A Dispatching Tool for Railway Transportation

Pascal Rebreyend

Computer engineering department, Dalarna University
Röda vägen 3, Borlänge, Sweden prb@du.se http://www.du.se/~prb

1 Introduction

Railway transportations are more and more used in many countries for many reasons (environment, costs,...). Like public and freight transportations, reliability (regarding timetables) is a crucial issue. With an increasing traffic, delays occurs more often. To minimize the effects of delays is the main goal of the dispatcher.

2 Problem description

Almost all trains have to run according a schedule. One exception may be some freight trains which may start at a random time. Unfortunatly, for some reasons, some minor problems occur and lead to have delays. Due to limited ressources available (tracks, cars,...), a small problem can lead to a chain of delays. To monitor the traffic and take decisions are the main goals of the train dispatcher. This person decides which train can run and on which track.

An example is the single-track problem. In this case, the schedule is built in order to let trains going in opposite directions to meet in some specific places (stations, yards,...). If one train is delayed, the dispatcher should decide how to organize the crossing. Basically, in most cases, he has to decide between to keep the crossing at the same location (by delaying the other train) or to move the crossing to another place (but may increase the delay of the already delayed train). The decision is difficult since it may introduce other conflicts or delays later.

Similar problems occur in general networks (dual-tracks,...) where the dispatcher has to choose a compromise. Since improving existing infrastucture is expensive and the traffic is increasing, the work of the dispatcher become more and more complex and difficult. Another factor is the wider range of trains used nowadays: suburban trains, freight trains, long-distance passengers, fast trains,... Another complex problem to handle is the connections between trains.

3 Simulation tool

Researchers on this dispatching problem need to have a simulation tool (and a benchmark too) to analyze different methods.

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 478–480. ISBN 80-210-3726-1.

3.1 Extended tool

This tool, called **Distrain**, is an extended tool since its main carateristic is to be open and flexible. This tool is also targeting the general problem, not only single-tracks or specific problems [1]. This openess and flexibility is useful to deal with complex problems and to handle all constraints. An example of constraint is the need to take into account pricing, i.e to charge each train with a price depending of its physical caracteristics and which level of priority is used. Therefore, the tool should be able to check for a future train which time will be required, and the potential effect of this train on the future traffic to fix the price.

This tool intends to *help* the dispatcher to choose a solution. Therefore, to be able to propose different possibilities is an interested feature. This can be easily achieved by Genetic Algorithms since in such method there is a population of solutions. Another interesting research issue is to let the software to learn which "kind" of solution is preferred by the dispatcher.

3.2 The model

To achieve this goal, data are represented using XML in different configuration files (networks, timetable, trains,. . .) [2]. The choice of the model is critical since it is used to represent data.

Data are decomposed into different parts. One part is the *network*. In this part we represent the fixed infrastucture used (tracks, stations, yards, switches,. . .) with all information needed to be able to simulate the network. The second part of the representation is dedicated to technical specifications of different trains. In order to run a simulation, technical caracteristics of the different trains need to be known. By technical caracteristics, we mean information like weigth, length, power, maximum speed, ... These informations are needed to compute traveling time (especially time needed to accelerate) as well to know if a train can stop in an upphill section. The third part is the *timetable* where precomputed timetables are stored. Then, we need to represent all specific constraints between trains, or between a train and another mean of transportations since inter-modal transportations are more and more popular. Dependancies between trains can be called "one-way", when a train cannot start from a station before the arrival of another one (plus the delay to let passengers to walk), or "two-ways" when two trains needs to exchange passengers and therefore stay together at the station. It's worth to mention that depedancies can either be with a single or a list of trains (usual case) or with a destination if the frequency of trains on the route is high and therefore it's worthless to delay such trains. In this second case, the system should only take care at the end of day and maybe delay the last train. The last part needed for the simulation is the list of *unexpected events* which occurs along the day. An unexpected event is composed of three different steps: discovery of a problem, information about the resolution of the event (when the problem will be solved) and resolution of the problem.

3.3 Methods and algorithms

In Sweden, dispatching is done mainly by humans with a computer system to inform them [3]. It is important to notice that they work on a small area. Their work is based on a priority list: on-time time trains have priority over delayed trains, passengers trains over freight trains,... By using a tool working at the country scale, the whole problem can be analyzed which is important with fast long-distance trains. This approach can be implemented using heuristics. But more complex optimization algorithms can be used. Since this tool is based to *help* the dispatcher to choose a compromise, it's important to have an algorithm which can easily be tuned by changing some parameters and which can provide different solutions. It's why algorithms like branch and bound or genetic algorithms are good candidates. Genetic algorithms are also interesting since they don't need to restart from scratch when a small modification of data is done.

4 Conclusion

The dispatching problem is a complex problem which involved a lot of human factors (passengers) [4]. This problem is connected to other complex problem like infrastructure dimensionning [5], ... At this stage, a prototype is under construction to deal with a section mixing single and dual tracks. This prototype includes simple heuristics.

References

1. Olivera, E.S.d.: Solving Single-Track Railway Scheduling Problem Using Constraint Programming. PhD thesis, School of Computing, The Univerisity of Leeds (2001)
2. Rebreyend, P.: Distrain: A simulation tool for train dispatching. In: ITSC, 8th Internation conference on Intelligent Transportation Systems. (2005) 801–806
3. Hellström, P.: Analysis and evaluation of systems and algorithms for computer-aided train dispatching. Master's thesis, Hgskolan Dalarna and Uppsala University (1998) Licentiate thesis, UPTEC 98 008R.
4. Bussieck, M., Winter, T., Zimmermann, U.: Discrete optimization in public rail transport. In Mathematical Programming **79** (1997) 415–444
5. Labouisse, V., Housni, D.: Demiurge: A tool for the optimisation and the capacity assessment for railway infrastructure. World Congress on Railway Research (WCRR), Köln (2001)