



Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

RALF BORNDÖRFER MARIKA KARBSTEIN

Metric Inequalities for Routings on Direct Connections with Application in Line Planning

Supported by the DFG Research Center MATHEON "Mathematics for key technologies", project B15.

ZIB Report 15-07 (2015)

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

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

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

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

Metric Inequalities for Routings on Direct Connections with Application in Line Planning*

Ralf Borndörfer** Marika Karbstein**

Abstract

We consider multi-commodity flow problems in which capacities are installed on paths. In this setting, it is often important to distinguish between flows on direct connection routes, using single paths, and flows that include path switching. We derive a feasibility condition for path capacities supporting such direct connection flows similar to the feasibility condition for arc capacities in ordinary multi-commodity flows. The concept allows to solve large-scale real-world line planning problems in public transport including a novel passenger routing model that favors direct connections over connections with transfers.

1 Introduction

Network design problems deal with installing capacities on edges to support a multi-commodity flow routing of a given demand. A key component of respective models are the *metric inequalities* by Iri [1] and Kakusho & Onaga [2] that characterize the feasible edge capacities, see Raack [3] and Dell’Amico, Maffioli, and Martello [4] for surveys. We consider network design problems in which capacities are installed on paths. This has applications, e. g., in public transport, where the paths correspond to lines, see Borndörfer and Karbstein [5] and Karbstein [6]. In such a setting it is important whether a commodity is routed on direct connections, i. e., on single paths, or whether the commodity has to switch paths, in which case a *switching penalty* arises. In line planning such a switching corresponds to a transfer and the switching penalty, hence, is a transfer penalty. Because of such penalties, direct connection routes are preferred, unless routes with path-switching are forced by a lack of capacity. The task is therefore to design a system of paths with associated capacities such that a weighted sum of path installation and demand routing costs, including switching penalties, is minimized.

This paper presents a tractable model to handle such problems. It is based on a novel concept of *metric inequalities for direct connections (dcmetric inequalities)*. Similar to the metric inequalities for the classical multi-commodity flow problem, the dcmetric inequalities completely characterize the feasible path capacities that support a given direct connection routing. They can be used to reduce such a direct connection routing to a classical multi-commodity flow routing. In this way, an additional important requirement can be handled using a model of roughly the same computational complexity. Applied to line planning, this approach allows an integrated optimization of line installations and passen-

*Supported by the DFG Research Center MATHEON “Mathematics for key technologies”.

**Zuse-Institute Berlin (ZIB), Takustr. 7, 14195 Berlin, Germany, Email {borndorfer, karbstein}@zib.de

ger routings, including accurate estimates of the number of direct travelers, for large-scale real-world instances.

The paper is structured as follows. In a first theoretical part we discuss a family of network design models that deal with direct connection routings. After an overview of the basic notation in Section 2, we start in Section 3 with an (explicit) *direct path connection model*, that associates direct connection routes with the corresponding paths. Section 4 derives the *metric inequalities for a direct connection routing*, that characterize the feasible path capacities. In Section 5, we use these constraints to construct an efficiently solvable *complete direct connection model*. We present an approximative *basic direct connection model* of polynomial size in Section 6 that involves a combinatorial subset of the metric inequalities for direct connection routings. The pricing of the non-direct connection route variables is discussed in Section 7. Section 8 concludes the theoretical part by giving a short comparison of all considered models. In a second part we apply these models to the line planning problem in public transport. Section 9 discusses the relevant literature and Section 10 shows that direct connection models work very well in practice.

2 Basic Notation

We use the following notation. Let $G = (V, E)$ be an undirected graph and \mathcal{P} a set of (explicitly given) elementary *paths* in G . On each path $p \in \mathcal{P}$ we can install one out of several *capacities* $\kappa_{p,k}$ at a *cost* $c_{p,k}$, $k \in K \subseteq \mathbb{N}$. Let $d \in \mathbb{Q}_{\geq 0}^{V \times V}$ be a *demand* for each pair of *origin-destination nodes* (OD-nodes) $s, t \in V$. We denote by $D = \{(s, t) \in V \times V : d_{st} > 0\}$ the set of all OD-pairs with positive demand. The demand is routed along a directed *routing graph* (V, A) that arises from the graph $G = (V, E)$ by replacing each edge $e \in E$ by two antiparallel arcs $a(e)$ and $\bar{a}(e)$; let conversely $e(a)$ be the undirected edge corresponding to such an arc $a \in A$. Each arc in the routing graph is associated with a *cost* τ_a . We also denote the routing graph by G . We say that a path $p \in \mathcal{P}$ that covers an edge e in the undirected graph covers the two antiparallel arcs $a(e)$ and $\bar{a}(e)$ in the directed routing graph. Let $\mathcal{P}(a) = \{p \in \mathcal{P} : e(a) \in p\}$ be the set of all paths that covers arc a . We denote by \mathcal{R}_{st} the set of all elementary directed *st-routes* from s to t in G (i. e., elementary *st-paths* in G), and by $\mathcal{R} = \bigcup_{(s,t) \in D} \mathcal{R}_{st}$ the set of all *routes*; they will be generated.

A *direct connection st-route* (*st-dcroute*) is an *st-route* $(s = v_0, a_0, v_1, \dots, v_j, a_j, v_{j+1} = t)$, where $a_i = (v_i, v_{i+1})$ and there exists some $p \in \mathcal{P}$ such that $e(a_i) \in p$ for $i = 0, \dots, j$, i. e., the demand can be routed along a single path from origin s directly to destination t without path-switching. Let \mathcal{R}_{st}^0 be the set of all *st-dcroutes*, $\mathcal{R}_{st}^0(a) = \{r \in \mathcal{R}_{st}^0 : a \in r\}$ the set of *st-dcroutes* that pass over arc a , and $\mathcal{R}^0 = \bigcup_{(s,t) \in D} \mathcal{R}_{st}^0$, $\mathcal{R}^0(a) = \bigcup_{(s,t) \in D} \mathcal{R}_{st}^0(a)$ their unions. We denote by $\mathcal{P}_r = \{p \in \mathcal{P} : r \subset p\}$ the set of all paths that support the dcroute r . We set the *cost* of a dcroute $r \in \mathcal{R}^0$ to the sum of the arc costs $\tau_{r,0} = \sum_{a \in r} \tau_a$. For all routes $r \in \mathcal{R}$ that involve a switching between paths we add a *switching penalty* $\sigma \in \mathbb{Q}_+$ and arrive at a *cost* of $\tau_{r,1} = \sigma + \sum_{a \in r} \tau_a$. The *direct connection network design problem* that we consider is to find a set of paths and associated capacities that (i) supports a routing of the demand by a set of routes and dcroutes and (ii) minimizes a weighted sum of path installation plus routing costs, including switching penalties.

3 Direct Path Connection Model

We first introduce a model for the direct connection network design problem that accounts for the demand on direct connections in an explicit way. To this purpose, we introduce flow variables $z_{r,0}^p$ and $y_{r,1}$ for the demand routed on dcroute r on path p and on route r with at least one path-switch, respectively. Introducing further variables $x_{p,k} \in \{0,1\}$ for installing capacity $\kappa_{p,k}$ on path p , we can state a *direct path connection model*

$$\text{(DPC)} \quad \min \lambda \sum_{p \in \mathcal{P}} \sum_{k \in K} c_{p,k} x_{p,k} + (1 - \lambda) \left(\sum_{r \in \mathcal{R}^0} \sum_{p \in \mathcal{P}_r} \tau_{r,0} z_{r,0}^p + \sum_{r \in \mathcal{R}} \tau_{r,1} y_{r,1} \right)$$

$$\sum_{r \in \mathcal{R}_{st}^0} \sum_{p \in \mathcal{P}_r} z_{r,0}^p + \sum_{r \in \mathcal{R}_{st}} y_{r,1} = d_{st} \quad \forall (s, t) \in D \quad (1)$$

$$\sum_{r \in \mathcal{R}^0(a)} \sum_{p \in \mathcal{P}_r} z_{r,0}^p + \sum_{r \in \mathcal{R}(a)} y_{r,1} \leq \sum_{p \in \mathcal{P}(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall a \in A \quad (2)$$

$$\sum_{r \in \mathcal{R}^0(a): p \in \mathcal{P}_r} z_{r,0}^p \leq \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall a \in A, \forall p \in \mathcal{P}(a) \quad (3)$$

$$\sum_{k \in K} x_{p,k} \leq 1 \quad \forall p \in \mathcal{P} \quad (4)$$

$$x_{p,k} \in \{0, 1\} \quad \forall p \in \mathcal{P}, \forall k \in K \quad (5)$$

$$z_{r,0}^p \geq 0 \quad \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r \quad (6)$$

$$y_{r,1} \geq 0 \quad \forall r \in \mathcal{R}. \quad (7)$$

Model (DPC) minimizes a weighted sum of path and routing costs; the weighing parameter is $\lambda \in [0, 1]$. Equations (1) enforce the demand flow. Inequalities (2) guarantee sufficient total path capacity on each arc. Constraints (3), the *direct path connection constraints*, ensure sufficient capacity for direct connection routes on each arc of each path. Inequalities (4) ensure that at most one capacity is installed on each path.

