Integer Programming Models for the Delay Management Problem

Anita Schöbel

Universität Kaiserslautern

and

Fraunhofer Institut für Techno-und Wirtschaftsmathematik

AMORE Research Seminar 2002
Plan of the talk

The delay management problem

Application
Introduction
Minimizing the total delay
The model with constant weights
Algorithms for (TDM-const)
Conclusions

September 27, 2002
Our Application

Why delay management?

Project: together with two traffic associations (VRN and VGS) and DB Regio

supported by: Stiftung Innovation Rheinland-Pfalz

Application 1:
On-line-decision for bus-drivers at the “Lautertalbahn”: wait for delayed trains or not?

Application 2:
Find perturbed timetables in case of known disturbances (construction areas, etc.) in the Saarland.
Size of the problem

Testregion

- 1100 stations
- 1350 vehicles within a day
- 11700 changing possibilities

resulting in a graph with

- 23000 nodes
- 25000 edges
Delay Management

Suppose that train $i$ arrives at some station $k$ with a delay. What should the connecting buses do?

Each bus $j$ may either

**WAIT** and therefore cause delay for
- the customers within the bus $j$,
- the customers who want to get on the bus $j$, and,
- for subsequent other buses, or

**DEPART** and therefore accept that customers who want to change from the delayed train $i$ onto the bus $j$ miss their connection.

QUESTION: **WAIT** or **DEPART**?
Wait or depart?

Two extreme solutions:

All vehicles **wait**:
- No connection is missed!
- Many vehicles are delayed.

All vehicles **depart in time**:
- Minimizes the number of delayed vehicles.
- Many connections are missed.

A solution between these extremes:

**Minimize the total delay!**
What is the total delay?

**Total delay** is the sum of all delays over all customers.

**Assumptions:**

- $T$ is the fixed time period before the next vehicle of the same type arrives.
- In the next period, all vehicles are in time.

Consider a customer traveling on a path

$$p = (\text{station, vehicle, station, vehicle, \ldots, station}).$$

- If all connections are maintained: Delay of the customer is the arrival delay of its last vehicle at its last station
- If at least one connection is missed, the delay is $T$.

**Consequence:** To calculate the total delay, a *perturbed timetable* has also to be found.
Find wait-depart decisions and a feasible perturbed timetable for all vehicles at all stations such that the total delay over all customers is minimized.

- What is a feasible perturbed timetable?
- How to calculate the total delay?
Introduction

**Literature review**

**For finding timetables in public transportation:** Many papers, e.g., Adamski, Bookbinder, Bowman, Burkhard, Brucker, Cai, Carey, Ceder, Chmiel, Daduna, Desilet, Domschke, Egmont, Fleischmann, Goh, Goberde, Higgins, Holz, Hüttmann, Hurink, Klemt, Krista, Nachtigall, Odijk, Serafini, Stemme, Turnquist, Ukovich, Voget, Voß, Weigand ...


Minimizing the total delay

Event-activity networks

see Nachtigall (1998)

\[ \mathcal{E} = \mathcal{E}_{arr} \cup \mathcal{E}_{dep}, \quad \mathcal{A} = \mathcal{A}_{drive} \cup \mathcal{A}_{wait} \cup \mathcal{A}_{change} \]
Variables and Parameters

Variables:

- $y_i$ is delay of event $i \in \mathcal{E}$.
- $z_a = \begin{cases} 
0 & \text{if connection } a \text{ is maintained} \\
1 & \text{if connection } a \text{ is missed} 
\end{cases}$

Parameters:

- $s_a$ slack time for all activities $a \in \mathcal{A}$.
- $d_i$ source delays for all $i \in \mathcal{E}$ (maybe 0)
- $\mathcal{P}$ set of customers paths, given as a sequence of events.
Feasible perturbed timetable

A perturbed timetable is feasible, if:

- for all $a = (i, j) \in \mathcal{A}_{wait} \cup \mathcal{A}_{drive}$ and all $a \in \mathcal{A}_{change}$ with $z_a = 0$:
  \[ y_i - y_j \leq s_a \]

- for all events:
  \[ y_i \geq d_i \]
Calculating the total delay

**Idea:** Calculate the delay on each activity and sum over all activities.

- The additional delay on activity \( a \) is given by the tension \( y_j - y_i \) (can be positive or negative).
- The additional delay, if \( a \) is missed is \( T - y_i = (T - y_j) + (y_j - y_i) \).
- The delay of a path is the sum of all additional delays over its activities.

