Automatically and Quickly Planning Platform and Route of Trains in Railway Stations

with a Case Study for Mechelen Station

P. Sels, T. Dewilde, D. Cattrysse, P. Vansteenwegen

Abstract—When creating a railway timetable, a sub-problem that occurs for every station is the Train Platforming Problem (TPP). We show that we are able to automatically and quickly solve all TPPs, and as such create the platform and route plans for all trains for all Belgian railway stations with our tool called Leopard. Leopard also evaluates the plan created by human planners, if it already exists. For both the human and the Leopard created plan, we produce a graphical image that is easily interpreted by humans. We are able to run Leopard on 232 stations in Belgium and all but one of them is solved in less than 1 second. Only one station takes more time to solve: 117 seconds.

Keywords—Train Platforming Problem (TPP), Train Routing, Mixed Integer Linear Programming (MILP).

I. INTRODUCTION

The Train Platforming Problem (TPP) is the problem of finding a platform and an IN-route for each train entering and an OUT-route for each train leaving a station. We developed a Mixed Integer Linear Programming (MILP) model. This model is implemented as a software tool, called Leopard, and integrated in the software architecture at Infrabel, the Belgian railway infrastructure management company. We show what results Leopard produces for a typical station.

In section II, we define the type of TPP we consider. In section III, we shortly describe the MILP model, its context and the advantage of using our tool. In section IV we zoom in on graphical representations of the results for Mechelen station: a picture based evaluation of the current plan and an automatic quick construction of an optimized plan and its similar evaluation. Also mentioned is how quickly we obtain results for all other stations in Belgium. Section V concludes and hints at some further work.

II. PROBLEM DEFINITION

A. Problem Definition

The TPP has been the subject of active research for some time [1-22,24-29,32,34,38-40]. We consider the particular type of TPP where one decides on platform tracks and routes for each train at the planning stage and not the real-time dispatching stage. Unlike [3-15,18-22,24-29,38-40] but

P. Sels is a self employed consultant with Logically Yours, Antwerp, Belgium (phone: +32 486 95 67 97, e-mail: sels.peter@gmail.com).

similar to [1,16,17], our model does not change any train arrival nor train departure time. Also, unlike [2-7,14-15,18-22,24-28,38-40] but alike [1,8-13,16,17], we consider only one route per (line,platform)-combination. We do however allow any number of incoming trains from any entry-lines to merge onto one platform as well as allow a train to split into any number of trains and drive out towards any exit-line. The objective function of our model is the number of train occupations assigned to real platform tracks and is being maximized. A fictive platform and fictive routes to and from it, from and to all IN respectively OUT lines, are defined. These fictive resources will hold all trains that cannot be assigned to real platforms and/or routes. In the case the TPP is infeasible, this allows us to generate a partial platforming rather than just reporting infeasibility.

B. Infrastructure Definition

Of course our TPP MILP model is set up so that it is as general as possible. Initially we defined train stations as they were defined in the databases of Infrabel, the Belgian railway infrastructure manager company. This is as a set of parallel platform tracks, juxtaposed by a grid on one or two sides. Each grid makes connections between lines and platform tracks. At first, we defined routes as belonging to a grid, which is somehow restrictive and does not allow all station topologies. In a second software architecture, we freed routes from having to go from one side of a grid to the other side. Routes are now simply sequences of components (signals, switches). Two routes are called *dependent* if they have one or more components in common and *independent* otherwise. In our model, we don't allow two dependent routes to be occupied at any time by more than one train. Each route has a length and a maximum speed. This definition and use of routes allows the most general type of station to be appropriately modeled.

Platforms tracks have a length and stopping trains stop at a position where the middle of the train is aligned to the middle of the platform.

C. Traffic Definition

We define the concept (train) *movement* as either an IN-movement or an OUT-movement. In an IN-movement a single train comes from a given open track line, arrives, with the middle of the train aligned to the middle of the - to be determined - platform track, at a given arrival time. A compatible route has to be determined too. In an OUT-movement, a train leaves the middle of the – to be determined - platform at a given departure time, takes a - to

be determined - route towards a given open track line exiting the station.