Model (DPC) includes a variable $z_{r,0}^p$ for the assignment of each direct connection route r to a direct connection path p . This makes the model fairly large. Indeed, a path of length l is usually a direct connection path for $\mathcal{O}(l^2)$ OD-pairs, such that the number of variables is much larger than the number of paths. Moreover, choices between several possible direct connection paths for every dcroute produce lots of degeneracy. We will show next how to overcome these problems and reduce the number of variables by relaxing the explicit assignment of dcroutes to direct connection paths. We will end up with a direct connection route variable $y_{r,0}$ for each route $r \in \mathcal{R}^0$.

4 Metric Inequalities for Direct Connections

We eliminate the assignment of direct routes to particular paths in model (DPC) by aggregating the dcroute variables as

$$y_{r,0} = \sum_{p \in \mathcal{P}_r} z_{r,0}^p, \quad (8)$$

i. e., we introduce *path-independent* dcroute variables $y_{r,0}$ for the demand routed directly on r . Such a substitution can be easily done in the objective of model (DPC) and in

the constraints (1) and (2). Skipping the direct path connection constraints (3) for the moment, we arrive at what we will call “*skeleton*” *direct connection model*

$$\begin{aligned} \text{(DC-skeleton)} \quad \min \lambda \sum_{p \in \mathcal{P}} \sum_{k \in K} c_{p,k} x_{p,k} + (1 - \lambda) \left(\sum_{r \in \mathcal{R}^0} \tau_{r,0} y_{r,0} + \sum_{r \in \mathcal{R}} \tau_{r,1} y_{r,1} \right) \\ \sum_{r \in \mathcal{R}_{st}^0} y_{r,0} + \sum_{r \in \mathcal{R}_{st}} y_{r,1} = d_{st} \quad \forall (s, t) \in D \end{aligned} \quad (9)$$

$$\sum_{r \in \mathcal{R}^0(a)} y_{r,0} + \sum_{r \in \mathcal{R}(a)} y_{r,1} \leq \sum_{p \in \mathcal{P}(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall a \in A \quad (10)$$

$$\sum_{k \in K} x_{p,k} \leq 1 \quad \forall p \in \mathcal{P} \quad (11)$$

$$x_{p,k} \in \{0, 1\} \quad \forall p \in \mathcal{P}, \forall k \in K \quad (12)$$

$$y_{r,0} \geq 0 \quad \forall r \in \mathcal{R}^0 \quad (13)$$

$$y_{r,1} \geq 0 \quad \forall r \in \mathcal{R}. \quad (14)$$

To replace the direct path connection constraints (3) in a way that is compatible with the aggregated dcroute variables, assume we are given a solution (x^*, y^*) of model (DC-skeleton). Such a direct connection routing is possible if and only if the following set of inequalities (C) is satisfied

$$\text{(C)} \quad \sum_{r \in \mathcal{R}^0(a): p \in \mathcal{P}_r} z_{r,0}^p \leq c^p (:= \sum_{k \in K} \kappa_{p,k} x_{p,k}^*) \quad \forall p \in \mathcal{P}, \forall a \in p \quad (\mu_a^p)$$

$$\sum_{p \in \mathcal{P}_r} z_{r,0}^p = y_{r,0}^* \quad \forall r \in \mathcal{R}^0 \quad (\omega_r)$$

$$z_{r,0}^p \geq 0 \quad \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r,$$

i. e., we can assign the path-independent dcroute variables $y_{r,0}$ to direct connection paths supporting the dcroutes r .

Using the Farkas Lemma either inequality set (C) has a solution or inequality set (\bar{C})

$$\begin{aligned} (\bar{C}) \quad \sum_{p \in \mathcal{P}} c^p \sum_{a \in p} \mu_a^p + \sum_{r \in \mathcal{R}^0} \omega_r y_{r,0}^* < 0 \\ \sum_{a \in r} \mu_a^p + \omega_r \geq 0 \quad \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r \\ \mu_a^p \geq 0 \quad \forall p \in \mathcal{P}, \forall a \in p. \end{aligned}$$

Consider some ω_r with $r \in \mathcal{R}^0$. Then

$$\begin{aligned} -\omega_r &\leq \sum_{a \in r} \mu_a^p \quad \forall p \in \mathcal{P}_r \\ \Leftrightarrow -\omega_r &= \min_{p \in \mathcal{P}_r} \left\{ \sum_{a \in r} \mu_a^p \right\} \quad (=:\text{dist}_{\mu}^{\mathcal{P}}(r)). \end{aligned}$$

We then get that (\bar{C}) has a solution if and only if there exists $\mu \in \mathcal{Q}_+$, $\mathcal{Q} := \{(a, p) : p \in \mathcal{P}, a \in p\}$, with

$$\sum_{p \in \mathcal{P}} c^p \sum_{a \in p} \mu_a^p < - \sum_{r \in \mathcal{R}^0} \omega_r y_{r,0}^* = \sum_{r \in \mathcal{R}^0} \text{dist}_{\mu}^{\mathcal{P}}(r) y_{r,0}^*.$$

This gives a necessary and sufficient feasibility condition for aggregated multi-commodity direct connection flows:

Theorem 4.1. *A capacity vector $c \in \mathbb{R}_+^{\mathcal{P}}$ supports a direct connection routing $y_{r,0}^*$ if and only if*

$$\sum_{p \in \mathcal{P}} c^p \sum_{a \in p} \mu_a^p \geq \sum_{r \in \mathcal{R}^0} \text{dist}_\mu^{\mathcal{P}}(r) y_{r,0}^* \quad \forall \mu \in \mathbb{Q}_+^{\mathcal{Q}}. \quad (15)$$

This result is similar to the Theorem of Iri [1], Kakusho and Onaga [2] for the feasibility of edge capacities to support a multi-commodity flow for a given demand to path capacities and direct connection routings. On the one hand, our setting is more general because it considers path capacities instead of arc capacities. On the other hand, direct connection routing is restrictive because it must follow paths. Hence, our result and the Theorem of Iri [1], Kakusho and Onaga [2] are related, but do not imply each other.

Note that we can restrict the weights μ_a^p in a dcmetric inequality to the set

$$\mathcal{M} := \{\mu \in \mathbb{Q}_+^{\mathcal{Q}} : \forall (a, p) \in \mathcal{Q} : \exists r \in \mathcal{R}(a) : \mu^p(r) := \sum_{b \in r} \mu_b^p = \text{dist}_\mu^{\mathcal{P}}(r)\}. \quad (16)$$

This means that arc a on path p is on a shortest direct connection for some dcroute r and hence cannot always be bypassed. In this sense, the weights $\mu \in \mathcal{M}$ define a kind of metric, a *dcmetric*, and the inequalities (15) can be interpreted as *dcmetric inequalities*. Note that \mathcal{M} is a cone such that we can, w.l.o.g., further restrict $0 \leq \mu_a^p \leq 1$ for all $p \in \mathcal{P}$, $a \in A$.

