Lecture Notes

Line Planning

Marc Pfetsch

Zuse Institute Berlin

pfetsch@zib.de

last change: 02/01/2007

TECHNISCHE UNIVERSITÄT BERLIN
Fakultät II, Institut für Mathematik
WS 2006/07

GANZZAHLIGE OPTIMIERUNG IM
ÖFFENTLICHEN VERKEHR

Ralf Borndörfer, Christian Liebchen, Marc Pfetsch

Contents

Chapter 4

Line Planning

4.1 Introduction

In this chapter we will treat mixed integer programming models developed for the *line planning problem* in public transport. The goal is to design line routes and their frequencies in a street/track network such that a transportation demand, given by an origin-destination matrix (OD-matrix), can be routed. The frequency of a line is supposed to indicate a basic timetable period and controls the lines' transportation capacity. There are two competing objectives: on the one hand to minimize the operating costs of lines and on the other hand to minimize user discomfort. User discomfort is usually measured by the total passenger traveling time or the number of transfers during the ride, or both. The planner has to balance these two objectives. Indeed, on the one extreme one would install a line for every passenger, which would be unaffordably expensive, on the other extreme a reduced core system would inflict extremely long traveling times for the passengers.

There are many approaches for line planning. A more detailed overview can be found in [1]. Here we only treat integer programming models for line planning. A very rough historically overview over line planning is as follows.

In the 1960ies experiments were started for line planning by constructing long line routes by adjoining small pieces, see [11]. In the 1980ies such routes were improved through local search, see [12]. Integer programming came into play in 1997, when Bussieck, Kreuzer, and Zimmermann [3] introduced their model for maximizing the number of direct travelers and Claessens, van Dijk, and Zwaneveld [4] presented their cost minimizing approach. The later model was improved by Goossens, van Hoesel, and Kroon [9, 8]. These models assume a so-called system split (see below), in which one assumes that the number of passengers that want to travel on each edge is known in advance. This assumption was dropped by Borndörfer, Grötschel, and Pfetsch [1] and Schöbel and Scholl [13]. In their models passengers are allowed to be freely routed in the network by using a path bases formulation for the passenger flow. This makes it necessary to use column generation to solve these models. We will present all four (mixed integer programming) models in this chapter.

Apart from this body of work, there are articles on line planning under the name of *transit network design*, which usually present meta heuristic approaches and often try to embed additional features like stop positioning.

In contrast to the some other problems discussed in this course, there are to the best of our knowledge no optimization methods in practical use for line planning. One reason is that strategic planning problems like line planning are usually multi-criteria optimization problems. This makes it much harder to see the value of an optimization approach. The vision of the methods presented in this chapter is to provide a decision support tool for the planner. With such tools many variants can be evaluated, which ultimately should result in better line plans, both for the customer and for the transportation companies.

4.1.1 Origin-Destination Matrices and System Split

All models of this chapter assume that an OD-matrix is given. Each entry in an OD-matrix gives the number of passengers that want to travel from one point in the network to another point within a fixed time horizon. Usually, OD-data are aggregated over one day, but it is similarly appropriate to consider, for instance, peak traffic in rush hours.

It is well known that such origin-destination data have certain deficiencies. For instance, OD-matrices depend on the geometric discretization used, they are highly aggregated, they give only a snapshot type of view, it is often questionable how well the entries represent the real situation, and they should only be used when the transportation demand can be assumed to be fixed. However, OD-matrices are at present the industry standard for estimating transportation demand. It is already quite an art and rather costly to assemble this data and there is currently no alternative in sight. All of the discussed models assume that the OD-matrices are fixed, i.e., do not depend on the planning steps that we are performing.

Many of the models in the literature are based on the so-called *system split*. Its starting point is a classification of the links of a transportation system into levels of different speed, as common in railway systems. Assuming that travelers are likely to change to fast levels as early and leave them as late as possible, the passengers are distributed onto several paths in the system, using Kirchhoff-like rules at the transit points, before any lines are known. This fixes the passenger flow on each individual link in the network, i.e., we know that there are ρ_e passengers traveling on edge e; this is also called the *passenger load*.

4.1.2 A Bit of Notation

For line planning problems (LPP) we are given a number M of transportation *modes* (bus, tram, subway, etc.), an undirected multigraph $G = (V, E) = (V, E_1 \dot{\cup} \ldots \dot{\cup} E_M)$ representing a multi-modal transportation network, *terminal sets* $\mathcal{T}_1, \ldots, \mathcal{T}_M \subseteq V$ of nodes for each mode where lines can start and

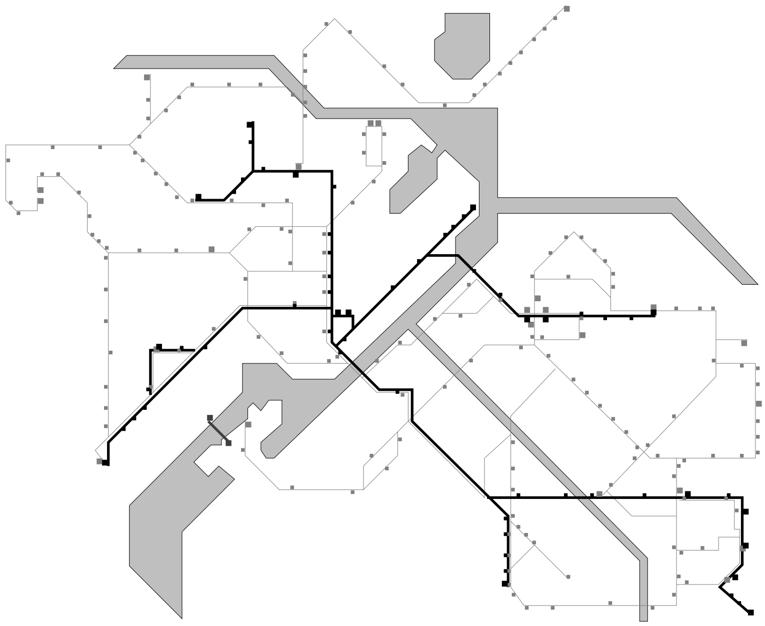

Figure 4.1: Multi-modal transportation network in Potsdam. Black: tram, lightgray: bus, darkgray: ferry, large nodes: terminals, small nodes: stations, grey: rivers and lakes.

end. Denote by $G_i = (V, E_i)$ the subgraph of G corresponding to mode i. See Figure 4.1 for an example network.

The problem formulation further involves a (not necessarily symmetric) *origin-destination matrix* (OD-matrix) $(d_{st}) \in \mathbb{Q}_+^{V \times V}$ of travel demands, i.e., d_{st} is the number of passengers that want to travel from node s to node t. Let $D := \{(s,t) \in V \times V \,:\, d_{st} > 0\}$ be the set of all *OD-pairs*.

A *line of mode* i is a path in G_i connecting two (different) terminals of $\mathcal{T}_i$. Note that paths are always *simple*, i.e., the repetition of nodes is not allowed; it is possible to consider additional constraints on the formation of lines such as a maximum length etc. Let $\mathcal{L}$ be a set of lines (the meaning depends on the models). Finally, we let $\mathcal{L}_e := \bigcup\{\ell \in \mathcal{L} \,:\, e \in \ell\}$ be the set of lines that use edge $e \in E$.

4.2 Models Based on System Split

In this section we discuss models that are based on a system split. Because in this case the number of passengers that want to travel on each edge is known or fixed in advance, these models have a different flavor than the models that allow free routing of passenger paths, see Section 4.3. The system-split models without loss of generality assume that there is only one mode, i.e., $G = (V, E)$ is the transportation graph. They furthermore assume that $\mathcal{L}$ is a pool of some predefined lines from which good lines have to be chosen.

Table 4.1: Notation and terminology for models based on system split.

