Real-time railway traffic management optimization and imperfect information: preliminary studies

(presented at the 6^{th} IESM Conference, October 2015, Seville, Spain) © I^4e^2 2015

Paola Pellegrini
Univ Lille Nord de France,
F-59000 Lille,
IFSTTAR, COSYS, LEOST,
F- 59650 Villeneuve dAscq
Email: paola.pellegrini@ifsttar.fr
Telephone: (+33) 320–438404 Fax: (+33) 320–438397

Grégory Marlière and Joaquin Rodriguez
Univ Lille Nord de France,
F-59000 Lille,
IFSTTAR, COSYS, ESTAS,
F- 59650 Villeneuve dAscq
Email: {gregory.marliere, joaquin.rodriguez}@ifsttar.fr

***Abstract*—Railway traffic is often perturbed by unexpected events and appropriate train routing and scheduling shall be applied to minimize delay propagation. A number algorithms for this routing and scheduling problem have been proposed in the literature and they have been tested in different traffic situations. Nonetheless, their performance are almost always studied considering perfect knowledge of future traffic conditions, which is almost impossible to achieve in reality. In this paper, we propose an experimental analysis assessing the usefulness of these algorithms in case of imperfect information. We consider RECIFE-MILP as a traffic management algorithm and advanced or delayed train entrance times in the control area as the source of imperfect information. The results show that the application of traffic management optimization allows outperforming the first-come-first-served management strategy even if the actual traffic conditions are not perfectly known by the optimization algorithm.**

I. Introduction

When railway traffic is perturbed by unexpected events, trains experience what is called primary delay. Primary delays may propagate with a snowball effect due to the emergence of conflicts: when a train is late, it may claim a track section in concurrence with another train and one of them must then slow down, or stop, to ensure safety. In this case, one of these trains will suffer a secondary delay due to the traffic perturbation.

Nowadays, operating to minimize delay propagation is the main task of railway dispatchers. They manually tackle conflicts by possibly re-scheduling trains, i.e., changing the planned train order at critical locations. Where alternative routes are available, also train routes may possibly be changed with respect to the planned ones (re-routing).

A noticeable number of academic studies have been devoted to finding effective algorithms for real-time railway traffic management [1], [2], [3], [4], [5], [6], [7], [8], [9]. These papers deal with this scheduling and routing problem based on different modeling choices as: the inclusion, limitation or exclusion of re-routing options; the consideration of train speed variation dynamics with different levels of simplification; the level of detail used for the infrastructure representation; the minimization of the total secondary delay, of the maximum one, or of the time to get back to the planned traffic situation. Among this works, we proposed a mixed-integer linear programming (MILP) formulation for solving to optimality the problem of routing and scheduling trains in case of railway traffic perturbation [7], [10], [11], [12], [13], [14]. Although very often this MILP formulation quickly finds the optimal solution to realistic instances, it fails sometime in delivering it within a computation time in line with real-time purposes. Hence, in the real-time applicable algorithm, we set a time limit to the MILP solution process so that when this limit is reached, the best solution found is returned. This algorithm is named RECIFE-MILP [15].

The great majority of the papers proposing and studying algorithms for real-time railway traffic management suppose a perfect knowledge of the initial perturbation to be tackled. If, for example, the perturbation studied consists in a set of trains entering the infrastructure considered late with respect to the timetable, then the entrance delays are supposed to be perfectly known in advance, when deciding on train re-routing and re-scheduling. However, this perfect knowledge is indeed an unrealistic assumption and the decisions made by the algorithms may have an unexpected impact on traffic. Along the just mentioned example, if a train A enters the considered infrastructure with a 5 min delay, it may be able to pass through a critical location before another train B without perturbing its trip, in case the headway time remains sufficiently large despite A's delay. The schedule imposing this order might be preferable (resulting in smaller delay propagation) with respect to the one allowing B passing first. In a real application, such a schedule would be implemented by constructing A's itinerary across the critical location first, and by constructing B's itinerary only after A has cleared the location itself. Nonetheless, if the expected delay is not fully precise and train A actually enters 5 min and 30 sec late, this schedule may result in a strong penalization of train B, which may have to stop and wait for A to pass. In this case, the assessment done by an optimization algorithm for selecting the preferable schedule would be based on an imprecise hypothesis (5 min delay rather than 5 min and 30 sec), and might then not be able to appropriately assess the possible alternatives.