5 Complete Direct Connection Model

Inserting the dcmetric inequalities into model (DC-skeleton) produces the following *complete direct connection model*

$$\begin{aligned} \text{(DC-complete)} \quad \min \quad & \lambda \sum_{p \in \mathcal{P}} \sum_{k \in K} c_{p,k} x_{p,k} + (1 - \lambda) \left(\sum_{r \in \mathcal{R}^0} \tau_{r,0} y_{r,0} + \sum_{r \in \mathcal{R}} \tau_{r,1} y_{r,1} \right) \\ & \sum_{r \in \mathcal{R}_{st}^0} y_{r,0} + \sum_{r \in \mathcal{R}_{st}} y_{r,1} = d_{st} \quad \forall (s, t) \in D \end{aligned} \quad (17)$$

$$\sum_{r \in \mathcal{R}^0(a)} y_{r,0} + \sum_{r \in \mathcal{R}(a)} y_{r,1} \leq \sum_{p \in \mathcal{P}(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall a \in A \quad (18)$$

$$\sum_{p \in \mathcal{P}} \sum_{a \in p} \mu_a^p \sum_{k \in K} \kappa_{p,k} x_{p,k} \geq \sum_{r \in \mathcal{R}^0} \text{dist}_\mu^{\mathcal{P}}(r) y_{r,0} \quad \forall \mu \in [0, 1]^{\mathcal{Q}} \quad (19)$$

$$\sum_{k \in K} x_{p,k} \leq 1 \quad \forall p \in \mathcal{P} \quad (20)$$

$$x_{p,k} \in \{0, 1\} \quad \forall p \in \mathcal{P}, \forall k \in K \quad (21)$$

$$y_{r,0} \geq 0 \quad \forall r \in \mathcal{R}^0 \quad (22)$$

$$y_{r,1} \geq 0 \quad \forall r \in \mathcal{R}. \quad (23)$$

Proposition 5.1. *Models (DPC) and (DC-complete) are equivalent. More precisely, each solution of model (DPC) can be transformed into a solution of model (DC-complete) with identical objective value and vice versa.*

We argue now that model (DC-complete) is algorithmically tractable, even though it contains a large number of dcmetric inequalities.

Consider the skeleton model (DC-skeleton) plus a (possibly empty) polynomially sized subset of the dcmetric inequalities (19). We denote this starting model by (DC-complete^{*}). Let (x^*, y^*) be an (optimal) LP solution for (DC-complete^{*}). If the objective value of the linear program (S)

$$\begin{aligned}
(S) \quad & \min \sum_{p \in \mathcal{P}} \sum_{a \in p} \mu_a^p \sum_{k \in K} \kappa_{p,k} x_{p,k}^* - \sum_{r \in \mathcal{R}^0} \omega_r y_{r,0}^* \\
\text{s.t.} \quad & \sum_{a \in r} \mu_a^p - \omega_r \geq 0 && \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r \\
& \mu_a^p \geq 0 && \forall p \in \mathcal{P}, \forall a \in p \\
& 1 \geq \omega_r \geq 0 && \forall r \in \mathcal{R}^0
\end{aligned}$$

is negative, then the objective function of (S) gives rise to a violated dcmetric inequality. The LP (S) arises from the system (\bar{C}) by inverting the sign of ω and by bounding ω by 1 to make the optimal objective value finite.

The feasible region of the separation program is a polyhedron that does not depend on the current solution (x^*, y^*) of (DC-complete^{*}). It therefore suffices to consider a finite number of dcmetric inequalities that correspond to the vertices and extremal rays of the feasible region. These arguments prove the following propositions.

Proposition 5.2. *The dcmetric inequalities can be separated in polynomial time.*

Proposition 5.3. *The LP-relaxation of the complete direct connection model (DC-complete) can be solved in polynomial time.*

We remark that one can also compute valid primal LP and IP solutions for model (DC-complete) from a given solution (x^*, y^*) of (DC-complete^{*}) *throughout the cutting plane algorithm* by some path-switching on connections that are currently mistaken as being direct, i. e., one can always easily produce a feasible solution; this is very convenient in practice. More precisely, the linear program

$$\begin{aligned}
(H) \quad & \max \sum_{r \in \mathcal{R}^0} \sum_{p \in \mathcal{P}_r} z_{r,0}^p \\
\text{s.t.} \quad & \sum_{r \in \mathcal{R}^0(a): p \in \mathcal{P}_r} z_{r,0}^p \leq \sum_{k \in K} \kappa_{p,k} x_{p,k}^* && \forall p \in \mathcal{P}, \forall a \in p \\
& \sum_{p \in \mathcal{P}_r} z_{r,0}^p \leq y_{r,0}^* && \forall r \in \mathcal{R}^0 \\
& z_{r,0}^p \geq 0 && \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r.
\end{aligned}$$

gives rise to the maximal demand that can be routed directly according to the path capacities defined by x^* and the estimated amount of direct demand $y_{r,0}^*$, $r \in \mathcal{R}^0$. In fact, $y_{r,0} := \sum_{p \in \mathcal{P}_r} z_{r,0}^p$,

$$y_{r,1} := \begin{cases} y_{r,1}^* + (y_{r,0}^* - \sum_{p \in \mathcal{P}_r} z_{r,0}^p) & \text{for } r \in \mathcal{R}^0 \\ y_{r,1}^* & \text{for } r \notin \mathcal{R}^0 \end{cases}$$

and $x := x^*$ is a valid LP/IP solution for (DC-complete). Of course, this routing is best possible for the capacities defined by x^* if $y_{r,0}^*$, $r \in \mathcal{R}^0$, is estimated correctly, i. e., the better (DC-complete^{*}) approximates (DC-complete) the better are the associated primal solutions.

The duality of the programs (S) and (H) can be seen as follows. We add the constant term $-\sum_{r \in \mathcal{R}^0} y_{r,0}^*$ to the objective of (H) and associate dual variables μ_a^p and ν_r with the set of inequalities. The dual of (H) is then

$$\begin{aligned} \min \quad & \sum_{p \in \mathcal{P}} \sum_{a \in p} \sum_{k \in K} \mu_a^p \kappa_{p,k} x_{p,k}^* + \sum_{r \in \mathcal{R}^0} y_{r,0}^* (\nu_r - 1) \\ \text{s.t.} \quad & \sum_{a \in r} \mu_a^p + \nu_r \geq 1 && \forall r \in \mathcal{R}^0, \forall p \in \mathcal{P}_r \\ & \mu_a^p \geq 0 && \forall p \in \mathcal{P}, \forall a \in p \\ & \nu_r \geq 0 && \forall r \in \mathcal{R}^0. \end{aligned}$$

It is easy to see that the above linear program has always an optimal solution with $\nu_r \in [0, 1]$, $r \in \mathcal{R}^0$. If we substitute $\omega_r := (1 - \nu_r)$, we have $1 \geq \omega_r \geq 0$, $r \in \mathcal{R}^0$, and get the linear program (S).

This relation also gives an interpretation of the objective value of (S). Namely, the optimal objective value of (S) amounts to the demand that cannot be routed directly with the current path capacities.

6 Basic Direct Connection Model

We finally show that a combinatorially motivated compact approximation of the complete direct connection model (DC-complete) provides a provable quality. More precisely, we will show that the ratio between the directly routed demand estimated by this approximation and the directly routed demand computed by model (DC-complete) can be bounded by half of the maximal length of a path.

We define a polynomial subset of dcmetric inequalities (15) as follows. For each route $r \in \mathcal{R}^0$ and each arc $a \in r$ we set

$$\mu_b^p := \begin{cases} 1 & \text{if } b = a \text{ and } p \in \mathcal{P}_r \\ 0 & \text{otherwise.} \end{cases}$$

Then $c^p = \sum_{k \in K} \kappa_{p,k} x_{p,k}$ and inequality (15) reads

$$\sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k} \geq \sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} y_{\tilde{r},0}. \quad (24)$$

We call these inequalities *basic direct connection constraints*.

Substituting inequalities (24) for the dcmetric inequalities, we obtain the following *basic*

direct connection model:

$$\begin{aligned} \text{(DC-basic)} \quad \min \lambda \sum_{p \in \mathcal{P}} \sum_{k \in K} c_{p,k} x_{p,k} + (1 - \lambda) \left(\sum_{r \in \mathcal{R}^0} \tau_{r,0} y_{r,0} + \sum_{r \in \mathcal{R}} \tau_{r,1} y_{r,1} \right) \\ \sum_{r \in \mathcal{R}_{st}^0} y_{r,0} + \sum_{r \in \mathcal{R}_{st}} y_{r,1} = d_{st} \quad \forall (s, t) \in D \end{aligned} \quad (25)$$

$$\sum_{r \in \mathcal{R}^0(a)} y_{r,0} + \sum_{r \in \mathcal{R}(a)} y_{r,1} \leq \sum_{p \in \mathcal{P}(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall a \in A \quad (26)$$

$$\sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} y_{\tilde{r},0} \leq \sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k} \quad \forall r \in \mathcal{R}^0, \forall a \in r \quad (27)$$

$$\sum_{k \in K} x_{p,k} \leq 1 \quad \forall p \in \mathcal{P} \quad (28)$$

$$x_{p,k} \in \{0, 1\} \quad \forall p \in \mathcal{P}, \forall k \in K \quad (29)$$

$$y_{r,0} \geq 0 \quad \forall r \in \mathcal{R}^0 \quad (30)$$

$$y_{r,1} \geq 0 \quad \forall r \in \mathcal{R}. \quad (31)$$

Model (DC-basic) minimizes, as model (DPC) or (DC-complete), a weighted sum of path and routing costs. Equations (25) enforce the demand flow. Inequalities (26) guarantee sufficient capacity on each arc. The direct connection constraints (27) approximate the sufficiency of capacities for direct connection routes on each arc.

