SPEED REGULATION IN RAIL NETWORKS

Dario Pacciarelli [*,1] **Marco Pranzo** [**]

[*] *Dipartimento di Informatica e Automazione, Università degli Studi Roma Tre, via della vasca navale, 79 - 00146 Roma, Italy. pacciarelli@dia.uniroma3.it*
[**] *Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Siena, Via Roma, 56 - 53100 Siena, Italy. pranzo@dii.unisi.it*

Abstract: This paper deals with the real time problem of determining feasible speed profiles for a number of trains circulating in a given area, compliant with train dynamics and railway safety rules, such that each train is able to reach given points in the network at given times. The results described in this paper are part of a research project on train traffic control systems, supported by the European Commission. Other results of the project include the development of new optimization models and algorithms for traffic management, and a general architecture for train traffic control, capable of managing both fixed block and moving block signalling safety concepts. Computational results are reported, based on a portion of the Dutch railway network, on the highspeed line Paris-Brussels-Amsterdam.

Keywords: Railway traffic control, speed regulation.

1. INTRODUCTION

This paper deals with the results of a research project on train traffic control systems supported by the European Community, entitled Project No. TR4004 IV FP DG XIII Telematics, acronym COMBINE. Its goal is to analyse opportunities and problems for traffic management related to the introduction of the moving block signalling system ERTMS level 3. The results of the project include the development of a general architecture for a train traffic control system and new optimization models and algorithms for real time traffic management. In this paper we focus in particular on the speed optimization issue.

There are two different technologies to ensure safety in the railway networks: the *fixed block* technology and the *moving block* technology. Since there are many different national standards, in this paper we refer to the Dutch NS54 fixed block signalling and to the European standard ERTMS for the moving block technology. Advanced traffic management systems ensure safety of railway traffic by controlling trains speed. Safety standards impose a maximum speed for each train, called *movement authority*. It depends on the infrastructure, on the signal status and must be sufficient for completely blocking the train in case of emergency, within the space available to it. Under the fixed block technology each train can

[1] Partially supported by the European Commission, grant number TR 4004, project COMBINE (enhanced COntrol center for a Moving Block sIgNalling systEm)

book a number of subsequent segments of rail line, called block sections. A block section is assigned to a train if it is empty and not assigned to other trains, the block section is then released as soon the train traversed it completely. An emergency braking is imposed to a train every time its speed is not sufficient to stop earlier than the end of the last block section it is assigned to it. Under the moving block technology there are not block sections any more, but the space available to each train coincides with the distance from the preceding train.

In any case, stopping or slowing a train causes a remarkable loss of time and energy, due to the long braking distances, followed by acceleration of large masses. Therefore, effective Traffic Management Systems (TMS) should suggest to train drivers smooth speed profiles such that trains do not suffer too many speed variations, and such that emergency braking is avoided as much as possible. This is the task of the speed regulator of the COMBINE TMS.

The paper is organized as follows. In Section 2 the architecture of the COMBINE TMS is defined. In Section 3 we describe the solution procedures implemented by the Speed Regulation system. Section 4 illustrates the computational experiences based on the Breda triangle in the Dutch part of the high-speed line Paris-Brussels-Amsterdam. Conclusions follow in Section 5.

2. TRAFFIC MANAGEMENT SYSTEM ARCHITECTURE

In this section we describe the architecture of the Traffic Management System (TMS) developed in COMBINE project. At the highest hierarchical level (Figure 1) there is the human dispatcher in charge of controlling the rail network. The TMS is allowed to take minor decisions on train rescheduling, in order to maintain a conflict-free schedule, compatible with the real time situation. The human dispatcher is also able to undertake major changes in the timetable (such that cancelling a connection or changing the route of a train) which can be required in order to restore a feasible situation.