Next, we define a (train) *occupation*, which bundles at least an IN-movement and an OUT-movement and realizes these on a single platform track. Multiple IN-movements can be bundled in a single occupation, which then defines the merging of multiple trains from different IN-lines onto a single platform track. Similarly, multiple OUT-movements can be bundled in a single occupation, which then defines the splitting of multiple trains from a single platform track towards different OUT-lines. A single occupation can also hold multiple IN-movements as well as multiple-OUT movements, which will then also all be assigned to a single platform track. The latter represents multiple trains all riding onto onto a single platform track and then being differently divided and attached and finally all riding towards different exit lines.

III. LEOPARD MODEL AND CONTEXT

A. MILP Model

With the definitions of the previous section, we constructed a MILP model of our TPP version. This model is described in detail in [34].

Based on the C++ library milp_logic [30], which we developed and open sourced, we were able to test our model against the three main commercial MILP solvers, Cplex 12.5, Gurobi 5.6.0 and Xpress 7.1. The sizes of the models remain quite small and solver times were very similar across the different solvers.

The MILP model used by Leopard [34] contains quite some boolean variables and expressions. These can be converted to 0,1-integers and linear constraints as for example described in [37]. After this conversion, the model is a true MILP model. This conversion is tiresome and error-prone when done manually. So in milp-logic [30], we implemented it as a layer on top of the solver abstraction layer. The result is that we can build MILP models extended with booleans and boolean constraints over them without worrying about linearity. The added bonus is that the models containing boolean expressions are more readable than the expanded ones after conversion.

B. Context and Advantage

From Infrabel databases, Leopard reads *macroscopic infrastructure* to determine lines and *microscopic infrastructure* to determine platforms and their physical connectivity from/to lines in a station.

In an earlier version of Leopard, *Train routes* along these physical connections could only be read from a database that only had contained this information for stations that had the most recent signaling system version (EBP), because only then, this information could be automatically read out and stored in the database. Now, in addition, Leopard can also read routes from a newly created XML export from Via-Cons train network simulation tool *LUKS* [23]. Manually inserting all possible route variants in

LUKS via the GUI is of course a very laborious process but the advantage of this import method is that routes are now available for all stations in Belgium. Note that Leopard, of all LUKS route variants, currently only uses the default route variants. Leopard selects these by taking the highest priority route variant amongst all route variants between the same endpoints.

Finally, Leopard also reads *train movements*, from either the database with the current planning, or the database with the planning under construction. In the latter case, all train movements that human planners have not assigned to a real platform track are put on the fictive platform in the '*original*' planning.

The idea and *advantage* of using Leopard on train movements of a new timetable is of course to quickly create an '*optimized*' platforming plan for all stations at a time when no other platforming plan could yet be manually created. As such one can more *quickly verify if the timetable is feasible* on the microscopic level as well. If not, the timetable needs to be changed and Leopard needs to be rerun on all affected stations. This process has to be repeated until all trains can be platformed in all stations.

The fact that Leopard can be used on train movements of a new timetable is a major contribution to the timetable planning process.

IV. RESULTS

We first report in a graphical way on results obtained for a single relatively large, relatively busy station: Mechelen. We show what human planners obtained as current platforming plan and contrast it with the plan Leopard generated. Next we show that we were also able to quickly generate similar platforming plans for 231 other stations in Belgium.