Let $r_1, r_2 \in \mathcal{R}^0$. We say that routes r_1 and r_2 are *comparable* if $\mathcal{P}_{r_1} \subseteq \mathcal{P}_{r_2}$ or $\mathcal{P}_{r_2} \subseteq \mathcal{P}_{r_1}$, i. e., the set of direct connection paths supporting r_1 is contained in the set of direct connection paths supporting r_2 or vice versa. A set of routes $\mathcal{R}' \subseteq \mathcal{R}^0$ is a *comparable set of routes* if every two elements $r_1, r_2 \in \mathcal{R}'$ are comparable.

Proposition 6.1. *Model (DC-basic) overestimates the directly routable demand (with respect to model (DC-complete)) by a factor of at most*

$$\max_{p \in \mathcal{P}} \left\lfloor \frac{|V \cap p|}{2} \right\rfloor.$$

Proof. Consider a solution (x^*, y^*) of model (DC-basic). For every $r \in \mathcal{R}^0$ and $p \in \mathcal{P}_r$ define

$$z_{r,0}^p := y_{r,0}^* \cdot \frac{\sum_{k \in K} \kappa_{p,k} x_{p,k}^*}{\sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^*}.$$

Then it holds

$$\begin{aligned} \text{(i)} \quad \sum_{p \in \mathcal{P}_r} z_{r,0}^p &= y_{r,0}^* & \forall r \in \mathcal{R}^0 \\ \text{(ii)} \quad \sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} \sum_{p \in \mathcal{P}_{\tilde{r}}} z_{\tilde{r},0}^p &\leq \sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^* & \forall r \in \mathcal{R}^0, \forall a \in r \\ \text{(iii)} \quad z_{r,0}^p &\leq \sum_{k \in K} \kappa_{p,k} x_{p,k}^* & \forall p \in \mathcal{P}_r. \end{aligned}$$

This can be seen as follows:

$$\begin{aligned}
\text{(i)} \quad \sum_{p \in \mathcal{P}_r} z_{r,0}^p &= \sum_{p \in \mathcal{P}_r} y_{r,0}^* \cdot \frac{\sum_{k \in K} \kappa_{p,k} x_{p,k}^*}{\sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^*} = y_{r,0}^* & \forall r \in \mathcal{R}^0 \\
\text{(ii)} \quad \sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} \sum_{p \in \mathcal{P}_r} z_{\tilde{r},0}^p &\stackrel{(i)}{=} \sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} y_{\tilde{r},0}^* \stackrel{(27)}{\leq} \sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^* & \forall r \in \mathcal{R}^0, \forall a \in r \\
\text{(iii)} \quad \text{inequality (27) yields } y_{r,0}^* &\leq \sum_{\substack{\tilde{r} \in \mathcal{R}^0(a) \\ \mathcal{P}_{\tilde{r}} \subseteq \mathcal{P}_r}} y_{\tilde{r},0}^* \leq \sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^* & \forall p \in \mathcal{P}_r \\
\Rightarrow z_{r,0}^p &= y_{r,0}^* \cdot \frac{\sum_{k \in K} \kappa_{p,k} x_{p,k}^*}{\sum_{p \in \mathcal{P}_r} \sum_{k \in K} \kappa_{p,k} x_{p,k}^*} \leq \sum_{k \in K} \kappa_{p,k} x_{p,k}^* & \forall p \in \mathcal{P}_r.
\end{aligned}$$

Let \tilde{z} be defined as

$$\tilde{z}_{r,0}^p := \left\lfloor \frac{1}{\lfloor \frac{|V \cap p|}{2} \rfloor} \right\rfloor z_{r,0}^p.$$

We show that \tilde{z} is a solution of (H). Consider a path $p \in \mathcal{P}$ and $a \in p$. Let $p = (v_1, \dots, v_m)$ be the nodes the path traverses in this order and let $a = (v_l, v_{l+1})$, $l \in \{1, \dots, m-1\}$. Further denote by $\mathcal{R}^0(a, p) = \{r \in \mathcal{R}^0(a) : p \in \mathcal{P}_r\}$ the set of routes that contain arc a and that are supported by p . Define the following sets of routes

$$\mathcal{R}_i = \begin{cases} \{r \in \mathcal{R}^0(a, p) : r \text{ starts in } v_i\} & i \in \{1, \dots, l\} \\ \{r \in \mathcal{R}^0(a, p) : r \text{ ends in } v_i\} & i \in \{l+1, \dots, m\} \end{cases}$$

Since all routes and the path p are elementary, the sets \mathcal{R}_i , $i = 1, \dots, m$, are comparable sets of routes, and $\mathcal{R}_1 \dot{\cup} \dots \dot{\cup} \mathcal{R}_l$ and $\mathcal{R}_{l+1} \dot{\cup} \dots \dot{\cup} \mathcal{R}_m$ are partitions of the set $\mathcal{R}^0(a, p)$, respectively. Either $l \leq \lfloor \frac{|V \cap p|}{2} \rfloor$ or $m-l \leq \lfloor \frac{|V \cap p|}{2} \rfloor$. Assume w.l.o.g. $l \leq \lfloor \frac{|V \cap p|}{2} \rfloor$. We then get

$$\begin{aligned}
\sum_{r \in \mathcal{R}^0(a, p)} \tilde{z}_{r,0}^p &= \sum_{i=1}^l \sum_{r \in \mathcal{R}_i} \tilde{z}_{r,0}^p = \sum_{i=1}^l \sum_{r \in \mathcal{R}_i} \left\lfloor \frac{1}{\lfloor \frac{|V \cap p|}{2} \rfloor} \right\rfloor z_{r,0}^p \\
&\stackrel{(ii), (iii)}{\leq} \sum_{i=1}^l \left\lfloor \frac{1}{\lfloor \frac{|V \cap p|}{2} \rfloor} \right\rfloor \sum_{k \in K} \kappa_{p,k} x_{p,k}^* \leq \sum_{k \in K} \kappa_{p,k} x_{p,k}^*,
\end{aligned}$$

i. e., inequality

$$\sum_{r \in \mathcal{R}^0(a) : p \in \mathcal{P}_r} \tilde{z}_{r,0}^p \leq \sum_{k \in K} \kappa_{p,k} x_{p,k}^* \quad \forall p \in \mathcal{P}, \forall a \in p$$

is satisfied. The inequality

$$\sum_{p \in \mathcal{P}_r} \tilde{z}_{r,0}^p \leq y_{r,0}^* \quad \forall r \in \mathcal{R}^0$$

follows immediately with (i).

Hence,

$$\begin{aligned}
y_{r,0} &:= \sum_{p \in \mathcal{P}_r} \tilde{z}_{r,0}^p, \\
y_{r,1} &:= \begin{cases} y_{r,1}^* + (y_{r,0}^* - \sum_{p \in \mathcal{P}_r} \tilde{z}_{r,0}^p) & \text{for } r \in \mathcal{R}^0 \\ y_{r,1}^* & \text{for } r \notin \mathcal{R}^0 \end{cases} \\
x &:= x^*
\end{aligned}$$

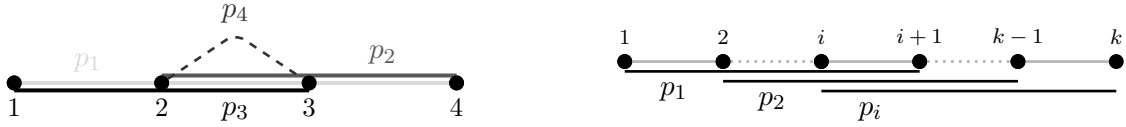


Figure 1: *Left:* Worst case example for Proposition 6.1 and approximation factor 2. *Right:* General worst case example for Proposition 6.1.