G	transport network	ρ_e	number of passengers on edge e
$\mathcal{L}$	line pool	$\mathcal{L}_e$	lines using edge e
D	set of OD-pairs	d_{st}	travel demand between s and t
$\underline{\Lambda}_e$	lower frequency bound	$\overline{\Lambda}_e$	upper frequency bound
κ	train capacity	K	vehicle capacity
T_ℓ	turn around time	d_ℓ	geometric length of line ℓ
C^t	train fixed costs	C^c	carriage fixed costs
c^t	train operating costs	c^c	carriage operating costs

4.2.1 The Direct Travelers Approach

The direct travelers approach for line planning was developed by Bussieck, Kreuzer, and Zimmermann [3] (see also Bussieck [2]). The goal is to choose lines from $\mathcal{L}$ such that the number of direct travelers, i.e., passengers that can travel on a line without transfering, is maximized.

Because of the system split there is only one mode and hence the capacities of all lines are equal and are denoted by κ. This determines a minimum frequency $\underline{\Lambda}_e$ that is necessary to transport the passengers that travel on edge $e \in E$, by

$$\underline{\Lambda}_e := \left\lceil \frac{\rho_e}{\kappa} \right\rceil.$$

The *direct travelers* model is the following:

$$\begin{aligned}
\max \; & \sum_{(s,t)\in D} \sum_{\ell\in\mathcal{L}} y_{\ell st} & & \\
s.t. \; & \underline{\Lambda}_e \leq \sum_{\ell\in\mathcal{L}_e} f_\ell \leq \overline{\Lambda}_e & \forall\, e \in E & \qquad (4.1)\\
& \sum_{\ell\in\mathcal{L},\, s,t\in\ell} y_{\ell st} \leq d_{st} & \forall (s,t) \in D & \qquad (4.2)\\
& \sum_{\substack{(s,t)\in D\\ s,t\in\ell}} y_{\ell st} \leq \kappa\, f_\ell & \forall\, \ell \in \mathcal{L} & \qquad (4.3)\\
& f_\ell \in \mathbb{Z}_+ & \forall\, \ell \in \mathcal{L} & \\
& y_{\ell st} \in \mathbb{Z}_+ & \forall\, (s,t) \in D,\ \ell \in \mathcal{L}. &
\end{aligned}$$

Variables f_ℓ determine the frequency of line ℓ, while $y_{\ell st}$ measures the number of passengers that *directly* travel from s to t via line ℓ; clearly, a variable $y_{\ell st}$ is only necessary if $s, t \in \ell$. The objective function gives the total number of direct travelers. Constraints (4.1) ensure that all passengers that travel on each edge can be transported and force upper bounds on the frequency on each edge, e.g., due to safety issues. Constraints (4.2) say that there cannot be more direct travelers on line ℓ from s to t than there is demand. Constraints (4.3) ensure that the frequency of each line is large enough in order to ensure enough capacity to transport all direct travelers that use this line directly.

This model simply ignores all passengers other than direct travelers. They are not accounted for in the capacity constraints and hence the frequencies will in general be too small. One can extend this model to also determine the number of transfers. This can be done by an appropriate construction of the graph – see the "change and go" model of Schöbel and Scholl in Section 4.3.1.

4.2.2 Finding Feasible Line Plans

An interesting subproblem of the above model is the *feasible line plan problem*: Decide whether there exists a subset $\mathcal{L}' \subseteq \mathcal{L}$ and integer frequencies $f_\ell \in \{1, \ldots, \min\{\overline{\Lambda}_e,\ e \in \ell\}\}$, $\ell \in \mathcal{L}'$, such that

$$\underline{\Lambda}_e \leq \sum_{\ell \in \mathcal{L}_e} f_\ell \leq \overline{\Lambda}_e \qquad \text{for all } e \in E. \tag{4.4}$$

Proposition 4.1 (Bussieck (1997))**.** *The feasible line plan problem is $\mathcal{NP}$-complete.*

Proof. For given $\mathcal{L}'$ and frequencies f_ℓ, condition (4.4) can be easily checked in polynomial time. Hence, the problem is in $\mathcal{NP}$.

Consider the $\mathcal{NP}$-complete problem *exact cover by* 3*-sets* (X3C) (see Garey and Johnson [7]): Given is a set X with $3q$ elements and a set $\mathcal{C}$ of 3-element subsets of X. The question is whether there is a subset $\mathcal{C}'$ of $\mathcal{C}$ such that each element of X is contained in *exactly one* set of $\mathcal{C}'$.

Let $(X, \mathcal{C})$ be an instance of X3C. We want construct an instance of the feasible line plan problem such that the answer for it is "yes" if and only if the answer to the X3C for $(X, \mathcal{C})$ is "yes". For this we construct the following graph $G = (V, E)$. For each element $x \in X$ we add a node x and a node x' to G. For ease of notation, we fix an arbitrary order of X. Then for each set $\{x_1, x_2, x_3\} \in \mathcal{C}$ with $x_1 < x_2 < x_3$, we add the two edges $\{x_1', x_2\}$, $\{x_2', x_3\}$, and for each element $x \in X$ we add an edge $\{x, x'\}$. The set of lines is the following:

$$\mathcal{L} := \{(x_1, x_1', x_2, x_2', x_3, x_3') \, : \, \{x_1, x_2, x_3\} \in \mathcal{C},\ x_1 < x_2 < x_3\}.$$

Note that the sequence of nodes in the previous definition form paths in G, see Figure 4.2. To make the specification of our instance for the feasible line plan problem complete, we define

$$\underline{\Lambda}_e = \begin{cases} 1 & \text{if } e = \{x, x'\},\ x \in X \\ 0 & \text{otherwise,} \end{cases} \qquad \text{and} \qquad \overline{\Lambda}_e = 1 \quad \text{for all } e \in E.$$

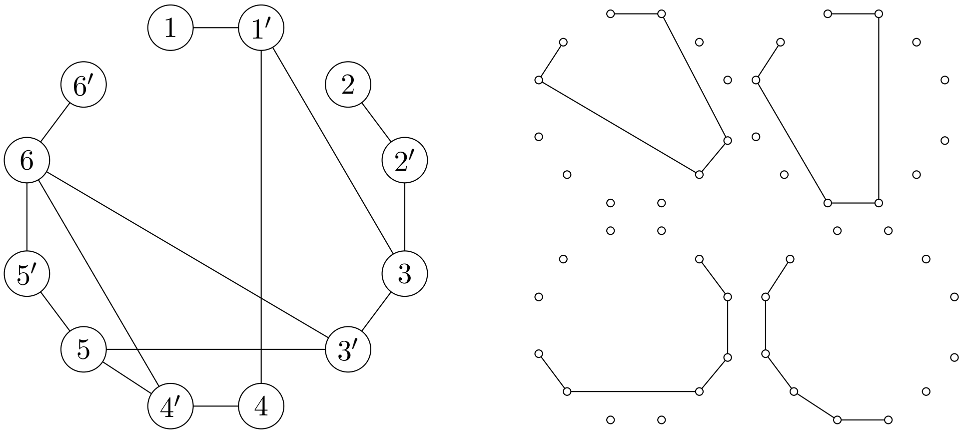

Figure 4.2: Example for the reduction in Proposition 4.1 with $X = \{1, 2, 3, 4, 5, 6\}$ and $\mathcal{C} = (\{1, 3, 6\}, \{1, 4, 6\}, \{2, 3, 5\}, \{4, 5, 6\})$.

Note that, by construction, the frequencies f_ℓ for each line ℓ are either 0 or 1, and will be 1 if a line is in a feasible line plan.

Now assume that $\mathcal{C}' \subseteq \mathcal{C}$ is an exact 3-cover for X. Then

$$\mathcal{L}' := \{(x_1, x_1', x_2, x_2', x_3, x_3') : \{x_1, x_2, x_3\} \in \mathcal{C}',\ x_1 < x_2 < x_3\} \subseteq \mathcal{L}$$

yields a feasible line plan by setting $f_\ell = 1$ for $\ell \in \mathcal{L}'$. This is true since each edge $\{x, x'\}$ is used exactly once and the other edges are used at most once.

