CETRA²⁰¹⁴

3ʳᵈ International Conference on Road and Rail Infrastructure
28–30 April 2014, Split, Croatia

Road and Rail Infrastructure III

Stjepan Lakušić – EDITOR

Proceedings of the
3rd International Conference on Road and Rail Infrastructures – CETRA 2014
28–30 April 2014, Split, Croatia

Road and Rail Infrastructure III

EDITOR
Stjepan Lakušić
Department of Transportation
Faculty of Civil Engineering
University of Zagreb
Zagreb, Croatia

ORGANISATION

CHAIRMEN

Prof. Stjepan Lakušić, University of Zagreb, Faculty of Civil Engineering
Prof. Željko Korlaet, University of Zagreb, Faculty of Civil Engineering

ORGANIZING COMMITTEE

Prof. Stjepan Lakušić
Prof. Željko Korlaet
Prof. Vesna Dragčević
Prof. Tatjana Rukavina
Assist. Prof. Ivica Stančerić
dr. Maja Ahac
Ivo Haladin
dr. Saša Ahac
Josipa Domitrović
Tamara Džambas

All members of CETRA 2014 Conference Organizing Committee are professors and assistants of the Department of Transportation, Faculty of Civil Engineering at University of Zagreb.

INTERNATIONAL ACADEMIC SCIENTIFIC COMMITTEE

Prof. Vesna Dragčević, University of Zagreb
Prof. Isfendiyar Egeli, Izmir Institute of Technology
Prof. Rudolf Eger, RheinMain University
Prof. Ešref Gačanin, Univeristy of Sarajevo
Prof. Nenad Gucunski, Rutgers University
Prof. Libor Izvolt, University of Zilina
Prof. Lajos Kisgyörgy, Budapest University of Technology and Economics
Prof. Željko Korlaet, University of Zagreb
Prof. Zoran Krakutovski, University of Skopje
Prof. Stjepan Lakušić, University of Zagreb
Prof. Dirk Lauwers, Ghent University
Prof. Zili Li, Delft University of Technology
Prof. Janusz Madejski, Silesian University of Technology
Prof. Goran Mladenović, University of Belgrade
Prof. Otto Plašek, Brno University of Technology
Prof. Vassilios A. Profillidis, Democritus University of Thrace
Prof. Carmen Racanel, Technical University of Civil Engineering Bucharest
Prof. Tatjana Rukavina, University of Zagreb
Prof. Andreas Schoebel, Vienna University of Technology
Prof. Mirjana Tomičić-Torlaković, University of Belgrade
Prof. Audrius Vaitkus, Vilnius Gediminas Technical University
Prof. Nencho Nenov, University of Transport in Sofia
Prof. Marijan Žura, University of Ljubljana

MICROSCOPIC SIMULATION OF RAILWAY OPERATION FOR DEVELOPING INTEGRATED TIMETABLES

Andreas Schöbel, Mark Volcic
OpenTrack Railway Technology GmbH, Kaasgrabengasse 19/8, 1190 Vienna, Austria

Abstract

OpenTrack is a user-friendly railroad network simulation program developed at the Swiss Federal Institute of Technology's Institute for Transportation Planning and Systems (ETH IVT). It is a microscopic model that simulates rail system operations based on user defined train, infrastructure, and timetable databases. OpenTrack functions as a railroad laboratory, by, for example, allowing users to define incidents and take infrastructure out of service to evaluate alternative scenarios. The program uses a mixed discrete/continuous simulation process that calculates both the continuous solution of train motion equations and the discrete processes of signal box states and delay distributions. It generates a wide variety data that can be easily presented in many formats including graphs (e.g. time-space diagrams), tables, and images. OpenTrack's main uses have been to evaluate and test infrastructure plans and operating schedules to optimize network and timetable design. It can be run on several different computer platforms and incorporates the benefits of object oriented programming language with a common data interface structure.

