Train Scheduling: models, decomposition methods and practice

Carlo Mannino
Dept. of mathematics and cybernetics, SINTEF
Dept. of mathematics, University of Oslo

www.schedulingseminar.com
Dispatchers at work
Oslo Central Station
Train Scheduling: Two basic versions

- Operational (real-time): train rescheduling (dispatching)
- Tactical/Strategical: train timetabling
- Train scheduling: a job-shop scheduling problem
- Job-shop scheduling problem arising in other applications
The tracks of the railway are segmented into elementary "blocks"

Each block can accommodate at most one train at a time
Modelling train movement

- A train runs through a sequence of blocks (its *route*).

- $t^i_q$ is the time train $i$ enters block $q$ (*schedule variable*).

- If $t_u$ is the time the train enters a block, and $t_v$ when it enters next one, then

\[ t_v - t_u \geq l_{uv}, \]

where $l_{uv}$ is the minimum running for the train through the block.
The route graph

- The train movement represented by route graph
- Nodes correspond to (the event) entering a block section.
- Edges represent time precedence constraints \( t_g^i - t_d^i \geq l_{dg}^i \)
A route graph in Oslo S
The time origin

- We add a node $o$ representing the start $t_o$ of the planning horizon.

- Event $d$ starting **not before** time $\Delta$: $t_d - t_o \geq 30$

- Event $a$ starting **at** time $\Delta$: $t_a - t_o = 10 \rightarrow \begin{cases} t_a - t_o \geq 10 \\ t_o - t_a \geq -10 \end{cases}$
(potential) Conflicts

- Trains compete for the same blocks
- Either train $i$ enters block $g$ before $j$ enters $d$: $t^j_d - t^i_g \geq 0$
- Or train $j$ enters block $c$ before $i$ enters $d$: $t^i_d - t^j_c \geq 0$

\[ t^j_d - t^i_g \geq 0 \lor t^i_d - t^j_c \geq 0 \]

Disjunctive constraint
Disjunctive arc

t^j_d - t^i_g \geq 0 \lor t^i_d - t^j_c \geq 0 \quad \text{Disjunctive constraint}

Disjunctive arc = pair of directed edges
Solving a conflict means deciding which term in $t_d^j - t_g^i \geq 0$ OR $t_d^i - t_c^j \geq 0$ to satisfy

train $i$ goes first

$T_d^i - T_g^i \geq 0$
Train scheduling problem

- Network $N$, set trains $I$ (with current position) and a wanted timetable $T$.

- $T_s^i$ is the arrival time of train $i$ at station $s$.

**WANT**

- Find a schedule $t^*$ satisfying all fixed and disjunctive precedence constraints.

- Minimize $f(t^*)$ (deviation from $T$)

PS. Fixed route case.
On the objective function $f(t)$

- Typically computed in special events, i.e. the arrival time at some stations $V^* \subset V$
- $f(t) = \sum_{u \in V^*} f_u(t_u)$ is often separable
- Typically $f_u(t_u)$ is non-decreasing.
Disjunctive formulation

\[
\begin{align*}
\min f(t) \\
t_v - t_u &\geq l_{uv} \quad (u,v) \in E \\
t_w - t_v &\geq 0 \text{ OR } t_u - t_z \geq 0 \quad \{(v,w), (z,u)\} \in D \\
t &\in \mathbb{R}^V
\end{align*}
\]

- \( V \) set of events (\( v \in V \) is a certain train entering a certain block or the origin),
  \( E \) set of precedence constraints, \( D \) set of disjunctive precedence constraints

- Train scheduling is a job-shop scheduling problem with blocking and no-wait constraints, Mascis & Pacciarelli (2002)
Disjunctive graph \( G = (V, E \cup E^D) \)

- \( V \) nodes (events), \( E \) directed edges, \( D \) disjunctive arcs (pairs of "conflict" edges \( E^D \))
- Each conflict edge corresponds to a specific term in a specific disjunction
Solving the disjunctive problem

For each disjunction, we must decide which term is satisfied by the solution $t$

- Equivalent to picking exactly one (conflict) edge for each disjunctive arc
- The set of conflict edges "picked" up is called (complete) selection.

$$G = (V, E \cup E^D)$$
Big-M formulation

\[
\begin{align*}
\min f(t) \\
y_{vw} + y_{zu} &= 1 \\
t_w - t_v &\geq -M(1 - y_{vw}) \\
t_u - t_z &\geq -M(1 - y_{zu}) \\
\end{align*}
\]

\[
\begin{align*}
t_v - t_u &\geq l_{uv} \\
t_w - t_v &\geq 0 \text{ OR } t_u - t_z &\geq 0 \\
t &\in \mathbb{R}^V, \ y \in \{0,1\}^{2D} \\
(u,v) &\in E \\
(v,w), (z,u) &\in D
\end{align*}
\]