Conversely, let $\mathcal{L}' \subseteq \mathcal{L}$ be a feasible line plan with frequencies f_ℓ, $\ell \in \mathcal{L}'$. We can assume that $f_\ell = 1$, since otherwise we can remove ℓ from $\mathcal{L}'$. Then

$$\mathcal{C}' := \{\{x_1, x_2, x_3\} : (x_1, x_1', x_2, x_2', x_3, x_3') \in \mathcal{L}'\} \subseteq \mathcal{C}$$

is an exact 3-cover because of the lower and upper bounds on the frequencies.

This shows that $\mathcal{C}'$ is a solution to X3C if and only if $(\mathcal{L}', \{f_\ell : \ell \in \mathcal{L}'\})$, is a solution for the feasible line plan problem. $\square$

If $\Lambda_e = \infty$ (or large enough), the feasible line plan problem is easy: We only need to check whether there exists a line covering each edge. If this is the case, we obtain a feasible line plan, otherwise there is none.

Corollary 4.2. *The direct travelers approach is $\mathcal{NP}$-hard.*

Proof. Assume that $\mathcal{L}' \subseteq \mathcal{L}$ with frequencies f_ℓ, $\ell \in \mathcal{L}'$, is a feasible line plan. Then we can define $f_\ell = 0$ for $\ell \in \mathcal{L} \setminus \mathcal{L}'$. Furthermore, we set $y_{\ell st} = 0$ for all $\ell \in \mathcal{L}$ and $s, t \in V$. This constitutes a feasible solution for the direct travelers approach. Conversely, if the direct travelers approach has a feasible solution $(\boldsymbol{f}, \boldsymbol{y})$, taking $\mathcal{L}' := \{\ell \in \mathcal{L} : f_\ell > 0\}$ gives us a feasible line plan.

Hence, the direct travelers approach has a feasible solution if and only if there exists a feasible line plan. This shows that if we could solve the direct travelers approach in polynomial time, we could solve the feasible line plan problem in polynomial time. $\square$

To solve the direct travelers model, Bussieck [2] performs preprocessing and adds valid inequalities. He uses data from the intercity networks of Germany and the Netherlands. We will no go into detail here.

4.2.3 Cost Minimal Models

While the direct travelers approach takes care of the passengers interest, namely to travel without transfers, it ignores the cost of the resulting line plan. Claessens, van Dijk, and Zwaneveld [4] introduced a model that takes care of the cost side, but ignores the travelers interests, e.g., the number of transfers. It, however, makes sure that all passengers are transported. The model was later refined by Goossens, van Hoesel, and Kroon [9]. As before we assume that a system split is performed and we are given a line pool $\mathcal{L}$.

A Nonlinear Integer Programming Formulation

We will first present a nonlinear integer programming formulation due to Claessens et al. We will then linearize this model to obtain a *linear* mixed integer programming formulation.

$$\begin{aligned}
\min \sum_{\ell \in \mathcal{L}} & \left\lceil \tfrac{f_\ell T_\ell}{T} \right\rceil (C^t + C^c z_\ell) + d_\ell f_\ell (c^t + c^c z_\ell) & \\
s.t.\ & \underline{\Lambda}_e \le \sum_{\ell \in \mathcal{L}_e} f_\ell \le \overline{\Lambda}_e & \forall e \in E \\
& \sum_{\ell \in \mathcal{L}_e} K f_\ell z_\ell \ge \rho_e & \forall e \in E \\
& \underline{z} \le z_\ell \le \overline{z} & \forall \ell \in \mathcal{L} \\
& f_\ell, z_\ell \in \mathbb{Z}_+ & \forall \ell \in \mathcal{L}.
\end{aligned}$$

Here, variables f_ℓ determine the frequency of line ℓ and z_ℓ gives the number of carriages for each train of line ℓ. The capacity of one carriage of the train is denoted by K, and ρ_e is the load of passengers on edge e as above. The parameters $\underline{z}$, $\overline{z}$ give lower and upper bounds on the number of carriages, respectively. The objective function is the sum of the following terms:

$$\underbrace{\left\lceil \tfrac{f_\ell T_\ell}{T} \right\rceil}_{\#\text{ trains}} \underbrace{(C^t + C^c z_\ell)}_{\text{fixed costs}} + \underbrace{d_\ell f_\ell (c^t + c^c z_\ell)}_{\text{operating costs}}$$

Here, T is the total time horizon, and T_ℓ is the turn-around time for line ℓ, i.e., the total time needed for ℓ in back and forth direction. Then the first term computes the number of trains needed. (Example: for $T = 60$ min, $f_\ell = 2$, $T_\ell = 31$ min, we need two trains, while for $T_\ell = 30$ min, we would need only one.) The other parts are as follows: C^t are fixed costs for one train, C^c are fixed costs for one carriage, d_ℓ is the geographic length of line ℓ, c^t are operating costs for one train per distance, c^c are operating costs for one carriage per distance.

The above model is nonlinear in two aspects. It contains a rounding operator and the product between variables z_ℓ and f_ℓ in the objective and in the capacity constraint.

Corollary 4.3 (Claessens et al. [4]). *The cost minimizing line planning approach is $\mathcal{NP}$-hard.*

Proof. By setting $\underline{z} = \overline{z} = 1$ and K large enough, the cost minimizing line planning approach contains the feasible line plan problem as a subproblem, i.e., if we could solve the cost minimizing line planning problem in polynomial time for these settings, we could solve the feasible line problem in polynomial time. □

A Linearized Cost Minimizing Model

The above model can be linearized as follows. Let $\mathcal{F}$ be the set of feasible frequencies; an example is $\mathcal{F} = \{1, \dots, F\}$, where F is an upper bound on the frequency of a line. Furthermore, let $\mathcal{C}$ be the set of feasible numbers of carriages, e.g., $\mathcal{C} = \{3, 4, 5\}$. Then define the set of all combinations of lines with frequencies and numbers of carriages per train:

$$\mathcal{R} = \mathcal{L} \times \mathcal{F} \times \mathcal{C}.$$

For $r \in \mathcal{R}$, we usually write $r = (r_\ell, r_f, r_z)$ to mark the components. Then we introduce variables y_r, $r \in R$, that determine which combination is used. The above nonlinear model becomes:

$$\begin{aligned}
\min \ & \sum_{r \in \mathcal{R}} k_r \, y_r \\
s.t. \ & \underline{\Lambda}_e \leq \sum_{r \in \mathcal{R}\colon e \in r_\ell} r_f \, y_r \leq \overline{\Lambda}_e && \forall\, e \in E \\
& \sum_{r \in \mathcal{R}\colon e \in r_\ell} K \, r_f \, r_z \, y_r \geq \rho_e && \forall\, e \in E \\
& \sum_{r \in \mathcal{R}\colon r_\ell = \ell} y_r \leq 1 && \forall\, \ell \in \mathcal{L} \\
& y_r \in \{0, 1\} && \forall\, r \in \mathcal{R}.
\end{aligned}$$

Here, we define

$$k_r = \left\lceil \frac{f_{r_\ell} T_{r_\ell}}{T} \right\rceil (C^t + C^c \, r_z) + d_{r_\ell} \, r_f \, (c^t + c^c \, r_z),$$

which is a constant for $r \in \mathcal{R}$.

Table 4.2: Computational results for the linearized cost minimizing model, due to Goossens et al. [9].

Name	n	m	$\lvert\mathcal{L}\rvert$	$\mathcal{F}$	$\mathcal{C}$	var.	cons.	gap
SP98IR	44	44	420	$\{1,2\}$	$\{3,\dots,12\}$	3 651	65	0.00%
SP98IC	41	46	627	$\{1,2\}$	$\{3,\dots,15\}$	10 894	63	0.56%
SP98AR	118	134	913	$\{1,2,3,4\}$	$\{2,\dots,10\}$	15 065	191	0.84%
SP97IC	40	52	831	$\{1,2\}$	$\{3,\dots,15\}$	12 497	60	1.37%
SP97AR	141	177	1 212	$\{1,2,3,4\}$	$\{1,\dots,5\}$	14 101	181	2.70%

This linearization comes at the cost of introducing many variables. In the nonlinear model we used two variables for each line, while here we have to consider each possible combination of frequencies and number of carriages. Using preprocessing, however, this number can be reduced (see below).

Clearly, Corollary 4.3 also holds for the linearized model, which is hence $\mathcal{NP}$-hard as well.