A. Evaluation of the Current Platforming Plan

For a given station, Leopard reads the current platforming plan, if available, and generates an occupation graph of all platform tracks. An example for the station of Mechelen is given in Fig. 1. The vertical axis shows the platform tracks. The horizontal axis shows time. Each light yellow rectangle represents an *occupation*, which is a time interval during which a platform track is occupied by a train. At the beginning of each occupation, a bLue (Left) rectangle represents a train entering the station, called an *IN-movement*. At the occupation ending, a bRown (Right) rectangle shows a train leaving the station, called an *OUT-movement*. We see that the occupation on platform track V from 06:52 until 07:15 also has a blue rectangle at 07:07. This means that a second train is then entering the station and joins the one already on platform V since 06:52. Both trains are coupled and leave as one *merged* train at 07:15. Similarly, splitting trains is possible as well, but Fig. 1 does not show an example of this. We see that no single occupation occurs on the fictive platform, which would only happen if the human planners have not fully planned all trains yet.

No pair of yellow rectangles overlaps, so we are sure that no two trains are using the same platform track at the same time, except if they are being merged or split, as this is obviously allowed - even necessary - then.

For any dependent route pair of which the second route is starting to get used within 5 minutes after the end of the usage interval of the first route, a green line is drawn between end of the first route and beginning of the second. If reused within 2 minutes we draw a light orange line, if reused within 1 minute a dark orange line and if reused within 0 minutes, which means that time overlap occurs, a red line. Fig. 1 does not show any red lines, which means there are no dependent route overlaps. This means that the route planning is feasible for the case of no train delays. There are three dark orange lines (from platform tracks VI to III, IV to IV and IV to V) with reuse times all equal to 0.9 minutes. They are to be considered as robustness attention points. Indeed, if the first train movement has a delay of 0.9 minutes the second train movement will have to be delayed too. There are three light orange lines with route reuse times of 1.8 minutes (VI to IV) and 1.9 minutes (VI to III). They are minor robustness attention points. Fig. 1 also shows 38 green lines which don't indicate any current problem. They only serve as a warning to the planner that when he changes platform times, train movement pairs that have green lines between them use dependent routes and should not be scheduled too closely together.

Note that, if any simultaneous use of the same platform by two occupations would occur, its necessarily dependent routes would also be in conflict and this would be marked as well. The dark orange line from platform IV to IV, between times 7:43 and 7:44 is such a case.

The colored lines in Fig. 1 also show a clear separation of platforms into three groups: I-VI, VII-VIII and IX-X. There are lines present amongst platform tracks of the same group, but not between platform tracks of different groups. Routes of different groups are independent. This is in part a consequence of the infrastructure of different groups being entirely separated. Indeed, in Mechelen station, platforms I-VI are at a different height than the platforms VII-X, so in the also separated grids, no routes exist from the first to the second group. In the grids attached to platforms VII to X, many interconnections are possible, but only straight through routes have been used, so no route conflicts arise here.

Also, for Fig. 1 to Fig. 3, when hovering over platform track lines, time division lines, occupation or movement rectangles, movement brown or blue text and route conflict lines, some extra information becomes available as tooltip text. For movement brown or blue rectangles and their text, the entire series of components (signals, switches) is shown. For route conflict lines, the first common component between the routes is shown as well as the reuse time of the dependent routes. The system of tooltips allows to more clearly see the complete text, which otherwise possibly partially overlaps with other text. It is also shown then in a larger font.
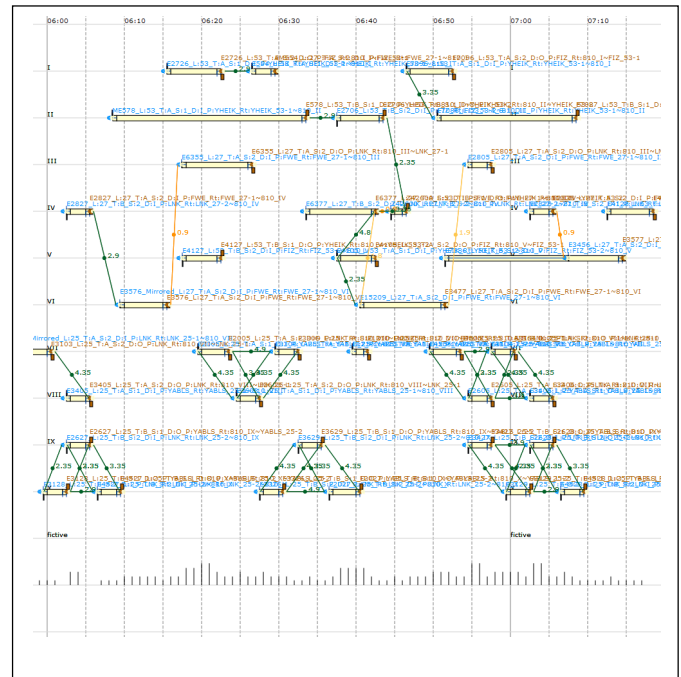