- Two binary (selection) variables \(y_{vw}, y_{zu}\) for each disjunction \((v, w), (z, u)\) \(\in D\)
- And the "big-M trick"!
Big-M formulation

\[ \min f(t) \]

Fixed precedence

\[ t_v - t_u \geq l_{uv} \quad (u, v) \in E \]

Disjunctive constraints

\[ \begin{align*}
    t_w - t_v & \geq -M(1 - y_{vw}) \\
    t_u - t_z & \geq -M(1 - y_{zu})
\end{align*} \quad \{(v, w), (z, u)\} \in D \]

Selection constraints

\[ y_{vw} + y_{zu} = 1 \]

\[ t \in \mathbb{R}^V, \ y \in \{0,1\}^{2D} \]

- Big-M formulations most used in the literature on train dispatching

- An alternative: time-indexed formulations (often used in train timetabling)

**Def. Feasible selections:**

\[ Y = \{ y \in \{0,1\}^{2D} : y_{vw} + y_{zu} = 1, \{(v, w), (z, u)\} \in D \} \]
**Big-M formulation**

\[
\begin{align*}
\text{Fixed precedence} & \quad \min f(t) \\
& \quad t_v - t_u \geq l_{uv} \quad (u, v) \in E \\
& \quad t_w - t_v \geq -M(1 - y_{vw}) \quad \{(v, w), (z, u)\} \in D \\
& \quad t_u - t_z \geq -M(1 - y_{zu}) \\
& \quad t \in R^V, \ y \in Y
\end{align*}
\]

For a given selection: \( \bar{y} \in Y \) let \( S(\bar{y}) \) be the set of selected terms. The problem becomes:

\[
\min \ \{ f(t) : t_v - t_u \geq l_{uv}, uv \in E \cup S(\bar{y}), t \in R^V \} \quad \text{Sched}(\bar{y})
\]

- Dual of a min-cost flow problem when \( f(t) \) is linear.
Benders’ decomposition(s)
Conflict edges

Each conflict edge \( e \in E^D \) is associated with a selection variable \( y_e \)

\[ y \in Y \text{ is the incidence vector of a set } S(y) \subseteq E^D \text{ of (conflict) edges} \]
What to do with routing?

- Add the alternative *routing edges* $E^R$ and binary (*routing*) variables $y_e, e \in E^R$

- Extend the set $Y$: new variables, multicommodity flow and coupling constraints.
Disjunctive graph and scheduling

- For $\bar{y} \in Y$ the disjunctive graph becomes a standard graph $G(\bar{y}) = (V, E \cup S(\bar{y}))$

- How does $G(\bar{y})$ relate to the associated scheduling problem $\text{Sched}(\bar{y})$?

\[
\text{min } f(t) \\
\text{Sched}(\bar{y}) \\
\quad t_v - t_u \geq l_{uv}, uv \in E \cup S(\bar{y}) \\
\quad t \in R^V
\]
Feasibility

Th. 1. For $\bar{y} \in Y$, $\text{Sched}(\bar{y})$ has a solution, if and only if $G(\bar{y})$ does not contain a directed cycle $C$ of positive length $l(C)$.

$C = \{(lq), (qu), (uv), (vw), (wr), (rl)\}$

$l(C) = 10 + 20 + 15 + 5 + 3 + 3 > 0$
Example of infeasible solution

Current time is 09:00
Train $i$ leaves the station at 9:10 (exactly)
Train $j$ can leave the station at any time from now
Suppose $j$ wins the conflict on $d$

$j$ wins $\rightarrow$ pick edge $(c^j, d^i)$

Cycle $C = \{c^j d^i, d^i o, o d^j, d^j c^j\}$, length $5 > 0$
Feasible solutions

\( \tilde{y} \in Y, G(\tilde{y}) \) no positive directed cycles.