Goossens et al. [9] consider line planning problems for the Dutch intercity network. Table 4.2 shows the relevant parameters; here, n is the number of nodes and m the number of edges in the network. Columns "var." and "cons." are the variables and constraints after preprocessing, respectively. For example: The largest instance (SP97AR) has $1\,212 \cdot 4 \cdot 6 = 29\,088$ variables. This is reduced to 14 101 variables by preprocessing. Goossens et al. further add valid inequalities and perform a branch-and-cut approach. Table 4.2 shows computational results. They could solve all instances of the Dutch networks within a small gap, i.e., the relative difference between the best lower and the best upper bound they could obtain.

4.3 Multi-Commodity Flow Models

In this section, we will discuss two models that allow passengers routes to be freely chosen, by using a multi-commodity flow formulation.

We use a directed *passenger route graph* (V, A) that arises from $G = (V, E)$ by replacing each edge $e \in E$ with two antiparallel arcs $a(e)$ and $\overline{a}(e)$. Let $e(a) \in E$ be the undirected edge corresponding to $a \in A$. For simplicity of notation, we denote this digraph also by $G = (V, A)$. We are given *traveling times* $\tau_a \in \mathbb{Q}_+$ for every arc $a \in A$. For an OD-pair $(s,t) \in D$, an (s,t)-*passenger path* is a directed path in (V, A) from s to t. Let $\mathcal{P}_{st}$ be the set of all (s,t)-passenger paths, $\mathcal{P} := \bigcup\{p \in \mathcal{P}_{st} : (s,t) \in D\}$ the set of all passenger paths, and $\mathcal{P}_a := \bigcup\{p \in \mathcal{P} : a \in p\}$ the set of all passenger paths that use arc a. The *traveling time* of a passenger path p is defined as $\tau_p := \sum_{a \in p} \tau_a$. See Table 4.3 for a list of all parameters that will be used in this section.

Table 4.3: Notation and terminology for multi-commodity flow models.

G	multi-modal transport network	G_i	subnetwork for mode i
$\mathcal{T}_i$	terminals for mode i	$\boldsymbol{c}^i$	line operating costs for mode i
c_ℓ	operating costs for line ℓ	C_i	line fixed costs for mode i
κ_i	vehicle capacity for mode i	κ_ℓ	vehicle capacity for line ℓ
$\mathcal{L}$	set of all lines	$\mathcal{L}_e$	lines using edge e
D	set of OD-pairs	d_{st}	travel demand between s and t
τ_a	traveling time on arc a	τ_p	traveling time on path p
$\mathcal{P}$	set of all passenger paths	$\mathcal{P}_{st}$	paths between s and t
y_p	passenger flow on path p	x_ℓ	whether line ℓ is used
f_ℓ	frequency of line ℓ	Λ_e	frequency bounds for edge e

4.3.1 The Change-and-Go Approach

The "change-and-go" approach for line planning was developed by Schöbel and Scholl [13, 14]. It considers the passengers interests by taking transfers into account. We will directly present a path formulation of their model.

Similar to the cost minimizing approach above, the model uses a line pool $\mathcal{L}$ and a set of feasible frequencies $\mathcal{F}$. We introduce binary variables x_r, $r = (r_\ell, r_f) \in \mathcal{R} := \mathcal{L} \times \mathcal{F}$, that decide whether line r_ℓ with frequency r_f is used. Additionally, we have continuous variables y_p that give the amount of passengers that travel on path $p \in \mathcal{P}$. The model is the following:

$$
\begin{aligned}
\min \quad & \sum_{p \in \mathcal{P}} \tau_p \, y_p && \\
s.t. \quad & \sum_{p \in \mathcal{P}_{st}} y_p = d_{st} && \forall\, (s,t) \in D \qquad (4.5)\\
& \sum_{p \in \mathcal{P}_a} y_p \le \sum_{r \in \mathcal{R}:\, e(a) \in r_\ell} \kappa_{r_\ell} \, r_f \, x_r && \forall\, a \in A \qquad (4.6)\\
& \sum_{r \in \mathcal{R}:\, r_\ell = \ell} x_r \le 1 && \forall\, \ell \in \mathcal{L} \qquad (4.7)\\
& \sum_{r \in \mathcal{R}} C_{r_\ell} \, x_r \le B && \qquad (4.8)\\
& x_r \in \{0,1\} && \forall\, r \in \mathcal{R}\\
& y_p \ge 0 && \forall\, p \in \mathcal{P}.
\end{aligned}
$$

Constraints (4.5) ensure that all passengers d_{st} are transported from s to t. The *capacity constraints* (4.6) provide the connection between passengers using arc a and the capacities of the lines that use the corresponding undirected edge $e(a)$. Hence, lines are taken as undirected, i.e., by running back and forth they can transport passengers in both directions. (It is easy to incorporate undirected lines into the model.) Constraints (4.7) make sure

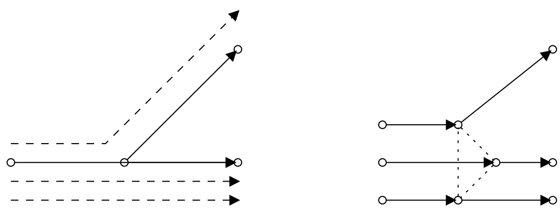

Figure 4.3: Construction of the expanded graph for the change-and-go approach. *Left:* A small piece of the transportation graph with three lines. *Right:* Expanded graph with transfer arcs (dotted, can be used in both directions).

that at most one frequency per line is picked and Constraint (4.8) provides a budget bound for the costs of the lines. Here, C_ℓ is a fixed cost for using line ℓ. Note that this model allows several modes of transportation.

Schöbel and Scholl propose to modify the original graph G to a *change-and-go* network in order to handle different aspects that are important in this setting. The resulting graph has a separate edge for each line passing over an edge in the original network. Furthermore, *transfer arcs* are inserted that allow the transfer of passengers at transfer nodes.; see Figure 4.3. Depending on the weights τ_a that are set on these arcs, one can either count the number of transfers ($\tau_a = 1$ on transfer arcs and $\tau_a = 0$ otherwise) or penalize transfers by setting a transfer time.

Theorem 4.4 (Schöbel and Scholl [13])**.** *The change-and-go line planning problem is $\mathcal{NP}$-hard.*

Proof. Consider an instance of the $\mathcal{NP}$-complete set covering problem, i.e., a finite set $X = \{1, \dots, m\}$, subsets $A_1, \dots, A_n$ of X, and a positive integer K. The question is whether there are at most K sets among $A_1, \dots, A_m$, such that each element of X is contained in at least one of these sets. We assume, without loss of generality, that X is ordered and the sets A_i are pairwise distinct.

From an instance for the set covering problem, an instance for the line planning problem is constructed as follows. The network contains $2m$ nodes $\{s_1, t_1, \dots, s_m, t_m\}$ with a complete set of arcs and edges E. The set of origin-destination pairs is

$$D := \{(s_i, t_i) \,:\, i \in \{1, \dots, m\}\},$$

and $d_{st} = 1$ for all $(s,t) \in D$. For each set $A_j = \{a_1, a_2, \dots, a_k\}$ (with $a_1 < a_2 < \dots < a_k$), we construct a line ℓ_j that visits the nodes

$$s_{a_1}, t_{a_1}, s_{a_2}, t_{a_2}, \dots, s_{a_k}, t_{a_k},$$

in this order. We furthermore set $\mathcal{F} = \{1\}$, $C_\ell = 1$, $\kappa_\ell = 1$, and $B = K$.

We claim that there exists a solution to the set covering problem if and only if there exists a feasible solution to the change-and-go line planning problem.

First assume that $A_{i_1}, \ldots, A_{i_k}$ ($i_1 < i_2 < \cdots < i_k$, $k \leq K$) form a solution to the set covering problem. Then we take the lines $\ell_{i_1}, \ldots, \ell_{i_k}$ and the budget constraint is fulfilled. Each edge $\{s_i, t_i\}$ is covered by a line, since we have started with a solution to the set covering problem. Hence, each passenger can travel on a line. Finally, the capacities $\kappa_\ell = 1$ suffice to carry all passengers.