**Define:** \( w_a \) as the number of customers using activity \( a \).

**The total delay is**

\[
f_{TDM} = \sum_{a=(i,j) \in A^s} w_a(y_j - y_i) + \sum_{a=(i,j) \in A_{change}} w_a z_a(T - y_j).
\]
Minimizing the total delay

The Model (TDM)

\[ \min \sum_{a=(i,j) \in A^s} w_a(y_j - y_i) + \sum_{a=(i,j) \in A_{change}} w_a \tilde{z}_a(T - y_j) \]

s.t. \[ y_i \geq d_i \text{ for all } i \in D \]
\[ y_i - y_j \leq s_a \text{ for all } a = (i, j) \in A_{wait} \cup A_{drive} \]
\[ -M \tilde{z}_a + y_i - y_j \leq s_a \text{ for all } a = (i, j) \in A_{change} \]
\[ \tilde{z}_a^p + \sum_{\tilde{a} \in p \cap A_{change} : \tilde{a} \prec a} \tilde{z}_\tilde{a} \geq 1 \text{ for all } p \in P^s \text{ and } a \in p \]
\[ \tilde{z}_a^p + \tilde{z}_\tilde{a} \leq 1 \text{ for all } p \in P^s \text{ and } a, \tilde{a} \in p \text{ with } \tilde{a} \prec a \]
\[ w_a = \sum_{p \in P^s : a \in p} w_p \tilde{z}_a^p \text{ for all } a \in A^s \]
\[ y_i, w_a \in \mathbb{N}^0 \]
\[ z_a, \tilde{z}_a^p \in \{0, 1\} \]
Model 2 is too complicated!!

Idea: Fix the weights $w_a$ as parameters by defining

$$w_a = \sum_{p \in P: a \in p} w_p,$$

i.e., assume that all customers can travel as they have planned.
The model with constant weights

\[ \text{(TDM-const)} \]

\[
\min_{a=(i,j) \in A^s} \sum_{a=(i,j) \in A^{\text{change}}} w_a (y_j - y_i) + \sum_{a=(i,j) \in A^{\text{change}}} w_a z_a (T - y_j)
\]

such that

s.t. \quad y_i \geq d_i \quad \text{for all } i \in D

\[ y_i - y_j \leq s_a \quad \text{for all } a = (i, j) \in A_{\text{wait}} \cup A_{\text{drive}} \]

\[ -M z_a + y_i - y_j \leq s_a \quad \text{for all } a = (i, j) \in A_{\text{change}} \]

\[ y_i \in \mathbb{N}^0 \quad \text{for all } i \in E \]

\[ z_a \in \{0, 1\} \quad \text{for all } a \in A^s \]
Note that

\[
\sum_{a=(i,j) \in A^s} w_a (y_j - y_i) = \sum_{i \in \mathcal{E}} w_i y_i
\]

if we define

\[
w_i = \sum_{p \in \mathcal{P} : \text{p ends at } i} w_p.
\]

This yields:

\[
f_{TDM} = \sum_{i \in \mathcal{E}} w_i y_i + \sum_{a=(i,j) \in A_{change}} w_a z_a (T - y_j)
\]

and forget about subtracting \(y_i\).

**Consequence: A linear model!!**
The model with constant weights

\[(TDM-\text{const})\]

\[
\min f_{TDM-\text{const}} = \sum_{i \in \mathcal{E}} w_i y_i + \sum_{a = (i, j) \in \mathcal{A}_{\text{change}}} w_a z_a T
\]

such that

\[
y_i \geq d_i \quad \text{for all } i \in \mathcal{D}
\]
\[
y_i - y_j \leq s_a \quad \text{for all } a = (i, j) \in \mathcal{A}_{\text{wait}} \cup \mathcal{A}_{\text{drive}}
\]
\[
-M z_a + y_i - y_j \leq s_a \quad \text{for all } a = (i, j) \in \mathcal{A}_{\text{change}}
\]
\[
y_i \in \mathbb{IN}^0 \quad \text{for all } i \in \mathcal{E}
\]
\[
z_a \in \{0, 1\} \quad \text{for all } a \in \mathcal{A}^s
\]
The model (TDM-const) is in general wrong.

