A methodology to optimise traffic management in railway nodes

M. Galaverna\textsuperscript{a}, A. Imovilli\textsuperscript{b}, G. Sciutto\textsuperscript{a}

\textsuperscript{a}Dipartimento di Ingegneria Elettrica, Università degli Studi di Genova, Genova, Italy
\textsuperscript{b}Istituto di Trasporti, Università degli Studi di Genova, Genova, Italy

Abstract

This paper deals with the methodological aspects involved in the realisation of a decision support system to be used in optimising traffic control of railway nodes. The following topics are treated: state of the art of decision support systems for rail traffic management, identification of traffic management strategies, performance indices suitable for comparison among management strategies, and use of object-oriented modeling and simulation of railway operation within the above-mentioned scope.

1 Introduction

In recent years, the traffic transported by rail has notably increased in many European countries. As a consequence, the problem of evaluating and improving the carrying capacity of railway systems (i.e., especially lines and nodes) is currently of the utmost importance.

Moreover, the realisation of the high-speed rail network will raise hard difficulties concerning train circulation in main stations and nodes, since environmental problems make it unfeasible to realise new terminal station in most of European greater towns. Consequently, a great interest in addressed to those technological solutions by means of which the maximum carrying capacity of existing railway infrastructures can be achieved. To this aim, optimised traffic
control is very important especially for railway nodes; indeed, it has been shown that a considerable percentage of train delays is due to node congestion. Besides, the theoretical carrying capacity of a station can be exploited to an extent that depends on criteria by which train platforming, i.e. the assignment of trains to station tracks, is performed. Consequently, optimised platforming plans can be used to achieve maximum carrying capacity in a given station.

Before trying to find a solution of the problem, the authors have found in [1] a full reference of Expert Systems Application in Rail Transport. The application of expert systems covers the following topics: Control (including train protection), Electrical supply, Logistics/scheduling, Maintenance, Nuisance (noise, pollution, etc.), Planning, Routening to platforms or tracks, Simulation/modelling, Track layout.

Expert systems are probably better established in the rail sector than they appear. The Japan country seems particularly well established in the use of fuzzy logic in knowledge-base systems, for costing control, and for automatic train stop control. Traffic control is the main area of applications. For railways, the scope of traffic control is particularly wide, starting from network wide train movements, either to the control of trains into and out of the stations, either to the control of individual locomotives or wagons. The management issues relate to the rolling stock, platforming scheduling, routing and planning such as the overlap with the traffic control area.

Figure 1: Example of a node
A support system to optimise traffic management in railway nodes

A railway node can be modeled as a set of singular points \( P_1, \ldots, P_n \) connected with each other through one-track or double-track links. Figure 1 shows an example of a node. Singular points are of three types: stations, junctions, and gate-stations. A generic station \( P_i \) is characterised by the set of links \( t_i(1), \ldots, t_i(n_i) \) with which it is connected, being \( n_i \) the number of such links, and the set of platforming tracks \( b_i(1), \ldots, b_i(m_i) \), being \( m_i \) the number of these. Figure 2 shows an example of a station represented by means of its track plan. Junctions are singular points, outside stations, where two or more links branch off; a junction \( P_j \) is characterised by the set of branching-off links only, because junctions do not have any platforming track. Gate-stations are ending elements which indicate the bound between the node examined and the surrounding, external railway network. Within the scope of this study, the hypothesis will be made that the bound of a railway node can be defined precisely in all cases; gate-stations will be treated as train sources or sinks where, respectively, trains enter the node considered or get out it.

Traffic management optimisation in railway nodes can be undertaken at two levels, that is a static level and a dynamic one.

The former concerns the compilation of timetables and it consists in assigning the most suitable platforming track in every station and the most convenient route, wherever alternative routes do exist, to each train scheduled in the timetable. The results of this off-line optimisation procedure can be conveniently used within the compilation of timetables and station platforming plans; such documents are usually produced twice a year [2].
210 Computers in Railways

The latter concern real-time train circulation management. To this regard, it is worth observing that no action would be required if every train should respect its timetable, because train scheduling is conceived as to avoid interferences among trains. Nevertheless it is unlikely that such an ideal situation may occur, therefore the indications provided by the timetable and station platforming plans (that is, routes and station tracks assigned to each train) may turn to be unsuitable; in this case, it is advisable to undertake an on-line optimisation procedure which, by taking into account the delays of trains, yields the optimum assignment of routes and platforming tracks to trains, which may be different from the previous, statically-determined one.