Conversely, let $\ell_{i_1}, \ldots, \ell_{i_k}$ be the chosen lines in the line plan. By the budget constraint we have $k \leq K$. We claim that $A_{i_1}, \ldots, A_{i_k}$ is a solution to the set covering problem. Indeed, every passenger from s_i to t_i is transported and hence each element $i \in X$ is covered. This proves the theorem. $\square$

Note that in the proof the construction of the change-and-go graph is not needed.

4.3.2 Column Generation for the Change-and-Go Approach

The above model contains exponentially many variables y_p, so we have to use a column generation procedure to solve the LP relaxation. The *LP relaxation* is obtained by removing the integrality constraints and replace them by the corresponding bounds. If one can solve the LP relaxation one can embed this into a branch-and-bound procedure to solve the original mixed integer programming model. We will discuss here only the solution of the LP relaxation.

In order to identify the pricing problems we will first derive the dual linear program for the LP relaxation of the above problem. For this we will write the LP relaxation original problem as follows:

$$
\begin{aligned}
\min \sum_{p \in \mathcal{P}} & \tau_p \, y_p && \\
& \sum_{p \in \mathcal{P}_{st}} y_p = d_{st} && \forall\, (s,t) \in D && (4.9)\\
& \sum_{r \in \mathcal{R}: e(a) \in r_\ell} \kappa_{r_\ell} \, r_f \, x_r - \sum_{p \in \mathcal{P}_a} y_p \geq 0 && \forall\, a \in A && (4.10)\\
& - \sum_{r \in \mathcal{R}: r_\ell = \ell} x_r \geq -1 && \forall\, \ell \in \mathcal{L} && (4.11)\\
& - \sum_{r \in \mathcal{R}} C_{r_\ell} \, x_r \geq -B && && (4.12)\\
& x_r \geq 0 && \forall\, r \in \mathcal{R} &&\\
& y_p \geq 0 && \forall\, p \in \mathcal{P}. &&
\end{aligned}
$$

Note that we can leave out the bounds $x_r \leq 1$, because they are dominated by inequalities (4.11) and the nonnegativity constraints $x_r \geq 0$.

We can now derive the dual program. To this end, let $\pi_{st} \in \mathbb{R}$ be the dual variables for Constraints (4.9), $\mu_a \geq 0$ the dual variables for Constraints (4.10), $\eta_\ell \geq 0$ the dual variables for Constraints (4.11), and let $\delta \geq 0$ be the dual variable for the Constraint (4.12). Then the dual of the above LP relaxation is:

$$
\begin{aligned}
\max \sum_{(s,t)\in D} d_{st}\,\pi_{st} - \sum_{\ell\in\mathcal{L}} \eta_\ell - B\delta & \\
\pi_{st} - \sum_{a\in p} \mu_a \leq \tau_p & \qquad \forall\, p \in \mathcal{P} \\
\sum_{e\in\ell_r} \kappa_{r_\ell}\, r_f \left(\mu_{a(e)} + \mu_{\overline{a}(e)}\right) - \eta_{r_\ell} - C_{r_\ell}\,\delta \leq 0 & \qquad \forall\, r \in \mathcal{R} \\
\mu_a \geq 0 & \qquad \forall\, a \in A \\
\eta_\ell \geq 0 & \qquad \forall\, \ell \in \mathcal{L} \\
\delta \geq 0. &
\end{aligned}
$$

Note that the variables x_r are static in the model and need not be generated. Hence, we only have to discuss the pricing problem for the variables y_p. Here, we have to decide whether there exists a path $p \in \mathcal{P}$ such that

$$\pi_{st} - \sum_{a\in p} \mu_a > \tau_p,$$

where (π_{st}, μ_a) are solutions of the dual LP above. Rewriting this condition we get:

$$\pi_{st} > \sum_{a\in p} \left(\mu_a + \tau_a\right).$$

Hence, this pricing problem can be solved using a shortest path algorithm. If there exists an $(s-t)$-path of length smaller than π_{st} with respect to the weights $\mu_a + \tau_a \geq 0$, then this path satisfies the above inequality and hence the corresponding variable has to be added to the master LP. Since the weights are nonnegative, shortest paths can efficiently be found by Dijkstra's algorithm, for instance (see, e.g., Korte and Vygen [10] for a description of Dijkstra's algorithm).

By the above discussion we have:

Corollary 4.5. *The LP relaxation of the change-and-go line planning approach can be solved in polynomial time.*

4.4 Variable Lines Approach

In this section we will go one step further and do not use a line pool anymore. That is, we will generate lines on the fly via a column generation approach. The passengers can be freely routed as in the change-and-go approach. Detailed information can be found in Borndörfer, Grötschel, and Pfetsch [1].

For the following, we need *operating costs* $\boldsymbol{c}^1 \in \mathbb{Q}_+^{E_1}, \ldots, \boldsymbol{c}^M \in \mathbb{Q}_+^{E_M}$ on the edges, *fixed costs* $C_1, \ldots, C_M \in \mathbb{Q}_+$ for the set-up of a line for each mode, *vehicle capacities* $\kappa_1, \ldots, \kappa_M \in \mathbb{Q}_+$ for each mode, and *edge capacities* $\boldsymbol{\Lambda} \in \mathbb{Q}_+^E$. Let $c_\ell := \sum_{e \in \ell} c_e^i$ be the operating cost of line ℓ of mode i, $C_\ell := C_i$ be its fixed cost, and $\kappa_\ell := \kappa_i$ be its vehicle capacity. See Table 4.3 for a list of all parameters.

With this notation, the line planning problem can be modeled using three kinds of variables:

$y_p \in \mathbb{R}_+$	the flow of passengers traveling from s to t on path $p \in \mathcal{P}_{st}$,
$f_\ell \in \mathbb{R}_+$	the frequency of line $\ell \in \mathcal{L}$,
$x_\ell \in \{0,1\}$	a decision variable for using line $\ell \in \mathcal{L}$.

$$\text{(LPP)} \qquad \min\ \lambda\Big(\sum_{\ell \in \mathcal{L}} C_\ell\, x_\ell + c_\ell\, f_\ell\Big) + (1-\lambda) \sum_{p \in \mathcal{P}} \tau_p\, y_p$$

$$\begin{aligned}
\sum_{p \in \mathcal{P}_{st}} y_p &= d_{st} && \forall\, (s,t) \in D && (i)\\
\sum_{p \in \mathcal{P}_a} y_p - \sum_{\ell : e(a) \in \ell} \kappa_\ell f_\ell &\le 0 && \forall\, a \in A && (ii)\\
\sum_{\ell \in \mathcal{L}_e} f_\ell &\le \Lambda_e && \forall\, e \in E && (iii)\\
f_\ell &\le F x_\ell && \forall\, \ell \in \mathcal{L} && (iv)\\
x_\ell &\in \{0,1\} && \forall\, \ell \in \mathcal{L} && (v)\\
f_\ell &\ge 0 && \forall\, \ell \in \mathcal{L} && (vi)\\
y_p &\ge 0 && \forall\, p \in \mathcal{P}. && (vii)
\end{aligned}$$

Here, $\lambda \in [0,1]$ is a weighing parameter. See below for more information.

The *passenger flow constraints* (i) and the nonnegativity constraints (vii) model a multi-commodity flow problem for the passenger flow, where the commodities correspond to the OD-pairs $(s,t) \in D$. This part guarantees that the demand is routed. The *capacity constraints* (ii) link the passenger paths with the line paths to ensure sufficient transportation capacity on each arc. The *frequency constraints* (iii) bound the total frequency of lines using an edge. Inequalities (iv) link the frequencies with the decision variables for the use of lines; they guarantee that the frequency of a line is 0 whenever it is not used. Here, F is an upper bound on the frequency of a line; for technical reasons, we assume that $F \ge \Lambda_e$ for all $e \in E$, see Section 4.4.2 for more information.

4.4.1 Discussion of the Model

Let us discuss some properties of the model before we investigate its algorithmic tractability.