is a solution for (DC-complete). Overall, we get

$$\begin{aligned} \sum_{r \in \mathcal{R}^0} y_{r,0} &= \sum_{r \in \mathcal{R}^0} \sum_{p \in \mathcal{P}_r} \tilde{z}_{r,0}^p = \sum_{r \in \mathcal{R}^0} \sum_{p \in \mathcal{P}_r} \frac{1}{\lfloor \frac{|V \cap p|}{2} \rfloor} z_{r,0}^p \\ &\geq \frac{1}{\max_{p \in \mathcal{P}} \lfloor \frac{|V \cap p|}{2} \rfloor} \sum_{r \in \mathcal{R}^0} y_{r,0}^*, \end{aligned}$$

i. e., the number of direct travelers is overestimated by at most

$$\max_{p \in \mathcal{P}} \left\lfloor \frac{|V \cap p|}{2} \right\rfloor.$$

□

The bound is tight for the example illustrated in the left of Figure 1. We have four paths and two routes $r_1 = (1, 2, 3)$ and $r_2 = (2, 3, 4)$ for two OD pairs $(1, 3)$ and $(2, 4)$ with demand 100 each. The routes are incomparable on arc $(2, 3)$, i. e., operating path p_1 and p_4 with capacity 100, respectively, yields a solution for the basic direct connection model (DC-basic) where all demand is assumed to be routed directly. But in fact either the demand from 1 to 3 or the one from 2 to 4 cannot be routed directly. This example can be extended to longer paths as illustrated on the right of Figure 1. Here, we have a (gray) path with k nodes and $\lfloor \frac{k}{2} \rfloor$ routes. Each route is covered by an individual path. Furthermore, there are paths for each edge $\{j, j+1\}$, $j = 2, \dots, k-2$. Choosing the gray path and the edge-paths with enough capacity yields a solution for the basic direct connection model (DC-basic) where all demand can be routed directly. But the capacity of the gray path is not necessarily sufficient to yield a direct connection for more than one route.

Corollary 6.2. *If each path contains at most 3 nodes then the dcmetric inequalities (15) are implied by inequalities (24), i. e., in this case models (DC-basic) and (DC-complete) are equivalent.*

7 Pricing Non-Direct Connection Routes

Models (DPC), (DC-complete), and (DC-basic) require an enumeration of the direct connection routes. For models (DC-complete) and (DC-basic), however, this number is much smaller than for model (DPC) and can usually be handled without problems. We remark that we have also considered a column generation algorithm for direct connection routes for a relaxation of model (DC-basic) in [5].

The non-direct connection routes can be handled by column generation involving a shortest path computation for all three models. The reduced cost $\bar{\tau}_{r,1}$ for the non-direct connection

route variable $y_{r,1}$ is

$$\bar{\tau}_{r,1} = -\pi_{st} + \sum_{a \in r} (\mu_a + (1 - \lambda)\tau_a) + (1 - \lambda)\sigma, \quad r \in \mathcal{R}_{st}. \quad (32)$$

The corresponding pricing problem is to find a route r such that $\bar{\tau}_{r,1} < 0$ or to conclude that no such route exists. This can be done by a shortest path algorithm. The arc weights are set to $\omega_a = \mu_a + (1 - \lambda)\tau_a \geq 0$ for $a \in A$. We then have to add $(1 - \lambda)\sigma$ to the weight of the path. If we have found a path with weight smaller than π_{st} , the associated variable has to be added to the problem.

Proposition 7.1. *The pricing problem for the non-direct connection route variables in models (DPC), (DC-complete), and (DC-basic) can be solved in polynomial time. In particular, finding, for given origin s an st -route with negative reduced cost and at least one path-switch or concluding that no such route exists, needs $\mathcal{O}(|V| \log |V| + |A|)$ time.*

8 Model Comparison

Let us denote by $v_R(M)$ the optimal objective value of relaxation R of an integer programming model M . Considering the IP values and the LP relaxation values of all models of this paper, we, finally, get the following picture.

Proposition 8.1.

$$\left. \begin{array}{l} v_{IP}(\text{DPC}) \\ \parallel \\ v_{IP}(\text{DC-complete}) \end{array} \right\} \geq v_{IP}(\text{DC-basic}) \geq v_{IP}(\text{DC-skeleton}),$$

$$\left. \begin{array}{l} v_{LP}(\text{DPC}) \\ \parallel \\ v_{LP}(\text{DC-complete}) \end{array} \right\} \stackrel{(*)}{\geq} v_{LP}(\text{DC-basic}) \geq v_{LP}(\text{DC-skeleton}).$$

Inequality (*) is an equality for $\max_{p \in \mathcal{P}} |V \cap p| \leq 3$. All LPs can be solved in polynomial time.

9 Application to Line Planning

The direct connection network design problem aims at applications in areas such as traffic and transport. We are particularly interested in line planning, where the treatment of transfers is a major challenge. Another great challenge is the integration of line planning and passenger routing; it discloses essential degrees of freedom. Existing models either relax one of these two important aspects or they are of extremely large scale, and seem to be computationally intractable. A comprehensive survey article on methods and models to deal with the line planning problem is given by Schöbel [7].

A first approach to maximize the number of *direct travelers*, i. e., travelers that do zero transfers, was taken by Bussieck, Kreuzer, and Zimmermann [8] (see also the thesis of Bussieck [9]). They proposed an integer programming model on the basis of a “system split” of the demand, i. e., an a priori distribution of the passenger flow on the arcs of the

transportation network. The direct travelers approach is therefore a sequential passengers-first lines-second routing method. However, the passenger flow strongly depends on the line plan which is to be computed. Hence, a number of approaches that integrate line planning and passenger routing have been developed. Schöbel and Scholl [10, 11] model travel and transfer times explicitly in terms of a "change-and-go graph" that is constructed on the basis of all potential lines. This model allows a complete and accurate formulation of travel and transfer time objectives; its only drawback is its enormous size, which seems to make this model computationally intractable. Borndörfer, Grötschel, and Pfetsch [12, 13] propose an integrated line planning and passenger routing model with a polynomial number of constraints. This model ignores transfers between lines of the same mode (transfers between, e. g., bus and tram lines are considered).

We now propose an application of the considered direct connection models for the integrated line planning and passenger routing problem in public transport with a focus on direct connection routes for passengers. The considered graph $G = (V, E)$ represents the infrastructure of a *public transportation network*. The nodes define stations or stops and the edges define streets and tracks, e. g., for busses and trams. Every edge $e \in E$ is associated with a certain transportation *mode* $i \in M$, i. e., M is the set of modes, e. g., bus or tram. An edge further has a *travel time* $\tau_e \in \mathbb{Q}_{\geq 0}$ and a *cost* $c_e \in \mathbb{Q}_{\geq 0}$. All possible *lines* are defined by the set \mathcal{P} . A line can be operated at a *frequency* k out of a finite set $K \subseteq \mathbb{N}$. Line p at frequency k has *transportation capacity* $\kappa_{p,k} = \kappa_i \cdot k$, where κ_i is a *standard capacity* of a line of mode $i \in M$, e. g., the size of a bus. The line further has *operation cost* $c_{p,k} = c_i + k \cdot \sum_{e \in p} c_e$, where c_i is a *standard fixed cost* of a line of mode $i \in M$. The path costs, hence, corresponds to the operating costs of the lines. D gives the transportation demand between two different origin and destination nodes (OD-nodes). OD nodes can be seen as virtual nodes representing a certain area (a traffic cell) and all stations (network nodes) within the cell are connected to this OD node by so-called OD edges. The direct routing graph or *passenger routing graph* for generating the passenger routes arises from the graph G by replacing each undirected edge by two antiparallel arcs. This graph can further involve additional transfer arcs, e. g., between lines or stations of different modes. These transfer arcs are associated with the transfer penalty σ . From $r \in \mathcal{R}^0$ then follows that r does not contain a transfer arc. The cost of a route in the line planning setting corresponds to the sum of the travel times on the arcs plus a possible transfer penalty.

All of the models (DPC), (DC-complete), and (DC-basic) choose a set of lines and a set of passenger routes such that the capacities of the lines support these passenger routes and a weighted sum of line operating cost and passenger travel times is minimized. The models subdivides routes for passengers that do not contain a transfer arc into those who are supported by direct connection lines with sufficient capacity and those who are only supported by several lines including at least one transfer. The latter routes are penalized by the transfer penalty σ regardless how many transfers on this route are necessary. A method to account for each transfer exactly is to extend the passenger routing graph to a change-and-go network as proposed by Schöbel and Scholl. In this approach, each node and each arc has to be copied for every line that contains this node and arc; further transfer arcs have to be added. Models (DPC), (DC-complete), and (DC-basic) can be seen as a "first order approximation" to this change-and-go approach: Models (DPC), (DC-complete), and (DC-basic) do not consider transfer penalties for the second, third, etc., transfer in a passenger route. All models except (DPC) further relax the assignment of direct connection routes to particular lines.

Models (DPC) and (DC-complete) can be interpreted as a passenger routing extension of

the direct traveler model of Bussieck, Kreuzer, and Zimmermann [8].