...not really surprisingly!
The never-meet-property

The delay management problem has the **never-meet-property** if in each (time-minimal) feasible solution with **zero slack times** for all $a \in A$:

For all $j \in \mathcal{E}$:

1. If $(i_1, j), (i_2, j) \in A$, and $y_{i_2} > 0$ then $y_{i_1} = 0$.

2. If $(i_1, j) \in A$, and $d_j > 0$ then $y_{i_1} = 0$.

\[
\text{i1} \xrightarrow{\text{delayed}} \text{j} \\
\text{i2} \xrightarrow{\text{delayed}} \text{j}
\]
Consequence of the never-meet-property

In all optimal solutions we have: If $\tilde{a} = (\tilde{i}, \tilde{j})$ and $z_{\tilde{a}} = 1$, then

1. $y_i = 0$ for all $i$ which can be reached from $\tilde{i}$.

2. $z_a = 0$ for all $a$ which can be reached from $\tilde{i}$. 
**Good news for (TDM-const)**

**Theorem:** (TDM-const) is correct if the never-meet-property holds

Proof: If $w_a$ is not the correct weight, the additional delay on $a$ is zero.
More good news for (TDM-const)

- The never-meet-property can be tested efficiently by the critical path method (forward phase)
- The never-meet property is in practice often almost satisfied

In our data (1100 stations, 1350 vehicles, $\geq 10000$ changing possibilities)

e.g.,

- 120 delayed vehicles
- source delay of 10 minutes each

results in only 148 conflicts!
The model with constant weights

The never-meet-property in practice

Example: 10 delayed vehicles
The never-meet-property in practice

Example: source delay of 10 minutes
Solving (TDM-const) for zero slack times

see [Scholl, 01]

- coefficient matrix is totally unimodular
- Solve a linear program

Numerical results

- on a 750 MHz Pentium III processor, 128 MB RAM
- with XPRESS (optimiser version 11.14)
- Size of LP: 70730 variables, 68250 constraints
- Even for 200 delayed vehicles not more than 3 minutes!
Solving (TDM-const) for arbitrary slack times

**Theorem:** (TDM-const) can be solved in $O(|\mathcal{A}|)$ time if the never-meet property holds.

**Idea of the enumeration-algorithm**

1. **Decompose:** At each changing activity $a$ decompose into problems $a_1, \ldots, a_L$ where there is no other changing activities between each of the $a_1, \ldots, a_L$ and $a$.

   **Lemma:** The problems $a_1, \ldots, a_L$ are independent of each other

2. **Compose:** The optimal solution for $a$ can be determined if we know the optimal solutions for $a_1, \ldots, a_L$, i.e.,
   
   - missing $a$ costs $w_a T$
   - maintaining $a$ costs $f_{a_1} + \ldots + f_{a_L} + C'(a)$. 

September 27, 2002
Conclusions

Summary

1. **Complicated** Model (TDM) to minimize the total delay

2. Simplification to a **linear integer program** (TDM-const), **but wrong**?

3. (TDM-const) **is correct** if the never-meet property holds!

4. The never-meet-property **can be tested efficiently**

5. (TDM-const) can be **solved in linear time** if the never-meet property holds.
Other aspects of delay management

- The enumeration algorithm can be extended to Branch and Bound for the general case (Schöbel, 2002)
- There exists a more general linear model for (TDM) (see Schöbel, 2001)
- Bicriterial delay management (see Ginkel and Schöbel, 2002) using methods of the DTCTP in the event-activity network
- Modeling railway specific requirements as fixed connections in alternative graphs, see (Schöbel, 2002)
- Allowing different time periods (see Schöbel and Scholl, 2002)