[
\begin{align*}
\text{cost}(V) &= \sum_{r \in V} \text{duration}(r) + \sum_{r \in V} \text{len}(r) + \text{cost}_{\text{veh}} \cdot |V| \\
&\leq \sum_{r \in V} \text{duration}(b(r)) + \sum_{r \in V} \text{len}(b(r)) + \text{cost}_{\text{veh}} \cdot |V'|
\end{align*}
\]

\[
= \text{cost}(V').
\]

We now use Lemmas 12, 13 and 16 to formulate convergence results for iteratively applying the Algorithms ReLinePlanning, ReTimetabling and ReVehicleScheduling. As the travel time is more difficult to improve, we can only guarantee convergence for applying ReTimetabling and ReVehicleScheduling although the objectives of both algorithms differ.

**Theorem 17.** Let \( P_0 \) be a feasible public transport plan with travel time \( t_0 \). Let \( P_i, i \in \mathbb{N}^+ \), be a public transport plan derived from \( P_{i-1} \) by applying either ReTimetabling or ReVehicleScheduling and let \( t_i \) be the travel time of \( P_i \). Then the sequence of travel time values \( (t_i)_{i \in \mathbb{N}} \) decreases monotonically and converges.

**Proof.** As all feasible activity durations are positive, the sequence is bounded from below by 0. From Lemmas 12 and 13 we get that the travel time is not increased by ReTimetabling while ReVehicleScheduling has no influence on it. Therefore, \( (t_i)_{i \in \mathbb{N}} \) is monotonic and bounded and converges by the monotone convergence theorem, see e.g. [Sut09].

For the operational costs, we can guarantee convergence if duration based costs are neglected, i.e., if \( \text{cost}_{\text{time}} = 0 \).

**Theorem 18.** Let \( P_0 \) be a feasible public transport plan with operational costs \( c_0 \) where duration based costs are neglected, i.e., with \( \text{cost}_{\text{time}} = 0 \). Let \( P_i, i \in \mathbb{N}^+ \), be a public transport plan derived from \( P_{i-1} \) by applying either ReLinePlanning, ReTimetabling or ReVehicleScheduling and let \( c_i \) be the operational costs of \( P_i \). Then the sequence of operational cost values \( (c_i)_{i \in \mathbb{N}} \) decreases monotonically and converges.

**Proof.** As all vehicle schedules have positive costs, the sequence is bounded from below by 0. From Lemmas 12, 13 and 16 we get that the operational costs are not increased by ReLinePlanning and ReVehicleScheduling as well as ReTimetabling if duration based costs
are neglected, i.e., if \( \text{cost}_{\text{time}} = 0 \) is satisfied. Therefore, \( (c_i)_{i\in\mathbb{N}} \) is monotonic and bounded and converges by the monotone convergence theorem, see e.g. [Sut09].

Especially, we get convergence for travel time and costs if duration based costs are neglected, i.e., if \( \text{cost}_{\text{time}} = 0 \) is satisfied, and only \text{ReTimetabling} and \text{ReVehicleScheduling} are applied.

**Corollary 19.** Let \( P_0 \) be a feasible public transport plan with travel time \( t_0 \) and operational costs \( c_0 \) where duration based costs are neglected, i.e., \( \text{cost}_{\text{time}} = 0 \) is satisfied. Let \( P_i, i \in \mathbb{N}^+ \), be a public transport plan derived from \( T_{i-1} \) by applying either \text{ReTimetabling} or \text{ReVehicleScheduling}. Let \( t_i \) and \( c_i \) be the travel time and the operational costs of \( P_i \), respectively. Then both the sequence of travel time values \( (t_i)_{i\in\mathbb{N}} \) and the sequence of operational cost values \( (c_i)_{i\in\mathbb{N}} \) decrease monotonically and converge.

**Proof.** The sequence \( (t_i)_{i\in\mathbb{N}} \) converges by Theorem 17 and \( (c_i)_{i\in\mathbb{N}} \) converges by Theorem 18. □

5 Computational Experiments

We test the iterative scheme to modify an existing public transport plan on two different data sets. The first one, \text{grid}, is a benchmark instance described in [FHSS17], while the second one, \text{regional}, is a close-to real-world data set derived from the regional train system in Lower Saxony, Germany. The public transportation network of \text{grid} is a \( 5 \times 5 \) grid network consisting of 25 stations and 40 edges. The PTN of \text{regional} consists of 35 stations and 36 edges. Both networks are depicted in Figure 10.

![PTN of data set grid](image1)

![PTN of data set regional](image2)

**Figure 10:** PTNs of data sets \text{grid} and \text{regional}.

We use data set \text{grid} as a case study with a fixed OD matrix described in [FHSS17]. For data set \text{regional} we apply the algorithms to ten different demand scenarios and report the average increases and decreases of the objectives.
The computations are conducted on a compute server with an Intel(R) Xeon(R) X5675 CPU @ 3.07 GHz and 132 GB of RAM.

To test the iterative algorithms, we at first compute an initial public transport plan using the LinTim software framework, see [SAP+18]. Here, the cost model of line planning, see [CvDZ98, Sch12], the standard periodic timetabling problem, see [SU89], and a cost-oriented vehicle scheduling model without a depot, see [BK09], are used. The timetabling problem is solved by a modulo simplex heuristic, see [GS13]. Afterwards, we apply one of the following iteration schemes:

- **forward** Iteratively compute a public transport plan by applying the Algorithms ReLinePlanning, ReTimetabling and ReVehicleScheduling.

- **backward** Iteratively compute a public transport plan by applying the Algorithms ReVehicleScheduling, ReTimetabling and ReLinePlanning.

- **mixed** Iteratively compute a public transport plan by applying the Algorithms ReLinePlanning, ReTimetabling, ReVehicleScheduling and again ReTimetabling.

- **passenger convenience** Iteratively compute a public transport plan by alternately applying the Algorithms ReTimetabling and ReVehicleScheduling.

We use two different cost parameter sets for the computations, either **normal** which reflects a close-to real-world cost evaluation or **convergence** which differs from normal by setting the duration based costs to 0, i.e., setting cost\textsubscript{time} = 0. Note that due to Theorem 18, cost parameter set convergence guarantees the convergences of the operational costs. For each public transport plan we compute the travel time on shortest paths according to the corresponding timetable and the operational costs depending on the cost parameter set that was used for the computation. Instead of the absolute values, we plot the relative values depending on the travel time and operational costs of the initial public transport plan, respectively. For both data sets, the runtime of each iteration is in the range of minutes. However using larger data sets for long-distance networks increases the runtime dramatically as not only the network size but also the trip length increases which both contribute to the problem size. Note that for Algorithm ReTimetabling we use the current timetable as starting solution to speed up the computation.
For data set grid we compare the influence of the different iteration schemes for cost parameter set normal on the convergence and the solution quality.

Figure 11 shows that although convergence is not guaranteed, both travel time and operational costs do not change anymore after a few iterations. However, the travel time does not decrease
monotonically. Especially for iteration scheme \textbf{backward}, depicted in Figure 11b, the travel time increases multiple times. Note that although for the operational costs monotonicity and convergence is not guaranteed as duration based costs are not neglected, i.e., for $c_{\text{time}} > 0$, the costs decrease monotonically for all iteration schemes considered here.

The solutions found by the different iteration schemes vary in respect to travel time and operational costs. While \textbf{backward} yields the highest operational cost decrease of 18\%, the travel time increases by 8\%. On the other hand \textbf{mixed} yields a lower decrease of 5\% of the initial operational costs but the increase in travel time is much lower, with only 5\%. Depending on the preference corresponding to the trade-off between travel time and operational costs, both solutions are interesting options. In contrast, the solution for iteration scheme \textbf{forward} is clearly worse than the one for iteration scheme \textbf{backward}, as both the decrease in operational costs is lower with 10\% and the increase in travel time is higher with 15\%.

Figure 12 shows the impact of convergence scheme \textbf{backward} on the line plan. The coverage of the PTN edges decreases, yielding the large improvements in operational costs but also the increase in travel time. While often lines are simply shortened, see, e.g. the orange dashed line or stay the same, see, e.g. the dark blue dotted line, also new lines are formed. The cyan dash-dotted line now directly connects station $v_6$ to the stations $v_{12}$, $v_{17}$ and $v_{22}$. In the initial line plan there is at least one transfer necessary to connect these stations.

For data set \textbf{regional}, we get even better results when considering iteration scheme \textbf{mixed} for the cost parameter sets \textbf{normal} and \textbf{convergence}. Although monotonically decreasing costs are only guaranteed for cost parameter set \textbf{convergence}, Figure 13 shows that the costs
Figure 13: Applying iteration scheme mixed for data set regional with different cost parameter sets.

decrease monotonically for both parameter sets. This can also be observed for data set grid, see Figure 11, showing that in practice Algorithm ReTimetabling does not often increase the costs even if duration based costs are considered. Furthermore, the costs decrease is even higher than for data set grid with 24% decrease for parameter set normal and 25% for parameter set convergence. Even though for both parameter sets the travel time does not decrease, the increase is relatively low compared to the reduction in operational costs with 6% and 7% for cost parameters sets normal and convergence, respectively. For parameter set normal there even is one instance where the travel time is slightly reduced by 2% while the operational costs are also reduced by 25%.

When considering iteration scheme passenger convenience with cost parameter set convergence, as depicted in Figure 14, we see that both the travel time and the operational
costs decrease monotonically as expected due to Corollary 19. Note that here only the first two iterations are illustrated as no further changes occur in the later iterations. For data set **grid** the improvement is relatively small with 1% decrease of travel time and 2% decrease in operational costs. However, for data set **regional** the travel time is decreased significantly by 9% with a small improvement of the operational costs by 2%. This makes the solution clearly preferable to the initial solution and makes for an interesting additional choice to the solution found by iteration scheme **mixed** for **regional** with the same cost parameter set **convergence** with lower costs but significantly higher travel time.

Figure 14: Applying iteration scheme passenger convenience with cost parameter set convergence.

In order to investigate the influence of the initial solution on the quality of the solution found the iteration schemes, we apply the iteration schemes **forward**, **backward** and **mixed** to two different initial solutions for data set **grid** with cost parameter set **normal**. **Initialization cost** is the initial solution described above, computed by using the cost model of line planning, a periodic timetabling model and a standard vehicle scheduling model. **Initialization direct** uses the direct travelers model of line planning, see [Bus98], combined with the same timetabling and vehicle scheduling models. Figure 15 shows that the solutions derived from applying the iteration schemes to initialization cost and initialization direct differ. Especially, the set of solutions found for initialization direct is preferable to the set of solutions found for initialization cost as for each solution derived from initialization cost there exists a strictly dominating solution derived from initialization direct. However, the solution found by the iterative schemes are all similar in travel time and operational costs, with average travel times varying from 23 to 25.8 and average operational costs varying from 890 to 984, although the initial solutions differ a lot with average travel times of 22.29 and 18.65 and average operational costs of 1144 and 2051.28, respectively.
Figure 15 especially shows that the iteration schemes forward, backward and mixed are mainly focused on minimizing operational costs instead of minimizing travel time.