Of course, in reality at least a slight deviation between train expected and actual delays is almost unavoidable. For coping with it, a possibility would be to include some techniques of robust optimization in the algorithms for real-time traffic management. However, such an inclusion would imply a strong increase in the complexity of the problem, which might be incompatible with real-time applications. An alternative possibility might consist in imposing to the algorithm the inclusion of additional buffer times between different train transits. However, too large buffer times might prevent from an effective exploitation of the infrastructure capacity. Indeed, the choice between these two possibilities must be done taking into account the trade-off between the potential improvement of the robustness of the solution thanks to the refinement of the model and its potential stronger sub-optimality due to the higher complexity of the model itself. However, before stepping into the unfolding of such a trade-off, it is necessary to understand how critical the robustness issue is. In other words, if the impact of the uncertainty on the quality of a real-time railway traffic management algorithm performance is high, then a deep investigation on the trade-off is a critical issue to be tackled urgently to preserve the usefulness of the algorithm. Otherwise, it can be seen as a refinement to be investigated with a lower priority. To the best of our knowledge, in the literature, only one paper [16] proposes an analysis of the impact of imperfect knowledge on algorithms decisions, where the imperfect knowledge is related to dwell times at station. This impact is assessed according to different metrics, mostly related to the stability of the algorithm decisions with respect to the progressive increase of the precision of its knowledge.

In this paper, we study the robustness of the re-routing and re-scheduling decisions made by RECIFE-MILP with respect to the uncertainty concerning the magnitude of the perturbations. In particular, we assess this robustness thanks to the microscopic railway simulator OpenTrack [17] on a section of line of 27 km around the station of Rouen-Rive-Droite, in France. Differently from [16], we assess the impact of the imperfect knowledge in terms of quality of the algorithm decisions, that is, in terms of total delay suffered by trains once these decisions are implemented. The imperfect knowledge concerns the train entrance delay in the infrastructure considered. Such an imperfect knowledge allows taking into account uncertainty sources as the the train driver behavior, which is actually impossible to predict with 100% precision in reality. Moreover, we consider this impact when changing the buffer time between trains.

Through this analysis, then, we aim to answer two research questions:

1) How robust is the improvement brought by the optimization with respect to other traffic management strategies when the perturbed situations tackled do not perfectly mirror the traffic evolution in terms of train entrance delays in the infrastructure considered?
2) Can we improve this robustness without increasing the computational burden due to the consideration of robust optimization techniques by varying the buffer times considered during the optimization?

To get to these answers, the rest of the paper is organized as follows. Section II details our approach for assessing the robustness of RECIFE-MILP decisions and Section III describes RECIFE-MILP. Sections IV and V present the experimental setup and the obtained results, respectively. Finally, Section VI reports the conclusions that can be drawn on account of these results and proposes some hints for future research.

II. Robustness assessment process

To assess the robustness of the decisions made by RECIFE-MILP we implement a two-step process.

For the explanation of this process, consider a generic instance I of the real-time railway management problem [7]. It represents the railway traffic situation to be tackled including a detailed description of the considered infrastructure and on the trains which will enter this infrastructure within a given time horizon. Let I_n be a realization of I when a noise factor of n seconds is applied: the train entrance times are perturbed, i.e., the trains are either advanced or delayed by adding to the intended entrance times a random number of seconds drawn from a uniform distribution in the interval $[-n, n]$.

In the first step of the robustness assessment process, we solve I through RECIFE-MILP. Let S be the solution obtained, i.e., the selected train routing and scheduling decisions.

In the second step, we use the microscopic railway simulator OpenTrack for quantifying the secondary delay resulting from the application of the decisions in S to the instance I_n: OpenTrack simulates the traffic managed by RECIFE-MILP in terms of routing and scheduling, but the trains enter the infrastructure in advance or delayed with respect to the instance which was actually tackled by RECIFE-MILP. In this way, we observe the performance of RECIFE-MILP when the hypothesis on which it based its decisions change, and we seek an answer to the first research question proposed in the introduction of this paper.