Given the current network status and the dispatcher suggestions, the aim of the Conflict Resolution system (CR) is to obtain in real time a conflict-free schedule, as close as possible to the planned timetable. The *alternative graph* formulation (Mascis and Pacciarelli, 2002) has been used within the COMBINE project to formulate all the relevant details of the conflict resolution problem, derived from the ERTMS concepts and from the timetable. The Conflict Resolution algorithm,

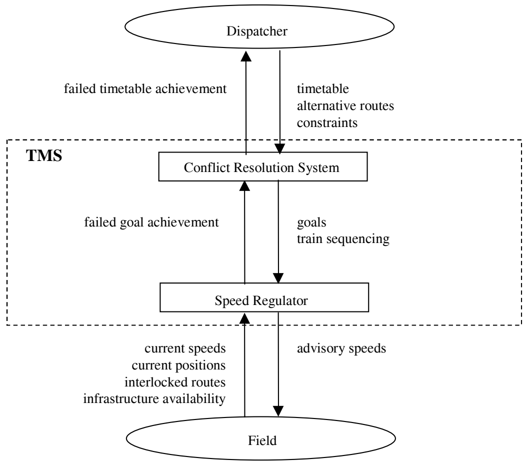

Fig. 1. The COMBINE TMS Architecture

described in (Mascis *et al.*, 2002), is based on the alternative graph formulation and produces as output a new conflict-free schedule (the *CR plan* in the following) whenever needed. The CR plan is a set of precedence constraints among trains and a set of *goals* for each train. A goal for a train specifies a relevant point along its path, and time windows of [minimum, maximum] arrival time and speed at the point. The CR plan is given as input to the Speed Regulator.

The Speed Regulator (SR) module is in charge for regulating the speed profile of each train in the network with the aim of respecting all goals and saving energy. Speed regulation is expected to become a significant aspect of traffic control under the moving block technology, whereas it is usually managed with simple static rules under the traditional fixed block technology. Finally, the output of the SR is sent to the field level.

In the COMBINE TMS the SR procedure is executed every time the rail network status is updated, whereas the CR is invoked, and a new feasible plan is obtained, only if the SR is not able to reach all the goals. If the CR is not able to respect all the planned timetable constraints then the help of the dispatcher is requested.

3. SPEED REGULATION

In this section we describe the Speed Regulator system. First, the relevant data to be exchanged by the CR and the SR are described, and then a detailed description of the SR algorithm is given.

At the start up of the process three look-up tables (Acceleration Table A, Braking Table B and Costing Table C) containing data on rolling stocks are loaded. In what follows we call $A(v)_S$ the space required by a train to reach its maximum velocity v_{max}, starting from speed v, and $A(v)_T$ the needed time to reach v_{max}, starting from v. Analogously, $B(v)_T$ and $B(v)_S$ are the minimum time and space needed by a train to stop, and $C(v)_T$ and $C(v)_S$ denote the time and space needed by a train to stop without braking, i.e., switching the engine off (slowing down without braking), starting from speed v. With this approach it is possibile to easily incorporate further aspect of running time calculation (not constant braking and acceleration curves, track gradients, additional curve resistance, train weight, etc.) by using more complex tables.

A train *goal* is associated to a relevant point of the rail network and contains the following information: the position, e.g. the end of the current resource, and the intervals [earliest, latest] time to reach the position and [minimum, maximum] speed at the goal position. More precisely, a train approaching point y has a goal (time, speed) = (t_y, v_y) which must be reached with a margin $(\pm\delta_t, \pm\delta_v)$. Reaching y at $(t_y + \delta_t, v_y - \delta_v)$ allows to reduce the energy consumption, but it may cause delays. Reaching y at $(t_y - \delta_t, v_y + \delta_v)$ allows to reduce the delays but causes an increasing of energy consumption.

The SR algorithm performs in sequence a safety check and a speed evaluation procedure. Every time a sequence is completed the resulting speed values and booking actions are sent to the field and a new sequence can start.

3.1 Safety Check

The safety check avoids that the underlying safety system takes the control of the train with undesired safety braking. This phase is performed differently in moving and fixed block technology. The *Movement Authority* (MA) of a train is the maximum speed allowed for the train. We also denote with MA_L the length of the Movement Authority, i.e. the space currently available to the train for completely stopping, and with $MA_D(v)$ the minimum distance necessary from the preceding train to stop, when the current train travels at speed v. Two different limitations on the maximum speed allowed to a train can be distinguished: a "static" limitation due to a route that has not been set yet and a "dynamic" limitation due to a preceding train having a smaller speed. In fixed block case the MA only depends on static limitations, while in the moving block case the MA depends on both static and dynamic limitations.

In moving block the check is performed by comparing the needed space to stop the train at the current speed v, i.e. $B(v)_S$, with the current distance from the end of the Movement Authority. In practice the Safety Check consists in checking whether $B(v)_S < MA_L + \sigma_1$ holds, where σ_1 is an internal technological parameter (Safety Parameter). If the inequality is FALSE then the train has two possibilities: (1) if the MA_L is due to a preceding train having smaller speed then the current train adapts its speed to the speed of the preceding train, (2) if the MA_L is due to a route that has not been set yet, then the current train starts stopping.