Objectives: The objective of the model has two competing parts, namely, to minimize total passenger traveling time

$$\sum_{p \in \mathcal{P}} \tau_p \, y_p = \boldsymbol{\tau}^{\mathrm{T}} \boldsymbol{y}$$

and to minimize costs

$$\sum_{\ell \in \mathcal{L}} C_\ell \, x_\ell + c_\ell \, f_\ell = \boldsymbol{C}^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{c}^{\mathrm{T}} \boldsymbol{f}.$$

Here, $\boldsymbol{C}^{\mathrm{T}} \boldsymbol{x}$ is the fixed cost for setting up lines and $\boldsymbol{c}^{\mathrm{T}} \boldsymbol{f}$ is the variable cost for operating these lines at frequencies $\boldsymbol{f}$. The model allows to adjust the relative importance of one part over the other by an appropriate choice of λ. For instance, if $\lambda = 0$, only the second part of the objective (measuring the total traveling time) is taken into account. Conversely, if $\lambda = 1$, only the first part (measuring the sum of fixed and operating costs) are used. Including fixed costs allows to consider objectives such as minimizing the number of lines; note that LPP is a linear program (LP) if all fixed costs are zero.

Passenger Routes: Since the traveling times $\boldsymbol{\tau}$ are nonnegative, we can assume passenger routes to be (simple) paths.

Our model does not fix passenger paths according to a system split, but allows to freely route passengers according to the computed lines. This is targeted at local public transport systems, where, in our opinion, people determine their traveling paths according to the line system and not only according to the network topology.

Our model computes a set of passenger paths that minimize the total traveling times $\boldsymbol{\tau}^{\mathrm{T}} \boldsymbol{y}$ in the sense of a system optimum.

The routing in our model allows for passengers paths of arbitrary travel times, which may force some passengers to long detours. We remark that this problem could be handled by introducing appropriate bounds on the travel times of paths. This would, however, turn the pricing problem for the passenger paths into an $\mathcal{NP}$-hard resource constrained shortest path problem; see the chapter on duty and vehicle scheduling. Note also that such an approach would measure travel times with respect to shortest paths in the underlying network (independent of any line system). Ideally, however, one would like to compare to the shortest paths using only arcs covered by the computed line system.

Line Routes: The literature generally takes line routes as (simple) bidirected paths, and we do the same. In fact, a restriction forcing some sort of simplicity is necessary in order to prevent repetitions around cycles.

It is easy to incorporate additional constraints on the formation of individual lines and constraints on sets of lines, e.g., that the length of a line should not deviate too much from a shortest path between its endpoints or bounds on the number of lines using an edge. Such constraints are important in practice. In the following we consider bounds on the number of edges in a line. Let us give two arguments why this case is practically relevant.

The first argument is based on an idea of a transportation network as a planar graph, probably of high connectivity. Suppose this network occupies a square, in which n nodes are evenly distributed. A typical line starts in the outer regions of the network, passes through the center, and ends in another outer region; we would expect such a line to be of length $\mathcal{O}(\sqrt{n})$.

Real networks, however, are not only (more or less) planar, but often resemble trees. In a *balanced* and preprocessed tree, where each node degree is at least 3, the length of a path between any two nodes is only $\mathcal{O}(\log n)$.

Transfers: Transfers between lines are currently ignored in our model, because constraints (iii) only control the total capacity on edges and not the assignment of passengers to lines. The problem are not transfers between different modes, which can be handled by linking the mode networks G_i with appropriate transfer edges, weighted by estimated transfer times. A similar trick could in principle be used for transfers between lines of the same mode, using an appropriate expansion of the graph. However, this greatly increases the complexity of the model, and it introduces degeneracy; it is unclear whether such a model remains tractable for practical data.

Frequencies: Frequencies indicate the (approximate) number of times vehicles need to be employed in order to serve the demand over the time horizon. In a real world line plan, frequencies often have to produce a regular timetable and hence are not allowed to take arbitrary fractional values. Our model, however, treats frequencies as continuous values. This is a simplification. We have introduced fixed costs in order to reduce the number of lines and decrease the likelihood of low frequencies. In addition, we could have forced our model to accept only a finite number of frequencies by enumerating lines with fixed frequencies in a similar way to the linear cost minimizing model in Section 4.2.3. The resulting model, however, would be much harder to solve. On the other hand, since the frequencies are mainly used to adjust line capacities, we do (at present) not care so much about "nice" frequencies and view the fractional values as approximations or clues to "sensible" values.

4.4.2 Column Generation for the Variables Lines Approach

The LP relaxation of (LPP) can be simplified by eliminating the $\boldsymbol{x}$-variables. In fact, since (LPP) minimizes over nonnegative costs, one can assume w.l.o.g. that inequalities (iv) are satisfied with equality, i.e., there is an optimal LP solution such that $Fx_\ell = f_\ell \Leftrightarrow x_\ell = f_\ell/F$ for all lines ℓ. Substituting for $\boldsymbol{x}$, we observe that the inequalities $f_\ell \leq F$ remaining after the elimination are dominated by inequalities (iii) and hence can be omitted (recall that we assumed $F \geq \Lambda_e$). Setting $\gamma_\ell = C_\ell/F + c_\ell$, we arrive at the following equivalent, but simpler, linear program:

$$
\text{(LP)} \qquad \min\ \lambda \sum_{\ell\in\mathcal{L}} \gamma_\ell\, f_\ell + (1-\lambda)\sum_{p\in\mathcal{P}} \tau_p\, y_p
$$

$$
\begin{aligned}
\sum_{p\in\mathcal{P}_{st}} y_p &= d_{st} && \forall\,(s,t)\in D && (i)\\
\sum_{p\in\mathcal{P}_a} y_p - \sum_{\ell:e(a)\in\ell} \kappa_\ell f_\ell &\leq 0 && \forall\, a\in A && (ii)\\
\sum_{\ell\in\mathcal{L}_e} f_\ell &\leq \Lambda_e && \forall\, e\in E && (iii)\\
f_\ell &\geq 0 && \forall\,\ell\in\mathcal{L} && (iv)\\
y_p &\geq 0 && \forall\, p\in\mathcal{P}. && (v)
\end{aligned}
$$

Note that (LP) contains only a polynomial number of inequalities (apart from the nonnegativity constraints (iv) and (v)).

We want to solve (LP) with a column generation approach and therefore investigate the corresponding pricing problems. We derive the following dual program, where the variables of the dual are as follows: $\boldsymbol{\pi} = (\pi_{st}) \in \mathbb{R}^D$ (flow constraints (i)), $\boldsymbol{\mu} = (\mu_a) \in \mathbb{R}^A$ (capacity constraints (ii)), and $\boldsymbol{\eta} \in \mathbb{R}^E$ (frequency constraints (iii)). The dual of (LP) is:

$$
\max \sum_{(s,t)\in D} d_{st}\,\pi_{st} - \sum_{e\in E} \Lambda_e\,\eta_e
$$

$$
\begin{aligned}
\pi_{st} - \sum_{a\in p} \mu_a &\leq (1-\lambda)\tau_p && \forall\, p\in\mathcal{P}_{st},\ (s,t)\in D\\
\kappa_\ell \sum_{e\in\ell} \big(\mu_{a(e)} + \mu_{\overline{a}(e)}\big) - \sum_{e\in\ell} \eta_e &\leq \lambda\,\gamma_\ell && \forall\,\ell\in\mathcal{L}\\
\mu_a,\ \eta_e &\geq 0 && \forall\, a\in A,\ e\in E.
\end{aligned}
$$

It will turn out that the pricing problem for the line variables f_ℓ is a longest path problem; the pricing problem for the passenger variables y_p, however, is a shortest path problem.

Pricing of the Passenger Variables

The pricing problem for the passenger path variables y_p is exactly the same as in the change-and-go approach and hence can be solved in polynomial time, see Section 4.3.2.