For interpreting the performance and hence assessing RECIFE-MILP robustness, we compare the total delay obtained applying this two-step process with the total delay obtained when the naive first-come-first-served rule (FCFS) is used for managing traffic.

Remark that this robustness assessment process is independent both from the algorithm used for solving I and from the simulator: any algorithm for the real-time railway traffic management problem and any railway microscopic simulator might be applied without impacting on the design or the applicability of the process.

III. RECIFE-MILP

The RECIFE-MILP heuristic consists in tackling a real-time railway traffic management problem instance by solving a MILP formulation for a limited computation time. If the optimal solution is found and proven before this computation time elapses, then the search process is stopped and the optimal solution is returned. Otherwise, the search process is interrupted after the available time has elapsed and the best solution identified is returned. For boosting the performance of the algorithm, a time of at most 30 seconds at the beginning of the search is devoted to a first-step optimization in which only the scheduling problem is tackled, i.e., the train routes are imposed to be those indicated in the timetable. If no feasible solution is found within 30 seconds, this first-step optimization continues until the detection of the first feasible solution. Although this possibility has to be accounted for in the design of the algorithm, we never encountered such a failure in finding at least a feasible solution to the pure scheduling problem within 30 seconds. The solution returned by the first-step optimization (the optimal one, the best one found within 30 seconds, or the first one found after 30 seconds) is used as starting solution for the second-step optimization in which the whole scheduling and routing problem is tackled.

In the MILP formulation at the basis of RECIFE-MILP, we model the infrastructure in terms of track-circuits, that is, into track sections on which the presence of a train is automatically detected. Sequences of track-circuits are grouped into block sections, which are opened by a signal indicating their availability. Before a train can enter (start the occupation of) a block section, all the track-circuits belonging to the same block section must be reserved for the train itself. The sum of reservation and occupation time is the utilization time. Without loss of generality, we consider the case of the signal opening the block section having three possible aspects (green, yellow and red). In the model, this translates into the need for the train to reserve two consecutive block sections before being allowed to enter the first of them. Any number of signal aspects can be considered by simply changing the number of consecutive block sections to be reserved in advance: being n the number of aspects, the number of block sections to be reserved is $n-1$. Moreover, each block section is reserved by the train some time before its entering, to allow the route formation, and it remains reserved after its leaving, to allow the route release. If additional buffer time is to be included between two trains traversing a same location, then such a buffer time is added to the route formation time.

Time-related variables are continuous (begin of the occupation of a track-circuit along a route, start and end of its utilization, and delay suffered there along a route), while the choice of the route to be used and the precedence relation between pairs of trains on track-circuits are represented through binary variables. These last variables ensure the disjunction between trains' track-circuit utilization through so-called big-M constraints.

The objective function which we consider in this study is the minimization of the sum of the delays suffered by all trains when they exit the control area considered.

For the details on the formulation, we refer the reader to our previous works [7], [10], [11], [12], [13], [14], [15].

IV. Experimental setup

The experimental analysis proposed in this paper is based on railway traffic management in the infrastructure representing a section of 27 km around the station of Rouen-Rive-Droite, in France. This section belongs to the line connecting Paris Saint Lazare station to Le Havre station, with its links to the port. The infrastructure is depicted in Figure 1. It includes six stations, with two to six platforms, and one bifurcation. The 190 track-circuits compose 189 block sections and 11347 routes. A one-day timetable includes 186 trains: 2 high-speed passenger trains, 107 conventional passenger trains, 33 freight trains and 44 local movements. The complexity of the instances representing traffic on this infrastructure is due to its large size, which implies the presence of a great amount of possible routes, and to the presence of mix traffic. The route formation and release times are 15 and 0 seconds for all block sections, respectively. The instances which we tackle in this paper include all the trains entering the control area between 7 and 8 am. In the instances, 20% of trains, randomly selected, suffer a random primary delay between 5 and 15 minutes [7]. By reiterating the random primary delay assignment we generate the 30 instances considered in the following experimental analysis. They include between 12 and 14 trains. Each train can use between 1 and 192 routes (mean 45), which translates into a MILP formulation with about 120000 continuous variables, 80000 binary variables and 370000 constraints. We run the experiments on an Intel(R) Xeon(R) CPU X5650 @ 2.67GHz, 12 cores, 24GB RAM, and we set the computation time available for RECIFE-MILP to three minutes [8]. The implementation is done using IBM ILOG CPLEX Concert Technology for C++ (IBM ILOG CPLEX 12.6, [18]).