3.2 Speed Evaluation

The speed evaluation phase consists of two basic subtasks: the first one (*goal check*) has to be executed in all the cycles of the Speed Regulator algorithm, the second one (*speed analysis*) can be executed with a lower frequency, in order to match the strict time requirements of the SR.

3.2.1. Goal Check The goal check simply verifies if the train can reach the goal (without taking into account the position of the other trains). If the train cannot reach the goal the SR sends a warning to the CR. Let t be the current time, x the current position of a train, v_x its current speed, y the goal position of the train, t_y the goal time, and v_y the goal speed. We define the following quantities:

$$T(v_1, v_2) = \max\{A(v_1)_T - A(v_2)_T, \; B(v_1)_T - B(v_2)_T\} \quad (1)$$

$$S(v_1, v_2) = \max\{A(v_1)_S - A(v_2)_S, \; B(v_1)_S - B(v_2)_S\} \quad (2)$$

where $T(v_1, v_2)$ is the time needed to switch from v_1 to v_2, and $S(v_1, v_2)$ is the space needed to switch from v_1 to v_2.

The goal check consists in verifying if there exists an intermediate speed v_c such that the train is able to reach the goal position and speed within the time $(t_y - t)$ switching from the speed v_x to v_c and, finally, to v_y. Note that, v_c can be either greater, smaller or equal to v_x and v_y. Using the equations 1 and 2 we can define the residual time Δt and the residual distance Δx to be covered at constant speed v_c as follows:

$$\Delta t = t_y - t - T(v_x, v_c) - T(v_c, v_y) \quad (3)$$

$$\Delta x = y - x + \sigma_2 - S(v_x, v_c) - S(v_c, v_y) \quad (4)$$

The parameter σ_2 is an internal parameter of the TMS (Safety Parameter). The goal check for a train is therefore successful if the following system has a feasible solution:

Procedure Goal Check

(1) $v_c' := \min\{MA, v_{max}\}$, $v_c'' := \min\{v_x, v_y\}$.
(2) Compute Δt and Δx for $v_c := v_c'$.
if $(\Delta t \geq 0) AND (\Delta x \geq 0)$ **then** go to 3,
else go to 4.
(3) check inequality $v_c \Delta t \geq \Delta x$.
if it is verified, **then** check:=TRUE,
else check:=FALSE ($y - x + \sigma_2$ is too long),
exit .
(4) Compute Δt and Δx for $v_c := v_c''$.
if $(\Delta t \geq 0) AND (\Delta x \geq 0)$ **then** go to 5,
else check:=FALSE ($t_y - t$ or $y - x + \sigma_2$ is too short), **exit** .
(5) check inequality $v_c \Delta t \geq \Delta x$.
if it is verified, **then** check:=TRUE, **exit** ,
else $v_c'' := (v_c' + v_c'')/2$ and go to 6.
(6) **if** $v_c' - v_c'' < \epsilon$ **then** check:=FALSE, **exit** ,
else Compute Δt and Δx for $v_c := (v_c' + v_c'')/2$.
if $(\Delta t \geq 0) AND (\Delta x \geq 0)$ **then** go to 5,
else go to 7.
(7) $v_c' := (v_c' + v_c'')/2$ and go to 5.

Fig. 2. The Goal Check procedure.

$$\begin{cases} v_c \Delta t \geq \Delta x \\ v_c \leq \min\{MA, v_{max}\} \\ \Delta t \geq 0 \\ \Delta x \geq 0 \end{cases} \tag{5}$$

where MA is the maximum speed allowed by the infrastructure between x and y, and v_{max} is the maximum speed allowed to the train. The solution can be easily found by means of the algorithm of Figure 2 (where ϵ is a small positive constant).