Figure 15: Comparing different initial solutions for iteration schemes forward, backward and mixed on data set grid with cost parameter set normal.

6 Outlook

There are several possible extensions to the models presented in this paper. First of all, the experiments show a clear tendency towards optimizing the cost, due to both vehicle scheduling and line planning both using costs as an objective. But especially for line planning, multiple possible models and objective functions are described in the literature. These could be adapted to serve as the last step of Algorithm 1, replacing the cost-optimization. This may lead to more balanced solutions, favouring the quality for the passengers.

Another possibility is to embed the iterative scheme in the eigenmodel approach discussed in [Sch17]. The problems described here form the “inner circle” of this model, see Figure 16. Therefore, it would be interesting to model the remaining problems that are not researched yet to create an meta-model for public transport planning. Several paths in the eigenmodel, representing different sequential solution approaches, are already researched (e.g. [MS09, PSSS17]), but there are still several challenges to discuss, considering new and already researched solution
Figure 16: Algorithmic scheme called eigenmodel. Nodes represent algorithms while edges represent possible concatenations of them. All possible sequential approaches to finding a public transport plan are shown, where the algorithms presented above are depicted in black. The classical sequential approach to public transport planning is depicted with dashed edges. For more information, see [Sch17].

approaches. In the end it would be interesting to determine good paths in the eigenmodel which approximate an integrated approach to public transport planning. One possibility would be to use machine learning techniques in developing a meta-algorithm for the planning process.
References


G. Cost-Minimal Public Transport Planning

M. Pätzold, A. Schiewe, A. Schöbel
Cost-Minimal Public Transport Planning
[Pätzold et al., 2019]
[Pätzold et al., 2018]
Cost-Minimal Public Transport Planning*

Julius Pätzold¹, Alexander Schiewe², and Anita Schöbel³

¹University of Göttingen, Lotzestr. 16-18, 37083 Göttingen, Germany, j.paetzold@math.uni-goettingen.de
²University of Göttingen, Lotzestr. 16-18, 37083 Göttingen, Germany, a.schiewe@math.uni-goettingen.de
³Technical University of Kaiserslautern, Gottlieb-Daimler-Straße, 67663 Kaiserslautern, Germany, schoebel@mathematik.uni-kl.de

Abstract

In this paper we investigate cost-optimal public transport plans, i.e., a line plan, a timetable and a vehicle schedule which can be operated with minimal costs while, at the same time, allowing all passengers to travel between their origins and destinations. We are hereby interested in an exact solution of the integrated problem. In contrast to a passenger-optimal public transport plan, in which there is a direct connection for every origin-destination pair, the structure or mathematical model for determining a cost-optimal public transport plan is not obvious and has not been researched so far.

We present three models which differ with respect to the structures we are looking for. If lines are directed and may contain circles, we prove that a cost-optimal schedule can (under weak assumptions) already be obtained by first distributing the passengers in a cost-optimal way. We are able to streamline the resulting integer program such that it can be applied to real-world instances. Additionally, solutions to this first model give bounds for the general case. In the second model we look for lines operated in both directions, but allow only simplified vehicle schedules. We show that this model yields a stronger lower bound than the first one. Our third and most realistic model looks for lines operated in both directions, and allows all structures for the vehicle schedules. This model, although theoretically being capable of determining general cost-optimal public transport plans, is only computable for small instances.

After introducing these three models and proving the mentioned bounds we compare their computational results and solution quality experimentally.

1 Introduction

Public transport planning is a challenging task since it consists of several stages including network design, line planning, timetabling, vehicle- and crew scheduling. In this paper we look for a line plan in combination with a timetable and a

*This work was partially supported by DFG under SCHO 1140/8-1.
vehicle schedule, i.e., a public transport plan. Apart from the different subproblems that need to be solved in an integrated way, there are also different objectives to be considered. A public transport plan should be passenger-friendly (mostly reflected by a short traveling time for the passengers) but also have low operating costs. For individual planning stages such as line planning or vehicle scheduling there exist models and algorithms but finding an integrated solution to this multi-stage problem is more challenging.

The goal of integrated planning is to find the set of pareto solutions with respect to costs and traveling time and then to choose a solution from this set that is affordable and good for the passengers. From an academic point of view it is interesting to find theoretical bounds on the two objective function values of the pareto solutions, i.e. finding the best achievable traveling time for the passengers, and finding the minimal costs (under the condition that all passengers can be transported). The former problem can be solved by a taxi-solution, providing a direct and fast connection for each origin-destination pair. Nevertheless, what a cost-optimal transportation plan would look like has not been studied so far and does not seem to be obvious.

Our contribution: In this paper we propose models for finding cost-optimal public transport plans. More precisely, for a given public transport network, passengers’ demand and a homogeneous fleet with a given vehicle capacity we design a line plan, a timetable, and a vehicle schedule under the constraint that all passengers can be transported, i.e., for each passenger there exists a possible (maybe non-desirable) connection from their origin to their destination such that none of the vehicles is overloaded. The three models presented are increasing in detail and complexity, allowing for quickly solvable approximations as well as a more detailed exact formulation, depending on the need of the planner. For the models computing approximations we prove bounds on their solution quality for the overall problem.

2 Literature Review

Traditionally, computing a public transport plan consists of solving a series of problems in a sequential order, as can be seen in [CW86, DH07, LM04]. A sequential approach, however, is unsatisfactory since the quality of the overall solution is dependent on all stages and can therefore often not be sufficiently approximated in early planning stages. Therefore integrated planning is an ongoing topic in mathematical public transport planning, see for example the recent special issue [MCZT18] and beyond, e.g., [TK00, DC18, KDC18]. Surprisingly, only a few papers evaluate both cost and traveling time for integrated public transport plans. A first approach in which line plans, timetables and vehicle schedules have been evaluated together under different criteria has been given in [GSS13]. More recently, [FHSS17] propose to measure costs and traveling time and evaluate public transport plans under these criteria (cf. Figure 7). Given a line pool, [BNP09] determine a line plan such that all origin-destination pairs can travel. The costs for the lines, however, are only approximated and not determined by the vehicle schedule. Furthermore, capacities are neglected. Other approaches often only integrate timetabling and vehicle scheduling while optimizing costs, see [vdHvdAvK08] or [DRB+17].
In contrast to these works, we take an integrated point of view and propose models for finding cost-optimal public transport plans including lines, timetables, and vehicle schedules. Additionally, we aim at solving the integrated system exactly, meaning that we do not provide iterative heuristics as in [BBLV17, Sch17, VKM17] or a sequential approach as in [PSSS17].

For the single planning stages line planning, timetabling, and vehicle scheduling models and algorithms are well-researched. For line planning cost-oriented models (e.g., [Zwa97, CvDZ98, GvHK06]) and passenger-oriented models (e.g., [Bas98, SS06, BGP07]) are known, see [Sch12] for a survey. (Periodic) timetabling focuses on the passengers and is the hardest of the three problems. Exact approaches to this problem can be found in [SU89, Nac98, PK03, Lie06] and heuristics in [NO08, GS13, PS16] and references therein. See [LLER11] for a survey. Integrating the passengers’ routes in timetabling is an ongoing problem, see [SS15, GGNS16, BHK17, Sch18]. For vehicle scheduling we refer to the survey in [BK09]. In this paper we consider periodic vehicle scheduling, which is equivalent to aperiodic planning under some assumptions as shown in [BKLL18].

3 A cost-optimal public transport plan

In this section we formally describe what a feasible public transport plan, consisting of a line plan, a timetable, and a vehicle schedule, is and how its quality can be evaluated. We restrict ourselves to periodic public transport plans (including periodic vehicle scheduling) in this paper.

**Notation 1.** The following input data is required:

- a public transport network PTN = \((V, E)\) with a set of stops \(V\) and direct connections \(E\) between them,
- for every edge \(e \in E\):
  - a length (in kilometers) \(\text{length}_e\),
  - a lower bound on the traveling time along the edge \(L_e^{\text{drive}}\),
- a lower bound \(L^{\text{wait}}\) for the time vehicles have to wait at every stop,
- a minimal turnaround time for vehicles \(L^{\text{turn}}\), denoting the minimal time a vehicle has to wait at the end of a line. We assume that \(L^{\text{wait}} \leq L^{\text{turn}}\),
- an OD-matrix \(W\) with entries \(W_{uv}\) for each pair of stops \(u, v \in V\), denoting how many passengers want to travel from an origin \(u\) to the destination \(v\) in a representative time period. A pair of stations \(u, v \in V\) with \(W_{uv} > 0\) is called an OD-pair.
- a capacity Cap being the maximal number of passengers each vehicle can transport,
- cost parameters
  - \(c_{\text{time}}\) costs per time period for a vehicle,
  - \(c_{\text{length}}\) costs per kilometer driven by a vehicle per time period.
We assume that the fixed costs (cost of a vehicle, administration, etc.) are included in the costs per time period and costs per kilometer as is often done in practice.

Network Design

- Line Planning
- Timetabling
- Vehicle Scheduling
- Crew Scheduling

Figure 1: Overview of the sequential planning procedure. The stages integrated here are highlighted with a grey box.

With this input data we then look for a public transport plan whose objects are described next. An overview on the sequential planning approach and the stages integrated here can be found in Figure 1.

Line plan

A line is a path in the PTN. A line plan is a set of lines $L$, each of them operated once in the planning period (often an hour). A line plan is feasible if every passenger can be transported, i.e., if for every OD-pair $(u, v)$ there exist

- a set of directed paths $P_{uv}$ from $u$ to $v$, $P_{all} = \bigcup_{u, v \in V} P_{uv}$

- weights $w_p$ for each path $p \in P_{uv}$

such that $\sum_{p \in P_{uv}} w_p = W_{uv}$ and such that for every edge $e$ it holds that

$$\sum_{p \in P_{all}, e \in p} w_p \leq \text{Cap} \cdot |\{l \in L : e \in l\}|.$$  \hspace{1cm} (1)

Note, that this notion of feasibility does not require the paths $P_{uv}$ to be good paths for the passengers, but only that all passengers can be transported, not necessarily on their shortest path in the network. See Section 7 for the effects on the computed solutions.

We furthermore assume that lines are simple paths and that every line is operated in both directions. We do not forbid identical lines, i.e., there may be multiple lines with the same path. In our setting we allow any such path to be a possible line (as also done in [BGP07]) in contrast to many papers which require a line pool of limited size.