Pricing of the Line Variables

The pricing problem for line variables f_ℓ is more complicated. The reduced cost $\overline{\gamma}_\ell$ for a variable f_ℓ is

$$\overline{\gamma}_\ell = \lambda\,\gamma_\ell - \sum_{e\in\ell} \big(\kappa_\ell\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e\big).$$

The corresponding pricing problem consists in finding a (simple) path ℓ of mode i such that

$$\begin{array}{rcl} 0 > \overline{\gamma}_\ell & = & \lambda\,\gamma_\ell - \sum_{e\in\ell}\big(\kappa_\ell\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e\big) \\ & = & \lambda\,C_\ell/F + c_\ell - \sum_{e\in\ell}\big(\kappa_\ell\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e\big) \\ & = & \lambda\,C_i/F + \sum_{e\in\ell} c_e^i - \sum_{e\in\ell}\big(\kappa_i\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e\big) \\ & = & \lambda\,C_i/F + \sum_{e\in\ell}\big(c_e^i - \kappa_i\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) + \eta_e\big) \\ \multicolumn{3}{l}{\Leftrightarrow\ \sum_{e\in\ell}(\kappa_i\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e - c_e^i) > \lambda\,C_i/F.} \end{array}$$

This problem turns out to be a maximum weighted path problem, since the weights $\big(\kappa_i\,(\mu_{a(e)} + \mu_{\overline{a}(e)}) - \eta_e - c_e^i\big)$ are not restricted in sign. Hence, the pricing problem for the line variables is $\mathcal{NP}$-hard [7]. This shows that solving the LP relaxation (LP) is $\mathcal{NP}$-hard as well.

If we restrict the lengths of the lines, however, we can show the following.

Theorem 4.6 (Borndörfer, Grötschel, and Pfetsch [1]). *The LP relaxation of (LPP) can be solved in polynomial time, if the lengths of the lines are most k, with $k \in \mathcal{O}(\log n)$.*

Here the *length* of a line are the number of edges contained in it.

4.4.3 Algorithm

We now present a column generation algorithm for the solution of the model (LPP) with length restricted lines. The algorithm solves the LP relaxation in a first phase and constructs a feasible line plan using a greedy type heuristic in a second phase.

To solve the LP relaxation, our algorithm iteratively prices out passenger and line path variables until no improving variables are found. We solve the master LP with the barrier algorithm and, towards the end of the process, with the primal simplex algorithm of CPLEX 9.1. We check for new passenger path variables for all OD-pairs using Dijkstra's algorithm, see Section 4.4.2, until no more improving passenger paths are found. If we don't find an improving passenger path, we price out line variables for all line modes and all feasible terminal pairs. We use enumeration for this step.

In the second phase, our algorithm tries to construct a good integer solution from a line pool consisting of the lines having nonzero frequencies in the optimal LP solution. The heuristic is motivated by the observation that the solution of the LP relaxation of a line planning problem often contains lines with very low frequencies. We try to remove these lines by a simple greedy method based on a strong branching selection criterion. In the beginning the $\boldsymbol{x}$-variables of all lines in the pool are set to 1. In each iteration, we tentatively remove a line (set its x-variable to 0), compute the objective $\lambda\, \boldsymbol{c}^{\mathrm{T}}\boldsymbol{f} + (1-\lambda)\, \boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{y}$ of the LP obtained by fixing the line variables as described, pricing passenger variables as needed, and add the fixed costs $\boldsymbol{C}^{\mathrm{T}}\boldsymbol{x}$ of all lines that are fixed to 1. After probing candidate lines with the smallest $\boldsymbol{f}$-values in this way, we permanently delete the line whose removal resulted in the smallest objective. We repeat this elimination as long as the remaining set of lines is still feasible, i.e., all demands can be routed, and the objective function decreases.

4.4.4 Computational Results

In this section we report on computational experience with line planning problems for the city of Potsdam, Germany. The experiments originate from a joint project with the two local public transport companies ViP Verkehrsgesellschaft GmbH and Havelbus Verkehrsgesellschaft mbH, the city of Potsdam, and the software company IVU Traffic Technologies AG.

Potsdam is a medium sized town near Berlin; it has about 150,000 inhabitants. Its public transportation system uses city buses and trams (operated by ViP) and regional buses (operated by Havelbus). Additionally, there are regional trains connecting Potsdam to its surroundings (operated by Deutsche Bahn AG) and a city railroad (operated by S-Bahn Berlin) which provides connections to Berlin. As regional trains and the city railroad are not operated by ViP and Havelbus, the associated lines routes are assumed to be fixed.

Data

Our data consists of a multi-modal traffic network of Potsdam and an associated OD-matrix, which had been used by IVU in a consulting project for planning the Potsdam network (Nahverkehrsplan). The data represents the line system of Potsdam of 1998. It has 27 bus lines and 4 tram lines. Including line variants, the total number of lines was 80. The network has 951 nodes, including 111 OD-nodes, and 1321 edges. The maximum length of a line is 47 edges.

The network was preprocessed as follows. We removed isolated nodes. Then we iteratively removed “leaves” in the graph, i.e., nodes which have only one neighbor, and iteratively contracted nodes with two neighbors. The preprocessed graph has 410 nodes, 106 of which were OD-nodes, and 891 edges. We remark that although such preprocessing steps are conceptually

Table 4.4: Experimental results of line planning for $\lambda = 0.9978$.

Optimized LP solution – enumeration:		
total traveling time:	108,360,036.33	[scaled: 238,392.08]
total line cost:	233,776.86	[scaled: 233,262.55]
LP objective value:	471,654.63	
active line/pass. var.:	60/4879	transfers: 8777/64607
Optimized integer solution – greedy heuristic:		
total traveling time:	112,581,291.50	[scaled: 247,678.84]
total line cost:	287,060.90	[scaled: 286,429.37]
integer objective value:	818,491.68	
active line/pass. var.:	30/4767	transfers: 8638/60539
Reference LP solution:		
total traveling time:	105,269,846.00	[scaled: 231,593.66]
total line cost:	501,376.24	[scaled: 500,273.21]
LP objective value:	731,866.87	
active line/pass. var.:	61/4857	transfers: 8618/63310
Reference integer solution – greedy heuristic:		
total traveling time:	106,952,869.00	[scaled: 235,296.31]
total line cost:	562,964.54	[scaled: 561,726.02]
integer objective value:	1,213,221.49	
active line/pass. var.:	44/4814	transfers: 9509/70525

easy, the data handling can be quite intricate in practice; for instance, our data included information on possible turnings of a line at road/rail crossings, which must be updated in the course of the preprocessing.

The OD-matrix was also modified. Nodes with zero traffic were removed. The original time horizon was one day, but we wanted to construct a line plan for the peak hour. We therefore scaled the matrix to 40% in an (admittedly rough) attempt to simulate afternoon traffic (3 p.m. to 6 p.m.). Note that the resulting matrix is still quite symmetric (the maximum difference between each of the two directions was 25) whereas a real afternoon OD-matrix would not be symmetric. The scaled OD-matrix had 4685 nonzeros and the total scaled travel demand was 42796.

All traveling times are measured in seconds and we always restricted the maximum length of a line to 55 edges. Since no data was available on line costs, we decided on $C_\ell = 10000$ (fixed costs) for each line ℓ and $c_e^i = 100$ (operating costs) for each edge e and mode i. Hence, we do not distinguish between costs of different modes (an unrealistic assumption in practice).

Experiments

Table 4.4 reports the results of several computational experiments with the data and implementation that we have described. All experiments were performed on a 3.4 GHz Pentium 4 machine running Linux. In the table, the *total traveling time* is $\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{y}$ and *total line cost* is $\boldsymbol{\gamma}^{\mathrm{T}}\boldsymbol{f}$, the *scaled* values are $(1-\lambda)\,\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{y}$ and $\lambda\,\boldsymbol{\gamma}^{\mathrm{T}}\boldsymbol{f}$, respectively; all four values refer to the LP relaxation (LP). The *LP objective value* is $\lambda\,\boldsymbol{\gamma}^{\mathrm{T}}\boldsymbol{f} + (1-\lambda)\,\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{y}$, the *integer objective value* refers to $\lambda\,(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{c}^{\mathrm{T}}\boldsymbol{f}) + (1-\lambda)\,\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{y}$. The last line in each block of results gives the number of active (i.e., nonzero) line and passenger variables, and the number of passenger transfers (first number) that were needed as well as the number of transfering passengers (second number). Note that we can compute transfers from passenger routes as an afterthought, although our optimization model is currently insensitive to them.