3.2.2. Speed Analysis Let TR_i be the ordered list of trains currently running on resource R_i. With moving block technology, the Speed Profile procedure calculates the speed profile for all the trains in a branch, in order of arrival at the next goal, i.e. starting from the first one to reach the next goal. A speed profile is computed for this train in order to reach the goal y at (t_y, v_y). If the required speed exceeds MA (i.e., if the train is late), then a speed profile is computed in order to reach y at $(t_y + \delta_t, v_y + \delta_v)$. If the train is too early, then a speed profile is computed in order to reach y at $(t_y - \delta_t, v_y - \delta_v)$. This computation is repeated one train at the time, for all the following trains, checking the compatibility with the preceding train, for which a speed profile has been computed already. If it turns out that the h-th train cannot reach the goal $(t_y^h + \delta_t, v_y^h + \delta_v)$ due to the preceding train, then a speed profile is re-computed for the $(h-1)$-th train using the goal $(t_y^{h-1} + \delta_t, v_y^{h-1} + \delta_v)$. In other words, we press the $(h-1)$-th train travelling at the maximum speed compatible with its target in order to give more space to the h-th train. If some train cannot reach its goal, this means that either it is too late, or it is forced to arrive late at the target due to a set of consecutive trains that cannot further anticipate their arrival at the goal position. In this case no feasible solution exists in which all trains reach their respective goals, and a message is sent to the CR, which must compute a new timetable and a new set of goals.

In what follows, we describe how to compute a speed profile for each train and how to check that its movement authority constraints are always satisfied. The speed profile of a train follows the general scheme of Section 3.2.1: starting from the current position x and speed v_x a train first switches to a intermediate speed v_c, then moves at constant speed v_c and finally it switches to the final speed v_y at its target y. We use tables A, B, C introduced at the beginning of Section 3 to compute the speed profile of a train when switching from one speed to another. We denote with $MA(z, t)$ the Movement Authority Speed at time t in position z. Verifying that a train speed is always smaller or equal than $MA(z, t)$ is performed differently with moving and fixed block technology.

In the moving block case the Movement Authority constraint can be verified as follows, once the speed profile of the preceding train is known. In fact, each train at position $x + S(v_x, v_c)$ switches its speed from v_x to the value v_c and, at position $y - S(v_c, v_y)$ switches to the goal speed v_y. Let indicate the preceding train with the superscript $h-1$ and the current train with the superscript h. x^{h-1} and t_y^{h-1} are therefore the current goal position and time for the preceding train, respectively. Note that train h cannot reach the goal y before time $t_y^{h-1} + k$, where k is a time interval depending on the speed of train h in proximity of the goal position y. For example, we can assume:

$$k = MA_D(v_c^h)/v_c^h + \min\{0, d(v_c^h - v_c^{h-1}\} \tag{6}$$

where $MA_D(v_c^h)$ is the Movement Authority Distance between the trains $h-1$ and h (at the speed v_c^h), and d is a safety parameter acting only if train h is faster than train $h-1$. In fact, this is the only case in which we must pay attention to the Movement Authority: the parameter d ensures that train h does not reach train $h-1$ *before* the goal. Note that, if $t_y^h < t_y^{h-1} + k$, then train h is not be able to reach its goal. In this case the SR must re-calculate the speed profile of train $h-1$ for the new target $t_y^{h-1} - \delta_t$, and possibly propagate back the computation to other preceding trains.

In the fixed block case, it is necessary to verify that, once the speed profile of train $h-1$ is computed, train h never violates the Movement Authority, which may happens if two trains have

Procedure Speed Analysis (t_x, v_x, t_y, v_y)

(1) $v'_c := \min\{MA, v_{max}\}$, $v''_c := \min\{v_x, v_y\}$, check:=FALSE.
(2) Compute $\Delta^C t$ and $\Delta^C x$ for $v_c := v'_c$.
if $(\Delta^C t \geq 0) AND (\Delta^C x \geq 0)$ **then** go to 3, **else** go to 4.
(3) Check inequality $v_c \Delta^C t - \Delta^C x \geq 0$.
if it is verified **then** check:=TRUE, find v_c, feasible for 5, by means of a binary search in the interval $(0, v'_c)$,
else check:=FALSE ($y - x$ is too long), **exit** .
(4) Compute $\Delta^C t$ and $\Delta^C x$ for $v_c := v''_c$.
if $(\Delta^C t \geq 0) AND (\Delta^C x \geq 0)$ **then** go to 5, **else** check:=FALSE ($t_y - t$ or $y - x$ is too short), **exit** .
(5) $v''_c := 0$; find a feasible value of v_c in the interval (v''_c, v'_c) such that $0 \leq v_c \Delta^C t - \Delta^C x \leq \epsilon$, $(\Delta^C t \geq 0) AND (\Delta^C x \geq 0)$ by means of a binary search.
if $v_c \Delta^C t - \Delta^C x > \epsilon$ **then** $v'_c := (v'_c + v''_c)/2$
else $v''_c := (v'_c + v''_c)/2$.
if v_c has been found **then** check:=TRUE.
if $v'_c - v''_c < \epsilon$ **exit** (the check is negative).
(6) **if** check:=FALSE **then** try a new search by using the values $\Delta^B t$ and $\Delta^B x$, and repeat steps 1 to 5.

