# Integrating Passengers' Assignment in Cost-Optimal Line Planning*

**Markus Friedrich[1], Maximilian Hartl[2], Alexander Schiewe[3], and Anita Schöbel[4]**

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

## 1 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

* This work was partially supported by DFG under SCHO 1140/8-1.


17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017).
Editors: Gianlorenzo D'Angelo and Twan Dollevoet; Article No. 5; pp. 5:1–5:16
Open Access Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

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 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 $\mathcal{L}$ together with their frequencies $f_l$ for all $l \in \mathcal{L}$. For the line planning problem, a set of potential lines, the so-called *line pool* $\mathcal{L}^0$ 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$= (V, E)$, $W_{uv}$ 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 For every edge $e \in E$ compute the lower edge frequency $f_e^{\min} := \lceil \frac{w_e}{\text{Cap}} \rceil$

3 Solve the line planning problem LineP$(f^{\min})$ and receive $(\mathcal{L}, f_l)$

**Algorithm 2:** Passengers' assignment algorithm.

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

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

  Compute a set of paths $P_{uv}^1, \ldots, P_{uv}^{N_{uv}}$ from $u$ to $v$ in the PTN

  Estimate weights for the paths $\alpha_{uv}^1, \ldots, \alpha_{uv}^{N_{uv}} \geq 0$ with $\sum_{i=1}^{N_{uv}} \alpha^i = 1$

**end**

**for** *every* $e \in E$ **do**

  Set $w_e := \sum_{u,v \in V} \sum_{\substack{i=1\ldots N_{uv}: \\ e \in P_{uv}^i}} \alpha_{uv}^i W_{uv}$

**end**

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

$$\text{cost}_l = c_{km} \sum_{e \in l} d_e + c_{\mathit{fix}}, \tag{1}$$

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 $c_{km} = 1$.

The demand is usually given in form of an OD-matrix $W \in \mathbb{R}^{|V| \times |V|}$, where $W_{uv}$ is the number of passengers who wish to travel between the stops $u, v \in V$. 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 $W_{uv} > 0$ the passenger-demand is assigned to possible paths in the PTN. Using these paths, for every edge $e \in E$ the *traffic loads* are computed. Given the capacity Cap of a vehicle, one can determine $f_e^{\min} := \lceil \frac{w_e}{\text{Cap}} \rceil$, i.e., how many vehicle trips are needed along edge $e$ to satisfy the given demand. These values $f_e^{\min}$ are called *lower edge frequencies*. They are finally used as input for determining the lines and their frequencies, Algorithm 1.

The problem LineP$(f^{\min})$ is the basic *cost model for line planning*:

$$\min \left\{ \sum_{l \in \mathcal{L}^0} f_l \cdot \text{cost}_l : \sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq f_e^{\min} \text{ for all } e \in E, f_l \in \mathbb{N} \text{ for all } l \in \mathcal{L}^0 \right\}. \tag{2}$$

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 $\alpha_{uv}^i$; 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 $P_{uv}^1 = P_{uv}$ 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 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 $SP_{uv} := \sum_{e \in P_{uv}} d_e$ 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 $e \in E$:

$$\sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq \left\lceil \frac{w_e}{\text{Cap}} \right\rceil \Longleftrightarrow \text{Cap} \sum_{l \in \mathcal{L}^0 : e \in l} f_l \geq w_e,$$

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

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

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

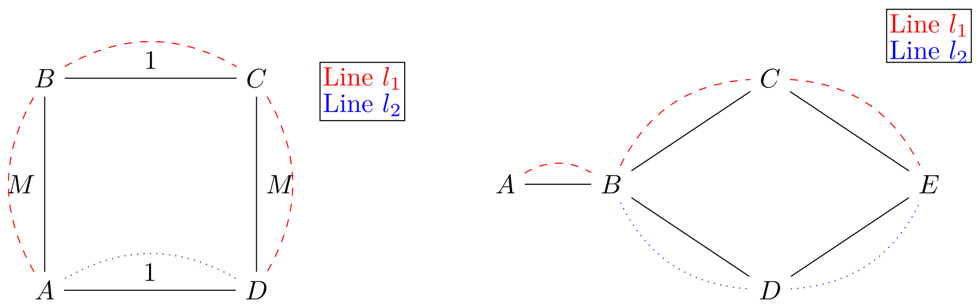
Note that the paths determined in Algorithm 3 will most likely not be the paths the passengers really take after (3) is solved and the line concept is known. This is known and has been investigated in case that the travel time of the passengers is the objective function: Travel time models such as [19] 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 $g^{\text{cost}} = c_{\text{fix}} + 2M + 1$. The sum of travel times for the passengers in this solution is $g^{\text{time}} = (\text{Cap} - 1) * (2M + 1) + 1$.