**Th. 2.** \( \bar{t}_u^* = \) length of longest path from \( o \) to \( u \in V \) in \( G(\tilde{y}) \) is feasible for \( Sched(\tilde{y}) \)
Optimal solutions

- \( \bar{y} \in Y, G(\bar{y}) \) no positive dicycles. For \( u \in V, \bar{t}_u^* = \) length of longest \( ou \)-path in \( G(\bar{y}) \)

**Th. 3.** If \( f(t) \) is non-decreasing then \( \bar{t}^* \) is an optimal solution for \( Sched(\bar{y}) \)

- \( t \in R^V \)

**Def.** \( H^*(\bar{y}) \) longest path tree in \( G(\bar{y}) \) then let \( c(H^*) = f(\bar{t}^*) \) be the cost of \( H^* \)
Find $\bar{y} \in Y$, such that $G(\bar{y})$ has no positive directed cycles and the cost $c(H^*)$ of a longest path tree $H^*$ in $G(\bar{y})$ is minimized.
The Path&Cycle formulation

- This led to a new \((Path&Cycle, 2019)\) formulation without annoying big-M constraints (but potentially many constraints)
- Based on disjunctive graph \(G = (V, E \cup E^D)\) \((G = (V, E \cup E^D \cup E^R))\)
- Binary variables \(y_e\) for \(e \in E^D\) \((e \in E^D \cup E^R)\),
- One real variable \(\mu\) representing the objective value.
- Two types of constraints: \textit{feasibility} and \textit{optimality}
- Feasibility constraints correspond to the positive lengths directed cycles of \(G\)
- Optimality constraints correspond to longest path trees of \(G\).
Feasibility constraints

Let $\Omega^+$ be the set of positive directed cycles of $G = (V, E \cup E^D)$

Feasibility constraint

$$\sum_{e \in C^D} y_e \leq |C^D| - 1, \text{ for } C^D = E^D \cap C, C \in \Omega^+$$
Optimality constraints

\[ \eta \geq c(H) \left( \sum_{e \in H^D} y_e - |H^D| - 1 \right), \quad \text{for } H^D = H \cap E^D, H \in \Pi^* \]

\[ G = (V, E \cup E^D) \]

\[ \Pi^* = \{ H^*(y) \text{ longest path tree in } G(y): y \in Y^+ \} \]

\[ Y^+ = \left\{ y \in Y: \sum_{e \in C \cap E^D} y_e \leq |C^D| - 1, \text{ for } C^D = C \cap E^D, C \in \Omega^+ \right\} \]
The Path and Cycle formulation

\[ \min \eta \]

**Feasibility**
\[ \sum_{e \in C^D} y_e \leq |C^D| - 1, \quad \text{for } C^D = E^D \cap C, C \in \Omega^+, \]

**Optimality**
\[ \eta \geq c(H)(\sum_{e \in H^D} y_e - |H^D| - 1), \quad \text{for } H^D = H \cap E^D, H \in \Pi^* \]
\[ \eta \in R, \quad y \in \{0,1\}^{E^D} \]

Many constraints: solve by delayed row generation

Problem infeasible → there exists a family \(\Omega \subseteq \Omega^+\) of positive directed cycles
\(G = (V, E \cup E^D)\) such every \(y \in Y\) «contains» at least a cycle in \(\Omega\):

For \(y \in Y\), \(\exists C^y \in \Omega\) such that \(S(y) \cap C^y = E^D \cap C^y\)
An example

Current time is 09:00
Train $i$ must leave at 9:10
Train $j$ must leave at 9:20

$G = (V, E \cup E^D)$

Disjunctive graph representation
Infeasibility proof

Problem infeasible: every \( y \in Y \) «contains» a cycle in \( \Omega = \{ C^1, C^2 \} \):

\[ S(y) \cap C^y = E^D \cap C^y, \text{for some } C^y \in \Omega \]

\[ G = (V, E \cup E^D) \]

\[ \bar{\Omega} = \{ C^1, C^2 \} \]

\( i \) wins

\( j \) wins
A real-life pilot application
Greater Oslo Area Railway

- We can solve the Big-M or P&C formulations for very small instances.
- Greater Oslo Area Railway is a combination of one large station (Oslo S) and 10 municipal lines incident to Oslo S.
- Almost 1000 trains daily.
- Need: more decomposition/reformulation.
Further decomposition

- One popular decomposition approach is the so-called *Macroscopic/Microscopic* decomposition.

- Subnetworks (as stations) are collapsed into "capacited" nodes.

- A solution is found for the collapsed (macroscopic) representation

- The solution is then extended to the original re-expanded (microscopic) areas

(Figure from Hansen and Pachl, *Railway Timetable & Operations*)
Collapsing Greater Oslo Railway

- Macroscopic solution = arrival and departure time for each train in each station (timetable)
- Can we extend the macro solution? = For each station, is the timetable feasible?
Macroscopic representation of train routes

Microscopic representation

Macroscopic representation
Block structure of constraint matrix

\[
\begin{align*}
\min f(t^T) \\
A t^L + \begin{bmatrix} B t^T \end{bmatrix} + 0 & \leq b - M^L y^L & \text{schedule on the line = macro network} \\
0 + \begin{bmatrix} C t^T \end{bmatrix} + D t^S & \leq d - M^S y^S & \text{schedule in stations = micro network(s)}
\end{align*}
\]

\[y^L, y^S \text{ binary, } t^L, t^T, t^S \text{ real}\]