Keywords: railway simulation, railway planning, railway operations analysis, RailML, XML, object oriented modelling

1 Introduction

At its most basic level a rail simulation program is a set of simplifying mathematical assumptions that attempt to duplicate actual train operations. It allows users to evaluate the impacts of user-defined changes on the rail system operation. There are many types of railroad simulation programs; each of which is designed to answer a different type of question. For example there are models designed to help dispatchers decide how to route trains through a network in real time. Similarly there are models designed to help planners identify infrastructure requirements for planned schedules ten years in the future. As with most planning processes, a critical component of railroad planning is choosing the best computer simulation tool to use in any given situation. Rail simulation models can be categorized as macroscopic or microscopic similar to other transportation models. Both types are outlined below.

1.1 Macroscopic Simulation Models

Macroscopic models use average or other statistical data to evaluate operation of the transportation system; they do not model individual unit (e.g. train) operations nor do they consider how trains are impacted by other trains or how safety systems impact train performance. A common type of macroscopic model is the NEMO model developed at the University of Hannover.

1.2 Microscopic Simulation Models

Microscopic simulation models attempt to replicate the actual operation of a railroad over time. They do this by modeling the operation of each individual train during a user-defined time step (often one second) and then repeating the process for the entire simulation period. Microscopic models consider the impact of trains on each other when they simulate train operations. For example, if during a particular time step, a train occupies a block that a second train wishes to occupy, then the program will model the second train as performing according to the (user defined) safety system parameters. In the case of a simple block signal system the second train would stop at the block entry signal and wait until the first train clears the block before proceeding. There are two types of microscopic models: synchronous and asynchronous. Synchronous models simulate all train operations in a single model run while asynchronous models simulate operations in a series of model runs.

In an asynchronous simulation the highest priority trains are modeled in the first run, then this schedule is locked and the second set of trains is modeled; the operation of the second set of trains does not impact the first set of trains, but is impacted by the first set (similarly, the operation of the third set of trains does not impact the first or second sets, and so forth). These types of models are often used for timetable construction since they can replicate ideal operational planning. A good example of this type of program is STRESI developed by RWTH in Aachen.

In contrast, synchronous models simulate all the trains operating in the modeled network at the same time, thus they provide a good way to simulate realistic operating situations (e.g. the impact of train delays on network operations). These models enable users to specify rules that the program uses to make dispatching decisions when there are conflicts between trains. These rules include simple train priorities (e.g. passenger trains before freight trains) as well as more complex rules designed to optimize some particular function (e.g. dynamic overtaking). In essence these models attempt to replicate good dispatching decisions.

OpenTrack is a microscopic synchronous railroad simulation model. It provides users with a great deal of flexibility for defining different dispatching logic as well as operational variables in a user-friendly manner.

2 OpenTrack Rail Simulation Program

OpenTrack was developed at the Swiss Federal Institute of Technology's Institute for Transportation Planning and Systems (ETH IVT). The project's goal was development of a user-friendly railroad simulation program that can run on different computer platforms and can answer many different questions about railway operations. Figure 1 illustrates the three main elements of OpenTrack: data input, simulation, and output.

OpenTrack is a microscopic synchronous railroad simulation model. As such it simulates the behaviour of all railway elements (infrastructure network, rolling stock, and timetable) as well as all the processes between them. It can be easily used for many different types of projects including testing the stability of a new timetable, evaluating the benefits of different long-term infrastructure improvement programs, and analyzing the impacts of different rolling stock.