Note that our costs are not realistic. Therefore the frequencies that we compute cannot be compared to the ones used in practice. To allow some adaptation to our cost model, we let the frequencies of all lines be variable, in particular, the frequencies of the city railroad and regional train lines.

In our first experiment, we solved the LP relaxation (LP) of the Potsdam problem. We set $\lambda = 0.9978$, which roughly balances the two parts of the objective function. The resulting LP had 5761 rows. Using enumeration, we obtained an optimal solution after 451 seconds and 283 iterations (i.e., solutions of the master LP), of which 15 were used to price lines. The pricing problems needed a total time of 183 seconds of which most was used for the pricing of line paths. Hence, more than half of the time is spent for solving the master LPs.

We also investigated the passenger routing of our LP solution for the enumeration variant in more detail. To connect the 4685 OD-pairs only 4879 paths are needed, i.e., most OD-pairs are connected by a unique path. The total traveling time is 108,360,036.33 seconds, see Table 4.4. For comparison, when we ignore capacities and route all passengers between every OD-pair on the fastest path in the final line system, the total traveling time is 95,391,460 seconds. This is a relative difference of 12%. This seems to be an acceptable deviation.

In our second experiment, we computed two integer solutions for (LPP) associated with the parameter $\lambda = 0.9978$, as above. The first solution is obtained by rounding all nonzero $\boldsymbol{x}$-variables in the solution of the LP relaxation, computed with the enumeration variant, to 1. The (integer) objective of this rounded solution is 1,058,079.69, which leads to a gap of 55% compared to the LP relaxation value of 471,654.63. The second solution is obtained by the greedy algorithm described in Section 4.4.3, starting from the same LP solution (only lines for city buses, trams, and regional buses were removed). It has 30 lines (17 bus lines and 2 tram lines), down from 60 in the first solution, see Table 4.4; it took 1368 seconds to compute. The final (scaled) operating costs are 286,429.37, while the final fixed costs are $\lambda \cdot 300,000 = 299,340$. The integer objective of 818,491.68 has a gap of 42% with respect to the LP relaxation value of 471,654.63. Note that the results heavily depend on the cost structure: decreasing the fixed costs automati-

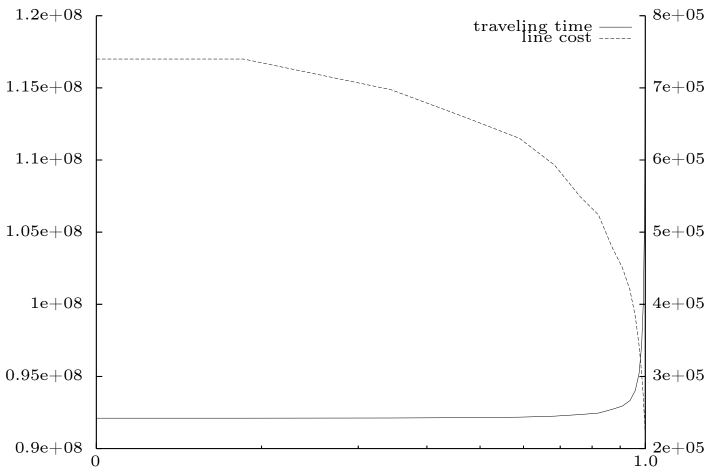

Figure 4.4: Total traveling time (solid, left axis) and total line cost (dashed, right axis) in dependence on λ (x-axis in logscale).

cally reduces the gap. In our context, with high fixed costs, emphasis is put on reducing the number of lines (recall that the costs were artificial). The result obtained seems to be quite good, given that the original line system contained 27 bus lines and 4 tram lines; it seems unlikely that one can reduce the number much further. Furthermore, the lower bound of the LP relaxations is typically very weak for such fixed cost problems. Still, more research is needed to provide better lower bounds and primal solutions.

We compare the LP and integer solutions to "reference solutions" shown in the lower part of Table 4.4. The reference LP solution is obtained by fixing the paths of the original lines of Potsdam and then solving the resulting LP relaxation without generating new lines, but allowing the frequencies of the lines to change. The reference integer solution is obtained by applying the greedy heuristic to the reference LP solution. The results show that allowing the generation of new line paths reduces line costs in both cases to roughly 50% and the total objective to roughly 2/3 of the original values, while the total traveling time increases by a few percent. Hence, in these experiments the greedy algorithm has not changed the relative improvement obtained from optimizing lines.

Our third experiment investigates the influence of the parameter λ on the solution. We computed the solutions to the LP relaxation for 21 different values of λ_i, taking $\lambda_i = 1 - \big(1 - i/20\big)^4$, for $i = 0, \ldots, 20$. This collects increasingly more samples near $\lambda = 1$, a region where the total traveling time and the total line cost are about equal.

The results are plotted in Figure 4.4. This figure shows the total traveling time and the total line cost depending on λ. The extreme cases are as expected: For $\lambda = 0$, the line costs do not contribute to the objective and are therefore high, while the total traveling time is low. For $\lambda = 1$, only the total line cost contributes to the objective and is therefore minimized as much as possible at the cost of increasing the total traveling time. With increasing λ, the total line cost monotonically decreases, while the total traveling time increases. Note that each computed pair of total traveling time and line cost constitutes a Pareto optimal point, i.e., is not dominated by any other attainable combination. Conversely, any Pareto optimal solution of the LP relaxation can be obtained as the solution for some $\lambda \in [0,1]$, see, e.g., Ehrgott [6].

Bibliography

[1] R. Borndörfer, M. Grötschel, and M. E. Pfetsch, *A path based model for line planning in public transport*, Report ZR-05-18, Zuse Institute Berlin, 2005.

[2] M. R. Bussieck, *Optimal lines in public rail transport*, PhD thesis, TU Braunschweig, 1997.

[3] M. R. Bussieck, P. Kreuzer, and U. T. Zimmermann, *Optimal lines for railway systems*, Eur. J. Oper. Res. **96**, no. 1 (1997), pp. 54–63.

[4] M. T. Claessens, N. M. van Dijk, and P. J. Zwaneveld, *Cost optimal allocation of rail passanger lines*, Eur. J. Oper. Res. **110**, no. 3 (1998), pp. 474–489.

[5] J. R. Daduna, I. Branco, and J. M. P. Paixão, eds., *Proc. of the Sixth International Workshop on Computer-Aided Scheduling of Public Transport (CASPT), Lisbon, Portugal, 1993*, Lecture Notes in Economics and Mathematical Systems 430, Springer-Verlag, Berlin, Heidelberg, 1995.

[6] M. Ehrgott, *Multicriteria optimization*, Springer-Verlag, Berlin, 2nd ed., 2005.

[7] M. R. Garey and D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, 1979.

[8] J.-W. H. M. Goossens, S. van Hoesel, and L. G. Kroon, *On solving multi-type line planning problems*, METEOR Research Memorandum RM/02/009, University of Maastricht, 2002.

[9] J.-W. H. M. Goossens, S. van Hoesel, and L. G. Kroon, *A branch-and-cut approach for solving railway line-planning problems*, Transportation Sci. **38**, no. 3 (2004), pp. 379–393.

[10] B. Korte and J. Vygen, *Combinatorial optimization. Theory and algorithms*, Algorithms and Combinatorics 21, Springer, Berlin, 2nd ed., 2002.

[11] W. Lampkin and P. D. Saalmans, *The design of routes, service frequencies and schedules for a municipal bus undertaking: A case study*, Operational Res. Quart. **18** (1967), pp. 375–397.

[12] C. E. Mandl, *Evaluation and optimization of urban public transportation networks*, Eur. J. Oper. Res. **5** (1980), pp. 396–404.

[13] A. Schöbel and S. Scholl, *Line planning with minimal travelling time.* Preprint, 2005.

[14] S. Scholl, *Customer-Oriented Line Planning*, PhD thesis, Universität Göttingen, 2005.