For assessing the RECIFE-MILP robustness, we consider five different noise factors, i.e., 20, 40, 60, 80 and 100 seconds. So, for each instance I representing the peak-hour of a randomly perturbed one-day traffic, we obtain a realization I_{20} which includes the same set of trains as I, each with a slightly different entrance time in the infrastructure. The difference for each train is a random number of seconds drawn from the uniform distribution with extremes -20 and 20. The same holds for realizations I_{40}, I_{60}, I_{80} and I_{100}.

As mentioned in Section II, for answering to the first research question, which wonders whether it is appropriate to use RECIFE-MILP even in case of imperfect information on train entrance times in the infrastructure considered, we compare the total delay obtained when simulating through OpenTrack the RECIFE-MILP traffic management decisions and the FCFS traffic management strategy. However, in several instances, the application of this latter strategy results in the occurrence of deadlocks. In particular, the most critical areas are those right around Sotteville and Rouen-Rive-Droite stations. Around these stations a two-track line must cope with traffic in both directions. On the one hand, at Sotteville station, the trains performing local movements must enter and exit the depot located above the commercial platforms. Their routes, hence, often cross those of the trains performing commercial services on both tracks. On the other hand, at Rouen-Rive-Droite station, the trains are scheduled to use all the six platforms in both directions, sometimes due to constrains related to the train and platform lengths. Hence, the two tracks right around the station have to be used in both directions. Moreover, the trains traveling towards Sotteville (i.e., the trains traveling in the direction of Paris Saint Lazare) must cope with the trains traveling in the opposite direction coming from three different origins: Paris Saint Lazare station through Oissel station, the depot at Sotteville station and Serqueux station through Darnetal bifurcation. To be able to perform a comparison of the total delay obtained with the two traffic management strategies, we eliminate all the instances for which the application of the FCFS strategy results in one or more deadlocks. As an order of magnitude, this elimination concerns slightly more than half of the instances.

Finally, for answering to the research question which wonders whether the increase of buffer times might increase the robustness of RECIFE-MILP in case of imperfect information

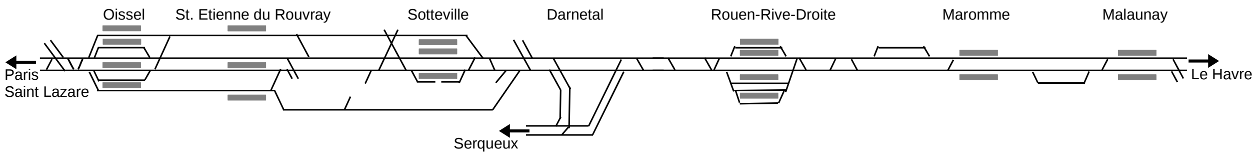


Figure 1. Representation of the Rouen infrastructure. The distances between stations are shrunk for ease of visualization.

Table I. MEAN DIFFERENCE BETWEEN THE TOTAL DELAY OF FCFS AND THE ONE OF RECIFE-MILP IN ITS THREE CONFIGURATIONS FOR THE MINIMUM HEADWAY TIME (SECONDS). WE INDICATE IN BOLD FONT THE BEST RESULT.

noise factor	RECIFE-MILP-h15	RECIFE-MILP-h25	RECIFE-MILP-h35
0	**124**	74	87
20	**136**	85	103
40	**179**	141	146
60	197	**218**	208
80	**214**	190	110
100	129	109	**133**