Fig 1.   Occupation graph of the current, human made, platform and route plan for Mechelen station for 6/11/2013. Hours to days of work for human planners.

At the bottom of Fig. 1, the number of occupations present at the station for every minute has been plotted graphically. This shows a measure of how station *business* varies over time.

B.   Generation and Evaluation of the Optimized Platforming Plan

For a given station, Leopard also automatically generates an optimized platforming and route plan. The maximum of train occupations is planned on real platform tracks without generating any train pair conflict between same platforms nor between dependent routes. Fig. 2 shows this plan, again for Mechelen station for the same date and time interval as in Fig. 1. We see that the same occupations as in Fig. 1 are present and that here too, no red lines occur, so no two dependent routes are used simultaneously.

As in Fig. 1, there are 9 i.o. 3 dark orange lines, again all with reuse times of 0.9 minutes. There are also 3 i.o. 2 light orange lines with reuse times of 1.8, 1.9 and 1.9 minutes. So the number of robustness attention points has somewhat increased compared to the original plan. This is because the goal function now only considers the number of occupations platformed and does not consider any measure of (minimal) resource reuse time. We plan to extend the objective function so that it also takes care of this.

Fig. 2 shows that, compared to Fig. 1, some occupations have changed platform track. For example, the train E2827 between times 06:02 and 06:06 was originally on platform track IV and is now on II. Both platform tracks IV and II are usually used by trains driving through the station in the same direction, so this is fine. Train occupation E3576_Mirrored between 06:09 and 06:15 was moved from VI to III. Platform tracks VI and III are currently not commonly used in opposite directions. Even though the Leopard solution is correct, this would not be currently preferred. A goal function term could be added to penalize usage of platform tracks in their non-default direction.
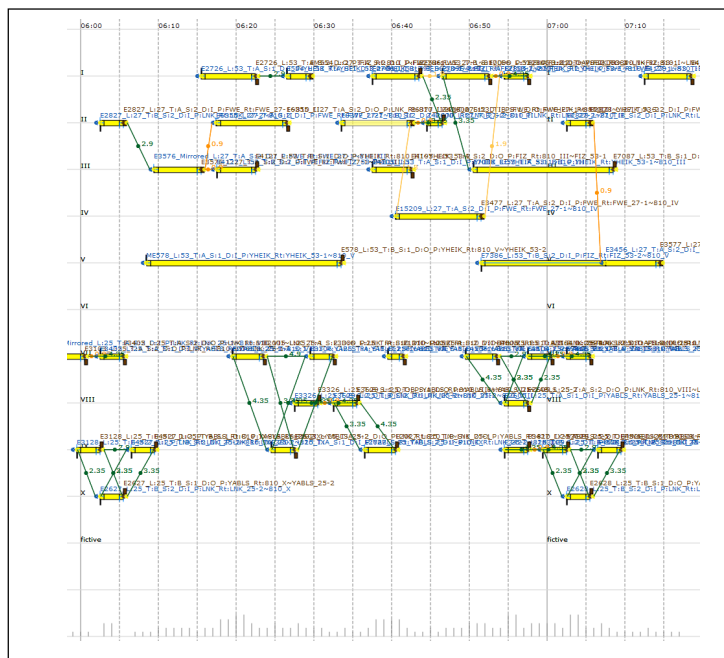
generated in the scalable vector graphics (SVG) format, so that when loaded in a web-browser it allows some dynamic functionality. A user can click on a certain occupation of the original platform plan and the cursor will jump to the same occupation in the optimized platform plan. This allows a user to easily verify to which platform the occupation has changed.