2.1 Input Data

OpenTrack administers input data in three modules: rolling stock (trains), infrastructure, and timetable. Users enter input information into these modules and OpenTrack stores it in a database structure. Once data has been entered into the program, it can be used in many different simulation projects. For example, once a certain locomotive type has been entered into the database, that locomotive can be used in any simulation performed with OpenTrack. Similarly, different segments of the infrastructure network can be entered separately into

the database and then used individually to model operations on the particular segment or together to model larger networks.
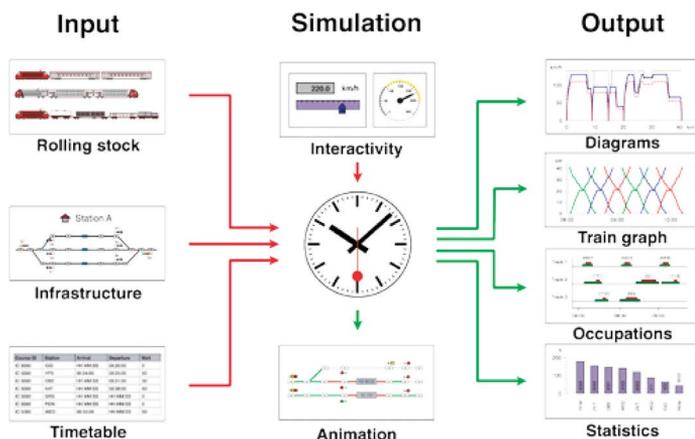

Figure 1 Data flow during a simulation project

Train data (locomotive and wagons) is entered into the OpenTrack database with easy to use forms displayed using pull down menus. Infrastructure data (e.g. track layout, signal type/location) is entered with a user-friendly graphical interface; quantitative infrastructure data (e.g. elevation) is added using input forms linked to the graphical elements. Following completion of the RailML data structure for rolling stock and infrastructure, OpenTrack will be modified to enable train and infrastructure data to be directly imported from RailML data files. Timetable data is entered into the OpenTrack database using forms. These forms include shortcuts that enable data input to be completed efficiently. For example, users can designate hourly trains that follow the same station stopping pattern an hour later. Since OpenTrack uses the RailML structure for timetable data, timetable data can also be entered directly from various different program output files as well as database files.

One advantage of OpenTrack is that it enables users to adjust many variables that impact railroad operations. For example, users can simulate the impact of weather on traction by specifying the adhesion scenario (good, normal, bad). OpenTrack then estimates locomotive traction power using a percentage (also user-defined) of that calculated using the Curtius and Kniffler formula. While OpenTrack provides standard default values for all variables, having the ability to adjust variables makes the program quite useful.

2.2 OpenTrack Simulation Process

In order to run a simulation using OpenTrack the user specifies the trains, infrastructure and timetable to be modeled along with a series of simulation parameters (e.g. animation formats) on a preferences window. During the simulation, OpenTrack attempts to meet the user-defined timetable on the specified infrastructure network based on the train characteristics. OpenTrack uses a mixed continuous/discrete simulation process that allows a time driven running of all the continuous and discrete processes (of both the vehicles and the safety systems) under the conditions of the integrated dispatching rules.

The continuous simulation is dynamic calculation of train movements based on Newton's motion formulas. For each time step, the maximum force between the locomotive's wheels and the tracks is calculated and then used to calculate acceleration. Next, the acceleration function is integrated to provide the train's speed function and is integrated a second time to provide the train's position function.

The discrete simulation process models operation of the safety systems; in other words, train movements are governed by the track network's signals. Therefore, parameters including occupied track sections, signal switching times, and restrictive signal states all influence the train performance. OpenTrack supports traditional multi-aspect signaling systems as well as new moving block train control systems (e.g. European Train Control System – ETCS signaling).

2.3 Dynamic Rail Simulation

OpenTrack is a dynamic rail simulation program. As such, the simulated operation of trains depends on the state of the system at each step in the process as well as the original user-defined objective data (e.g. desired schedule).

A simple way of describing dynamic rail simulation is that the program decides what routes trains use while the program is running. For example, when building the network, users identify various different routes that trains can use between two points; OpenTrack decides, during the simulation, which route the train will use by assigning the train the highest priority route available. If the first priority is not available, OpenTrack will assign the train the second highest priority route and so on.