Timetable

Given a set of lines $L$, a timetable assigns a time to every departure and arrival of each line at each of its stops. Determining a (periodic) timetable is the hardest of
the three problems line planning, timetabling, and vehicle scheduling, and even finding a feasible timetable that respects the upper and lower bounds on driving, waiting, transfer and turnaround activities is intractable. Since we neglect the passengers, no upper bounds on transfer activities are required and hence a feasible timetable exists for every possible line plan \( L \) (since the timetable for each line can then be determined separately). Since we are only interested in minimizing the costs we furthermore need not care about optimizing the traveling time of the passengers, meaning that any feasible timetable is sufficient. More precisely, we can neglect the timetabling as a separate planning stage in cost-optimal planning by setting the duration of all drive and wait activities to their lower bounds and simply using the arrival and departure times which are determined by the vehicle schedule.

**Vehicle schedule**

Given a line plan a *vehicle schedule* determines the number of vehicles and the exact routes of the vehicles for operating the lines. We construct a set of trips \( L' \) which contains two directed lines for every (undirected) line \( l \in L \), one in forward and the other one in backward direction.

A route of a vehicle is given by the sequence of (directed) lines it passes,

\[
r = (l'_1, \ldots, l'_k), \quad l'_i \in L',
\]

whereby requiring all \( l'_i, \ i = 1, \ldots, k \) to be pairwise distinct. We assume that the vehicle, after having taken the last trip \( l'_k \) in a route, starts again with \( l'_1 \).

This sequence \( r \) is interpreted as follows: A vehicle starts with operating line \( l'_1 \) at some point in time \( x \). At the end of line \( l'_1 \) it drives to the beginning stop of line \( l'_2 \), operates this line, and so on. At the end of line \( l'_k \) the vehicle returns to the beginning stop of \( l'_1 \) and starts again at time \( y \). In order to ensure the required periodicity of the schedule the vehicle needs to start after an integer multiple of the period \( T \), i.e., \( y = x + d_r \cdot T \) with \( d_r \) being the number of periods needed for a complete operation of the route \( r \).

A vehicle schedule thus consists of a set of routes \( R \). It is feasible if each directed line in \( L' \) is contained in exactly one route, i.e., if

\[
| \{ r \in R : l' \in r \} | = 1 \quad \forall l' \in L'.
\]

With these assumptions in place we can now define a *public transport plan*.

**Definition 2.** A feasible public transport plan is a tuple \(( L, R )\), such that

- \( L \) is a feasible line plan, i.e., it satisfies (1),
- \( R \) is a feasible vehicle schedule for the directed lines \( L' \) constructed by the line plan \( L \), i.e., \( R \) satisfies (2).

**Costs of a public transport plan**

The costs of a public transport plan are given by the distance driven by all vehicles and its total duration. Since we compute a periodic schedule, we consider the costs per planning period \( T \).
A vehicle route $r$ consists of (directed) lines $l' \in \mathcal{L}'$. Hence, we first determine time and duration of a line $l'$, that is

$$\text{length}_{l'} = \sum_{e \in l'} \text{length}_e$$

$$\text{dur}_{l'} = (|l'| - 1)L_{\text{wait}} + \sum_{e \in l'} L_{\text{drive}},$$

where $|l'| := \{ e \in E | e \in l' \}$ and (4) uses the fact that it is always cheaper to operate a line as fast as possible. For the empty rides between a pair of lines $l'_1$ and $l'_2$ we can use the PTN to determine the parameters

$$\text{length}_{l'_1, l'_2} = \text{length when driving from last station of } l'_1 \text{ to first station of } l'_2$$

$$\text{time}_{l'_1, l'_2} = \text{time for driving from last station of line } l'_1 \text{ to first station of } l'_2$$

The minimum turnaround time (usually accounting for a driver’s break) has to be added to the duration of an empty ride. This yields

$$\text{dur}_{l'_1, l'_2} = L_{\text{turn}} + \text{time}_{l'_1, l'_2}.$$  

The number of kilometers covered by a given public transport plan is determined by summing up the kilometers of each single route, i.e.,

$$\text{length}(\mathcal{L}, \mathcal{R}) = \sum_{l' \in \mathcal{L}'} \text{length}_{l'} + \sum_{r=(l'_1, \ldots, l'_k) \in \mathcal{R}} \sum_{i=1}^{k_r} \text{length}_{l'_i, l'_{i+1}}$$

$$= \sum_{l \in \mathcal{L}} 2 \cdot \text{length}_l + \sum_{r=(l'_1, \ldots, l'_k) \in \mathcal{R}} \sum_{i=1}^{k_r} \text{length}_{l'_i, l'_{i+1}}$$

with $l'_{k_r+1} := l'_1$. The duration of a route $r = (l'_1, \ldots, l'_k) \in \mathcal{R}$ is measured by the number of time periods $\text{dur}_r$ needs. Formally, this can be computed by

$$\text{dur}_r = \left[ \sum_{i=1}^{k_r} \text{dur}_{l'_i} + \text{dur}_{l'_i, l'_{i+1}} \right] T$$

with $[a]_T := \min \{ n \in \mathbb{N} | n \cdot T \geq a \}$ for any $a \in \mathbb{R}$ and $l'_{k_r+1} := l'_1$. The overall duration is hence given as

$$\text{dur}(\mathcal{L}, \mathcal{R}) = \sum_{r \in \mathcal{R}} \text{dur}_r.$$  

Finally, the cost function is defined as

$$g(\mathcal{L}, \mathcal{R}) := c_{\text{time}} \cdot \text{dur}(\mathcal{L}, \mathcal{R}) + c_{\text{length}} \cdot \text{length}(\mathcal{L}, \mathcal{R}).$$

The number of required vehicles is determined by the number of time periods used in $(\mathcal{L}, \mathcal{R})$, i.e., by $\text{dur}(\mathcal{L}, \mathcal{R})$. Once again, any fixed costs per vehicle can be included by being added to $c_{\text{time}}$. Since this does not change the structure of the cost function we assume vehicle costs to already be included in $c_{\text{time}}$. 203
The cost function defined above allows us to define the optimization problem we are concerned with in this paper.

**Problem (cost-opt):** Given the input data from Notation 1, find a feasible public transport plan \((\mathcal{L}, \mathcal{R})\), i.e., satisfying (1) and (2), with minimal costs \(g(\mathcal{L}, \mathcal{R})\). We denote the optimal objective value with \(z^{\text{opt}}\).

The rest of this paper is structured as follows: In order to find the exact cost minimum of the integrated problem (cost-opt) we present three different models (see Figure 2). The first model, presented in Section 4, aims at distributing the OD-pairs in a cost-optimal way (called load generation). Although the first model considers only this very first step, we can show that under certain conditions it already determines the minimal costs of an integrated public transport plan. Section 5 presents the second model that integrates load generation and line planning while minimizing a cost function that approximates (now in greater detail) the costs of a resulting public transport plan. Finally, Section 6 presents a third model, an exact IP formulation for integrating load generation, line planning, timetabling, and vehicle scheduling; it hence provides an exact model for (cost-opt).

<table>
<thead>
<tr>
<th>Model 1: Load Generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 2: Integrating up to Line Planning</td>
</tr>
<tr>
<td>Model 3: Integrating up to Timetabling and Vehicle Scheduling, i.e., solving it all</td>
</tr>
</tbody>
</table>

Figure 2: Three proposed models for solving (cost-opt)

## 4 Model 1: Creating a Cost-efficient Load

Line planning is often decomposed into two steps. In the first step, all OD-pairs \((u, v)\) are routed through the PTN resulting in paths \(P_{uv}\) with \(P_{\text{all}} = \bigcup_{u, v \in \mathcal{V}} P_{uv}\) and weights \(w_p\) for every path \(p \in P_{uv}\) (with \(\sum_{p \in P_{uv}} w_p = W_{uv}\)). This data is then used to define the load

\[
 f_{e}^{\text{min}} = \left[ \sum_{p \in P_{\text{all}} : e \in p} w_p \cdot \frac{1}{\text{Cap}} \right],
\]

specifying how often an edge \(e \in E\) in the PTN has at least to be served by some vehicle. In the second step, the line planning problem, i.e., finding a line plan \(\mathcal{L}\) satisfying \(f_{e}^{\text{min}} \leq |\{l \in \mathcal{L} : e \in \mathcal{L}\}|\), is solved using these minimal frequencies.

For our first model we only consider the first one of these two steps: calculating a load. Normally the load \(f_{e}^{\text{min}}\) is calculated assuming that all passengers are
able to travel on their shortest path in the PTN to their destination. Since we are interested in finding a cost-minimal public transport plan, we do not want to work with such a fixed assumption. Instead, in our system we want to admit just enough capacities to ensure that every passenger has some possibility to travel to their destination. We use this insight to find a load that eventually even leads to a cost-minimal public transport plan.

Of course, in this early planning stage we do not yet have all information to exactly determine the costs of the resulting public transport plan since they depend on the line plan and the vehicle schedule. Nevertheless, we can already approximate the costs with the following model.

**Model 1.** Given the input data from Notation 1, calculate a load (i.e., \( f_{e}^{\text{min}} \) for all \( e \in E \)) that aims at minimizing the cost of a public transport plan.

\[
\begin{align*}
\text{min} & \quad c_{\text{time}} \cdot \text{dur} + c_{\text{length}} \sum_{e \in E} 2 \cdot \text{length}_{e} \cdot f_{e}^{\text{min}} \\
\text{s.t.} & \quad \sum_{e \in E} 2f_{e}^{\text{min}}(L_{e}^{\text{drive}} + L^{\text{wait}}) \leq T \cdot \text{dur} \quad (10) \\
& \quad \sum_{u \in V} f_{(i,v),u} \leq f_{e}^{\text{min}} \cdot \text{Cap} \quad \forall i, j \in V \text{ with } \{i, j\} \in E \quad (11) \\
& \quad \sum_{i \in V : (i,v) \in E} f_{(i,v),u} = W_{uv} + \sum_{i \in V : (v,i) \in E} f_{(v,i),u} \quad \forall u \in V \quad \forall v \in V \setminus \{u\} \\
& \quad \sum_{i \in V : (u,i) \in E} f_{(u,i),u} = \sum_{v \in V} W_{uv} \quad \forall u \in V \\
\end{align*}
\]

Variables:
- \( f_{(i,j),u} \) – number of passengers starting from stop \( u \in V \) traveling on arc \((i,j)\) for some \( i, j \in V \) with \( \{i, j\} \in E \) (non-negative, continuous)
- \( f_{e}^{\text{min}} \) – load for edge \( e \), i.e., how often \( e \) has to be covered (integer)
- \( \text{dur} \) – total duration (counted in periods) (integer)

In this model we define from every stop \( u \in V \) in the PTN some passenger flow going to all destinations \( v \in V \). In order to not mix up passengers starting from different stations we have to define \( |V| \) different flows. The constraints (12) and (13) describe the flow conservation constraints. In order to restrict the number of passengers traveling on a certain edge in the network we define capacity constraints in (11). Note, that the flow variables \( f_{(i,j),u} \) for \( u \in V \) are defined on directed edges \((i,j)\) whereas the minimal frequencies \( f_{e}^{\text{min}} \) are defined on undirected edges \( \{i,j\} = e \in E \). Finally, constraint (10) rounds up the minimal duration to the next multiple of time period \( T \) and the objective function amounts the costs required in the best case, that is, for a vehicle schedule without any empty ride and as less time loss (by the periodicity rounding) as possible. We will call the optimal objective value to this model \( z_{1}^{\text{opt}} \).