At the bottom of Fig. 3, the station business graphs for the original and optimized platform plan can be compared. They are equal since Leopard does not currently perform time shifts.
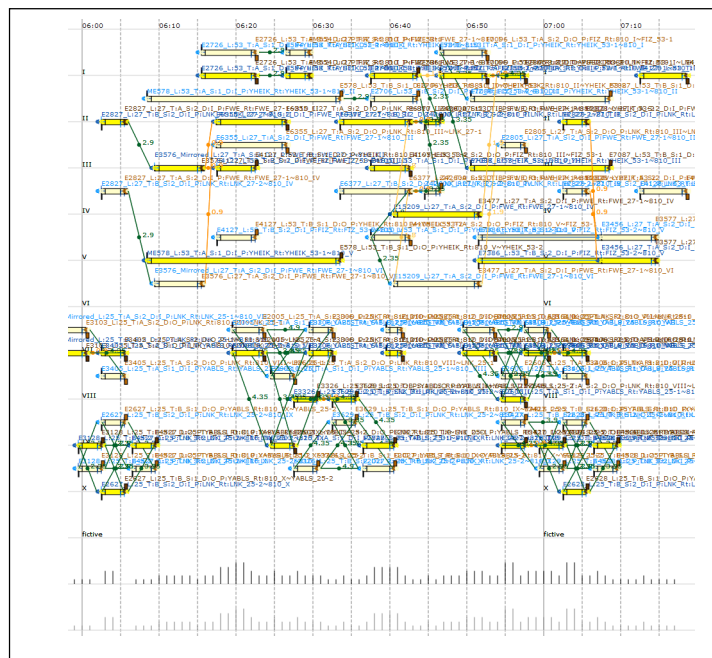

Fig 2. Occupation graph of the optimized, Leopard produced, platform and route plan for Mechelen station for 6/11/2013. Less than a second of work for our tool Leopard.

Fig 3. Comparison of occupation graphs of both, the human made (as in Fig. 1) and Leopard made (as in Fig. 2), platform and route plan for Mechelen station for 6/11/2013.

In contrast to Fig.1, Fig. 2 now also shows (4) green lines between platform tracks VII and IX. This indicates that some routes between platform track groups VI-VII and IX-X are dependent. The human plan in Fig.1 just did not use those.

C. Comparison of Original and Optimized Platforming Plan

The human solution in Fig. 1 and the Leopard solution in Fig. 2 both platform all trains, so when taking the criterium of our current goal function being number of trains platformed, both solutions can be considered optimal. Robustness in the human solution is still better here than in the Leopard solution. By taking robustness into account in the goal function, we want to improve on that.

Fig. 3 compares the solutions of Fig.1 and Fig. 2, by interleaving them into one picture. To avoid overlap, the original platform plan has been shifted upwards with half the height of a platform to platform division. The picture is

D. Application to All Belgian Passenger Stations and Traffic

Leopard has been tested on 232 stations in total. These are all Belgian stations which have platforms for passengers. When selecting all station traffic between 7am and 9am, Leopard is able to solve the generated MILP model for each station platforming problem in less than a second. This excludes some seconds to read in traffic from databases. Only one station, which has 22 platform tracks, an abundance of route possibilities and very dense train traffic, takes 117 seconds to solve.

For some stations, like Mechelen, Leopard can platform and route all traffic. For others this is not the case and the user then has the option to manually change train arrival or departure times until Leopard can platform all required traffic. Currently, Leopard cannot automatically change platform times. They are fixed in the model. However, this forces the user to more consciously select particular trains

and time shifts. This is important since any change of a train arrival or departure time in a station has to be propagated to the macroscopic timetable. If not enough reserves on the open lines are available, the change to the timetable can render it infeasible.