OpenTrack's dynamic nature allows users to assign certain attributes to specified times in the simulation. Thus, users can assign a delay to a particular train at a given station and time, rather than being limited to assigning a delay at the start and using it through the entire simulation. Similarly, users can define other types of incidents (e.g. infrastructure failures, rolling stock breakdowns) for particular times and places.

Finally, dynamic simulation enables users to run OpenTrack in a step-by-step process and monitor results at each step. Users can also specify exactly what results are displayed on the screen. Running OpenTrack in a step-by-step mode with real time data presented on screen helps users to identify problems and develop alternative solutions.

2.4 OpenTrack Output

One of the major benefits of using an object oriented language is the great variety of data types, presentation formats, and specifications that are available to the user. During the OpenTrack simulation each train feeds a virtual tachograph (output database), which stores data such as acceleration, speed, and distance covered. Storing the data in this way allows users to perform various different evaluations after the simulation has been completed.

OpenTrack allows users to present output data in many different formats including various forms of graphs (e.g. time-space diagrams), tables, and images. Similarly, users can choose to model the entire network or selected parts, depending on their needs. Output can be used either to document a particular simulation scenario or as an interim product designed to help users identify input modifications for another model run.

3 Application of OpenTrack at NeusiedlerSeeBahn (NSB)

The NeusiedlerSeeBahn is responsible for the line from Neusiedl am See to Pamhagen. At Neusiedl am See the branch line from/to Eisenstadt is located. Pamhagen is the last station in Austria before the Hungarian border. From Neusiedl am See a line leads to Parndorf Ort where there is the connection to the Eastern line of Austrian Railways which is connecting Vienna and Budapest. People living at the villages located between Neusiedl am See and Pamhagen typically go for work to Vienna every single day of the week. Therefore the NSB is interested in offering shorter traveling times to Vienna. To achieve this goal some investments were done in the last years to increase the track speed. OpenTrack has been successfully used to evaluate the shortenings of running times for local trains and regional trains.
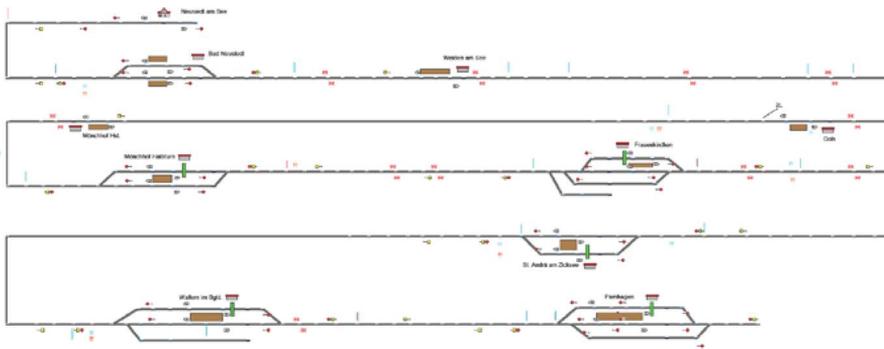
Figure 2 Infrastructure layout of the line from Neusiedl am See to Pamhagen

Figure 2 shows the topology of the line from Neusiedl am See to Pamhagen. Crossing opportunities are located at Bad Neusiedl, Mönchhof-Halbturn, Frauenkirchen, St. Andrä am Zicksee, Wallern and Pamhagen. The first step in the project was the check of the model with the existing timetable. This step allowed the exact calculation of running times reserves included in the existing timetable.
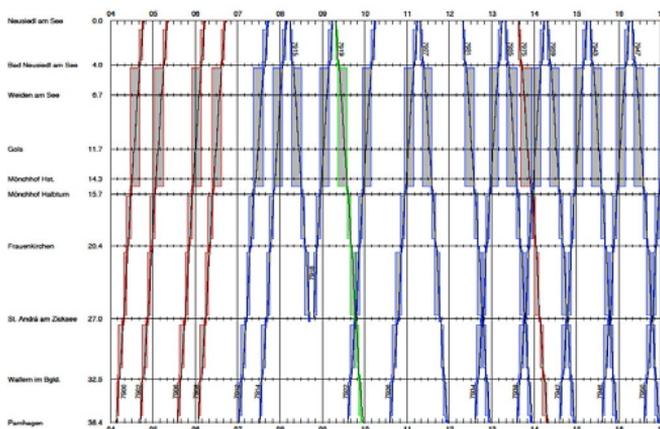

