Models in Transportation

Tim Nieberg
Transportation Models

- large variety of models due to the many modes of transportation
  - roads
  - railroad
  - shipping
  - airlines
- as a consequence different type of equipment and resources with different characteristics are involved
  - cars, trucks, roads
  - trains, tracks and stations
  - ships and ports
  - planes and airports
- consider two specific problems
Basic Characteristics

- consider the problem from the view of a company
- the planning process normally is done in a 'rolling horizon' fashion
- company operates a fleet of ships consisting of
  - own ships \(\{1, \ldots, T\}\)
  - chartered ships
- the operating costs of these two types are different
- only the own ships are scheduled
- using chartered ships only leads to costs and these costs are given by the spot market
Basic Characteristics (cont.)

- each own ship $i$ is characterized by its
  - capacity $cap_i$
  - draught $dr_i$
  - range of possible speeds
  - location $l_i$ and time $r_i$ at which it is ready to start next trip
  - ...
Basic Characteristics (cont.)

- the company has \( n \) cargos to be transported
- cargo \( j \) is characterized by
  - type \( t_j \) (e.g. crude type)
  - quantity \( p_j \)
  - load port \( port_j^l \) and delivery port \( port_j^d \)
  - time windows \([r_j^l, d_j^l]\) and \([r_j^d, d_j^d]\) for loading and delivery
  - load and unload times \( t_j^l \) and \( t_j^d \)
  - costs \( c_j^* \) denoting the price which has to be paid on the spot market to transport cargo \( j \) (estimate)
Basic Characteristics (cont.)

- there are \( p \) different ports
- port \( k \) is characterized by
  - its location
  - limitations on the physical characteristics (e.g. length, draught, deadweight, \ldots) of the ships which may enter the port
  - local government rules (e.g. in Nigeria a ship has to be loaded above 90\% to be allowed to sail)
- \ldots
Basic Characteristics (cont.)

- the objective is to minimize the total cost of transporting all cargos
- hereby a cargo can be assigned to a ship of the company or ‘sold’ on the spot market and thus be transported by a chartered ship
- costs consist of
  - operating costs for own ships
  - spot charter rates
  - fuel costs
  - port charges, which depend on the deadweight of the ship
ILP modeling

- straightforward choice of variables would be to use 0–1-variables for assigning cargos to ships
- problem: these assignment variables do not define the schedule/route for the ship and thus feasibility and costs of the assignment can not be determined
- alternative approach: generate a set of possible schedules/routes for each ship and afterwards use assignment variables to assign schedules/routes to ships
- problem splits up into two subproblems:
  - generate schedules for ships
  - assign schedules to ships
Tanker Scheduling

ILP modeling - generate schedules

- a schedule for a ship consist of an assignment of cargos to the ship and a sequence in which the corresponding ports are visited
- generation of schedules can be done by ad-hoc heuristics which consider
  - ship constraints like capacity, speed, availability, …
  - port constraints
  - time windows of cargos
- each schedule leads to a certain cost
- for each ship enough potential schedules should be generated in order to get feasible and good solutions for the second subproblem
ILP modeling - generate schedules (cont.)

- the output of the first subproblem is
  - a set $S_i$ of possible schedules for ship $i$
  - each schedule $l \in S_i$ is characterized by
    - a vector $(a_{i1}^l, \ldots, a_{in}^l)$ where $a_{ij}^l = 1$ if cargo $j$ is transported by ship $i$ in schedule $l$ and 0 otherwise
    - costs $c_i^l$ denoting the incremental costs of operating ship $i$ under schedule $l$ versus keeping it idle over the planning horizon
    - profit $\pi_i^l = \sum_{j=1}^n a_{ij}^l c_j^* - c_i^l$ by using schedule $l$ for ship $i$ instead of paying the spot market
ILP modeling - generate schedules (cont.)

Remarks:
- all the feasibility constraints of the ports and ships are now within the schedule
- all cost aspects are summarized in the values $c_i$ resp. $\pi_i$
- the sequences belonging to the schedules determine feasibility and the costs $c_i$ but are not part of the output since they are not needed in the second subproblem
ILP modeling - assign schedules to ships

- variables $x_i^l = \begin{cases} 
1 & \text{if ship } i \text{ follows schedule } l \\
0 & \text{else} 
\end{cases}$

- objective: $\max \sum_{i=1}^{T} \sum_{l \in S_i} \pi_i^l x_i^l$

- constraint:
  \begin{align*}
  \sum_{i=1}^{T} \sum_{l \in S_i} a_{ij}^l x_i^l & \leq 1; \quad j = 1, \ldots, n \text{ (each cargo at most once)} \\
  \sum_{l \in S_i} x_i^l & \leq 1; \quad i = 1, \ldots, T \text{ (each ship at most one schedule)}
  \end{align*}
the ILP model is a set-packing problem and well studied in the literature

- can be solved by branch and bound procedures

- possible branchings:
  - chose a variable $x_i^l$ and branch on the two possibilities $x_i^l = 0$ and $x_i^l = 1$
    - select $x_i^l$ on base of the solution of the LP-relaxation: choose a variable with value close to 0.5
  - chose a ship $i$ and branch on the possible schedules $l \in S_i$
    - selection of ship $i$ is e.g. be done using the LP-relaxation: choose a ship with a highly fractional solution
ILP modeling - assign schedules to ships (cont.)