The following theorem shows that Model 1 is indeed an approximation of \( \text{(cost-opt)} \) as its optimal solution yields a lower bound.
Theorem 3. Model 1 is a relaxation of (cost-opt), i.e.,

\[ z_{1}^{\text{opt}} \leq z^{\text{opt}}. \]

Proof. Let \((L, R)\) be some feasible solution to (cost-opt). Since the line plan \(L\) is feasible, we can construct some feasible flow from it by setting \(f_{e}^{\text{min}} = \{|l \in L| e \in l\}\) and \(f_{e,u} = \sum_{p \in P_{\text{all}}: e \in p} w_{p}\) with \(P_{\text{all}}\) and \(w_{p}\) obtained from (1).

Now we get for all \(i, j \in V\) with \{(i, j) \in E\}

\[
\sum_{u \in V} f_{(i,j),u} = \sum_{p \in P_{\text{all}}: (i, j) \in p} w_{p} \leq f_{e}^{\text{min}} \cdot \text{Cap}
\]

by definition of feasibility of a line plan, i.e., constraint (11) is satisfied. Since the \(w_{p}\) correspond to paths in the PTN the flow conservation constraints (12) and (13) are also satisfied. By setting

\[
dur = \left\lceil \sum_{e \in E} 2f_{e}^{\text{min}}(L_{e}^{\text{drive}} + L_{e}^{\text{wait}}) \right\rceil_T
\]

we have constructed a feasible solution to Model 1.

We now show that the objective function value of the constructed solution is better than \(g(L, R) = c_{\text{time}} \cdot \text{dur}(L, R) + c_{\text{length}} \cdot \text{length}(L, R)\).

We first consider \(\text{length}(L, R)\): We know that for the constructed solution it holds that \(f_{e}^{\text{min}} = |\{l \in L| e \in l\}|\), hence

\[
\text{length}(L, R) \geq \sum_{l' \in L'} \text{length}_{l'} = \sum_{l \in L} \sum_{e \in l} 2\text{length}_{e} \geq \sum_{e \in E} 2\text{length}_{e}f_{e}^{\text{min}}.
\]

For \(\text{dur}(L, R)\) we calculate

\[
\text{dur}(L, R) = \sum_{r \in R} \text{dur}_{r} = \sum_{r \in R} \left[ \sum_{l' \in r} (\text{dur}_{l'} + L_{l'}^{\text{turn}}) \right]_T
\]

\[
\geq \left[ \sum_{r \in R} \sum_{l' \in r} (\text{dur}_{l'} + L_{l'}^{\text{turn}}) \right]_T
\]

\[
\geq \left[ \sum_{r \in R} \sum_{l' \in r} ((|l| - 1)L_{l'}^{\text{wait}} + L_{l'}^{\text{turn}} + \sum_{e \in l'} (L_{e}^{\text{drive}} + L_{e}^{\text{wait}}) \right]_T
\]

\[
= \left[ \sum_{l' \in L} (L_{l'}^{\text{turn}} - L_{l'}^{\text{wait}} + \sum_{e \in l'} (L_{e}^{\text{drive}} + L_{e}^{\text{wait}}) \right]_T
\]

\[
\geq \left[ \sum_{l \in L} 2 \left( L_{l}^{\text{turn}} - L_{l}^{\text{wait}} \right) + \sum_{e \in l} (L_{e}^{\text{drive}} + L_{e}^{\text{wait}}) \right]_T
\]

\[
\geq \left[ \sum_{e \in E} 2f_{e}^{\text{min}}(L_{e}^{\text{drive}} + L_{e}^{\text{wait}}) \right]_T = \text{dur}.
\]
Overall it holds that
\[ g(L,R) = c_{\text{time}} \text{dur}(L,R) + c_{\text{length}} \text{length}(L,R) \]
\[ \geq c_{\text{time}} \text{dur} + c_{\text{length}} \sum_{e \in E} 2 \text{length}_e \cdot f_{\text{min}}^e. \]

Thus every feasible solution to (cost-opt) can be transformed to a solution for Model 1 whose objective is smaller than \( g(L,R) \). Hence, Model 1 is a relaxation of (cost-opt).

For large problem instances a speed-up of the solution process is possible by adding the following valid inequalities to Model 1.

**Lemma 4.** Let \((X, Y)\) be some cut, i.e., some disjoint partition of all nodes in the PTN with \( E_{\text{cut}} = \{ \{i,j\} = e \in E \mid i \in X \text{ and } j \in Y \} \) being all cut edges. Then it holds that
\[ \sum_{u \in X} \sum_{v \in Y} W_{uv} \leq \text{Cap} \cdot \sum_{e \in E_{\text{cut}}} f_{\text{min}}^e. \]

**Proof.** We start with constraint (12), i.e.,
\[ \sum_{i \in V} \sum_{\{i,v\} \in E} f_{(i,v),u} = W_{uv} + \sum_{i \in V \setminus \{v\}} f_{(v,i),u} \quad \forall u \in V \forall v \in V \setminus \{u\} \]
and argue that for any \( u \in X \) it holds that
\[ \sum_{v \in Y} \sum_{i \in X \setminus \{v\}} f_{(i,v),u} = \sum_{v \in Y} \left( W_{uv} + \sum_{i \in X \setminus \{v\}} f_{(v,i),u} \right) \]
\[ \overset{\forall u \in X \forall v \in V \setminus \{u\}}{\Rightarrow} \sum_{v \in Y} \sum_{i \in X \setminus \{v\}} f_{(i,v),u} = \sum_{v \in Y} \left( W_{uv} + \sum_{i \in X \setminus \{v\}} f_{(v,i),u} \right) \]
\[ \overset{(\ast) \text{ cancel out}}{\Rightarrow} \sum_{v \in Y, i \in X \setminus \{v\}} f_{(i,v),u} = \sum_{v \in Y} W_{uv} + \sum_{i \in X \setminus \{v\}} f_{(v,i),u} \]

Hence we can conclude
\[ \sum_{v \in Y, i \in X \setminus \{v\}} f_{(i,v),u} \geq \sum_{v \in Y} W_{uv} \quad \forall u \in X. \] (14)
Thus we get that

\[ \text{Cap} \cdot \sum_{e \in E_{\text{cut}}} f_e^{\min} \geq \sum_{i \in X, v \in Y: \{i, v\} \in E_{\text{cut}}} \sum_{u \in V} f(i, v), u \geq \sum_{X \subseteq V} \sum_{i \in X, v \in Y: \{i, v\} \in E_{\text{cut}}} \sum_{u \in V} W_{uv}. \]

(11)

In the computational experiments, see Section 7, we investigated adding these valid inequalities, which resulted in an improvement of the runtime of up to 50%.

In order to find an upper bound for \( \text{cost-opt} \) instead of a lower bound, we slightly modify Model 1.

**Definition 5.** We define an adjusted version of Model 1, where \( L^{\text{wait}} \) is replaced by \( L^{\text{turn}} \) in constraint (10), to be Model 1*. We call the optimal objective value of this model \( z_{1^*}^{\text{opt}} \).

Using this new model, we are able to compute an upper bound to \( \text{cost-opt} \). Note, that in the following of this chapter we always assume the graph \( G = (V, \bar{E}) \) with \( \bar{E} = \{ e \in E: f_e^{\min} > 0 \} \) to be connected for an optimal solution to Model 1*. This is for example the case, when the graph \( (V, W') \) with \( W' = \{ \{ u, v \} \subseteq V : W_{uv} > 0 \} \) of the OD pairs is connected.

**Theorem 6.** For every feasible solution to Model 1* where \( G \) is connected, there is a feasible solution to \( \text{cost-opt} \) with the same objective value, i.e.,

\[ z^{\text{opt}} \leq z_{1^*}^{\text{opt}} \]

**Proof.** For every solution to Model 1*, i.e., for some feasible \( (f^{\min}, f) \), we can construct some feasible solution \( (L, R) \) to \( \text{cost-opt} \) as follows: We define the line plan \( L \) that contains for each edge \( e \in E \) exactly \( f_e^{\min} \) lines containing exactly this one edge \( e \), i.e., \( L := \{ l^1, \ldots, l^{f_e^{\min}} : e \in E \} \). Since \( f_e^{\min} = |\{ l \in L | e \in l \}| \) and \( f_e^{\min} \) admits a feasible load, e.g., corresponding to \( f \), the line plan \( L \) is feasible.

For this line plan we now generate a vehicle schedule \( R \) that consists of only one large route. To this end, we consider the resulting set of directed lines \( L' \)

\[ L' = \left\{ (i, j)^1, \ldots, (i, j)^{f_e^{\min}}, (j, i)^1, \ldots, (j, i)^{f_e^{\min}} : e = \{ i, j \} \in E \right\} \]

which contains \( f_e^{\min} \) copies of both directions of every edge \( e \in E \). This is a set of directed edges which creates a directed multigraph \( (V, L') \). Due to the assumption that \( G = (V, \bar{E}) \) with \( \bar{E} = \{ e \in E : f_e^{\min} > 0 \} \) is connected, this graph is strongly connected and every node in \( (V, L') \) has the same indegree as outdegree. Hence we can find an Eulerian Cycle on it (see, e.g., [Fle91]). This means that we can form a route containing all directed lines \( r = (l'_1, \ldots, l'_k) \) (with \( |r| = |L'| \)) such that length \( l'_i, l'_{i+1} = 0 \) and time \( l'_i, l'_{i+1} = 0 \). We set the vehicle schedule \( R = \{ r \} \) to contain exactly this route \( r \).
We hence have constructed some solution \((\mathcal{L}, \mathcal{R})\) to (cost-opt) with

\[
\text{length}(\mathcal{L}, \mathcal{R}) = \sum_{l \in \mathcal{L}} \text{length}_l + \sum_{r = (l_1', \ldots, l_k')} \sum_{i=1}^{k_r} \text{length}_{l_i} \cdot f_{e_{l_i}} = \sum_{l \in \mathcal{L}} 2 \cdot \text{length}_l = \sum_{e \in E} 2 f_{\min} (L_{\text{drive}}^e + L_{\text{turn}}^e) \cdot f_{\min}
\]

and

\[
\text{dur}(\mathcal{L}, \mathcal{R}) = \sum_{r \in \mathcal{R}} \text{dur}_r = \sum_{l \in \mathcal{L}} (\text{dur}_l + L_{\text{turn}}) \cdot T = \sum_{e \in E} 2 f_{\min} (L_{\text{drive}}^e + L_{\text{turn}}^e) \cdot T = \text{dur}.
\]

Hence, for every solution to Model 1 we can construct a solution \((\mathcal{L}, \mathcal{R})\) to (cost-opt) such that \(g(\mathcal{L}, \mathcal{R}) = c_{\text{time}} \cdot \text{dur} + c_{\text{length}} \cdot \sum_{e \in E} 2 f_{\min} \cdot f_{\min}\). Together with Theorem 3 the solution \((\mathcal{L}, \mathcal{R})\) is optimal for (cost-opt) and hence Model 1 has the same objective value as (cost-opt).