Model (DC-skeleton) is quite similar to the column generation model of Borndörfer, Grötschel, and Pfetsch [12]. The set of routes \mathcal{R}^0 includes all routes that contain no transfer arc. If there are no additional constraints for direct connection passenger routes, an optimal solution will use a path-switching route $y_{r,1} > 0$ only if r contains a transfer arc. Such routes do not allow for a direct connection, i. e., there exist no variables $y_{r,0}$. We can therefore combine both types of variables $y_{r,0}$ and $y_{r,1}$ into variables y_r ; a transfer penalty is added if and only if r contains a transfer arc. In this case, model (DC-skeleton) is equal to the basic dynamic model introduced by Karbstein [6]; she shows the equivalence to the column generation model of Borndörfer, Grötschel, and Pfetsch [12]. In this way, models (DPC), (DC-complete), and (DC-basic) can also be seen as a "transfer improvement" of the model of Borndörfer, Grötschel, and Pfetsch.

10 Computational Results

In this section, we present computational results for models (DC-complete) and (DC-basic). Model (DC-complete) is equivalent to model (DPC), but has the advantage that we can handle direct connection routings more implicitly by using a cutting plane method for dcmetric inequalities. In this way, we do not have to add all dcmetric inequalities in every node, i. e., we can keep the size of the LP relatively small.

We consider four transportation networks that we denote as China, Dutch, SiouxFalls, and Potsdam. The instance SiouxFalls uses the graph of the street network with the same name from the Transportation Network Test Problems Library of Bar-Gera [14]. Instances China, Dutch, and Potsdam correspond to public transportation networks. The Dutch network was introduced by Bussieck in the context of line planning [15]. The China instance is artificial; we constructed it as a showcase example, connecting the twenty biggest cities in China by the 2009 high speed train network. The Potsdam instances are real multi-modal public transportation networks for 1998 and 2009.

For China, Dutch, and SiouxFalls all nodes are considered as terminals, i. e., nodes where lines can start or end. We constructed a line pool by generating for each pair of terminals all lines that satisfy a certain length restriction. To be more precise, the number of edges of a line between two terminals s and t must be less than or equal to f times the number of edges of the shortest path between s and t . For each network, we increased f in three steps to produce three instances with different line pool sizes. For Dutch and China instance number 3 contains all lines, i. e., all paths that are possible in the network. The line pools for the Potsdam network of 1998 are generated for different restrictions on the length of the lines considering the given turning restrictions on crossings. We defined all nodes as terminals that are terminals of operating lines in the year 1998. The Potsdam 2009 instance arose within a project with the Verkehr in Potsdam GmbH (ViP) [16] to optimize the 2010 line plan [17, 18]. The line pool contains all possible lines that fulfill the ViP requirements. The Potsdam instances are the only instances that are based on a real multi-modal network of bus, tram, regional, and commuter traffic, i. e., the passenger routing graphs for the Potsdam instances contain transfer arcs between lines of different modes. The lines for regional and commuter trains are not operated by ViP and we therefore fix them to given frequencies in our computations.

The other lines can be operated at frequencies 3, 6, 9, and 18; this corresponds to a cycle

Table 1: Statistics on the line planning instances and models. The first six columns list the instances, the number of non-zero OD pairs, number of OD nodes, number of nodes and edges of the preprocessed passenger routing graph. The remaining columns list the number of line variables which is equal in all considered models, the number of dcroutes in model (DC-complete) and (DC-basic) (in brackets number of dcroute variables in models (DPC)), the number of constraints of the skeleton model (DC-skeleton), and the number of constraints of model (DC-basic).

problem	$ D $	$ V_O $	$ V $	$ A $	$ \mathcal{L} $	x -vars	dc-vars	(lines)	(DC-s) cons	(DC-b) cons
Dutch1	420	23	23	106	402	1 608	1 280	(6 856)	1 076	2 931
Dutch2	420	23	23	106	2 679	10 716	6 450	(11 0648)	3 339	15 844
Dutch3	420	23	23	106	7 302	29 208	12 426	(457 182)	7 943	30 136
China1	379	20	20	98	474	1 896	1 656	(14 386)	1 179	3 583
China2	379	20	20	98	4 871	19 484	12 824	(387 226)	5 458	31 809
China3	379	20	20	98	19 355	77 420	38 965	(2 657 962)	19 932	92 653
SiouxF1	528	24	24	124	866	3 464	2 652	(23 924)	1 769	5 545
SiouxF2	528	24	24	124	9 397	37 588	20 208	(566 610)	10 196	48 427
SiouxF3	528	24	24	124	15 365	61 460	44 950	(1 283 254)	16 145	129 736
Potsd98a	7 734	107	344	2 746	207	775	19 106	(63 133)	9 964	18 185
Potsd98b	7 734	107	344	2 746	1 907	7 561	58 357	(1 853 030)	11 975	70 920
Potsd98c	7 734	107	344	2 746	4 342	17 301	106 228	(7 441 199)	14 353	132 948
Potsd2010	4 443	236	851	5 542	3 433	13 745	34 039	(904 965)	11 601	63 134

time of 60, 30, 20, and 10 minutes in a time horizon of 3 hours. Line costs are proportional to line lengths and the frequency plus a fixed cost term that is used to reduce the number of lines. The costs and the capacities of the lines depend on the mode of transportation (e. g., bus, tram). In the instances each edge is associated with exactly one mode, i. e., all lines on an edge have the same capacity, see Karbstein [6] for more details. The objective weighing parameter was set to $\lambda = 0.8$ and the transfer penalty was set to $\sigma = 15$ minutes.

Table 1 gives some statistics about the test instances and the models. The columns labeled $|D|$, $|V_O|$, $|V|$, $|A|$, and $|\mathcal{L}|$ list the number of OD pairs with non-zero demand, OD nodes, nodes, arcs, and lines after some preprocessing. The column labeled x -vars gives the number of line variables which is equal for all considered models. Column dc-vars gives the number of all direct connection routes for models (DC-complete) and (DC-basic); in brackets we list the number of all direct connection line routes, i. e., the variables based on the routes and the lines, for model (DPC). The second last column lists the number of starting constraints for model (DC-complete) after preprocessing of SCIP. This number involves no dcmetric inequalities (since they are separated afterwards) but all preprocessing inequalities described below. The last column lists the number of constraints for model (DC-basic) after preprocessing of the IP solver SCIP. In all instances, the number of basic direct connection constraints is higher than the number of other constraints. The ratio of modeling constraints for (DC-skeleton) and basic direct connection constraints varies from around 1:2 for Dutch1 to 1:8 for Potsdam98c.

The instances were solved with a column generation algorithm implemented on the basis of the CIP framework SCIP [19], version 2.1.0, see Achterberg [20], using CPLEX 12.4 as LP-solver (in single core mode). Line/frequency variables and the direct connection passenger route variables were enumerated, non-direct connection passenger route variables were priced with Dijkstra’s shortest path algorithm and a labeling algorithm for the constrained shortest path case which results if routes can contain transfer arcs. We mainly used the default settings of SCIP with few exceptions: We turned off all SCIP heuristics (they usually yield non-valid solutions for model (DC-complete) due to the implicit handling

of direct connection constraints), we reduced the value for a cut in the root node to be efficacious to 0.004, we changed the LP pricing parameter to “quickstart steepest edge”, and we increased the number of rounds within separation of cuts where the objective value may stall to 20. We further implemented a special rounding heuristic and preprocessing cuts that account for the demand on single arcs that disconnect at least two OD-nodes as well as the out-going and in-coming demand of an OD-node. Namely, we scale the capacity constraints associated with these demand sets by $\kappa^i k$, for each frequency $k \in K$, and apply mixed integer rounding. We also added violated cuts of the form

$$\sum_{p \in \mathcal{P}_r : r \in \mathcal{R}_{st}^0(a)} \sum_{k \in K} \min\{\kappa_{p,k}, d_{st}\} x_{p,k} \geq \sum_{r \in \mathcal{R}_{st}^0(a)} y_{r,0} \quad \forall (s, t) \in D, \forall a \in A_{st} \quad (33)$$

in each branching node in model (DC-complete); here $A_{st} := \{a \in A : r \in \mathcal{R}_{st}^0(a)\}$. These cuts can be related to the basic direct connection constraints (27) as follows. Instead of setting $\mu_a^p = 1$ for all paths that cover some dcroute r via arc a , we set

$$\mu_b^p := \begin{cases} 1 & \text{if } b = a \text{ and } p \in \{\mathcal{P}_r : r \in \mathcal{R}_{st}^0(a)\} \\ 0 & \text{otherwise,} \end{cases}$$

i.e., we set $\mu_a^p = 1$ for all paths that cover some st -dcroute containing arc a . This yields the following dcmetric inequality

$$\sum_{p \in \mathcal{P}_r : r \in \mathcal{R}_{st}^0(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \geq \sum_{\tilde{r} \in \mathcal{R}^0(a) : \mathcal{P}_{\tilde{r}} \subseteq \{\mathcal{P}_r : r \in \mathcal{R}_{st}^0(a)\}} y_{r,0} \quad \forall (s, t) \in D, \forall a \in A_{st}. \quad (34)$$