Table II. NUMBER OF INSTANCES IN WHICH FCFS AND RECIFE-MILP IN ITS THREE CONFIGURATIONS FOR THE MINIMUM HEADWAY TIME ACHIEVE THE BEST RESULT AND NUMBER OF INSTANCES ON WHICH THE COMPARISON IS BASED. WE INDICATE IN BOLD FONT THE BEST RESULT.

noise factor	# inst.	RECIFE-MILP-h15	RECIFE-MILP-h25	RECIFE-MILP-h35	FCFS
0	12	**8**	4	1	3
20	12	**8**	3	1	2
40	14	**5**	4	**5**	0
60	13	3	**7**	5	0
80	13	**6**	3	5	0
100	14	4	5	**6**	1

Table III. NUMBER OF INSTANCES IN WHICH FCFS AND RECIFE-MILP IN ITS THREE CONFIGURATIONS FOR THE MINIMUM HEADWAY TIME ACHIEVE THE WORST RESULT AND NUMBER OF INSTANCES ON WHICH THE COMPARISON IS BASED. WE INDICATE IN BOLD FONT THE BEST RESULT.

noise factor	# inst.	RECIFE-MILP-h15	RECIFE-MILP-h25	RECIFE-MILP-h35	FCFS
0	12	**0**	4	5	6
20	12	**1**	3	3	7
40	14	4	1	**0**	13
60	13	4	**0**	**0**	13
80	13	4	**1**	**1**	11
100	14	3	**1**	2	11

on train entrance times in the infrastructure considered, we varied the value of the route formation time for all block sections. We consider 15 seconds, which is the actual route formation time and is hence to be considered in case of perfect information, 25 and 35 seconds.

In the next section, we describe the results of these experiments. We refer to the results of the simulation applying the FCFS traffic management strategy simply as FCFS. Similarly, we will refer to the results of the simulation applying RECIFE-MILP traffic management decisions as RECIFE-MILP-h15, RECIFE-MILP-h25 and RECIFE-MILP-h35 for the solutions obtained when considering a buffer time of 15, 25 and 35 seconds, respectively.

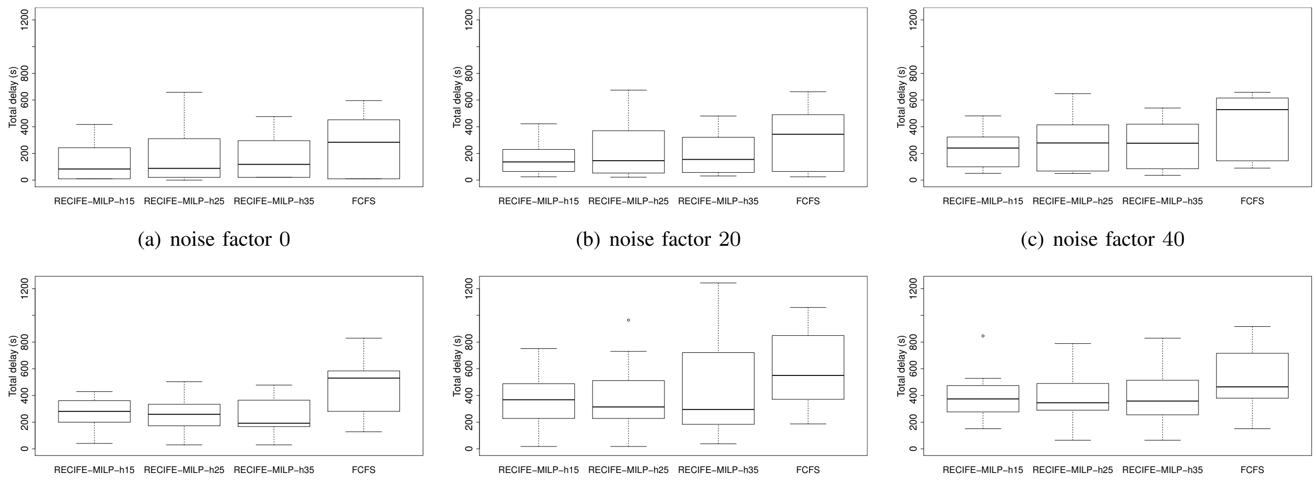
V. EXPERIMENTAL RESULTS

The experimental results of the analysis performed on the line around Rouen-Rive-Droite station show that RECIFE-MILP in general allows a better traffic management than the FCFS strategy. This is true even when the train entrance times in the infrastructure considered is not perfectly known when making scheduling and routing decisions.