We can now compute a gap between Model 1 and Model 1*. This allows us to estimate the objective value to (cost-opt) by only computing a solution to Model 1.

**Theorem 7.** Let \((\text{dur}, f, f_{\min})\) be an optimal solution to Model 1. Then the gap between the optimal objective values to Model 1 and Model 1* is bounded, i.e.,

\[
\zeta_{1*}^{\text{opt}} - \zeta_{1}^{\text{opt}} \leq \left( 2 \cdot \frac{L_{\text{turn}} - L_{\text{wait}}}{T} \cdot \sum_{e \in E} f_{\min} \right) \cdot c_{\text{time}}.
\]

**Proof.** Let \((\text{dur}, f, f_{\min})\) be an optimal solution to Model 1. For every optimal solution to Model 1, it holds that

\[
\text{dur} = \left( \sum_{e \in E} 2 f_{\min} \cdot (L_{\text{drive}}^e + L_{\text{wait}}) / T \right).
\]

By setting

\[
\text{dur}^* := \left( \sum_{e \in E} 2 f_{\min} \cdot (L_{\text{drive}}^e + L_{\text{turn}}^e) / T \right),
\]

\((\text{dur}, f, f_{\min})\) can be transformed to a feasible solution \((\text{dur}^*, f, f_{\min})\) for Model 1*. 
With this and the fact that \( \lceil x + y \rceil \leq \lceil x \rceil + \lceil y \rceil \) for all \( x, y \in \mathbb{R} \) it holds that

\[
\begin{align*}
z_{1^*}^{\text{opt}} - z_1^{\text{opt}} &= c_{\text{time}} \cdot \left( \frac{\sum_{e \in E} 2 \cdot f_{\text{min}}(L_{e}^{\text{drive}} + L_{e}^{\text{turn}})}{T} - \frac{\sum_{e \in E} 2 \cdot f_{\text{min}}(L_{e}^{\text{drive}} + L_{e}^{\text{wait}})}{T} \right) \\
&\leq c_{\text{time}} \cdot \left( \frac{\sum_{e \in E} 2 \cdot f_{\text{min}}(L_{e}^{\text{turn}} - L_{e}^{\text{wait}})}{T} \right).
\end{align*}
\]

This bound can be extended to a gap to \( (\text{cost-opt}) \).

**Corollary 8.** The absolute error of solving Model 1 or Model 1* is bounded by

\[
\begin{align*}
z_{1^*}^{\text{opt}} - z_{1}^{\text{opt}} &\leq 2 \cdot \frac{L_{e}^{\text{turn}} - L_{e}^{\text{wait}}}{T} \cdot \sum_{e \in E} f_{\text{min}}(e) \cdot c_{\text{time}} \\
z_{1}^{\text{opt}} - z_{1}^{\text{opt}} &\leq 2 \cdot \frac{L_{e}^{\text{turn}} - L_{e}^{\text{wait}}}{T} \cdot \sum_{e \in E} f_{\text{min}}(e) \cdot c_{\text{time}}.
\end{align*}
\]

Additionally, this bound allows an optimality condition, where the optimal objective value of Model 1 and Model 1* is the optimal objective value of \( (\text{cost-opt}) \).

**Corollary 9.** Let \( L_{e}^{\text{wait}} = L_{e}^{\text{turn}} \). Then the optimal objective of Model 1 and Model 1* is equal to the optimal objective of \( (\text{cost-opt}) \).

If we allow that lines do not have to be bidirectional and simple paths in the PTN, we can always obtain an optimal solution to \( (\text{cost-opt}) \) by just solving Model 1. This can be done by converting the Eulerian Cycle constructed in the proof of Theorem 6 into one big line.

**Corollary 10.** Let \( L_{e}^{\text{wait}} \leq L_{e}^{\text{turn}} \). Then the optimal objective value of Model 1 is equal to the optimal objective of \( (\text{cost-opt}) \) if we allow directed and non-simple lines.

This, of course, may lead to non-practical lines, as can be seen in the following example.

![Figure 3: Solution of Model 1 for Example 11](image)

**Example 11.** We examine the solution provided by Corollary 10 on a small example. Consider the PTN given in Figure 3, with Cap passengers traveling from \( v_1 \) to \( v_5 \) and 1 passenger traveling from \( v_2 \) to \( v_3 \). Then the solution provided
by Model 1 is given by lower bounds of \([1,2,1,1]\) and the vehicle schedule of Corollary 10 is depicted in Figure 3, where the edges are numbered in order of their usage. As can be seen here, the resulting line structure, that is, if the whole vehicle schedule is transformed into a single line, is not suitable for a practical public transport system, since it contains a cycle.

5 Model 2: Integrating Load Generation and Line Planning

Although we can already find a solution to (cost-opt) using Model 1, it is only cost-minimal in the case of \(L_{\text{wait}} = L_{\text{turn}}\). For \(L_{\text{wait}} < L_{\text{turn}}\), however, we have seen that if we want to obtain a cost-minimal solution, the resulting line plan may consist of directed lines (without their symmetric counterparts) and the lines may contain circles. We hence want to incorporate the next steps of public transport planning to resolve this issue and ensure that the lines satisfy the usual requirements. To this end, we combine the load generation of Model 1 with line planning to improve the approximation of the cost objective of the overall plan. This idea is approached by the following model.

**Model 2.** Given the input data from Notation 1, calculate a load \(f_{e}^{\text{min}}\) and a line plan \(\mathcal{L}\) that aim at minimizing the costs of a public transport plan.
\[
\begin{align*}
\min & \quad c_{\text{time}} \cdot \text{dur} + c_{\text{length}} \sum_{l \in [L]} \sum_{e \in E} 2x_{e,l} \cdot \text{length}_e \\
\text{s.t.} & \quad (11) - (13) \\
& \quad \sum_{l \in [L]} \left( 2z_l (L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} (2L_{\text{drive}} + L_{\text{wait}}) \cdot x_{e,l} \right) \leq \text{dur} \cdot T \\
& \quad \sum_{l \in [L]} \sum_{e \in E} x_{e,l} \geq f_{\text{min}}^e \quad \forall e \in E \\
& \quad x_{e,l} \leq z_l \quad \forall e \in E \forall l \in [L] \\
& \quad \sum_{e \in E} x_{e,l} \geq z_l \quad \forall l \in [L] \\
& \quad \sum_{e \in E, s \in e} x_{e,l} \leq 2 \quad \forall s \in V \forall l \in [L] \\
& \quad 2x_{e,l} \leq y_{i,l} + y_{j,l} \quad \forall l \in [L] \forall (i,j) = e \in E \\
& \quad \sum_{s \in V} y_{s,l} = \sum_{e \in E} x_{e,l} + z_l \quad \forall l \in [L] \\
& \quad \sum_{(i,j) = e \in E, i \in C \text{ and } j \in C} x_{e,l} \leq |C| - 1 \quad \forall \text{circles } C \subseteq E \forall l \in [L] \\
\end{align*}
\]

Coefficients:

- \(L\) – maximal possible number of lines (integer) and \([L] := \{1, ..., L\}\).

Variables:

- \(z_l\) – is 1 if line \(l\) is non-empty. (binary)
- \(y_{s,l}\) – is 1 if stop \(s\) is contained in line \(l\). (binary)
- \(x_{e,l}\) – is 1 if edge \(e\) is contained in line \(l\). (binary)
- \(\text{dur}\) – total duration of all lines (counted in periods) (integer)
- \(f_{\text{min}}^e\) – as in Model 1, including the variables \(f_{e,u}\) and constraints (11) - (13) from Model 1.

This model finds some feasible line plan. First the \(z_l\)-variables determine if line number \(l\) is a line or empty. Constraint (18) and (19) ensure this. Now we need for every index \(l\) that for every stop of some line there are at most two incident edges (constraint (20)). This ensures that the \(x_{e,l}\) variables form circles or paths. To ensure that they form only one connected path we could consider them as flow variables. Here, we decided to add \(y\)-variables for every visited stop and count the number of stops that a line visits. The \(y\)-variables are set to one for the incident nodes of all edges the line visits in (21). We then can ensure that there is some connected path by requiring that there exists exactly one more stop than edges in a line in constraint (22). Finally we need to rule out subtours which is done by constraint (23). (As usual they are added by constraint generation procedures). The variables \(f_{\text{min}}^e\) taken from Model 1 help...
us to determine feasibility of the line plan, which is done by constraint (17). Finally we round up the duration to the next multiple of a time period, which is done by (16). We call an optimal objective value to this model $z_2^{\text{opt}}$.

The objective function is again a lower bound on the exact costs of a public transport plan which is shown in the next theorem. Note, that the choice of the size of $L$ is crucial for the quality of the model and will be discussed later.