**(a)** Infrastructure network for Example 1. **(b)** Infrastructure network for Example 3.

**Figure 1** Example infrastructure networks.

- 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 $g^{\text{time}} = (\text{Cap} - 1) * 1 + 1 = \text{Cap}$.

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 $\text{SP}_{uv}$. The integrated problem can be modeled as integer program (LineA)

$$
\begin{aligned}
\text{(LineA)}\ \ \min g^{\text{cost}} &:= \sum_{l \in \mathcal{L}^0} f_l \left( \sum_{e \in E} d_e + c_{fix} \right) \\
\text{s.t.}\ \ \text{Cap} \sum_{l \in \mathcal{L}^0 : e \in l} f_l &\geq \sum_{u,v \in V} x_e^{uv} \text{ for all } e \in E \\
\Theta x^{uv} &= b^{uv} \text{ for all } u, v \in V \\
\sum_{e \in E} d_e x_e^{uv} &\leq \beta SP_{uv} W_{uv} \\
f_l &\in \mathbb{N} \text{ for all } l \in \mathcal{L}^0 \\
x_e^{uv} &\in \mathbb{N} \text{ for all } l \in \mathcal{L}^0
\end{aligned}
$$

where

- $x_e^{uv}$ 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

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

- $b^{uv} \in \mathbb{R}^{|V|}$ 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.** *(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

$$w_e := \sum_{u,v \in V: e \in P_{uv}} W_{uv}. \tag{4}$$

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 $\mathcal{L}^0 = \{l_1 = ABCE, l_2 = BDE\}$. Since the lines have different lengths their costs differ: $cost_{l_1} = 3$ and $cost_{l_2} = 2$ (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.

**Input:** PTN$=(V,E)$, $W_{uv}$ for all $u,v \in V$
**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 edge lengths $d$
**end**
**for** *every* $e \in E$ **do**
  Set $w_e := \sum_{\substack{u,v \in V \\ e \in P_{uv}}} W_{uv}$
**end**

▶ **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 $c_{fix} = 0$. Let $f_l$, $l \in \mathcal{L}$, be the cost optimal line concept for LineP($w$) and $f'_l$, $l \in \mathcal{L}'$, be the cost optimal line concept for LineP($w'$). Then $g^{\text{cost}}(w) \leq |OD| g^{\text{cost}}(w')$.*

**Proof.** See [9]. ◀

If we drop the assumption of choosing a common path for every OD-pair, the factor increases to the number $|W|$ of passengers. However, if we solve the relaxation of LineP($w$) the passengers' assignment has no effect:

▶ **Theorem 6.** *Consider a line planning problem with complete pool and without fixed costs (i.e. $c_{fix} = 0$). Then the objective value of the LP-relaxation of LineP($w$) is independent of the choice of the traffic assignment if it is shortest-path based. More precisely:*

*Let $w$ and $w'$ be two shortest-path based traffic assignments with $\tilde{g}^{cost}(w), \tilde{g}^{cost}(w')$ the optimal values of the LP-relaxations of LineP($w$) and LineP($w'$). Then $\tilde{g}^{cost}(w) = \tilde{g}^{cost}(w')$.*

**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 *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 lenghts.

**Algorithm 5:** Passengers' Assignment: Reduction.

**Input:** PTN= $(V, E)$, $W_{uv}$ for all $u, v \in V$
i := 0
$w_e^0 := 0\, \forall e \in E$
**repeat**
  **for** *every* $u, v \in V$ *with* $W_{uv} > 0$ **do**
    Compute a shortest path $P_{uv}^i$ from $u$ to $v$ in the PTN, w.r.t.