Table I shows the mean difference between the performance of FCFS and the one of RECIFE-MILP in its three buffer-time configurations. This difference is computed as the total delay of the solution obtained when applying the FCFS minus the one obtained when applying RECIFE-MILP. We show the results separately for each noise factor tested. With a similar structure, Tables II and III report the number of instances on which the comparison is based for each noise factor, i.e., the number of instances on the original 30 for which FCFS did not end into deadlocks, and the number of instances in which each traffic management strategy obtains either the best (Table II) or the worst (Table III) result. The instances for which the best (or the worst) result is found by more than one strategy enter in the count for all of them.

A further representation of the results is depicted in Figure 2. This figure reports the boxplots of the distribution of total delay of the four traffic state management strategies which we consider in this paper, for each noise factor. In particular, each box represents the distribution of the total delay corresponding to the instances on which the comparison is made for each factor. For example, for factor 0 (Figure 2(a)), only 12 of the 30 original instances are shown, since FCFS ends up in deadlocks in the other 18 ones. In the figure, the horizontal line within each box represents the median of the distribution, while the extremes of the box represent the first and third quartiles, respectively; the whiskers show the smallest and the largest non-outliers in the data-set and dots correspond to the outliers.

The three tables show, first of all, that when the information considered by RECIFE-MILP for deciding the traffic management strategy is precise (noise factor 0), the best performance are obtained by using it with a buffer time of 15 seconds, i.e., the value actually imposed during the simulation. In this case, RECIFE-MILP-h15 allows the decrease of the mean total delay of 124 seconds with respect to FCFS, it is the best strategy in 8 over 12 instances, and it is never the worst one. In this case of perfect information, RECIFE-MILP improves over FCFS also when a higher buffer time is imposed, with a mean reduction of total delay of more than one minute, corresponding at least to a 28% improvement of the objective function value. This conclusion is supported also by Figure 2(a), where the box representing RECIFE-MILP-h15 is the one with the lowest quartile and whisker lines.



(a) noise factor 0 (b) noise factor 20 (c) noise factor 40

(d) noise factor 60 (e) noise factor 80 (f) noise factor 100

Figure 2. Boxplots of the distribution of total delay of the four traffic management strategies considered with the different noise factors.

When the noise increases, the results indicate similar qualitative conclusions for what concerns the comparison with FCFS. In particular, RECIFE-MILP is always better than FCFS in terms of mean total delay (Table I). However, the relative performance of RECIFE-MILP with the three buffer-time configurations are different for the different noise factors.

For a factor 20, i.e., close to perfect information, the actual value of 15 seconds (RECIFE-MILP-h15) is undoubtedly the best one from all the points of view considered. Figure 2(b) suggests, however, that the difference with respect to the other configurations is not as clear as in the case of perfect information: the first quartile of the distribution of total delay is slightly higher for RECIFE-MILP-h15 (67 seconds) than for RECIFE-MILP-h25 (62 seconds) and RECIFE-MILP-h35 (63 seconds). Nevertheless, the total delay distribution of RECIFE-MILP-h15 remains the best as a whole (median 137 seconds and third quartile 208 seconds compared to 146 and 367 of RECIFE-MILP-h25 and 155 and 301 of RECIFE-MILP-h35), and this allows this configuration to remain the best one.

For a noise factor 40, RECIFE-MILP-h15 starts suffering some performance issue. It is still the best configuration for mean difference of total delay (Table I) and for number of instances in which the best result is achieved (Table II). However, in the latter case, the best performance is shared with RECIFE-MILP-h35. Furthermore, according to Table III, it obtains the worst result in 4 out of 14 instances, while RECIFE-MILP-h25 is the worst in only one case and RECIFE-MILP-h35 never is. As shown in Figure 2(c), however, the median of the RECIFE-MILP-h15 distribution is still the lowest, as the line representing the third quartile.