**Theorem 12.** For sufficiently large $L$, the optimal objective value of Model 2 is a lower bound on the optimal objective value of (cost-opt) and an upper bound to the optimal objective value of Model 1, i.e.,

$$z_2^{\text{opt}} \leq z_2^{\text{opt}} \leq z^{\text{opt}}.$$

**Proof.** Let $(\mathcal{L}, \mathcal{R})$ be some feasible solution to (cost-opt). Then we know that we can set $f_{e}^{\text{min}} = |\{l \in \mathcal{L}[e \in l]\}|$ (and $f_{e,u}$ accordingly) as in the proof of Theorem 3 to some feasible flow which satisfies (17). Furthermore we can enumerate all lines with some bijective mapping $\varphi : \mathcal{L} \rightarrow |\mathcal{L}|$ such that $x_{e,\varphi(l)} = 1$ iff $e \in l$ for all $l \in \mathcal{L}$ and also $y_{s,\varphi(l)} = 1$ iff $s \in e$ for some $e \in l$. Finally, we have to set $z_i = 1$ for all $i \in |\mathcal{L}|$ and 0 for all $i \in |\mathcal{L}| \setminus |\mathcal{L}|$. Since $\mathcal{L}$ was some feasible line plan, all lines are simple paths and hence also constraints (18) to (23) are satisfied. Now for the objective function it holds that

$$\text{length}(\mathcal{L}, \mathcal{R}) = \sum_{l \in \mathcal{L}} |\mathcal{L}| + \sum_{l \in \mathcal{L}} \sum_{e \in l} \text{length}_{e,l} \geq \sum_{l \in \mathcal{L}} \sum_{e \in l} 2|\mathcal{L}| \cdot \text{length}_{e,l} = \sum_{l \in |\mathcal{L}|} \sum_{e \in \mathcal{E}} 2|\mathcal{L}| \cdot \text{length}_{e,l}.$$  

For the duration we get

$$\text{dur}(\mathcal{L}, \mathcal{R}) = \sum_{r = (t_1', \ldots, t_k') \in \mathcal{R}} \left[ \sum_{i=1}^{k_r} \text{dur}_{i, t_i', t_{i+1}} \right] \geq \left[ \sum_{r \in \mathcal{R}} \sum_{l \in \mathcal{L}} \left( \text{dur}_{l} + \sum_{e \in l} L_{e, \text{drive}} + L_{e, \text{wait}} \right) \right] \geq \left[ \sum_{l \in |\mathcal{L}|} \left( 2L_{\text{turn}} - L_{\text{wait}} + \sum_{e \in \mathcal{E}} 2L_{e, \text{drive}} + L_{e, \text{wait}} \right) \cdot x_{e,l} \right].$$

Hence, by finally setting

$$\text{dur} = \left[ \sum_{l \in |\mathcal{L}|} \left( 2L_{\text{turn}} - L_{\text{wait}} + \sum_{e \in \mathcal{E}} 2L_{e, \text{drive}} + L_{e, \text{wait}} \right) \cdot x_{e,l} \right],$$

we conclude that from any feasible solution $(\mathcal{L}, \mathcal{R})$ to (cost-opt) we can construct some feasible solution to Model 2 such that

$$g(\mathcal{L}, \mathcal{R}) \geq c_{\text{time}} \cdot \text{dur} + c_{\text{length}} \sum_{l \in |\mathcal{L}|} \sum_{e \in \mathcal{E}} 2x_{e,l} \cdot \text{length}_{e,l},$$

213
which means that the objective function value of Model 2 is a lower bound to (cost-opt).

On the other hand every feasible solution to Model 2 is a feasible solution to Model 1. This can be seen by setting the three types of variables, \( f_{\text{min}}, f_{\text{e,u}} \) and \( \text{dur} \), that are contained in both models, to be the same. Hence constraints (11) - (13) are satisfied, and also (10) is satisfied since

\[
dur \cdot T \geq \sum_{l \in [L]} \left( 2z_l (L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} 2(f_e^{\text{drive}} + L_{\text{wait}}) \cdot x_{e,l} \right) \geq 0
\]

\[
\geq \sum_{e \in E} 2f_{\text{min}}^{e} (L_{e}^{\text{drive}} + L_{\text{wait}}).
\]

For the objective functions it additionally holds that

\[
\sum_{l \in [L]} \sum_{e \in E} 2x_{e,l}\text{length}_e = \sum_{e \in E} 2f_{\text{min}}^{e}\text{length}_e.
\]

This means that every solution to Model 2 can be projected to a solution of Model 1 with smaller objective value in Model 1, meaning that Model 2 is an upper bound to Model 1. \( \square \)

We can again construct a feasible solution for (cost-opt) from the solution of Model 2 in the case that we are only interested in line-pure vehicle schedules. In such schedules, every vehicle serves the same line, alternating between its forward and its backward direction. More formally:

**Definition 13.** A solution to (cost-opt) is called line-pure if \( R = \{r_l : l \in L\} \), with \( r_l = (l^+, l^-) \) being the route that contains only the forward and backward direction of line \( l \in L \).

Again, we do not only want to find a lower, but also an upper bound to (cost-opt). To this end we slightly modify Model 2. Instead of measuring the overall duration of all lines in constraint (16), we track each line individually by using the constraints

\[
2z_l (L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} 2(f_e^{\text{drive}} + L_{\text{wait}}) \cdot x_{e,l} \leq d_l \cdot T \quad \forall l \in [L] \quad (24)
\]

\[
\sum_{l \in [L]} d_l = \text{dur} \quad (25)
\]

\[
d_l \in \mathbb{N}. \quad (26)
\]

By doing so, we implicitly evaluate our lines using a line-pure vehicle schedule.

**Definition 14.** Consider Model 2 and replace constraint (16) by constraints (24)-(26). We call this modified version Model 2* and its optimal objective value \( z^*_{\text{opt}} \).

Restricting ourselves to the special structure of line-pure vehicle schedules, we are still able to obtain the optimal solution to (cost-opt) by simply considering loads and lines. This is the main result of this section.
Theorem 15. An optimal solution to Model 2* solves (cost-opt) under the restriction that only line-pure vehicle schedules are allowed.

Proof. Let $\mathcal{L}, \mathcal{R}$ be some line-pure feasible solution to (cost-opt). For the objective value of $(\mathcal{L}, \mathcal{R})$ we know that

$$\text{length}(\mathcal{L}, \mathcal{R}) = \sum_{r=(l_1', \ldots, l_k') \in \mathcal{R}} \sum_{i=1}^{k_r} \text{length}_{l_i'} = \sum_{l \in \mathcal{L}} 2\text{length}_l = \sum_{l \in \mathcal{L}} \sum_{e \in l} 2\text{length}_e,$$

and that

$$\text{dur}(\mathcal{L}, \mathcal{R}) = \sum_{r \in \mathcal{R}} \left[ \sum_{l \in r} (\text{dur}_l + L_{\text{turn}}) \right]_T = \sum_{l \in \mathcal{L}} \left[ 2(\text{dur}_l + L_{\text{turn}}) \right]_T$$

$$= \sum_{l \in \mathcal{L}} \left[ 2(L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E \colon e \in l} 2(L_e^{\text{drive}} + L_e^{\text{wait}}) \right]_T.$$

We can extend the line plan $\mathcal{L}$ to some feasible solution to Model 2* by again defining a bijective mapping $\phi : \mathcal{L} \to [\vert \mathcal{L} \vert]$ such that $x_{e,\phi(l)} = 1$ if $e \in l$ for $l \in \mathcal{L}$ for all $e \in E$. Analogously a solution $x_{e,l}$ can be transformed into some feasible line plan $\mathcal{L}$ by defining a line $l$ to contain exactly all edges $e \in E$ if $x_{e,l} = 1$.

Thus there exists a bijection between the set of feasible solutions between (cost-opt) and Model 2* as well as the same objective function for both problems since

$$\sum_{l \in \mathcal{L}} \sum_{e \in l} 2\text{length}_e = \sum_{l \in \mathcal{L}} \sum_{e \in E} 2x_{e,\phi(l)} \text{length}_e = \sum_{l \in \mathcal{L}} \sum_{e \in E} 2x_{e,l} \text{length}_e$$

and

$$\sum_{l \in \mathcal{L}} \left[ 2(L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E \colon e \in l} 2(L_e^{\text{drive}} + L_e^{\text{wait}}) \right]_T$$

$$= \sum_{l \in [\mathcal{L}]} \left[ 2(L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} 2x_{e,l} \text{length}_e (L_e^{\text{drive}} + L_e^{\text{wait}}) \right]_T = \sum_{l \in [\mathcal{L}]} d_l.$$

Hence their optimal objective values coincide.

For the general case of (cost-opt), i.e., without the restriction of line-pure vehicle schedules, Model 2* still finds a feasible solution and therefore provides an upper bound to (cost-opt).

Corollary 16. The optimal objective value to Model 2* imposes an upper bound on the optimal objective value of (cost-opt), i.e.,

$$z_{\text{opt}} \leq z_{\text{opt}}^{2*}.$$

Additionally, for an $L$, that is known to be sufficiently large, we can provide an a priori bound between $z_{\text{opt}}^{2*}$ and $z_{\text{opt}}^{2}$.

Theorem 17. The gap between the optimal objective values of Model 2 and Model 2* is bounded by $c_{\text{time}} \cdot (L - 1)$, i.e.,

$$z_{\text{opt}}^{2*} - z_{\text{opt}}^{2} \leq c_{\text{time}} \cdot (L - 1).$$
Proof. Let \((\text{dur}, f_{\text{min}}, f, x, z)\) be an optimal solution to Model 2. Since the solution is optimal, we know that
\[
\text{dur} = \left\lceil \frac{\sum_{l \in [L]} \left( 2z_l (L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} 2(L_e^{\text{drive}} + L_e^{\text{wait}}) \cdot x_{e,l} \right)}{T} \right\rceil.
\]
Denote
\[
a_l := 2z_l (L_{\text{turn}} - L_{\text{wait}}) + \sum_{e \in E} 2(L_e^{\text{drive}} + L_e^{\text{wait}}) \cdot x_{e,l},
\]
i.e.,
\[
\text{dur} = \left\lceil \sum_{l \in [L]} a_l \right\rceil.
\]
Since the only difference between Model 2 and Model 2* is the replacement of constraint (16) by constraints (24) and (25), \((\text{dur}^*, f_{\text{min}}, f, x, z)\) with
\[
d_l = [a_l]
dur^* = \sum_{l \in [L]} d_l
\]
is a feasible solution for Model 2*. Therefore
\[
z_{2*}^{\text{opt}} \leq c_{\text{time}} \cdot \text{dur}^* + c_{\text{length}} \cdot \sum_{l \in [L]} \sum_{e \in E} 2x_{e,l} \cdot \text{length}_e
\]
holds. Let
\[
\bar{a}_l = a_l - \lfloor a_l \rfloor.
\]
be the non-integer part of \(a_l\). Without loss of generality there exists an \(l \in \{1, \ldots, L\}\) with \(\bar{a}_l > 0\), because otherwise the gap would be 0. Then it holds that
\[
z_{2*}^{\text{opt}} - z_{2}^{\text{opt}} \leq c_{\text{time}} \cdot (\text{dur}^* - \text{dur})
\]
\[
= c_{\text{time}} \cdot \left( \sum_{l \in [L]} [a_l] - \sum_{l \in [L]} a_l \right)
\]
\[
= c_{\text{time}} \cdot \left( \sum_{l \leq l < L} [\bar{a}_l] - \sum_{l \geq l < L + 1} a_l \right)
\]
\[
\leq c_{\text{time}} \cdot (L - 1)
\]
Using this gap, Model 2 can provide an a priori bound on the objective value of \((\text{cost-opt})\).
Corollary 18. The absolute error of solving Model 2 or Model 2* is at most \( c_{\text{time}} \cdot (L - 1) \), i.e.,

\[
\begin{align*}
  z_{2^*}^{\text{opt}} - z^{\text{opt}} & \leq c_{\text{time}} \cdot (L - 1) \\
  z^{\text{opt}} - z_{2}^{\text{opt}} & \leq c_{\text{time}} \cdot (L - 1).
\end{align*}
\]

The following example shows that the bound provided in Corollary 18 can be attained.