- lower bounds can be achieved by generating feasible solutions via clever heuristics (feasible solution = lower bound since we have a maximization problem)

- upper bounds can be obtained via relaxing the integrality constraints and solving the resulting LP (note, that this LP-solution is also used for branching!)

- for a small example, the behavior of the branch and bound method is given in the handouts
Remarks Two Phase Approach

- In general, the solution after solving the two subproblems is only a heuristic solution of the overall problem.
- If in the first subproblem all possible schedules/routes for each ship are generated (i.e., $S_i$ is equal to the set $S_i^{all}$ of all feasible schedules for ship $i$), the optimal solution of the second subproblem is an optimal solution for the overall problem.
- For real-life instances, the cardinalities of the sets $S_i^{all}$ are too large to allow a complete generation (i.e., $S_i$ is always a (small) subset of $S_i^{all}$).
- Column generation can be used to improve the overall quality of the resulting solution.
General Remarks

- in the railway world lots of scheduling problems are of importance
  - scheduling trains in a timetable
  - routing of material
  - staff planning
  - ...

- currently lots of subproblems are investigated

- the goal is to achieve an overall decision support system for the whole planning process

- we consider one important subproblem
Decomposition of the Train Timetabling

- mostly the overall railway network consists of some major stations and 'lines/corridors' connecting them

![Diagram of train stations]

- a corridor normally consists of two independent one-way tracks
- having good timetables for the trains in the corridors makes it often easy to find timetables for the trains on the other lines

DH: Den Haag Centraal
As: Amsterdam Centraal
Am: Amersfoort
R: Rotterdam Centraal
U: Utrecht Centraal
Train Timetabling

Scheduling Train on a Track

- consider a track between two major stations
- in between the two major stations several smaller stations exists

![Diagram of a train track with stations labeled: R, RN, RA, CS, NI, G, GG, W, U.]

- trains may or may not stop at these stations
- trains can only overtake each other at stations
Problem Definition Track Scheduling

- time period $1, \ldots, q$, where $q$ is the length of the planning period (typically measured in minutes; e.g. $q = 1440$)
- $L + 1$ stations $0, \ldots, L$
- $L$ consecutive links;
- link $j$ connects station $j-1$ and $j$
- trains travel in the direction from station 0 to $L$
- $T$: set of trains that are candidates to run during planning period
- for link $j$, $T_j \subseteq T$ denotes the trains passing the link
Problem Definition Track Scheduling (cont.)

- train schedules are depicted in so-called time-space diagrams

- diagrams enable user to see conflicts
Problem Definition Track Scheduling (cont.)

- Train schedules are depicted in so-called time-space diagrams.

- Diagrams enable users to see conflicts.
Problem Definition Track Scheduling (cont.)

- each train has an most desirable timetable (arrivals, departures, travel time on links, stopping time at stations), achieved e.g. via marketing department
- putting all these most desirable timetables together, surely will lead to conflicts on the track
- possibilities to change a timetable:
  - slow down train on link
  - increase stopping time at a station
  - modify departure time at first station
  - cancel the train
Problem Definition Track Scheduling (cont.)

- cost of deviating from a given time $\hat{t}$:
  - specifies the revenue loss due to a deviation from $\hat{t}$
  - the cost function has its minimum in $\hat{t}$, is convex, and often modeled by a piecewise linear function

- piecewise linear helps in ILP models!
Variables for Track Scheduling

- Variables represent departure and arrival times from stations
  - $y_{ij}$: time train $i$ enters link $j$
    - $= \text{time train } i \text{ departs from station } j - 1$
  - $z_{ij}$: time train $i$ leaves link $j$
    - $= \text{time train } i \text{ arrives at station } j$

- $c_{ij}^d(y_{ij}) \ (c_{ij}^a(z_{ij}))$ denotes the cost resulting from the deviation of the departure time $y_{ij}$ (arrival time $z_{ij}$) from its most desirable value
Variables for Track Scheduling (cont.)