Quite the same behavior is observable for noise factor 80, when RECIFE-MILP-h15 is the best configuration for Tables I and II, but not for Table III. Observing Figure 2(e), though, we can detect a qualitatively different behavior with respect to the previously mentioned case of noise factor 40: RECIFE-MILP-h15 keeps its status of best performing configuration mostly thanks to the upper part of the total delay distribution.

The situation is different for noise factor 60, when RECIFE-MILP-h25 is the best configuration according to all criteria. Figure 2(d) shows that, if the improvement with respect to RECIFE-MILP-h15 concerns the whole total delay distribution, the one with respect to RECIFE-MILP-h35 is due only to its upper part.

Almost the same relative behavior of the three configurations is observable for noise factor 100. Here, anyway, the better upper part of the total delay distribution of RECIFE-MILP-h25 (Figure 2(f)) does not allow this configuration to be the best: according to all criteria RECIFE-MILP-h35 achieves the best performance.

As a whole, these results suggest that it is not possible to detect a clear trend for identifying the best RECIFE-MILP configuration depending on the noise factor. In other words, in these experiments, a relation between the noise affecting the expected entrance time of the trains in the infrastructure and the buffer time considered in the optimization.

VI. Conclusions

In this paper we assessed the robustness of the real-time traffic management solutions found by RECIFE-MILP with respect to the uncertainty on train entrance times in the considered infrastructure. To assess this robustness, we implemented RECIFE-MILP's traffic management decisions in situations slightly different from the optimized ones, in terms of train entrance times. To this aim, we used the microscopic railway simulator OpenTrack. We repeated the same procedure while varying the buffer time considered by RECIFE-MILP, to assess the impact of this factor on the robustness.

The results of this analysis allow answering the two research questions which we proposed in the introduction of this paper:

1) The improvement of performance allowed by RECIFE-MILP is quite robust to the noise perturbing the traffic situation to be tackled, when this noise impacts the train entrance times in the infrastructure. In particular, in the great majority of the cases, it outperforms the FCFS.

2) This robustness does not seem to be improvable without using robust optimization techniques, which necessarily increase the computational burden of the algorithm. In fact, the augmentation of the buffer time considered by RECIFE-MILP does not have a clear positive impact on the results.

Hence, the lesson learned in this paper is that RECIFE-MILP is useful in practical railway traffic management, even if indeed reality is affected by some noise. However, how to effectively deal with this noise is not clear yet.

Future research will be devoted to the widening of the experimental analysis presented in this paper, and to the study of other types of perturbations. Moreover, we will extend this work by considering the optimization in a closed-loop framework, as tested in the European FP7 funded project ON-TIME [19]. In this framework, railway traffic is controlled by following a rolling horizon approach. In this approach, current traffic information is gathered from the field at regular time intervals. Based on this information and on the traffic management strategy being implemented, a short term prediction is made to identify train positions and speeds in the near future, for example two minutes ahead. The time lapse considered for the prediction is the prediction horizon. The prediction is used as input of an optimization run, to automatically compute a traffic management strategy for a given time interval in the future starting at the moment considered in the prediction. The time available for the optimization is set slightly shorter than the prediction horizon. The produced management strategy is then implemented into the field starting exactly at the moment which was considered in the prediction. At the same time, a new prediction is made and a new optimization run is started, so that a new traffic management strategy is computed regularly with a time step equal to the prediction horizon chosen. The new traffic management strategies provided subsequently consider the traffic evolution throughout the whole considered optimization period, a day for example, being able to revise previous decisions if unexpected conditions occur. Of course, the short term traffic prediction is crucial in this framework, although its perfect correspondence to reality is hardly possible. The robustness of the optimized traffic management strategy as studied in this paper is then particularly important. However, this importance might be reduced by the periodic revision of the traffic management strategy to be implemented. In future research we will work for understanding whether this reduction is sufficient for eliminating the need of using robust optimization techniques for effectively tackling the real-time railway traffic management problem.

REFERENCES

[1] R. Acuna-Agost, P. Michelon, D. Feillet, and S. Gueye, "A MIP-based local search method for the railway rescheduling problem," *Networks*, vol. 57, no. 1, pp. 69–86, 2011.

