

Solving the World's Largest Crew Scheduling Problem

Niklas Kohl, Ph.D.
Senior Consultant
Carmen Consulting Denmark
niklas.kohl@carmenconsulting.com

Airline crew scheduling has for decades been a major industrial application area of mathematical programming and operations research. The goal of crew scheduling is to assign the tasks to be carried out (the flights) to named individuals such that rules and regulations are respected and costs minimized.

The problem naturally decomposes into the *crew pairing problem*, in which trips, i.e. sequences of flights, starting and ending at a crew base are constructed and the *crew rostering problem*, where the anonymous trips are assigned to named individuals. In this article we limit the scope to the crew pairing problem.

Virtually all major airlines and most midsize airlines use optimization software as part of their crew pairing process. More recently railways have started to use optimization. There are a lot of similarities between the crew pairing problems faced at airlines and railways, but also important differences.

The purpose of this article is to give some insight into the particularities of the railway crew pairing problem and to briefly outline how we at Carmen have chosen to solve some of the difficulties. In particular we will discuss the crew pairing problem at Europe's largest railway Deutsche Bahn, the German state railways.

About Carmen

Carmen Systems is the leading developer of crew scheduling software for the transportation industry. The

crew pairing system discussed in this article is in use at all major European airlines as well as Singapore Airlines, Northwest Airlines and other major international airlines. In addition to Deutsche Bahn, railway customers include the Swedish State Railways and the freight operator Green Cargo.

About Deutsche Bahn

Deutsche Bahn (DB) is the state owned railway company in Germany and one of the worlds largest transportation companies. DB Regio, consists of many partly independent companies, which operate regional and commuter traffic. DB Fernverkehr operates long haul passenger traffic. Total crew (train drivers and conductors) number around 30,000. The DB Fernverkehr and the DB Regio timetable contains roughly 30,000 trains per day and more than 100 crew bases are in use. A train is operated by one train driver and from zero to seven conductors depending on the type of train, the expected number of passengers as well a number of other factors.



Mathematical Model of the Crew Pairing Problem

In its simplest form the crew pairing problem can be described as follows; The timetable specifies a number of *atomic* tasks which must be staffed by a crew. In case of a railway an atomic task consist of staffing the train between two stations where crew change can take place. Following the

airline terminology these tasks are denoted *legs*. A *trip* is a sequence of legs, which define a possible working pattern for a crew. It must start and end at a crew base, must respect legal and contractual rules regarding work and rest time as well as operational rules ensuring that the trip can be operated in practice. The cost of a trip is the crew cost associated with operating the trip as well as other direct costs such as hotel costs and costs for passive transports (for example taxis). Mathematically, the crew pairing problem formulates as a set partitioning problem

$$\begin{array}{ll} \text{Minimize} & \sum_t C_t X_t \\ \text{Subject to} & \sum_t A_{lt} X_t = 1, \forall l \in L \\ & X_t \in \{0,1\}, \forall t \in T \end{array}$$

Where the sets T and L are the sets of possible trips and legs, respectively. C_t is the cost of trip t and the coefficient A_{lt} is 1 if trip t contain leg l and 0 otherwise. The binary decision variable X_t is 1 if trip t is part of the optimal solution and 0 otherwise. The constraints set expresses, that each leg should be contained in exactly one trip. Numerous extensions and modifications to this model have been proposed, see e.g. Andersson et al. (1998). Most importantly, the partitioning constraints (=) are usually replaced with covering constraints (\geq), and base constraints, which specify the distribution of work time between crew bases, are added.

Though simple, this model suffers from two drawbacks. The set partitioning problem is NP-hard and even worse – the set T is extremely large for virtually all problems of reasonable size.

Differences between the Airline and the Railway Crew Pairing Problem

The vast majority of academic and industrial research on the crew pairing problem has focused on the problem faced by airlines. In the airline problem

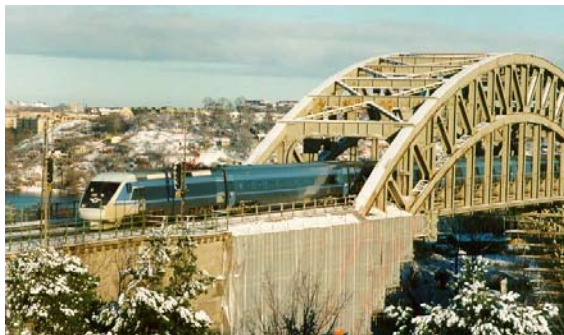
the set L is rather small – hundreds or a few thousands of legs. On the contrary the long haul traffic of DB, i.e. DB Fernverkehr, consists of more than 1000 trains per day and many trains correspond to more than 10 legs since it is possible to change crew at most major stations. Several conductor positions must be staffed on a train, and since preparation times and qualification requirements depend on the position, each position must be considered separately. However, a crew member can change position between legs within a trip, so all positions must be considered as one problem. Hence, the set L for a weekly pairing problem is more than 100,000.

Further, the number of 1-coefficients per variable is higher than in the airline case. An airline trip will typically contain 2-4 legs per day and rarely have a duration of more than 4 days. A railway trip is never more than two days long, but since the average duration of a leg is only 30 minutes for the long haul traffic and 20 minutes for the regional traffic, pairings with 20 or more legs are not uncommon.