- variables resulting from the departures and arrivals times:
  - $\tau_{ij} = z_{ij} - y_{ij}$: travel time of train $i$ on link $j$
  - $\delta_{ij} = y_{i,j+1} - z_{ij}$: stopping time of train $i$ at station $j$

- $c^\tau_{ij}(\tau_{ij})$ ($c^\delta_{ij}(\delta_{ij})$) denotes the cost resulting from the deviation of the travel time $\tau_{ij}$ (stopping time $\delta_{ij}$) from its most desirable value

- all cost functions $c^d_{ij}, c^a_{ij}, c^\tau_{ij}, c^\delta_{ij}$ have the mentioned structure
Train Timetabling

Objective function

\[
\text{minimize } \sum_{j=1}^{L} \sum_{i \in T_j} (c_{ij}^d(y_{ij}) + c_{ij}^a(z_{ij}) + c_{ij}^T(z_{ij} - y_{ij})) + \sum_{j=1}^{L-1} \sum_{i \in T_j} c_{ij}^\delta(y_{i,j+1} - z_{ij})
\]
Constraints

- minimum travel times for train $i$ over link $j$: $\tau_{ij}^{min}$
- minimum stopping times for train $i$ at station $j$: $\delta_{ij}^{min}$
- safety distance:
  - minimum headway between departure times of train $h$ and train $i$ from station $j$: $H_{hij}^d$
  - minimum headway between arrival times of train $h$ and train $i$ at station $j$: $H_{hij}^a$
- lower and upper bounds on departure and arrival times:
  $y_{ij}^{min}$, $y_{ij}^{max}$, $z_{ij}^{min}$, $z_{ij}^{max}$
Constraints (cont.)

- To be able to model the minimum headway constraints, variables have to be introduced which control the order of the trains on the links

\[ x_{hij} = \begin{cases} 
1 & \text{if train } h \text{ immediately precedes train } i \text{ on link } j \\
0 & \text{else} 
\end{cases} \]

- Using the variables \( x_{hij} \), the minimum headway constraints can be formulated via 'big M'-constraints:

\[ y_{i,j+1} - y_{h,j+1} + (1 - x_{hij})M \geq H_{hij}^d \]

\[ z_{ij} - z_{hj} + (1 - x_{hij})M \geq H_{hij}^a \]
Constraints (cont.)

- two dummy trains 0 and * are added, representing the start and end of the planning period (fix departure and arrival times appropriate ensuring that 0 is sequenced before all other trains and * after all other trains)
Constraints (cont.)

\[
\begin{align*}
y_{ij} & \geq y_{ij}^{\min} & j = 1, \ldots, L; \ i \in T_j \\
y_{ij} & \leq y_{ij}^{\max} & j = 1, \ldots, L; \ i \in T_j \\
z_{ij} & \geq z_{ij}^{\min} & j = 1, \ldots, L; \ i \in T_j \\
z_{ij} & \leq z_{ij}^{\max} & j = 1, \ldots, L; \ i \in T_j \\
z_{ij} - y_{ij} & \geq \tau_{ij}^{\min} & j = 1, \ldots, L; \ i \in T_j \\
y_{i,j+1} - z_{ij} & \geq \delta_{ij}^{\min} & j = 1, \ldots, L - 1; \ i \in T_j \\
y_{i,j+1} - y_{h,j+1} + (1 - x_{hij})M & \geq H_{hij}^d & j = 0, \ldots, L - 1; \ i, h \in T_j \\
z_{ij} - z_{hj} + (1 - x_{hij})M & \geq H_{hij}^a & j = 1, \ldots, L; \ i, h \in T_j \\
\sum_{h \in T_j \setminus \{i\}} x_{hij} & = 1 & j = 1, \ldots, L; \ i \in T_j \\
\sum_{i \in T_j \setminus \{h\}} x_{hij} & = 1 & j = 1, \ldots, L; \ h \in T_j \\
x_{hij} & \in \{0, 1\} & j = 1, \ldots, L; \ i, h \in T_j
\end{align*}
\]
Remarks on ILP Model

- the number of 0-1 variables gets already for moderate instances quite large
- the single track problem is only a subproblem in the whole time tabling process and needs therefore to be solved often
- as a consequence, the computational time for solving the single track problem must be small
- this asks for heuristic approaches to solve the single track problem
Train Timetabling

Decomposition Approach: General Idea

- schedule the trains iteratively one by one
- initially, the two dummy trains 0 and * are scheduled
- the selection of the next train to be scheduled is done on base of priorities
- possible priorities are
  - earliest desired departure time
  - decreasing order of importance (importance may be e.g. measured by train type, speed, expected revenue, ...)
  - smallest flexibility in departure and arrival
  - combinations of the above
Decomposition Approach: Realization