A possible future extension of Leopard that would change platform times automatically would also have to possess information about how much time shift can be applied without rendering the macroscopic timetable infeasible. Inspired by our other work in macroscopic timetabling [18,31,33-36], we envision that passing this information from timetabling to platforming and also even simultaneous optimization of them might one day become possible.

V. CONCLUSION

Leopard uses a MILP model which allows to quickly and automatically decide on a platform and a route for each train in a station. Train merges and splits are supported. A picture showing all double uses of the same platform and all double uses of dependent routes is automatically generated for the current platforming plan. Leopard generates a plan which assigns a maximum of train occupations to real platform tracks. The plans are correct by construction. For each but one Belgian station, they are generated in less than one second. Both the correctness and the speed of resolution are advantages compared to the manual planning.

Concerning the softer requirements of the platforming plan, the Leopard generated platform plans can still have more robustness attention points than the manually constructed plan. Also, the current practice of preferring platform tracks to be mainly used in one direction is not enforced yet in the Leopard model. We will add both of these softer requirements in the Leopard objective function. Our hard constraints can stay unchanged.

REFERENCES

[1] Billionnet, A., 2003. Using Integer Programming to Solve the Train Platforming Problem. Transportation Science 37, 213–222.

[2] Burkolter, D.M., 2005. Capacity of Railways in Station Areas using Petri Nets. Ph.D. thesis. Swiss Federal Institute of Technology Zürich.

[3] Caimi, G., 2009. Algorithmic decision support for trains scheduling in a large and highly utilised railway network. Ph.D. thesis. ETH Zürich.

[4] Caprara, A., 2010. Almost 20 Years of Combinatorial Optimization for Railway Planning: from Lagrangian Relaxation to Column Generation. ATMOS 2010 1, 1–12.

[5] Caprara, A., Galli, L., Toth, P., 2011a. Solution of the train platforming problem. Transportation Science 45, 246–257.

[6] Caprara, A., Kroon, L., Monaci, M., Peeters, M., Toth, P., 2007. Passenger Railway Optimization. Handbooks in Operations Research and Management Science. Volume 14. Elsevier, Amsterdam, 129-187.

[7] Caprara, A., Kroon, L., Toth, P., 2011b. Optimization Problems in Passenger Railway Systems. Wiley Encyclopedia of Operations Research and Management Science 6, 3896–3905.

[8] Carey, M., 1994a. A model and strategy for train pathing with choice of lines, platforms and routes. Transportation Research 28B(5), 333–353.

[9] Carey, M., 1994b. Extending a train pathing model from one-way to two-way track. Transportation Research 28B(5), 395–400.

[10] Carey, M., Carville, S., 2000. Testing schedule performance and reliability for train stations. Journal of the Operational Research Society 51(6), 666–682.

[11] Carey, M., Carville, S., 2003. Scheduling and platforming trains at busy complex stations. Transportation Research 37A(3), 195–224.

[12] Carey, M., Crawford, I., 2005. Scheduling trains on a network of busy complex stations. Transportation Research 41B, 159–178.

[13] Carey, M., Lockwood, D., 1995. A model, algorithms and strategy for train pathing. Journal of the Operation Research Society 46B(8), 988–1005.

[14] Chakroborty, P., Vikram, D., 2008. Optimum assignment of trains to platforms under partial schedule compliance. Transportation Research Part B, Methodological 42, 169–184.

[15] Cordeau, J.F., Toth, P., Vigo, D., 1998. A Survey of Optimization Models for Train Routing and Scheduling. Les Cahiers du GERAD 1, 1–42.

[16] De Luca Cardillo, D., 1998. k L-List Colouring of Graphs. European Journal of Operational Research 106, 160–164.