3 Traffic optimisation in railway nodes

Several objective functions to be minimised can be assumed to perform optimum traffic control within a railway node, as listed below.

a) sum of the delays of trains circulating within the node considered, expressed as

\[ S_U = \sum_{i \in U} \left( \sum_{j \in N(i)} (t_{ij} - t_{ij}^*) \right) \]  \hspace{1cm} (1)

b) weighted sum of the same quantities as above mentioned,

\[ WS_U = \sum_{i \in U} w_i \left( \sum_{j \in N(i)} (t_{ij} - t_{ij}^*) \right) \]  \hspace{1cm} (2)

c) irregularity in train running, expressed as

\[ I_U = \sum_{i \in U} \left( \sum_{j \in L(i)} (v_{ij}^* - v_{ij}) \right) \]  \hspace{1cm} (3)

where:
- \( t_{ij} \) is the real departure instant of train \( TR_i \) from station \( P_j \)
- \( t_{ij}^* \) is the theoretical departure instant of train \( TR_i \) from station \( P_j \), as it can be found in timetable
- \( N(i) \) is the set of stations where train \( TR_i \) has to stop
- \( U \) is the set of trains circulating within the node considered in a given time interval
- \( w_j \) is a weighting factor that indicates the relative importance of train \( TR_j \)
- \( v_{ij} \) is the theoretical average speed of train \( TR_i \) along the \( j \)-th line section, as it can be deduced from timetable
- \( v^*_{ij} \) is the real average speed of train \( TR_i \) along the \( j \)-th line section
- \( L(i) \) is the set of line sections (i.e., links connecting two singular points with each other) that form the path of train \( TR_i \) within the node considered.

The following parameters can be assumed as free variables:
- the train successions \( T_{Sk} = \{TR_1, TR_2, ..., TR_i, ..., TR_N\} \), that indicate how the sequences of trains which have to enter or to leave a singular point (either a station or a junction) are arranged; the on-line optimisation procedure may result in different successions from those previously scheduled and written in timetables, if such differences lead to minimising one of the above-mentioned objective functions;
- the alternative routes \( L^k_{ij} \) that connect two singular points \( P_i \) and \( P_j \) with each other;
- platforming tracks of stations.

It is very difficult to analytically describe the relation among each of the above-mentioned objective functions and these free variables. Nevertheless, at the present some heuristic methods to perform optimisation have been found. Such methods are based on scanning the list of trains that enter the node considered; this list can be ordered in according to the theoretical sequence of trains as it is written in timetables or in according to the real order of appearance of trains at the gate-stations of the node under examination, if on-line optimisation is being undertaken. For each train present in such a list, one of the following criteria can be selected to carry out the optimisation:

- search for the route with maximum remaining traffic capacity
- search for the platforming track with minimum impedance.

The former criterion is based on considering all the different routes \( L^k_{ij} \) that link two points \( P_i \) and \( P_j \). Let \( v^k_{ij} \) be the average speed of trains along \( L^k_{ij} \) (also taking into account dwell times in intermediate stations); then, the traffic capacity \( C^k_{ij} \) of \( L^k_{ij} \), expressed in trains/hour in one direction, can be evaluated as

\[
C^k_{ij} = \frac{T}{T_{ij}^k + p_{ij}^k}
\]  

(4)
where:
- \( T \) is the time unit (for example, an hour)
- \( L^k_{ij} \) is the length of \( L^k_{ij} \)
- \( T^k_{ij}=k_{ij}/v^k_{ij} \) is the average time necessary to cover \( L^k_{ij} \)
- \( p^k_{ij} \) is the dead time related to the operation of the signalling system of \( L^k_{ij} \).

The remaining traffic capacity ratio \( RC^k_{ij}(t) \), in a given instant \( t \), can be evaluated as

\[
RC^k_{ij}(t) = 1 - \frac{NT^k_{ij}(t)}{C^k_{ij}}
\]  (5)

being \( NT^k_{ij}(t) \) the number of trains running along \( L^k_{ij} \) in the instant, with the exception of those dwelling in \( P_i \) or \( P_j \). This criterion, based on searching for the route with maximum remaining traffic capacity consists in assigning the route with the largest value of \( RC^k_{ij}(t) \) to the train which has to leave from \( P_i \) at the instant \( t \), if several alternative routes connecting \( P_i \) with \( P_j \) exist.