$$\text{cost}_i(e) = d_e + \gamma \cdot \frac{d_e}{\max\{w_e^{i-1}, 1\}}$$

  **end**
  **for** *every* $e \in E$ **do**
    Set $w_e^i := \sum_{\substack{u,v \in V \\ e \in P_{uv}^i}} W_{uv}$
  **end**
  i = i + 1
**until** $\sum_{e \in E}(w_e^{i-1} - w_e^i)^2 < \epsilon \quad or \quad i > max\_it$;
Compute a shortest path $P_{uv}$ from $u$ to $v$ in the PTN, w.r.t.

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

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

## 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 $\mathcal{L}^0$, CGN=$(\tilde{V}, \tilde{E})$ is a graph in which every node is a pair $(v, l)$ of a station $v \in V$ and a line $l \in \mathcal{L}^0$ 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 $(u, v) \in E$ 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 $u,v \in V$
i := 0
**repeat**
  i = i + 1
  $w_e^i := w_e^{i-1} \, \forall e \in E$
  **for** *every* $u,v \in V$ *with* $W_{uv} > 0$ **do**
    Compute a shortest path $P^i_{uv}$ from $u$ to $v$ in the PTN, w.r.t.

$$\text{cost}_i(e) = \max\{d_e \cdot \left(1 - \gamma \cdot (w_e^{i-1} \mod \text{Cap})/(\text{Cap})\right), 0\}$$

    **for** *every* $e \in P^{i-1}_{uv}$ **do**
      Set $w_e^i := w_e^i - W_{uv}$
    **end**
    **for** *every* $e \in P^i_{uv}$ **do**
      Set $w_e^i := w_e^i + W_{uv}$
    **end**
  **end**
**until** $\sum_{e\in E}(w_e^{i-1} - w_e^i)^2 < \epsilon$ *or* $i > max\_it$;

**Algorithm 7:** CGN routing for Algorithm 4.

**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.

$$\text{cost}(\tilde{e}) = \begin{cases} d_e & \text{if } \tilde{e} \text{ is a driving edge which corresponds to } e \\ \text{pen} & \text{if } \tilde{e} \text{ is a transfer edge, where pen is a transfer penalty} \end{cases}$$

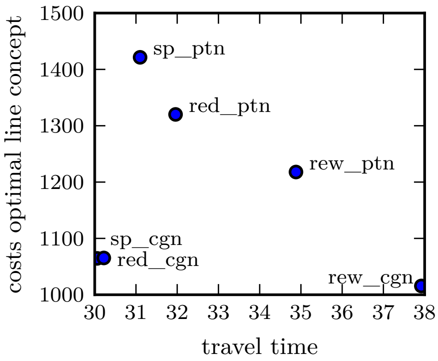
**end**
**for** *every* $e \in E$ **do**
  Set $w_e := \sum_{\substack{\tilde{e}\in\tilde{E}: \\ \tilde{e} \text{ corr. to } e}} \sum_{\substack{u,v\in V: \\ \tilde{e}\in\tilde{P}_{uv}}} W_{uv}$
**end**

where we use

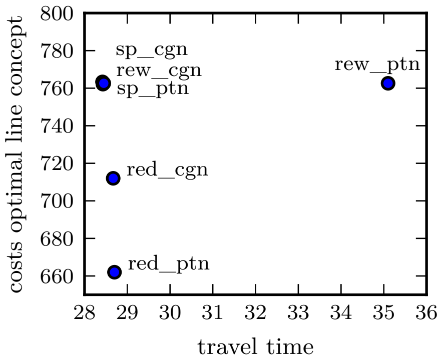
$$\text{cost}(\tilde{e}) = \begin{cases} cost_i(e) & \text{if } \tilde{e} \text{ is a driving edge which corresponds to } e \\ \text{pen} & \text{if } \tilde{e} \text{ is a transfer edge, where pen is a transfer penalty} \end{cases}$$

as costs in the CGN. We still compare the weights $w_e^i$ and $w_e^{i-1}$ 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 $w_e^i$ in the PTN at the end of every iteration in order to compare the weights $w_e^i$ and $w_e^{i-1}$ in the PTN for deciding if we end or repeat the loop. For the detailed version see Algorithm 9 in Appendix A.