- $T_0$: set of already scheduled trains
- initially $T_0 = \{0, \ast\}$
- after each iteration a schedule of the trains from $T_0$ is given
- however, for the next iteration only the sequence in which the trains from $T_0$ traverse the links is taken into account
- $S_j = (0 = j_0, j_1, \ldots, j_{n_j}, j_{n_j+1} = \ast)$: sequence of trains from $T_0$ on link $j$
- if train $k$ is chosen to be scheduled in an iteration, we have to insert $k$ in all sequences $S_j$ where $k \in T_j$
- this problem is called $\text{Insert}(k, T_0)$
ILP Formulation of Insert$(k, T_0)$

Adapt the 'standard' constraints and the objective to $T_0$: 

$$
\begin{align*}
\min & \sum_{j=1}^{L} \sum_{i \in T_j} (c_{ij}^d(y_{ij}) + c_{ij}^a(z_{ij}) + c_{ij}^\tau(z_{ij} - y_{ij})) \\
+ & \sum_{j=1}^{L_1} \sum_{i \in T_j} c_{ij}^\delta(y_{i,j+1} - z_{ij}) \\
\text{subject to} & \\
& y_{ij} \geq y_{ij}^{\min} \quad j = 1, \ldots, L; \quad i \in T_0 \cap T_j \\
& y_{ij} \leq y_{ij}^{\max} \quad j = 1, \ldots, L; \quad i \in T_0 \cap T_j \\
& z_{ij} \geq z_{ij}^{\min} \quad j = 1, \ldots, L; \quad i \in T_0 \cap T_j \\
& z_{ij} \leq z_{ij}^{\max} \quad j = 1, \ldots, L; \quad i \in T_0 \cap T_j \\
& z_{ij} - y_{ij} \geq \tau_{ij}^{\min} \quad j = 1, \ldots, L; \quad i \in T_0 \cap T_j \\
& y_{i,j+1} - z_{ij} \geq \delta_{ij}^{\min} \quad j = 1, \ldots, L - 1; \quad i \in T_0 \cap T_j 
\end{align*}
$$
ILP Formulation of $\text{Insert}(k, T_0)$ (cont.)

- adapt $y_{i,j+1} - y_{h,j+1} + (1 - x_{hij})M \geq H_{hij}^d$ for trains from $T_0$

  $$y_{j+1} - y_{j} \geq H_{j,j+1,j-1}^d$$ for $j = 1, \ldots, L, i = 0, \ldots, n_j$

- adapt $z_{ij} - z_{hj} + (1 - x_{hij})M \geq H_{hij}^a$ for trains from $T_0$

  $$z_{j+1} - z_{j} \geq H_{j,j+1,j}^a$$ for $j = 1, \ldots, L, i = 0, \ldots, n_j$
ILP Formulation of $\text{Insert}(k, T_0)$ (cont.)

- insert $k$ on link $j$ via variables

$$x_{ij} = \begin{cases} 1 & \text{if train } k \text{ immediately precedes train } j_i \text{ on link } j \\ 0 & \text{else} \end{cases}$$

- new constraints for $j = 1, \ldots, L, i = 0, \ldots, n_j$:
  - $y_{k,j} - y_{j_i,j} + (1 - x_{ij})M \geq H_{j_i,kj}^d$
  - $y_{j_i+1,j} - y_{k,j} + (1 - x_{ij})M \geq H_{k_{j_i+1},j}^d$
  - $z_{k,j} - z_{j_i,j} + (1 - x_{ij})M \geq H_{j_i,kj}^a$
  - $z_{j_{i+1},j} - z_{k,j} + (1 - x_{ij})M \geq H_{k_{j_{i+1}},j}^a$

- 0-1 constraints and sum constraint on $x_{ij}$ values
Remarks on ILP Formulation of $\text{Insert}(k, T_0)$

- the ILP Formulation of $\text{Insert}(k, T_0)$ has the same order of continuous constraints $(y_{ij}, z_{ij})$ but far fewer 0-1 variables than the original MIP.
- A preprocessing may help to fix $x_{ij}$ variables since on base of the lower and upper bound on the departure and arrival times of train $k$ many options may be impossible.
- Solving $\text{Insert}(k, T_0)$ may be done by branch and bound.
Solving the overall problem

- an heuristic for the overall problem may follow the ideas of the shifting bottleneck heuristic
  - select a new train $k$ (machine) which is most 'urgent'
  - solve for this new train $k$ the problem $Insert(k, T_0)$
  - reoptimize the resulting schedule by rescheduling the trains from $T_0$

- rescheduling of a train $l \in T_0$ can be done by solving the problem $Insert(l, T_0 \cup \{k\} \setminus \{l\})$ using the schedule which results from deleting train $l$ from the schedule achieved by $Insert(k, T_0)$