- Constraint matrix with quasi-block structure
- Station tracks and line tracks share only timetable variables \(t^T\)
- The objective function is only in \(t^T\)
- Station constraints decompose

\[
\begin{pmatrix}
C^1, D^1 \\
C^2, D^2 \\
C^3, D^3 \\
\vdots
\end{pmatrix}
\begin{pmatrix}
M^1 \\
M^2 \\
M^3 \\
\vdots
\end{pmatrix}
\]
Logic Benders' Reformulation

$$\min f(t^T)$$

<table>
<thead>
<tr>
<th>MASTER</th>
<th>SLAVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$At^L + Bt^T \leq b - M^L y^L$ schedule on the line</td>
<td>$C^T t^T + d^S \leq d - M^S y^S$ schedule in stations</td>
</tr>
</tbody>
</table>

$y^L, y^S$ binary, $t^L, t^T, t^S$ real

Reformulation

$$\min f(t^T)$$

<table>
<thead>
<tr>
<th>MASTER</th>
<th>SLAVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$At^L + Bt^T \leq b - M^L y^L$ schedule on the line</td>
<td>$C^T t^T \leq d' - M^S y^S$ logic Benders' cuts</td>
</tr>
</tbody>
</table>

$y = (y^L, y^S)$ binary, $t^L, t^T$ real
Solving the Train Scheduling Problem

- Apply row generation

Diagram:
- Macro Problem: Line scheduling
- Micro Problem(s): Station scheduling

Solve the current restricted master

Solve slave(s)

Add $C^q t^T \leq d^q - M^q y^S$

Feasible?

NO

Optimal solution $(t^L, t^T, t^S, y^L, y^S)$
The slave feasibility problem

- The slave problem decomposes in many independent feasibility problem

Station feasibility problem:

A. Given a station and arrival and departure times for all trains (a timetable), does a feasible solution (in the station) exist?

B. If the problem is infeasible, what are the constraints to return to the master?

- We exploit the feasibility conditions of the P&C formulation
Individual station problem

- Station problem: given arrival times $T^A_1, T^A_2, ...$, departure times $T^D_1, T^D_2, ...$, does there exist a feasible solution?

$$G = (V, E \cup E^D \cup E^R)$$ disjunctive graph representing problem instance

$$Y = \{y^1, y^2, \ldots \}$$ set of (incident vectors of) edge selections

Station problem infeasible:

$G$ contains a family $\bar{\Omega} = \{C^1, C^2, \ldots \}$ of positive lengths cycles such that every selection $y \in Y$ "contains" a cycle, i.e.

$$S(y) \cap C^i = E^D \cap C^i, \quad \text{for some } C^i \in \bar{\Omega}$$
Combinatorial Benders' cuts

\( \Omega = \{ C_1, C_2, \ldots \} \). Suppose \( C \in \Omega \) contains a timetable edge.

Then \( C \) contains the origin \( o \) and exactly two timetable edges.

\[ l(C) > 0 \rightarrow T^i_A - T^j_A + \delta > 0 \rightarrow T^j_A - T^i_A < \delta \]

To prevent \( l(C) > 0 \) a timetable must satisfy

\[ t^j_A - t^i_A \geq \delta \]
Combinatorial Benders' cuts

\( \bar{\Omega} \): every selection \( y \in Y \) "contains" a cycle in \( \bar{\Omega} \)

\( \Omega^T \subseteq \bar{\Omega} \) subset of "timetable" cycles of \( \bar{\Omega} \)

For \( C_i \in \Omega^T \), \( t_i^-, t_i^+ \) time variable associated with (the other endpoint of) the non-positive edge and non-negative edge,

Then, for any feasible timetable \( t \), we must have:

\[
 t_1^- - t_1^+ \geq \delta_1 \quad \text{OR} \quad t_2^- - t_2^+ \geq \delta_2 \quad \text{OR} \ldots
\]

Again, a disjunction of time precedence constraints!
The full reformulation

\[ \min f(t) \]
\[ t_v - t_u \geq l_{uv} \quad (u, v) \in E \]

other line constraints ...

\[ \bigvee_{c_i \in \Omega^k} t_i^- - t_i^+ \geq \delta_i \]
\[ \Omega^k \in \Lambda \]
\[ t \in R^V \]

- \( \Lambda \) is the set of all families of timetable cycles (for all stations!)
- Disjunctions can be linearized by introducing binary variables and big \( M \)s.
Dispatching system in Oslo

- We developed a real-time scheduling system for dispatching trains in Oslo Greater Oslo Region
References