Fig. 3. The Speed Analysis procedure.

distance comparable with the length of some block section. Let us define the following quantities:

$$\hat{T}(v_1, v_2) = \max\{A(v_1)_T - A(v_2)_T, C(v_1)_T - C(v_2)_T\} \quad (7)$$

$$\hat{S}(v_1, v_2) = \max\{A(v_1)_S - A(v_2)_S, C(v_1)_S - C(v_2)_S\} \quad (8)$$

where $\hat{T}(v_1, v_2)$ and $\hat{S}(v_1, v_2)$ are the time and the space needed to switch from v_1 to v_2 without braking, respectively. The residual time and space to be covered at constant speed, can then be computed as follow in the cases of braking and costing, respectively:

$$\Delta^B t = t_y - t - T(v_x, v_c) \quad (9)$$

$$\Delta^B x = y - x - S(v_x, v_c) \quad (10)$$

$$\Delta^C t = t_y - t - \hat{T}(v_x, v_c) \quad (11)$$

$$\Delta^C x = y - x - \hat{S}(v_x, v_c) \quad (12)$$

Note that, $\Delta^C x \geq \Delta^B x$ and $\Delta^C t \geq \Delta^B t$ since $C(v)_S > B(v)_S$ and $C(v)_T > B(v)_T$ for all values of v.

Note that Procedure Speed Analysis (Figure 3) is executed at most three times for each pair (train, goal), and more precisely for the goals (t_y, v_y), $(t_y - \delta_t, v_y + \delta_v)$ and $(t_y + \delta_t, v_y - \delta_v)$.

4. COMPUTATIONAL EXPERIENCES

In this section, we report on our experience with a test site located in the Dutch part of the high-speed line Paris-Brussels-Amsterdam (Figure 4), called the Breda triangle. It is assumed that there are maximum speed limits but no power supply limitations. There are fast trains (TGV) running on the main line from Amsterdam to Brussels, and shuttle trains running from Rotterdam to Breda and from Brussels to Breda. The COMBINE TMS, which includes the speed regulator, has been tested by using a detailed rail simulator of the area. In this section we compare the COMBINE TMS performance with those of the First In - First Out dispatching rule. We report, in particular, on two sets of tests. In the first set of experiments the timetables have been defined with trains circulating at maximum allowed speeds, and no significant margins are planned in the timetables to recover entry delays. in the second set of experiments the timetables include some buffer time, i.e. trains are planned to travel at slightly less than their maximum speeds.

4.1 Tests with Maximum Speed Timetable

For evaluation purposes, several traffic conditions have been considered: Normal Traffic (NT) representing the traffic planned over the high speed line for year 2015, Heavy Traffic (HT) as the Normal Traffic but with more Shuttles, Extreme Traffic (ET) as the Heavy Traffic but with additional national traffic on the secondary lines between Rotterdam and Breda. For each traffic condition, three disturbance scenarios have been considered, namely small stochastic disturbances, large stochastic disturbances, and small stochastic disturbances, and large deterministic disturbance. In order to collect sufficient data for a statistically sound analysis, each test consisted of 4 replications of 5 consecutive hours, where the first hour (warm-up time) has been discarded in the analysis of the results.