It is known that the average case difficulty of a set partitioning problem increase with the average number of 1-coefficients per variable, so the railway set partitioning problems are not only more difficult than their airline counterparts due to their size, but also due to their inherently complex structure.

It is not only the mathematical structure of the problem, which is more complex for railways than for airlines. The calculation of legality and cost of a trip is substantially more complex at Deutsche Bahn. There are several reasons for this. Most importantly train crew carries out a number of duties in addition to driving trains. In particular train drivers have a lot of preparation tasks, which must be derived from the locomotive and wagon rotations.

Rest calculations are also complicated by the possibility of taking a rest while the train is running - obviously this is only possible for conductors! Such rest will be paid, but will extend the possible working time. Normal rest at a station is generally unpaid if the rest is required by the working time rules, but will be paid if it is not required. Since rest occurs naturally in connection with changing trains, a trip with more work can actually produce less pay than a trip with less work because the rest will be unpaid in the first case.



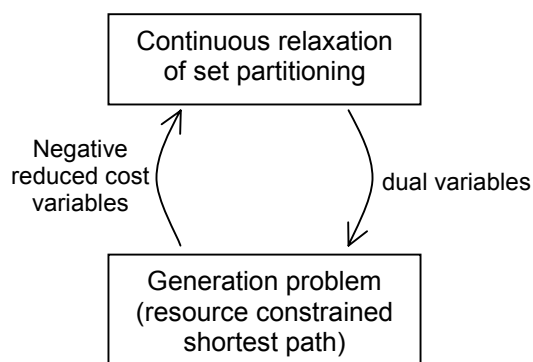
A further complication is the amount and quality of data. The time table is huge. Wagon and locomotive rotations are needed to calculate preparation and termination activities as well as to calculate connection times. And these data must be consistent with the time table. In some cases infrastructure data is needed as well. For a locomotive driver it can for example be important which route the train takes between station A and B, since each route will require a specific qualification profile.

How to Generate the Variables

As previously stated, the number of possible trips will be enormous for a large crew pairing problem. Early crew pairing algorithms tried to reduce the number of trips considered by heuristically generating some thousands of trips, solving the set partitioning problem, updating the solution and repeating generation. One implementation of this idea is to randomly select a small number of trips and generate all possible trips,

which “cover” a subset of legs covered by the legs “covered” by the selected trips. This can be repeated as long as time permits. The number of trips selected in each iteration can be chosen to ensure that the set partition problems remain solvable. There are many other ways to limit the number of generated trips.

More recent work on the crew pairing problem has focused on the column generation (or Dantzig-Wolfe) approach. In this approach the generation problem is to find the variable (trip) which will minimize the reduced cost with respect to the solution to the continuous relaxation of the set partitioning problem defined on the currently known variables. In theory, this can be repeated until no variable with negative reduced cost exist and at this point the continuous relaxation of the set partitioning problem, defined on all trips, has been solved without explicitly generating all possible trips. However, two problems remain. The solution obtained is generally not integer and – in practice much worse – it is not trivial to find the variable which minimize the reduced cost. In fact the latter problem turns out to be a resource constrained shortest path problem.



In many cases dynamic programming can solve the resource constrained shortest path problem. Two different methods are in use. Either multidimensional state variables are used, where each dimension represent a resource such as work time, or a k -shortest path algorithm is used. In the

later case the k shortest paths are generated and those paths which corresponds to resource infeasible trips are discarded, see Hjorring and Hansen for a further discussion of this idea. Recent work by Carmen focus on integrating these two approaches.

Technologies Used

Due to the size of the integer programs to be solved, it is not possible to obtain acceptable solution times with commercial integer programming solvers such as ILOG CPLEX or XPRESS-MP. Instead we use the proprietary PAQS optimizer developed in cooperation with Chalmers University of Technology. PAQS has been designed to yield near-optimal solutions to huge set covering type problems and has consistently outperformed the standard commercial optimizers on this class of problems. An early reference on PAQS is Wedelin (1995)

Another important technology used is the Rule And Value Evaluator (RAVE). RAVE is a programming language used to define legality, costs and other objectives. Compared to a normal programming language such as C++ or Java, RAVE programming is simple because it has been developed to reason about sequences of activities and because the programmer only need to state the definitions and rules, but do not need to consider how the rules will be tested during the optimization.

Results Obtained

Compared to current, manually constructed solutions there are optimized solutions with a significant lower crew need. In addition to this some more "soft" quality aspects have been improved and the inevitable rule violations associated with manual planning have been eliminated.

References

Erik Andersson, Efthymios Housos, Niklas Kohl, and Dag Wedelin. *Crew Pairing Optimization*. In G. Yu editor, *Operations Research in the Airline Industry*, Kluwer Academic Publishers, 1998.

Curt A. Hjorring and Jesper Hansen. *Column Generation with a Rule Modelling Language for Airline Crew Pairing*. In Proceedings of the 34th Annual Conference of the Operational Research Society of New Zealand, December 10-11, 1999, Hamilton, New Zealand. This paper is available from the URL below.

Dag Wedelin. *An algorithm for large scale 0-1 integer optimization*. *Annals of Operations Research* 57. 1995.

A lot of papers on crew scheduling, authored or co-authored by Carmen employees, can be found at

http://www.carmen.se/research_development/research_reports.htm