[17] Delorme, X., Rodriguez, J., Gandibleux, X., 2001. Heuristics for railway infrastructure saturation. Electronics Notes in Theoretical Computer Science 50, 39–53.

[18] Dewilde, T., Sels, P., Cattrysse, D., Vansteenwegen, P., 2013. Improving the robustness in railway station areas. European Journal of Operational Research (EJOR), URL: http://dx.doi.org/10.1016/j.ejor.2013.10.062.

[19] Fuchsberger, M., 2007. Solving the train scheduling problem in a main station area via a resource constrained space-time integer multi-commodity flow. Master's thesis. Institute of Operations Research, ETH Zürich.

[20] Galli, L., 2009. Combinatorial and Robust Optimisation Models and Algorithms for Railway Applications. Ph.D. thesis. Univ. of Bologna.

[21] Ghoseiri, K., Szidarovski, F., Asgharpour, M.J., 2004. A multi-objective train scheduling model and solution. Transportation Research 38B, 927–952.

[22] Herrmann, T.M., 2006. Stability of Timetables and Train Routings through Station Region. Ph.D. thesis. Swiss Federal Institute of Technology Zürich.

[23] Janecek, D.; Weymann, F., LUKS - Analysis of lines and junctions, Proceedings of the 12th World Conference on Transport Research (WCTR), July 11-15, 2010, Lisbon, Session C2.2.

[24] Kroon, L., Romeijn, H., Zwaneveld, P.J., 1997. Routing Trains Through Railway Stations: Complexity Issues. European Journal of Operational Research 98, 485–498.

[25] Lusby, R.M., 2008. Optimization Methods for Routing Trains through Railway Junctions. Ph.D. thesis. University of Auckland.

[26] Lusby, R.M., Larsen, J., Ehrgott, M., Ryan, D., 2011a. Railway track allocation: models and methods. OR Spectrum 2011 33, 843–883.

[27] Lusby, R.M., Larsen, J., Ehrgott, M., Ryan, D., 2013. A set packing inspired method for real-time junction train routing. Computers and Operations Research 40, 713–724.

[28] Lusby, R.M., Larsen, J., Ryan, D., Ehrgott, M., 2011b. Routing Trains Through Railway Junctions: A New Set-packing Approach. Transportation Science 45, 228–245.

[29] Miao, J., Yu, Y., Meng, L., Yang, Z., 2012. Stability-oriented Optimization of Train Platforming Problem on High Speed Railway. Journal of Transportation Systems Engineering and IT 12, 115–121.

[30] Sels, P., 2012. milp-logic: a C++ MILP Solver Abstraction Layer with a C++ Boolean Modeling Layer on Top. URL: http://code.google.com/p/milp-logic.

[31] Sels, P., Dewilde, T., Cattrysse, D., Vansteenwegen, P., 2011. Deriving all Passenger Flows in a Railway Network from Ticket Sales Data. Proceedings of 4th International Seminar on Railway

Operations Modeling and Analysis (IAROR): RailRome2011, February 16-18, Rome, Italy. URL: http://4c4u.com/RR2011.pdf.

[32] Sels, P., Waquet, B., Dewilde, T., Cattrysse, D., Vansteenwegen, P., 2011. Calculation of Realistic Railway Station Capacity by Platforming Feasibility Checks. Proceedings of 2nd International Conference on Models and Technologies for Intelligent Transport Systems (MT-ITS): MTITS2011, June 22-24, Leuven, Belgium. URL: http://4c4u.com/ITS2011.pdf.

[33] Sels, P., Dewilde, T., Cattrysse, D., Vansteenwegen, P., 2013. Expected Passenger Travel Time as Objective Function for Train Schedule Optimization. Proceedings of 5th International Seminar on Railway Operations Modeling and Analysis (IAROR): RailCopenhagen2013, May 13-15, Copenhagen, Denmark. URL: http://4c4u.com/RC2013.pdf.