The results for these tests are summarized in Table 1, where train delays and energy consumption for each traffic condition are reported. For sake of readability, the "total tardiness" columns show the sum of the exit delays as percentage normalized to the FIFO case. Similarly, the "energy consumption" columns express the energy consumption as percentage normalized to the FIFO case. In this case, the exit delays with COMBINE TMS are slightly larger than in the FIFO case, while the energy consumption is slightly smaller. This is due to the fact that the COMBINE TMS aims at reducing both delays and energy consumption, and since there is no possibility to recover delays, the FIFO rule provides the best solution in terms of

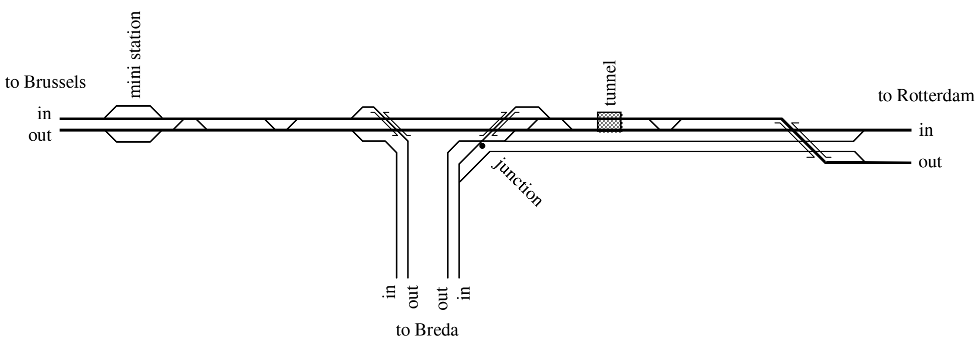

Fig. 4. The test site (Breda junction)

	Normal Traffic		Heavy Traffic		Extreme Traffic	
	Total Tardiness	Energy Consumption	Total Tardiness	Energy Consumption	Total Tardiness	Energy Consumption
FIFO	100%	100%	100%	100%	100%	100%
TMS	111.1%	99.5%	116.1%	98.3%	110.2%	98.3%

Table 1. Tests with maximum speed timetable.

punctuality while the COMBINE TMS provides the best solution in terms of energy consumption.

4.2 Tests with less than Maximum Speed Timetable

Two different test cases are analyzed in this case, a hindering and a convergence conflict. Also in this case, we compare the performance of the COMBINE TMS versus the FIFO control strategy. Each test consists of 4 replications of 5 consecutive hours.

(**Hindering conflict.**) A Shuttle from Belgium to Breda enters the control area with large delays, thus hindering a TGV from Belgium to Rotterdam. With the FIFO rule, the Shuttle hinders the TGV until the former leaves the high-speed line. This turns out into significant delays for the TGV, whereas the Shuttle is able to recover most of its initial delay. The TMS in this case re-routes the TGV through the secondary line of the mini-station in order to overtake the Shuttle, which is slowed down below its maximum speed allowed in the station. So doing, the TGV can leave the control area on schedule with a smaller delay.

(**Convergence conflict.**) In the second test case, a convergence conflict arises between a TGV from Rotterdam to Belgium, which enters the control area with large delay, and a Shuttle running from Breda to Belgium. Hence, a convergence conflict arises between the two trains when joining the high-speed line. With the FIFO case the Shuttle runs at the planned speed, thus approaching the convergence point before the delayed TGV. The TGV is hindered by the Shuttle in this case, and its exit delay is larger then the entry one. With the COMBINE TMS, the Shuttle is slowed down before the convergence point, so that it joins the high-speed line just behind the delayed TGV. Then, both the Shuttle and the TGV speeds are regulated to their maximum value, so that both can recover their initial delays.

	Hindering Conflict		Convergence Conflict	
	Total Tardiness	Energy Consumption	Total Tardiness	Energy Consumption
FIFO	100%	100%	100%	100%
TMS	79.5%	89.1%	78.4%	90.8%

Table 2. Tests with less than Maximum Speed Timetable.

With this second set of tests (Table 2) the COMBINE TMS performs significantly better than the FIFO rule, for both punctuality (more than 20% of improvement) and energy saving (up to 10% of reduction) indicators.

5. CONCLUSIONS

In this paper we discussed a Speed Regulator system developed within the COMBINE project. Performance tests aimed at showing whether advanced optimization algorithms are useful to manage railway traffic. Results showed that the COMBINE TMS provides valuable advantages in terms of punctuality and energy consumption if suitable buffer times are included in the train timetables. Besides this fact, the speed regulator is always able to control speeds effectively.

REFERENCES

Mascis, A. and D. Pacciarelli (2002). Job shop scheduling with blocking and no-wait constraints. *European Journal of Operational Research* **143**, 498–517.

Mascis, A., D. Pacciarelli and M. Pranzo (2002). Models and algorithms for traffic management of rail networks. Technical Report DIA-74-2002. Dipartimento di Informatica e Automazione, Università Roma Tre.