![Infrastructure network for Example 19](image)

Figure 4: Infrastructure network for Example 19

Example 19. Let \( L \) be known. Consider the PTN depicted in Figure 4 with \( L \) single edges, connecting two nodes each. Each edge has a length of \( L_{\text{drive}} = \epsilon \), one passenger travelling and let \( L_{\text{turn}} = \epsilon \). Then

\[
z_{2^*}^{\text{opt}} - z^{\text{opt}} = c_{\text{time}} \cdot L - c_{\text{time}} \cdot \left[ \frac{4\epsilon T}{T} \right] \epsilon \to 0 \to c_{\text{time}} \cdot (L - 1).
\]

As we have already mentioned, the presented theoretical results of this section only hold true if \( L \) is chosen sufficiently large. One possible upper bound for \( L \) is \( \sum_{u,v \in \text{OD}} \left\lceil \frac{W_{uv}}{\text{Cap}} \right\rceil \), giving every od pair the possibility to build its own lines. In practice, however, much smaller values for \( L \) are already feasible. Smaller values of \( L \), that are still large enough, can be computed with the following insight.

Theorem 20. Let \( \psi \) be a feasible solution to Model 2* with objective value \( \text{obj} \) with an arbitrary \( L \). Define

\[
L_{\text{ub}} := \frac{\text{obj} - c_{\text{length}} \sum_{u,v \in V} \text{SP}(u,v) \frac{W_{uv}}{\text{Cap}}}{c_{\text{time}}}.
\]

where \( \text{SP}(u,v) \) for \( u, v \in V \) is the shortest path from \( u \) to \( v \) with respect to the edge lengths \( \text{length}_e \), \( e \in E \). Then the number of lines of the optimal solution to Model 2* is bounded by \( L_{\text{ub}} \).

Proof. Assume \( \psi' \) to be an optimal solution to Model 2* with objective value
that uses \( L’ > L_{ub} \) lines. Then

\[ z_{2*}^{opt} = c_{time} \cdot \text{dur}_{\geq L’ > L_{ub}} + c_{\text{length}} \sum_{l \in |L|} \sum_{e \in E} 2 \cdot x_{el} \cdot \text{length}_e \]

\[ > c_{time} \cdot L_{ub} + c_{\text{length}} \sum_{l \in |L|} \sum_{e \in E} 2 \cdot x_{el} \cdot \text{length}_e \]

\[ \geq \text{obj} + c_{\text{length}} \left( \sum_{l \in |L|} \sum_{e \in E} 2 \cdot x_{el} \cdot \text{length}_e - \frac{\sum_{u,v \in V} \text{SP}(u,v) \cdot W_{uv}}{\text{Cap}} \right) \]

\[ \geq \text{obj} \quad (\ast) \]

which is a contradiction to \( \psi’ \) being optimal. Here, \( (\ast) \) holds due to \( \sum_{l \in |L|} \sum_{e \in E} 2 \cdot f_e^{\text{min}} \cdot \text{length}_e \) being the (passenger-weighted) length of a feasible flow and \( \sum_{u,v \in V} \text{SP}(u,v) \cdot W_{uv} \) \( \text{Cap} \) being the length of the corresponding shortest flow.

Using Theorem 20 we can now obtain a sufficiently large, but still reasonably low, choice of \( L \) by solving Model 2* only twice: For obtaining a first solution an arbitrarily chosen \( L \) is sufficient. With the objective value of this first solution we then can calculate \( L_{ub} \) by using (27). Now, if we solve Model 2* again with \( L_{ub} \), we can be sure that an optimal solution will be found.

Continuing our process of finding public transport plans of good quality, we investigate how Model 2* behaves when confronted with Example 11. It illustrates that the solutions of Model 2* are more usable than the solutions of Model 1*, i.e., the practical problems demonstrated at the end of Section 4 are solved by Model 2*.

**Example 21.** We continue Example 11 and now consider the solution constructed in Theorem 12. These now provide simple lines, resulting in the line-pure vehicle schedule depicted in Figure 5, improving on the line structure of Example 11. The first line is depicted in red, the second is dashed in green. The lines here look much more reasonable for practical implementation than the solution which was obtained by Model 1*.

![Figure 5: Solution of Model 2](image-url)
6 Model 3: Integrating Timetabling and Vehicle Scheduling

In Model 1 and Model 2 we did not consider all arising subproblems of (cost-opt) so far. Especially, we did not include a proper vehicle scheduling into the mathematical models. With the following model we want to overcome this issue and formulate the whole problem in an integrated way.

To formulate the integrated model, we need a notation for the event-activity network $\mathcal{N} = (\mathcal{E}, \mathcal{A})$ (see, e.g., [Lie06, LM04, Nac98, Pee03, PK01]). The set of events $\mathcal{E}$ consists of all departures and all arrivals of all lines at all stops and two additional OD-events $((u, \text{dep}), (u, \text{arr}))$ per stop $u$ for passengers to enter and leave the network, denoted as $\mathcal{E}_{OD}$. The set $\mathcal{A}$ connects the events by driving, waiting and transfer activities. The OD-events are connected to each departure event of the corresponding stop using OD-activities ($\mathcal{A}_{OD}$). Using this, we can now formulate the integrated model. Let further denote with $\mathcal{A}_l'$ all activities in $\mathcal{A} \setminus \mathcal{A}_{OD}$ that are included in a directed line $l' \in \mathcal{L}'$.

Model 3. Given the input data from Notation 1, find a feasible public transport plan $(\mathcal{L}, \mathcal{R})$ with minimal costs, i.e., minimizing $g(\mathcal{L}, \mathcal{R})$. 
min \sum_{r \in [R]} \text{cost}_r
\quad \text{s.t.} \quad \text{dur}_r \geq \frac{1}{T} \cdot \sum_{l' \in L'} x_{l',r} \cdot \text{dur}_l + \sum_{l'_1, l'_2 \in L'} x_{(l'_1, l'_2),r} \cdot \text{dur}_{l'_1, l'_2} \quad \forall r \in [R] \tag{28}

\text{length}_r \geq \sum_{l' \in L'} x_{l',r} \cdot \text{length}_l + \sum_{l'_1, l'_2 \in L'} x_{(l'_1, l'_2),r} \cdot \text{length}_{l'_1, l'_2} \quad \forall r \in [R] \tag{29}

\text{cost}_r \geq c_{\text{length}} \cdot \text{length}_r + c_{\text{time}} \cdot \text{dur}_r \quad \forall r \in [R] \tag{30}

\sum_{l' \in L'} x_{(l', r), r} = \sum_{l' \in L'} x_{b(l'), r} \quad \forall l' \in L', \forall r \in [R] \tag{31}

\sum_{r \in [R]} \sum_{l' \in L'} x_{l',r} = \sum_{r \in [R]} \sum_{u,v \in V} f_{a,(u,v)} \quad \forall l' \in L', \forall a \in A_r \tag{33}

\sum_{(i,j) \in \mathcal{A} \setminus \mathcal{E} \cap \mathcal{E}_{OD}} f_{a,(u,v)} = \sum_{(i,j) \in \mathcal{A} \setminus \mathcal{E} \cap \mathcal{E}_{OD}} f_{a,(u,v)} \quad \forall u, v \in V, \forall j \in \mathcal{E} \setminus \mathcal{E}_{OD} \tag{34}

\sum_{(i,j) = a \in \mathcal{A}_{OD}} f_{a,(u,v)} = W_{uv} \quad \forall u, v \in V, \forall j = (u, arr) \in \mathcal{E}_{OD} \tag{35}

\sum_{(i,j) = a \in \mathcal{A}_{OD}} f_{a,(u,v)} = W_{uv} \quad \forall u, v \in V, \forall j = (u, dep) \in \mathcal{E}_{OD} \tag{36}

\sum_{(l'_1, l'_2) \in U'} x_{(l'_1, l'_2),r} \leq |U'| - 1 \quad \forall U' \subseteq L' \times L', \forall r \in [R] \tag{37}

dur_r \in \mathbb{N} \quad \forall r \in [R] \tag{38}

**Coefficients:**

- \( R \): number of possible vehicle routes, we assume it to be sufficiently large
- \( L' \): the set of all possible directed lines in the network, \( b(l') \) denotes the backwards direction for a directed line \( l' \), \( l \) is the corresponding undirected line.

**Variables:**

- \( x_{l',r} \): is 1 iff the directed line \( l' \) is part of route \( r \)
- \( x_{(l'_1, l'_2),r} \): is 1 iff lines \( l'_1 \) and \( l'_2 \) are served directly after each other in route \( r \)
- \( \text{cost}_r \): the costs of route \( r \)
- \( \text{dur}_r \): the duration of route \( r \)
- \( \text{length}_r \): the length of route \( r \)
• $f_a(u,v)$ – the number of passengers traveling from $u$ to $v$ using activity $a$

This model finds a cost-optimal public transport plan (i.e., line plan, timetable and vehicle schedules). The $f$ variables determine the passenger flow, satisfying the classical flow conservation constraints ((34)-(36)) and creating coupling constraints for the vehicle routes $r$ in (33), determined by the $x$-variables. The duration and length of the routes are determined in (28) and (29) and then combined in (30) to determine the costs. Of course, the vehicle routes need to satisfy flow conservation as well (see (31)). (37) are the subtour elimination constraints. Constraint (32) ensures that every line is served in both directions. Since this is a rather large program, we prove formally that it is working as intended.

**Theorem 22.** Model 3 is a correct formulation for (cost-opt).

**Proof.** We prove the theorem in the following three steps:

1. For every optimal solution for Model 3 there is a feasible solution for (cost-opt)
2. For every feasible solution for (cost-opt) there is a feasible solution for Model 3
3. The objective values coincide for optimal solutions

**Step 1:** Let $(x,f,cost,dur,length)$ be an optimal solution for Model 3. We construct a feasible solution to (cost-opt), i.e., a feasible public transport plan $(L,R)$. For the line concept, set

$$f_l = \sum_{r \in [R]} x_{l,r}$$

and let $l \in L$ if $f_l > 0$. Due to (32), this is well defined and only both or no direction of a line will be served. The vehicle routes for the vehicle schedule can easily be constructed using the $x$ variables.