[34] Sels, P., Dewilde, T., Cattrysse, D., Vansteenwegen, P., 2013. The Train Platforming Problem: The Infrastructure Management Company Perspective, submitted for review to Transportation Research, Part B: Methodological. URL: http://4c4u.com/TRB2013.pdf.

[35] Vansteenwegen, P., Van Oudheusden, D., 2006. Developing Railway Timetables that Guarantee a Better Service. European Journal of Operational Research 173, 337–350.

[36] Vansteenwegen, P., Van Oudheusden, D., 2007. Decreasing the passenger waiting time for an intercity rail network. Transportation Research 41B, 478–492.

[37] Williams, H., 1994. Model Building in Mathematical Programming. volume 1. 3 ed., John Wiley and Sons Ltd, England.

[38] Zwaneveld, P.J., 1997. Railway Planning and Allocation of Passenger Lines. Ph.D. thesis. Rotterdam School of Management.

[39] Zwaneveld, P.J., 2001. Routing Trains through a Railway Station based on a Node Packing Model. EJOR 128, 14–33.

[40] Zwaneveld, P.J., Kroon, L., Romeijn, H., Salomon, M., Dauzère-Perès, S., Van Hoesel, S.P.M., Ambergen, H.W., 1996. Routing Trains Through Railway Stations: Model Formulation and Algorithms. Transportation Science 30, 181–194.

P. Sels obtained his M.Sc. in Electrical Engineering at Group T, Institute of Technology, Leuven, Belgium in 1994 and his M.Sc. in Computation at Oxford University, U.K in 1996. He was a digital chip design engineer at Easics and was a consultant to Altran, Alcatel Space, Camco Technologies and Philips Research. He is currently consulting to the Belgian Railway Infrastructure Management Company Infrabel as well as performing research towards his Ph.D. at KU Leuven, Belgium. His current research interests are timetabling, platforming and the interaction between both.

T. Dewilde is Master of Science in Mathematics (KU Leuven, 2009). Being an IWT-fellow, he is currently active as PhD researcher at the Centre for Industrial Management, Traffic and Infrastructure of the department of Mechanical Engineering of the KU Leuven. His current research interests are applications of operations research in the field of public transportation and logistics.

D. Cattrysse is Master of Science in civil engineering (KU Leuven, 1983), Master of Engineering in industrial management (KU Leuven, 1984), and holds a Ph.D. in industrial management (KU Leuven, 1990). He studied as a visiting fellow at the MIT Sloan school of management (USA, 1988-1989), where he did research in the area of applied operations research. He is currently full professor at the KULeuven and was visiting professor at several universities. He was also part-time appointed at the University of Stellenbosch (South Africa, 1998-2004), department of Industrial Engineering. He was president (SOGESCI-B.V.W.B., 1998-1999) and member of the board of the Belgian OR-society until 2008. For five years, he was member of the board of the VIL (Flemish Institute for Logistics) and is still member of the board of the VLW (Transport and logistics workgroup). He is chairman of the Centre for Industrial Management, Traffic and Intelligent Transportation Systems, Department of Mechanical Engineering and programme director of the Master of Engineering: Traffic, Logistics and Intelligent Transportation Systems. He is especially interested in applications of operations research and the development of decision support systems in the area of logistics (production planning, distribution and reverse logistics).

P. Vansteenwegen has a master degree in mechanical engineering (KU Leuven, 2002) and a PhD in operations research (KU Leuven, 2008). In October 2010, he became an assistant professor at the department of industrial engineering at UGent and from October 2012, he is an assistant professor at the Centre of Industrial Management, Traffic and Infrastructure at KU Leuven (Belgium). His research aims at optimizing the operational planning for public transportation and developing metaheuristics for solving complex logistic problems. He is a member of the executive committee and secretary of ORBEL (Belgian Society of Operations Research) and member of the international Institute for Operations Research and the Management Sciences (INFORMS).