It takes some effort to identify the set of routes on the right-hand side of this inequality. A weaker but simpler version considers only the st -dcroutes covering arc a

$$\sum_{p \in \mathcal{P}_r : r \in \mathcal{R}_{st}^0(a)} \sum_{k \in K} \kappa_{p,k} x_{p,k} \geq \sum_{r \in \mathcal{R}_{st}^0(a)} y_{r,0} \quad \forall (s, t) \in D, \forall a \in A_{st}. \quad (35)$$

Note that the right-hand side $\sum_{r \in \mathcal{R}_{st}^0(a)} y_{r,0}$ is smaller than d_{st} , such that we can reduce the coefficients on the left-hand side; this yields (33).

For model (DC-complete) we implemented a constraint handler that ensures that each solution satisfies all dcmetric inequalities. We implemented an exact separation of these inequalities in the root node and one round of separation at every 100 nodes. In the root node we first search for violated basic direct connection constraints (27). We then search for violated dcmetric inequalities as follows: We optimize the program (H) that maximizes the number of direct travelers for a solution (x^*, y^*) of model (DC-complete*). If $\sum_{p \in \mathcal{P}_r} z_{r,0}^p = y_{r,0}^*$ for all $r \in \mathcal{R}^0$, we are done. Otherwise, we try to identify several violated cuts by solving a linear program for each route r with $\sum_{p \in \mathcal{P}_r} z_{r,0}^p + w = y_{r,0}^*$, $w > 0$, i. e., each route whose complete demand was not routed directly in the optimal solution of program (H). We further restrict this LP to the set of arcs $A_r = \{a \in A : a \in r\}$ covered by r and the set of routes $\mathcal{R}_r = \{\tilde{r} \in \mathcal{R}^0 : \exists a \in A_r \text{ s.t. } a \in \tilde{r}\}$ that share at least

Table 2: Statistics on the computations for models (DC-basic) and (DC-complete).

instance	nodes	(DC-basic)			nodes	(DC-complete)		
		LP-val.	best sol	gap		LP-val.	best sol	gap
Dutch1	687	2 613 065	2 613 065	opt	929	2 613 065	2 613 065	opt
Dutch2	8 666	2 609 666	2 609 666	opt	28 872	2 609 666	2 609 666	opt
Dutch3	6 520	2 609 097	2 609 097	opt	33 055	2 609 097	2 609 097	opt
China1	843 859	2 611 982	2 616 970	0.19%	501 433	2 616 795	2 626 661	0.38%
China2	33 701	2 517 394	2 530 929	0.54%	44 500	2 517 831	2 543 261	1.05%
China3	5 826	2 516 702	2 530 929	0.57%	1 817	2 516 767	2 543 261	1.09%
SiouxFalls1	539 278	646 140	646 592	0.07%	289 142	646 257	649 201	0.68%
SiouxFalls2	10 969	639 366	640 146	0.12%	21 687	639 382	643 129	0.58%
SiouxFalls3	4 042	638 825	639 401	0.09%	9 691	638 837	643 129	0.67%
Potsdam98a	34 558	1 017 129	1 018 656	0.15%	14 703	1 017 326	1 018 617	0.13%
Potsdam98b	207	979 975	984 095	0.42%	2 568	980 498	984 683	0.43%
Potsdam98c	51	978 815	984 169	0.55%	864	979 067	984 683	0.57%
Potsdam2010	1 474	208 577	209 783	0.58%	1 752	208 831	209 198	0.18%

one arc with route r . If the program

$$\begin{aligned}
 (S_r) \quad & \min \sum_{p \in \mathcal{P}} \sum_{a \in A_r \cap p} \mu_a^p \\
 \text{s.t.} \quad & \sum_{p \in \mathcal{P}} \sum_{a \in A_r \cap p} \mu_a^p \sum_{k \in K} \kappa_{p,k} x_{p,k}^* - \sum_{\tilde{r} \in \mathcal{R}_r} \omega_{\tilde{r}} y_{\tilde{r},0}^* \leq -w \\
 & \sum_{a \in \tilde{r}} \mu_a^p - \omega_{\tilde{r}} \geq 0 \quad \forall \tilde{r} \in \mathcal{R}_r, \forall p \in \mathcal{P}_{\tilde{r}} \\
 & \mu_a^p \geq 0 \quad \forall p \in \mathcal{P}, \forall a \in A_r \cap p \\
 & 1 \geq \omega_{\tilde{r}} \geq 0 \quad \forall \tilde{r} \in \mathcal{R}_r.
 \end{aligned}$$

has a solution, then the first inequality gives rise to a violated cut. LP (S_r) is a heuristic restriction of the separation LP (S) , that turned out to be much more efficient than the entire model (S) or model (S_r) for $A_r = A$. Note that there are rare cases possible where (S_r) is infeasible; we then consider $A_r = A$.

Finally, we included additional auxiliary *branching variables* $h_{a,i} \in \mathbb{Z}_{\geq 0}$, $a \in A$, $i \in K$, that account for the number of lines on arc a with frequency greater than or equal to i , and the corresponding *branching constraints* $\sum_{p \in \mathcal{P}: e(a) \in p} \sum_{k \in K: k \geq i} x_{p,k} = h_{a,i} \forall a \in A, k \in K$. Including these branching variables and constraints combines the possibility to branch on those constraints with the sophisticated branching rules implemented in the SCIP framework. This works well especially for the Dutch instances, e.g., it needs nearly a thousandfold of branching nodes to solve instance Dutch1 with model (DC-basic) without branching variables compared to the solution with the branching variables.

We set a time limit of 5 hours for all instances. All computations were done on computers with an Intel(R) Xeon(R) CPU X5672 with 3.20 GHz, 12 MB cache, and 48 GB of RAM.

Table 2 shows statistics on the number of branching nodes, LP-value, best solution, and the integrality gap for model (DC-basic) and (DC-complete). Our implemented heuristic computes a direct connection routing with model (H) for (DC-complete), compare with Section 5. Hence, this solution may be improvable by computing the exact direct connection routing for the given line capacities. This can be done by fixing the line capacities

Table 3: Evaluation of the best solutions of models (DC) and (DC-complete). The columns 2 to 5 list travel times (in minutes), costs, objective value, and number of direct travelers¹ by computing a system optimum passenger routing in the change-and-go graph. The second last column shows the number of direct traveler predicted² by the considered model while the last column gives the over- or under-estimation of direct travelers.

problem	travel times	costs	obj.	dir. trav. ¹	dir. trav. ²	diff. (%)
Dutch1 (DCc)	$1.279 \cdot 10^7$	69500	2613305	179 536	179 536	-
Dutch1 (DCb)	$1.279 \cdot 10^7$	68900	2613305	179 376	179 376	-
Dutch2 (DCc)	$1.279 \cdot 10^7$	64400	2609841	179 796	179 796	-
Dutch2 (DCb)	$1.279 \cdot 10^7$	64400	2609841	179 796	179 796	-
Dutch3 (DCc)	$1.278 \cdot 10^7$	65600	2609271	180 234	180 234	-
Dutch3 (DCb)	$1.278 \cdot 10^7$	65600	2609271	180 234	180 234	-
China1 (DCc)	$1.200 \cdot 10^7$	284 172	2626661	749 571	749 571	-
China1 (DCb)	$1.226 \cdot 10^7$	275 624	2672905	731 333	749 924	2.54%
China2 (DCc)	$1.166 \cdot 10^7$	262 387	2542900	759 680	759 452	-0.03%
China2 (DCb)	$1.196 \cdot 10^7$	250 824	2592971	743 271	759 933	2.24%
China3 (DCc)	$1.166 \cdot 10^7$	262 387	2542900	759 680	759 452	-0.03%
China3 (DCb)	$1.196 \cdot 10^7$	250 824	2592971	743 271	759 933	2.24%
SiouxFalls1 (DCc)	$3.198 \cdot 10^7$	12 007	649162	360 600	360 585	-0.00%
SiouxFalls1 (DCb)	$3.226 \cdot 10^7$	8 801	652238	360 408	360 600	0.05%
SiouxFalls2 (DCc)	$3.181 \cdot 10^7$	10 312	644432	360 600	360 578	-0.01%
SiouxFalls2 (DCb)	$3.197 \cdot 10^7$	6 094	644365	360 600	360 600	-
SiouxFalls3 (DCc)	$3.181 \cdot 10^7$	10 312	644432	360 600	360 578	-0.01%
SiouxFalls3 (DCb)	$3.211 \cdot 10^7$	5 200	646392	360 566	360 600	0.01%
Potsdam98a (DCc)	$5.077 \cdot 10^6$	28 063	1037770	70 676	71 324	0.92%
Potsdam98a (DCb)	$5.055 \cdot 10^6$	22 113	1033137	71 275	71 166	-0.15%
Potsdam98b (DCc)	$4.848 \cdot 10^6$	34 604	997429	78 814	78 955	0.18%
Potsdam98b (DCb)	$4.815 \cdot 10^6$	28 794	991858	79 357	79 246	-0.14%
Potsdam98c (DCc)	$4.848 \cdot 10^6$	34 604	997429	78 814	78 955	0.18%
Potsdam98c (DCb)	$4.815 \cdot 10^6$	28 794	991858	79 357	79 212	-0.18%
Potsdam2010 (DCc)	$1.021 \cdot 10^6$	9 464	211816	38 284	38 118	-0.43%
Potsdam2010 (DCb)	$1.024 \cdot 10^6$	9 663	212631	38 036	37 984	0.14%