In order to check feasibility of the line concept, we transform the passenger weights $f$ in the EAN to weights $w_p$ in the PTN for each passenger. Then for every $e \in E$ it holds that

$$\sum_{p \in P_{\text{all}}}^{e \in p} w_p = \sum_{l' \in L'} \sum_{a \in A_{l'}}^{a \text{ corr. to } e} \sum_{u,v \in V} f_{a,(u,v)}$$

$$\leq \text{Cap} \sum_{l' \in L'} \sum_{a \in A_{l'}}^{a \text{ corr. to } e} \sum_{r \in [R]} x_{l',r}$$

$$= \text{Cap}|\{l \in L : e \in l\}|$$

Therefore constraint (1) is satisfied and the constructed line concept is feasible. Regarding the feasibility of the vehicle schedule, the subtour elimination constraints (37) ensure that all lines in a route are distinct and every line is covered exactly once due to the construction of $L$ and the optimality of the solution.
Step 2: Let now \((\mathcal{L}, \mathcal{R})\) be a feasible public transport plan with corresponding passenger paths \(P_{\text{all}}\). Then there exist passenger flows \(f_{u,v}\) in the EAN for all OD-pairs such that

\[
\sum_{u,v \in V} f_{u,(u,v)} \leq \text{Cap} \quad a \in A_{l'}, l' \in \mathcal{L}', \tag{39}
\]

since (1) is satisfied and passengers can choose an arbitrary line for each edge in their path. Set \(x\) variables according to \(\mathcal{R}\), i.e., set \(x_{l',r} = 1\) iff line \(l' \in \mathcal{L}'\) is covered in \(r \in \mathcal{R}\) and \(x_{(l', l''),r} = 1\) iff line \(l''\) is directly behind line \(l'\) in \(r \in \mathcal{R}\). Then the constructed solution is feasible for Model 3, since (34)-(36) are satisfied due to the construction of \(f\) and \(P_{\text{all}}\), (37) holds since the given vehicle routes are feasible, (33) holds due to the construction of \(x\) and (39), (32) holds due to the construction of \(x\) and \(\mathcal{L}'\), (31) holds due to the construction of \(x\) and the feasibility of \(\mathcal{R}\). The remaining constraints (28)-(30) are no feasibility constraints.

Step 3: The objective value of the solutions does not change when using above constructions. Note, that the \(\lceil \cdot \rceil\) operator is replaced by multiplication with \(\frac{1}{T}\) and the integer constraint of \(\text{dur}_r\). Together, these three steps prove the correctness of the proposed Model 3.

With this, the following relations between the Models 1, 2 and 3 can be formally stated.

Corollary 23. Model 1 and Model 2 are relaxations of Model 3.

Proof. Directly follows from the proof of Theorems 3, 12 and 22.

Model 3 is too large to be solved for realistic instances. As can be seen in the computational experiments in Section 7, the integrated problem cannot be solved – even for instances of small size. This is due to its enormous number of variables including a trip for every possible line in the network. Nevertheless, Model 3 can be used if enough variables are fixed. We hence can combine it with Model 2 by fixing the lines in Model 3 to the optimal lines computed by Model 2. This means that we only need to consider the constraints (28)-(31) and (37), additionally guaranteeing that every trip in \(\mathcal{L}'\) is covered exactly once. The result is a tractable model for medium-sized instances.

Other possibilities to reduce the size of Model 3 would be to start with a line pool of limited size (e.g., as generated in [GHS17] or from Model 2) or to use column generation approaches as in [BGP07].

7 Experiments

In the computational experiments we implemented the three proposed models with the open source library LinTim (see [APS+, GSS13, SAP+18]) and tested them on four different datasets. These datasets are described in Table 1 and depicted in Figure 6.

We implemented Model 1, Model 1*, Model 2, Model 2* and Model 3 using Gurobi 8.0 as a MIP solver with default settings. We tested all implementations on a compute server (6 cores of Intel(R) Xeon(R) CPU X5650 @ 2.67GHz, 78 GB RAM) with a time limit of 3 hours per test case. For each model and each
instance we considered two different cases: Either $L_{\text{turn}} = L_{\text{wait}}$ or $L_{\text{turn}} > L_{\text{wait}}$ to distinguish the cases where Model $1^*$ is able to find an optimal solution and where it is not. We obtained the results depicted in Tables 2 and 3. A symbol ◦
denotes that the problem has not been solved to optimality and hence only the
best found upper or lower bound is presented.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Model 1</th>
<th>Model 1*</th>
<th>Model 2</th>
<th>Model 2*</th>
<th>lb</th>
<th>ub</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>130</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>Toy</td>
<td>1424</td>
<td>1424</td>
<td>1424</td>
<td>1696</td>
<td>1270°</td>
<td>1460°</td>
</tr>
<tr>
<td>Grid</td>
<td>1034</td>
<td>1034</td>
<td>1034</td>
<td>1034</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Germany</td>
<td>73321°</td>
<td>84694°</td>
<td>54148°</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 2: Objective values for the case of $L_{\text{turn}} = L_{\text{wait}}$

<table>
<thead>
<tr>
<th>Instance</th>
<th>Model 1</th>
<th>Model 1*</th>
<th>Model 2</th>
<th>Model 2*</th>
<th>lb</th>
<th>ub</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>80</td>
<td>130</td>
<td>130</td>
<td>130</td>
<td>130</td>
<td>130</td>
</tr>
<tr>
<td>Toy</td>
<td>1424</td>
<td>1474</td>
<td>1424</td>
<td>1696</td>
<td>1288°</td>
<td>1539°</td>
</tr>
<tr>
<td>Grid</td>
<td>1034</td>
<td>1134</td>
<td>1030°</td>
<td>1140</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Germany</td>
<td>74462°</td>
<td>85612°</td>
<td>54148°</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 3: Objective values for the case of $L_{\text{turn}} > L_{\text{wait}}$

For each of the three models there exist two columns. The left column contains
a lower bound to (cost-opt), whereas the right column contains an upper bound,
i.e., the objective value of the best found feasible solution.

We observe for Model 1 that in the case $L_{\text{turn}} = L_{\text{wait}}$ it almost always finds the
optimal objective value within the specified time limit of 3 hours. Only in our
biggest instance we cannot get an optimal solution within the time limit (we
still have a gap of 13.7% here). For the case $L_{\text{turn}} > L_{\text{wait}}$ there exists a gap
between the lower and the upper bound of Model 1, but this model still obtains
the best solutions.

Model 2 can solve the two smallest instances easily, but starts having trouble
with the time limit for Grid. For Germany it is not able to find a feasible solution
within the specified time limit. Regarding the solution quality, we see that the
lower bound given by Model 2 is only in a single case sharper than the lower
bound given by Model 1. On the other hand, the upper bounds found by Model
2* never have smaller objective values than Model 1*. Note, that the values for
$L$ provided by Theorem 20 are close to the number of used lines in the optimal
solutions found by Model 2*, e.g., for dataset Grid, $L_{\text{ub}}$ is 15 and 13 lines are
used in the computed optimal solution.

Model 3 is already on the toy instance not able to find an optimal solution
within 3 hours. The obtained objective values for Linear and the bounds for
Toy are consistent with the values given in Models 1 and 2. For the bigger
instance, even the precomputation of the complete line pool for Model 3 was
not possible anymore.

We illustrate our results on the dataset Grid (see [FHSS17, FOR]) and compare
them to previously known solutions on this dataset. All solutions are evaluated
Figure 7: Multiple solutions for Grid (see [FOR]), evaluated by their cost per hour and traveling time (perceived journey time meaning traveling time plus a time penalty for every occurring transfer). With our models we were able to find a cost-minimal solution. Its objective value is depicted by a red line.

with respect to their costs and their traveling times. The solutions shown in Figure 7 have been computed sequentially, contrary to the integrated approach presented in this work. We see that the sequential solutions with smallest costs are A4 (computed in [PSSS17]) and P5 (computed in [Lie18]). For this instance of the dataset Grid it holds that $L_{\text{turn}} = L_{\text{wait}}$. Hence, we were able to compute a cost-minimal solution by using Model 1. Its objective value is depicted as a red line, since the traveling times are not computed for this model. The optimal solution improves the costs by 23% compared to the best existing solution.

The traveling time of the cost-minimal solution is hard to evaluate: Assigning passengers to travel on their shortest paths in the EAN, as done for the other solutions in Figure 7, would lead to a traveling time of only 20.57. We did not depict this objective value in the figure since in this solution the passengers are far away from using the paths computed for them in Model 1 and hence the solution would have heavily overloaded vehicles. On the other hand, using a capacitated evaluation, i.e., finding a system optimal solution for the passengers, where no overcrowding in the vehicles occur, will lead to a perceived travel time of 23.86. But since this evaluation is not consistent with the evaluation strategy used for the other solutions depicted in Figure 7, we chose to only depict the cost value in the figure.

We finally investigate the influence of valid inequalities introduced in Lemma 4 on the runtime of Model 1. We restricted this investigation to Grid, since the runtime for the smallest two instances is already less than a second, and for Germany it is already non-trivial to determine “good” cuts of the network. For Grid, however, we took all horizontal and vertical cuts of the network, whose PTN is depicted in Figure 6, into the model. With this improvement we were able to speed up the solution process significantly with respect to runtime and
number of explored MIP nodes, as can be seen in Table 4.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>No Cuts</th>
<th>Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nodes explored</td>
<td>46557</td>
<td>2398</td>
</tr>
<tr>
<td>Runtime in sec</td>
<td>23.18</td>
<td>8.99</td>
</tr>
</tbody>
</table>

Table 4: Runtime improvements with Lemma 4 on Grid for $L_{\text{turn}} > L_{\text{wait}}$

8 Outlook

In this work we propose three models to compute cost-optimal public transport plans. For an overview, see Table 5. For the first two models we derived optimality conditions and bounds to the optimal solution. With the third model we present an IP formulation for the integrated exact model. The computational experiments show that the implementation of the models is computationally tractable.

<table>
<thead>
<tr>
<th>Model</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1</td>
<td>Very low computation time, able to provide solutions for real-world instances</td>
<td>Low theoretical bound quality</td>
</tr>
<tr>
<td>Find optimal solution under (weak assumptions)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model 2</td>
<td>Low computation time</td>
<td>May not find optimal solution for non line-pure vehicle schedules</td>
</tr>
<tr>
<td>Finds optimal line-pure solution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Better bound quality than Model 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model 3</td>
<td>Integrated model, finding the optimal solution to the problem</td>
<td>High computation time, only able to provide solutions for small instances</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Overview of the different models presented in this paper

Model 1 is able to compute cost-optimal solutions up to Grid outperforming previous approaches to tackle this problem. For large networks the model provides bounds of good quality in a reasonable amount of time. Model 2 finds optimal line-pure public transport plans and constitutes a trade-off between computation time and solution quality. Finally, Model 3 yields a cost-optimal public transport plan without requiring any further assumptions.

For future work we plan to sharpen the formulation of Model 1 by identifying good cuts. It would hopefully be the case that better cuts lead to a further decrease of the computation time, especially for the large instances. Furthermore it would be interesting to not only find a solution with minimal costs, but to find a lexicographic solution, i.e., the cost-optimal solution with
the best traveling time for the passengers. To this end, we can include the passengers’ traveling time in Model 3 which will most likely further increase the computation time of the model. To use this model effectively, more work in speed-up techniques is necessary. Promising ideas include column generation and decomposition techniques, similar to the methods presented in [LPSS].

References