**(a)** Solution results for a line pool with 33 lines.




**(b)** Solution results for a line pool with 275 lines.

**Figure 2** Solution results for a small and a big line pool.

# 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:

- $g^{\text{cost}} = \sum_{l \in \mathcal{L}^0} f_l \cdot \text{cost}_l$, i.e., the objective function of (LineP($w$)) and (LineA) that we used before, and
- $g^{\text{time}} = \sum_{u,v \in V} SP_{uv} + \text{pen} \cdot \#\text{transfers}$, 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.

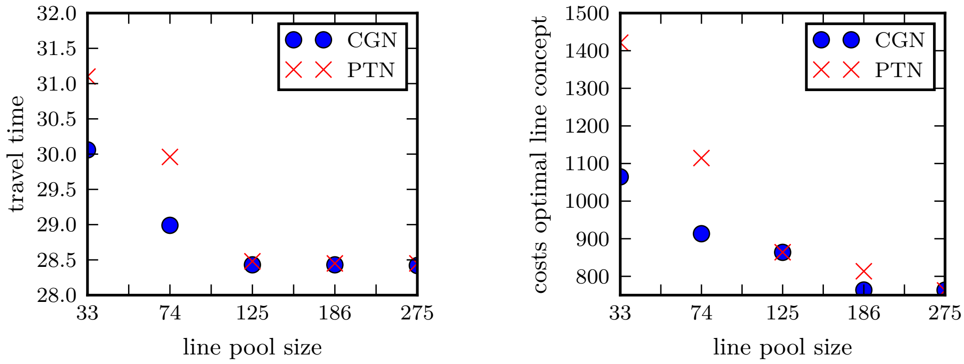
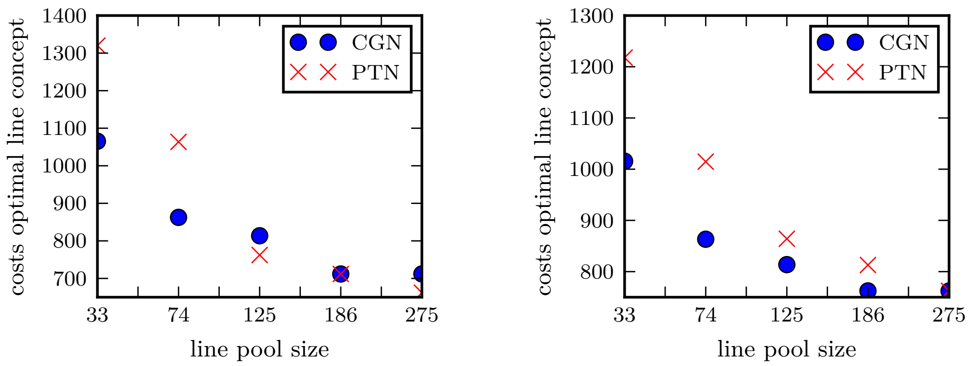



**Figure 3** Travel time and cost of Shortest Path solutions for increasing line pool size.




**(a)** Cost of Reduction.

**(b)** Cost of Reward.

**Figure 4** Cost of Reward and Reduction solutions for increasing line pool size.

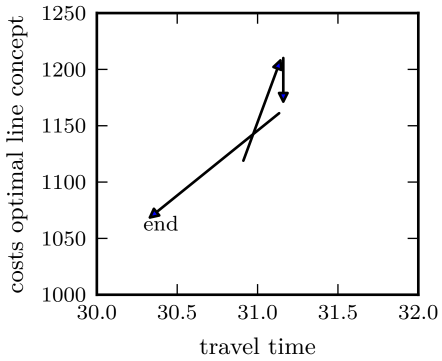
Figure 2b shows the behaviour for a larger line pool. Still, the solution with lowest travel time is received by Shortest Path, and it is still better in the CGN than in the PTN but the difference is less significant compared to the small line pool. The lowest cost for larger line pools are received 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

