Finding and evaluating robust train timetabling solutions

Matteo Fischetti
Domenico Salvagnin
Arrigo Zanette

DEI, University of Padova
Utrecht, April 19, 2007

Robust planning and Rescheduling in Railways

Work supported by the Future and Emerging Technologies unit of the EC (IST priority), under contract no. FP6-021235-2 (project ARRIVAL)
Problem Definition

- single one-way line
- aperiodic daily timetable to be designed

Minimize the timetable cost computed as follows…
Shift

Ideal Departure Instant

Station 1

Ideal timetable

Actual timetable

Station 2

Stop

Station 3

Stop

Station 4

Stretch
An event-scheduling MIP model

Variables:
- Arrival and departure times (event times) \( t_i \)
- Binary variables modeling event precedences \( x_{ij} \)

Constraints:
- Minimum travel times \( d_{ij} \)
- Safety constraints:
  - buffer times, no overtaking outside stations, etc.
  - Typical constraints of the type: \( t_i - t_j \geq d_{ij} - Mx_{ij} \)

Objectives:
- Minimize the cost of the schedule
- Robustness (whatever it means)
Timetable robustness …

- … is not concerned with major disruptions
- … is not intended to cope with heavy truck breaks or alike

- to be handled by REAL TIME CONTROL SYSTEMS

- … is a way to control delay propagation
- … has to favor delay compensation without heavy actions from the traffic control center

- no overtaking allowed to prevent delay propagation
- no train cancellation
- train precedences unchanged w.r.t. to the planned timetable
Non-robust and robust timetables
Our approach

- Take a **feasible timetable** -> near-optimal solution of the “nominal” timetable problem
- Fix a maximum **price of robustness** → the cost of the robust solution cannot exceed by more than XX% the optimal cost of the nominal problem
- **Fix all train precedences** (binary var.s $x_{ij}$ in the MIP model)
- **Relax the integrality** on the event-time var.s $t_i$ (the only unknowns)
- Enforce **robustness** in the resulting LP by using alternative techniques
- Evaluate the achieved robustness through a common **validation model**
- Compare the results
Pursuing robustness in the LP/MIP context

• **Stochastic Programming**
  - Take first-stage decisions
  - Pay for restoring feasibility afterwards (second-stage recourse var.s)
  - Applied successfully by the Kroon’s group to periodic timetabling
  - Very flexible but computationally heavy in scenario-based approaches

• **Robustness à la Bertsimas-Sim**
  - Kind of worst-case analysis of robustness
  - Limits the moves of the adversary (just a few coefficients can change in each constraint)
  - Feasibility deterministic (if adversary behaves as expected) or with high probability (otherwise)
  - Very simple model
  - Unfortunaltely, of no use in the timetable context (infeasible or very inefficient solutions)

• **Light Robustness**
  - “Light” version of Bertsimas-Sim using slack variables for “too conservative” constr.s
  - Linear or quadratic objective function (minimize slack var.s)
  - Very well suited for timetabling → talk in the afternoon…
Stochastic programming model

- Two-stage model with **recourse** var.s (unabsorbed delay)

**Nominal constraint**

\[ t_i - t_j + s_{ij}^{(\omega)} \geq d_{ij} + \delta_{ij}^{(\omega)} \]

**Recourse** \hspace{1cm} **Disturbance**

- Deterministic model through **scenario** expansion

\[ t_i - t_j + s_{ij}^{(r)} \geq d_{ij} + \delta_{ij}^{(r)}, \quad \forall r \in [1 \ldots N] \]

- Objective function: minimize the average unabsorbed delay

- **big LP model to be solved** (though each scenario actually introduces just a few new var.s and constr.s)
Scenario Generation

- Delay model:
  - Random cumulative train delay
  - Scaled by time band factors
  - Distributed across lines with section factors
Validation model

• Simulation tool used to **evaluate** the actual robustness of a given timetable \((\tilde{x}, \tilde{t})\).

• Uses information on the line to generate a delay scenario for each run.

• For each run, solve an **LP model** to **absorbe as much delay as possible**
  
  - Fixed precedences \( x_{ij} := \tilde{x}_{ij} \)
  
  - Continuous event-time var.s only \( t_i = \) actual times in the delayed schedule
  
  - Cannot anticipate with respect to the input solution to evaluate \( t_i \geq \tilde{t}_i \)
  
  - Minimize sum of delays (event-time shifts) \( \min \sum_i (t_i - \tilde{t}_i) \)

• Gather **statistical information**
Test bed

- Real world instances from RFI
  - PD-BO: 17 stations, ~35 trains
  - BZ-VR: 27 stations, ~130 trains
  - Mu-VR: 48 stations, ~50 trains
  - Br-BO: 48 stations, ~70 trains

- For each instance, 5 almost-optimal (non-robust) timetables computed by DEIS
Validation results

![Graph showing validation results for Line BZ–VR, nominal solution sol01.xml. The graph plots average cumulative delay (min) against efficiency loss (%). The graph compares stochastic model solution (400 scenarios) and LR (quadratic objective).]
Computing times

![Bar chart showing computation times for different scenarios and models. The x-axis represents different scenarios and models: Br-BO, BZ-VR, MU-VR, PD-BO. The y-axis represents time in seconds. The bars are color-coded for Scenario generation (blue), Stochastic model (red), and Quadratic LR model (green).]
Restoring integrality on the timetable var.s
Importance of the quadratic LR function
Computing times (updated)
Thanks