The latter criterion concern the choice of a platforming track for a generic train \( TR^k_i \) that has to enter a station \( P_i \). This criterion can be summarised as below (we refer to the symbols indicated in Figure 2). For a train \( TR^k_i \) running along the line section \( t^i(j) \) to the station \( P_i \) it is necessary to identify, among platforming tracks \( b^i(j),...,b^i(m^i) \), the subset of tracks which are clear and connected with \( t^i(j) \), let \( S \) be this subset. Then, all the routes \( (t^i(j),b^i(r)) \), being \( b^i(r) \) a generic element of \( S \), can be enumerated and ordered according to one of the following key parameters: length (from the shortest to the longest route), maximum alignment speed (from the highest to the lowest speed), or number of switches (from the largest to the smallest number). The criterion consists in platforming the train \( TR^k_i \) on the track \( b^i(r) \) that compares in the route \( (t^i(j),b^i(r)) \) which results to be optimal according to one or more of the above-mentioned key parameters; a weighted sum of these parameters can be regarded as an impedance.

A comparison between the optimisation criteria so far described is unlikely to be undertaken by an analytical approach, since the relations among free variables and objective functions are quite complex. Conversely, such a comparison can be obtained through computer simulation of railway operation. For this reason the author have undertaken the realisation of a software tool especially conceived for the simulation of railway node operation. The main features of this software tool, as far as modeling of railway operation is concerned, are dealt with in the next section.
4 A computer model of railway nodes

An object-oriented approach to the modelling of railway-operation processes turns out to be particularly suitable, as it is possible to identify, in railway systems, set of objects suited to being described as abstract data (e.g., trains, signals, stations) [3].

The model developed by the authors is based on a hierarchical data structure, as four logical levels can be identified; these are associated to the following main processes:

a) train running;

b) operation of interlocking in stations and junctions;

c) operation of block and signalling system;

d) centralised traffic control.

Within the computer code of the simulation program each level is associated with a basic class of objects. At the train level, the basic class is the class Train. The procedures associated with this object class include all the routines for simulating train-running phases (i.e., starting, constant-speed running, coasting, braking) and speed regulation as a function of the layout characteristics, of the train schedule, and of signals along the route.

At the level b, the basic class is Interlocking. Within the code, when simulation starts an object belonging to this class is instantiated for each station or junction present in the node considered. The class Interlocking includes the procedures for traffic control inside stations; these procedures reproduce the setting, the utilisation and the clearing of routes. The algorithm for the simulation of interlocking operation is based on the concept of Finite-State-Automaton (FSA); for each station or junction Pj an AC-type automaton is instantiated within the simulation program (Figure 3).

Level c concerns the operation of block systems. Because almost all the operating systems of European railways are based on distance spacing of trains, a central role is played by distancing posts which generally coincide with singular points Pj, with the exception of automatic intermediate block post of those line sections that are equipped with automatic electric block. Also the algorithm for the simulation of block systems is based on the concept of FSA. For each distancing post Pj (station or junction) as many pairs of FSA are instantiated as the number of section lines connected with Pj; these pairs of FSA consist of a R-type automaton (Figure 4) and a C-type automaton (Figure 5). Within the simulation program the commutation of such automata from a state to another one is caused by an exogenous event, such as the announcement of an arriving train or the clearing of a
track circuit. It can be proved that almost all the operation systems currently used in European railway networks can be described by this FSA-based model.

Figure 3: AC-type automaton

Level d concerns traffic control (i.e., giving right of way, changing train-passing points, modifying train sequences). The procedures used at this level simulate the operations that, in real railways, are performed by the dispatchers of railway lines operated by centralised systems. At this level, the basic class is the class Dispatcher. An object of this class is instantiated for each node or portion of network to which a traffic manager is attributed.

The above-described model makes it easy to test different optimised traffic control strategies without modifying the implementation of the other software modules.
Figure 4: R-type automaton

Conclusions

This paper has described some methodological aspects involved in the realisation of a decision support system for the optimisation of train circulation in railway nodes. A set of performance indices suitable to compare different traffic management strategies has been proposed. The comparison can be carried out by a program especially designed for computer simulation of operation of railway nodes; the main
features of this software tools, which are object-oriented data structure and modeling of railway operation based on finite-state automata, are discussed.

![C-type automaton](image)

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₀</td>
<td>ZERO STATE</td>
</tr>
<tr>
<td>S₁</td>
<td>WAITING FOR GIVING A BLOCK CONSENT</td>
</tr>
<tr>
<td>S₂</td>
<td>BLOCK CONSENT ALREADY GIVEN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>g₁</td>
<td>A REQUEST FOR BLOCK CONSENT HAS COME</td>
</tr>
<tr>
<td>g₂</td>
<td>THE REQUEST CAN BE SATISFIED</td>
</tr>
<tr>
<td>g₃</td>
<td>THE LINE SECTION HAS BEEN CLEARED BY THE TRAIN FOR WHICH THE CONSENT HAS BEEN GIVEN UP</td>
</tr>
</tbody>
</table>

Figure 5: C-type automaton

References