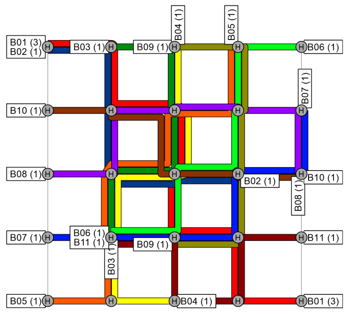
We have already seen that for larger line pools, cost optimal solutions are obtained by Reduction and for smaller line pools by Reward. Figures 3 and 4 now study further the dependence of the line pool.

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.




**(a)** Iterations for Reduction, $\gamma = 75$, 186 lines.




**(b)** Solution evaluated by VISUM.

**Figure 5**

### 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


1 S. Albert, J. Pätzold, A. Schiewe, P. Schiewe, and A. Schöbel. LinTim – Integrated Optimization in Public Transportation. Homepage. see `http://lintim.math.uni-goettingen.de/`.

2 R. Borndörfer, M. Grötschel, and M. E. Pfetsch. A column generation approach to line planning in public transport. *Transportation Science*, 41:123–132, 2007.

3 M. R. Bussieck. *Optimal lines in public transport*. PhD thesis, Technische Universität Braunschweig, 1998.

4 M. R. Bussieck, P. Kreuzer, and U. T. Zimmermann. Optimal lines for railway systems. *European Journal of Operational Research*, 96(1):54–63, 1996.

5 M. T. Claessens. De kost-lijnvoering. Master's thesis, University of Amsterdam, 1994. (in Dutch).

6 M. T. Claessens, N. M. van Dijk, and P. J. Zwaneveld. Cost optimal allocation of rail passenger lines. *European Journal on Operational Research*, 110:474–489, 1998.

7 H. Dienst. *Linienplanung im spurgeführten Personenverkehr mit Hilfe eines heuristischen Verfahrens*. PhD thesis, Technische Universität Braunschweig, 1978. (in German).

8 M. Friedrich, M. Hartl, A. Schiewe, and A. Schöbel. Angebotsplanung im öffentlichen Verkehr – planerische und algorithmische Lösungen. In *Heureka'17*, 2017.

9 M. Friedrich, M. Hartl, A. Schiewe, and A. Schöbel. Integrating passengers' assignment in cost-optimal line planning. Technical Report 2017-5, Preprint-Reihe, Institut für Numerische und Angewandte Mathematik, Georg-August Universität Göttingen, 2017. URL: `http://num.math.uni-goettingen.de/preprints/files/2017-5.pdf`.

10 P. Gattermann, J. Harbering, and A. Schöbel. Line pool generation. *Public Transport*, 2016. accepted.

11 M. Goerigk, M. Schachtebeck, and A. Schöbel. Evaluating line concepts using travel times and robustness: Simulations with the lintim toolbox. *Public Transport*, 5(3), 2013.

12 J. Goossens, C. P. M. van Hoesel, and L. G. Kroon. On solving multi-type railway line planning problems. *European Journal of Operational Research*, 168(2):403–424, 2006.

13 R. Hüttmann. *Planungsmodell zur Entwicklung von Nahverkehrsnetzen liniengebundener Verkehrsmittel*, volume 1. Veröffentlichungen des Instituts für Verkehrswirtschaft, Straßenwesen und Städtebau der Universität Hannover, 1979.

14 S. Kontogiannis and C. Zaroliagis. Robust line planning through elasticity of frequencies. Technical report, ARRIVAL project, 2008.

15 S. Kontogiannis and C. Zaroliagis. Robust line planning under unknown incentives and elasticity of frequencies. In Matteo Fischetti and Peter Widmayer, editors, *ATMOS 2008 – 8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, volume 9 of *Open Access Series in Informatics (OASIcs)*, Dagstuhl, Germany, 2008. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/OASIcs.ATMOS.2008.1581`.