and generating all violated dcmetric inequalities. This procedure sometimes needs several rounds to terminate. We did this for each final solution of model (DC-complete) and limited the number of separation rounds to 300. The integrality gap for model (DC-basic) after 5 hours is smaller than the one for model (DC-complete) for all instances but Potsdam98a and Potsdam2010. All Dutch instances can be solved to optimality by both models, while the computation with (DC-basic) needs less branching nodes and, hence, less time than the computation with model (DC-basic). However, the numbers show that both models can be solved efficiently.

We evaluate the quality of the solutions of model (DC-complete) and (DC-basic) by computing an optimal passenger routing, including penalties for all transfers, in a change-and-go graph similar to that of Schöbel and Scholl [10]. Recall that this graph contains a copy of each node and arc for every line that contains this node and arc. Further transfer arcs are added between two nodes of different lines whenever a transfer between these two lines is possible on this node. The travel time of all arcs is set to the travel time of the associated arc in G , transfer arcs are additionally penalized by σ . We then fix the frequencies of the lines according to the computed line plan and route the passengers to minimize the total travel and transfer times, i.e., we compute the correct number of transfers for all passengers in a system optimum routing. Table 3 shows the result of this

evaluation. More precisely, columns 2, 3, and 4 give the travel times, costs, and objective values of this system optimum for the best solutions computed with model (DC-complete) and model (DC-basic), respectively. Column 5 lists the number of direct travelers in the change-and-go-system-optimum while column 6 lists the number of direct travelers predicted with the considered model. The last column gives the relative difference between these two numbers.

It can be seen that the exact number of direct travelers is very close to the number of direct travelers predicted by models (DC-complete) and (DC-basic), respectively. The number of direct travelers for the instances Dutch, China, and SiouxFalls predicted by model (DC-complete) is equal to or underestimates the exact number of direct travelers. Each passenger path in the solution of these instances is either a direct connection path or involves exactly one transfer. The underestimation is because of a primal solution computation based on a separation of direct connection constraints which may cause some numerical problems. Here, we stopped the separation after 300 rounds. The solutions of the Potsdam instances contain passenger paths with 2 or more transfers. Hence, an overestimation of direct connection passenger paths by model (DC-complete) is possible since it may be better to route passengers on paths with transfers instead of direct connection paths to obtain shorter connections for passengers with 2 or more transfers. For the Dutch instances the basic direct connection constraints (27) suffice to define a correct direct connection routing. Considering the objective values of all instances for both models, there is no clear winner. While model (DC-complete) computes better solutions for the China instances, model (DC-basic) yields better solutions for the Potsdam98 instances.

In a final test, we tried to solve the change-and-go model of Schöbel and Scholl using the best solution of model (DC-complete) as a starting solution. Within five hours of computation time the root node could not be solved for Dutch3, China3, SiouxFalls2, SiouxFalls3, and all Potsdam instances. After five hours of branch-and-bound only the solution for SiouxFalls1 could be improved by 0.18% from 649 162 to 647 985.

Our results show that the direct connection approach is an efficient method to solve large-scale real-world line planning problems in public transport. Moreover, the basic direct connection model with its combinatorially motivated basic direct connection constraints is a computationally strong and computationally (relatively) simple approximation for the complete model (DC-complete).

References

- [1] M. Iri, On an extension of the maximum-flow minimum-cut theorem to multicommodity flows, *Journal of the Operations Research Society of Japan* 13 (3) (1971) 129–135.
- [2] K. Onaga, O. Kakusho, On feasibility conditions of multicommodity flows in networks, *Transactions on Circuit Theory* 18 (4) (1971) 425–429.
- [3] C. Raack, Capacitated network design – multi-commodity flow formulations, cutting planes, and demand uncertainty, Ph.D. thesis, TU Berlin (2012).
- [4] A. Balakrishnan, T. L. Mangnanti, P. Mirchandani, Network design, in: M. Dell’Amico, F. Maffioli, S. Martello (Eds.), *Annotated Bibliographies in Combinatorial Optimization*, Wiley, Chichester, 1997, Ch. 18, pp. 311–334.

- [5] R. Borndörfer, M. Karbstein, A direct connection approach to integrated line planning and passenger routing, in: D. Delling, L. Liberti (Eds.), ATMOS 2012 - 12th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems, OpenAccess Series in Informatics (OASISs), Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2012, pp. 47–57. doi:<http://dx.doi.org/10.4230/OASISs.ATMOS.2012.47>. URL <http://drops.dagstuhl.de/opus/volltexte/2012/3702>
- [6] M. Karbstein, Line planning and connectivity, Ph.D. thesis, TU Berlin (2013).
- [7] A. Schöbel, Line planning in public transportation: models and methods, OR Spectrum (2011) 1–20. URL <http://dx.doi.org/10.1007/s00291-011-0251-6>
- [8] M. R. Bussieck, P. Kreuzer, U. T. Zimmermann, Optimal lines for railway systems, Eur. J. Oper. Res. 96 (1) (1997) 54–63.
- [9] M. R. Bussieck, Optimal lines in public rail transport, Ph.D. thesis, TU Braunschweig (1997).
- [10] A. Schöbel, S. Scholl, Line planning with minimal traveling time, in: L. G. Kroon, R. H. Möhring (Eds.), Proc. 5th Workshop on Algorithmic Methods and Models for Optimization of Railways, 2006.
- [11] S. Scholl, Customer-oriented line planning, Ph.D. thesis, University of Göttingen (2005).
- [12] R. Borndörfer, M. Grötschel, M. E. Pfetsch, A column-generation approach to line planning in public transport, Transportation Science 1 (41) (2007) 123–132. URL <http://www.zib.de/Publications/abstracts/ZR-05-18/>
- [13] R. Borndörfer, M. Grötschel, M. E. Pfetsch, Models for line planning in public transport, in: M. Hickman, P. Mirchandani, S. Voß (Eds.), Computer-aided Systems in Public Transport, Vol. 600 of Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, 2008, pp. 363–378.
- [14] Transportation network test problems, <http://www.bgu.ac.il/~bargera/tntp/>.
- [15] GAMS, M. Bussieck, lop.gms: Line optimization, <http://www.gams.com/modlib/libhtml/lop.htm>.
- [16] Stadtwerke Potsdam – ViP Verkehrsbetrieb Potsdam GmbH, Website, <http://vip-potsdam.de>.
- [17] R. Borndörfer, I. Friedow, M. Karbstein, Optimierung des Liniensplans 2010 in Potsdam, Der Nahverkehr 30 (4) (2012) 34–39. URL <http://vs24.kobv.de/opus4-zib/frontdoor/index/index/docId/1448>
- [18] R. Borndörfer, M. Neumann, Linienoptimierung – reif für die Praxis?, in: HEUREKA’11, FGSV Verlag, 2011.
- [19] SCIP – Solving Constraint Integer Programs, documentation, <http://scip.zib.de>.
- [20] T. Achterberg, SCIP: Solving Constraint Integer Programs, Mathematical Programming Computation 1 (1) (2009) 1–41.