[2] G. Caimi, M. Fuchsberger, M. Laumanns, and M. Lüthi, "A model predictive control approach for discrete-time rescheduling in complex central railway station approach," *Computers & Operations Research*, vol. 39, pp. 2578–2593, 2012.

[3] A. D'Ariano, F. Corman, D. Pacciarelli, and M. Pranzo, "Reordering and local rerouting strategies to manage train traffic in real-time," *Transportation Science*, vol. 42, no. 4, pp. 405–419, 2008.

[4] M. Dessouky, Q. Lu, J. Zhao, and R. Leachman, "An exact solution procedure to determine the optimal dispatching times for complex rail networks," *IIE Transactions*, vol. 38, no. 2, pp. 141–152, 2006.

[5] R. Lusby, J. Larsen, M. Ehrgott, and D. Ryan, "A set packing inspired method for real-time junction train routing," *Computers & Operations Research*, vol. 40, no. 3, pp. 713–724, 2012.

[6] M. Mazzarello and E. Ottaviani, "A traffic management system for real-time traffic optimisation in railways," *Transportation Research Part B*, vol. 41, pp. 246–274, 2007.

[7] P. Pellegrini, G. Marlière, and J. Rodriguez, "Optimal train routing and scheduling for managing traffic perturbations in complex junctions," *Transportation Research Part B*, vol. 59, pp. 58–80, 2014.

[8] J. Rodriguez, "A constraint programming model for real-time train scheduling at junctions," *Transportation Research Part B*, vol. 41, pp. 231–245, 2007.

[9] J. Törnquist Krasemann, "Design of an effective algorithm for fast response to re-scheduling of railway traffic during disturbances," *Transportation Research Part C*, vol. 20, pp. 62–78, 2012.

[10] P. Pellegrini, G. Marlière, and J. Rodriguez, "Real Time Railway Traffic Management Modeling Track-Circuits," in *12th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, ser. OpenAccess Series in Informatics (OASIcs), D. D. and L. L., Eds., vol. 25. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2012, pp. 23–34.

[11] P. Pellegrini, G. Douchet, G. Marlière, and J. Rodriguez, "Real-time train routing and scheduling through mixed integer linear programming: Heuristic approach," in *IESM 2013, 5th International Conference on Industrial Engineering and Systems Management,Rabat, Morocco*, 2013.

[12] P. Pellegrini, G. Marlière, and J. Rodriguez, "Boosting the performance of a MILP formulation for railway traffic management in complex junctions," in *MT-ITS 2013, Models & Technologies for Intelligent Transportation Systems, Dresden, Germany*, 2013, pp. 419–428.

[13] ——, "A mixed-integer linear program for the real-time railway traffic management problem modeling track-circuits," in *RailCopenhagen 2013, 5th International Seminar on Railway Operations Modelling and Analysis, Copenhagen, Denmark*, 2013.

[14] ——, "Optimal train routing and scheduling in case of traffic perturbations: improving solution time through parameter tuning," in *TRA2014 Transport Research Arena 2014*, 2014.

[15] P. Pellegrini, G. Marlière, R. Pesenti, and J. Rodriguez, "Recife-milp: An effective milp-based heuristic for the real-time railway traffic management problem," *Intelligent Transportation Systems, IEEE Transactions on*, 2015, to appear.

[16] E. Quaglietta, F. Corman, and R. Goverde, "Stability analysis of railway dispatching plans in a stochastic and dynamic environment," *Journal of Rail Transport Planning & Management*, vol. 3, no. 4, pp. 137–149, 2013.

[17] A. Nash and D. Huerlimann, "Railroad simulation using OpenTrack," in *Computers in Railways IX*, C. Brebbia, J. Allan, G. Sciutto, and S. Scone, Eds. WIT Press, Southampton, United Kingdom, 2004, pp. 45–54.

[18] IBM Corporation, "User's manual for cplex," http://publib.boulder.ibm.com/infocenter/cosinfoc/v12r6, 2014.

[19] On-Time Consortium, "Optimal networks for train integration management across Europe – on-time," 2014, http://www.ontime-project.eu/.