16 PTV. Visum. `http://vision-traffic.ptvgroup.com/de/produkte/ptv-visum/`.

17 M. Sahling. Linienplanung im öffentlichen Personennahverkehr. Technical report, Universität Karlsruhe, 1981.

**18** A. Schöbel. Line planning in public transportation: models and methods. *OR Spectrum*, 34(3):491–510, 2012.

**19** A. Schöbel and S. Scholl. Line planning with minimal travel time. In *5th Workshop on Algorithmic Methods and Models for Optimization of Railways*, number 06901 in Dagstuhl Seminar Proceedings, 2006.

**20** A. Schöbel and S. Schwarze. A Game-Theoretic Approach to Line Planning. In *ATMOS 2006 – 6th Workshop on Algorithmic Methods and Models for Optimization of Railways, September 14, 2006, ETH Zürich, Zurich, Switzerland, Selected Papers*, volume 6 of *Open Access Series in Informatics (OASIcs)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2006. doi:10.4230/OASIcs.ATMOS.2006.688.

**21** A. Schöbel and S. Schwarze. Finding delay-resistant line concepts using a game-theoretic approach. *Netnomics*, 14(3):95–117, 2013. doi:10.1007/s11066-013-9080-x.

**22** C. Simonis. Optimierung von Omnibuslinien. *Berichte stadt–region–land, Institut für Stadtbauwesen, RWTH Aachen*, 1981.

**23** H. Sonntag. *Linienplanung im öffentlichen Personennahverkehr*, pages 430–439. Physica-Verlag HD, 1978.

**24** H. Wegel. *Fahrplangestaltung für taktbetriebene Nahverkehrsnetze*. PhD thesis, TU Braunschweig, 1974. (in German).

**25** P. J. Zwaneveld. *Railway Planning – Routing of trains and allocation of passenger lines*. PhD thesis, School of Management, Rotterdam, 1997.

**26** P. J. Zwaneveld, M. T. Claessens, and N. M. van Dijk. A new method to determine the cost optimal allocation of passenger lines. In *Defence or Attack: Proceedings of 2nd TRAIL Phd Congress 1996, Part 2*, Delft/Rotterdam, 1996. TRAIL Research School.

# 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

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

$\mathrm{i} := 0$

$w_e^0 := 0 \, \forall e \in E$

**repeat**

  $\mathrm{i} = \mathrm{i} + 1$

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

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

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

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

  **end**

  **for** *every* $e \in E$ **do**

    Set $w_e^i := \sum_{\substack{\tilde{e} \in \tilde{E} \\ e \text{ corr. to } \tilde{e}}} \sum_{\substack{u,v \in V \\ \tilde{e} \in \tilde{e}_{uv}^i}} W_{uv}$

  **end**

**until** $\sum_{e \in E} (w_e^{i-1} - w_e^i)^2 < \epsilon$ *or* $i > max\_it$;

**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.

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

**end**

**for** *every* $e \in E$ **do**

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

**end**

---

**Algorithm 9:** CGN routing version of Algorithm 6.

---

**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

$w^0_{\tilde{e}} := 0 \, \forall \tilde{e} \in \tilde{E}$

**repeat**

  i = i + 1

  $w^i_{\tilde{e}} := w^{i-1}_{\tilde{e}} \, \forall \tilde{e} \in \tilde{E}$

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

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

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

    **for** *every* $\tilde{e} \in \tilde{P}^{i-1}_{uv}$ **do**

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

    **end**

    **for** *every* $\tilde{e} \in \tilde{P}^i_{uv}$ **do**

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

    **end**

  **end**

  **for** *every* $e \in E$ **do**

    Set $w_e := \sum_{\substack{\tilde{e} \in \tilde{E}: \\ \tilde{e} \text{ corr. to } e}} \sum_{\substack{u,v \in V: \\ \tilde{e} \in \tilde{P}_{uv}}} W_{uv}$

  **end**

**until** $\sum_{e \in E} (w^{i-1}_e - w^i_e)^2 < \epsilon \quad or \quad i > max\_it$;