Figure 3 Simulation result of the existing timetable

Figure 3 shows the result from the simulation of the existing timetable. It has to be noted that one course during the day is only going to St. Andrä am Zicksee and not to the terminal station at Pamhagen. Additionally there is no service during the morning hours towards the region between Neusiedl am See and Pamhagen. Exactly the same happens during the evening when there is no service towards Vienna. The reason for this asymmetry in the timetable can be only explained by the saving of operational costs. For pupils there are two additional services during the day to allow them traveling to and back from school. Unfortunately the infrastructure does not allow a crossing between Bad Neusiedl and Mönchhof-Halbturn which would be required to run a 30 minutes interval in both directions. Furthermore the infrastructure model had to be extended with the upgraded track speed limits which had been indicated by markers at the related vertices. Due to the increase of the track speed the crossing shifts from St. Andrä am Zicksee towards Wallern because of keeping the arrival and departure times at Neusiedl am See (see Figure 4).
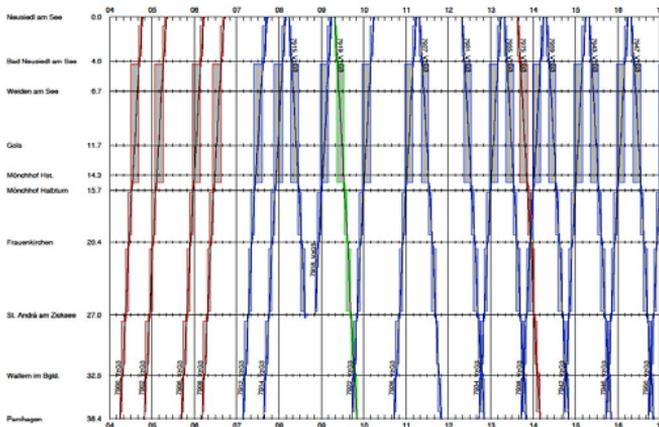
Figure 4 Result of the simulation for upgraded infrastructure

Local and regional trains benefit from the increase of track speed by a shortening of running times of 5 minutes each. Local trains which stop at every single station run in 33 instead of 38 minutes and regional trains run in 28 instead of 33 minutes. A shortening of 5 minutes in terms of running time could be also achieved by introducing today regional trains. Of course the increase of the track speed in combination with the introduction of regional trains leads to an overall reduction of running time of 10 minutes for regional trains. An open point for further investigations is the integration of both services at the same time because each crossing of two trains will lead to an increase of running time while one train has to take the siding track in a crossing station. This could require the demand to upgrade also the frequently used switches to allow higher speeds for the siding track.

4 Conclusions

OpenTrack is an efficient and effective railroad simulation program. It has been successfully used in many different railway planning projects throughout the world. The program's use of object oriented programming and the RailML data structure makes it particularly effective since the program can be modified relatively easily to address specific applications and since data can be transferred easily to and from other programs based on RailML.

References

[1] Huerlimann, D.: Object oriented modeling in railways; ETH Dissertation Nr. 14281; 2001 (in German).

[2] Huerlimann, D., Nash, A.: OpenTrack – Simulation of Railway Networks, User Manual Version 1.6; OpenTrack Railway Technology GmbH; Zürich, Switzerland; 2010; 160 pages.

[3] For more information on OpenTrack see the project website: www.opentrack.ch