A Capacity Test for Shunting Movements

John van den Broek[1] and Leo Kroon[2]

[1] Dept. of Mathematics and Computer Science,
Eindhoven University of Technology,
NS Reizigers, Utrecht,
The Netherlands
j.j.j.v.d.broek@tue.nl
[2] Rotterdam School of Management,
Erasmus University Rotterdam,
NS Reizigers, Utrecht,
The Netherlands
L.Kroon@rsm.nl

Abstract. One of the bottlenecks in the logistic planning process at Netherlands Railways is the capacity of the infrastructure at the larger railway stations. To provide passenger trains with the right composition of rolling stock, many shunting movements between platform tracks and shunting areas are necessary, especially just before and after the peak hours. These shunting movements use the same infrastructure as the timetabled passenger and cargo trains.

In this paper we describe a capacity test that has been developed to test at any moment during the planning process, whether the capacity of the infrastructure between the platform tracks and the shunting areas is sufficient for facilitating all the shunting movements that have to be planned in between the already timetabled train movements. With this test it is not necessary anymore to plan every detail of the shunting movements far before the actual operations.

The capacity test is based on a mixed integer programming model. The running time of the Branch-and-Bound algorithm of CPLEX 9.0 is sufficiently small, as was observed in computational experiments related to three stations in the Netherlands.

1 Introduction

In this paper we focus on shunting processes related to passenger trains. Shunting processes belong to the backstage processes in a railway system. They are carried out in and around the large stations in a railway network in order to provide passenger trains with the right composition of rolling stock, and to facilitate the inside and outside cleaning and the short term maintenance of the rolling stock. During the rush hours, passenger trains are usually operated at full capacity. However, outside the rush hours, an operator of passenger trains usually has a surplus of rolling stock. This surplus has to be parked at a shunting area in order to be able to fully exploit the main railway infrastructure.

F. Geraets et al. (Eds.): Railway Optimization 2004, LNCS 4359, pp. 108–125, 2007.

At the larger railway stations, timetable related shunting movements between platform tracks and shunting areas are necessary in the following cases:

1. Extending a train. Rolling stock is added to a passenger train in order to increase the train's capacity. A shunting movement is necessary to bring the rolling stock from a shunting area to a platform track.
2. Shortening a train. Rolling stock is uncoupled from an arriving passenger train. A shunting movement is necessary to bring the uncoupled rolling stock from a platform track to a shunting area.
3. Starting trains. These are the first trains in the morning that have to depart from a station. To provide these trains with rolling stock, a shunting movement from a shunting area to a platform track is necessary.
4. Ending trains. These are trains which arrive at a station and, after arrival, the rolling stock of these trains is not used anymore on the same day. The rolling stock has to be brought from a platform track to a shunting area.

Besides the timetable related shunting movements, also many shunting movements related to the cleaning and maintenance of the rolling stock have to be carried out. However, these shunting movements usually take place at the shunting areas themselves, without too much interaction with the timetabled trains. We only have to consider these shunting movements if they use infrastructure outside the shunting areas.

Shunting processes involve highly complex routing and scheduling problems with capacity restrictions on time, space, and personnel. Especially the limited capacity of time and space (routing and storage) leads to several bottlenecks in the railway process. The routing and scheduling aspects of the shunting processes are strongly interrelated. Usually, the capacity of the required shunting crew is less a bottleneck, since personnel is a relatively cheap resource.

For each shunting movement, an appropriate route over the railway infrastructure and an appropriate time instant have to be determined. This is particularly relevant for the shunting movements between the shunting area and the platform area of a station. These shunting movements have to take place between the timetabled passenger and cargo trains. They should not disturb these trains. From a robustness point of view, it is desirable that the shunting movements are scheduled as far (in time and space) as possible from the train movements of the passenger and cargo trains.

Shunting processes are highly dependent on the timetable and on the rolling stock circulation of a railway operator. First, as was indicated already, the shunting movements share the capacity of the railway infrastructure with the timetabled passenger and cargo trains. Moreover, as soon as the planned timetable or the planned rolling stock circulation is modified, the number, the order, and the compositions of the shunting movements usually change as well. Therefore also the shunting plans have to be modified in such cases. Currently this requires a lot of manual planning work.

In the current planning process, every detail of the shunting movements is planned as soon as possible, sometimes months before the actual operations. Adding one train to a plan or a small change in the rolling stock circulation may result in many changes in the original shunting plans of a number of railway stations, which means a lot of replanning. The time spent on detail planning has to be reduced by postponing the detail planning. The only reason to plan every detail of the shunting movements far before actual operations, is to be certain that the capacity of the infrastructure is sufficient.

Our first contribution is the recognition that in practice it is useful to make a distinction between a capacity test to be carried out a relatively long time before the operations, and a planning tool to be used for finalizing the detailed plans briefly before the operations. The main contribution is the capacity test. It verifies whether the capacity of the infrastructure between the shunting areas and platform tracks is sufficient. Two mixed integer programming models have been developed which both minimize the number of shunting movements that can not be planned because of lack of capacity of the infrastructure. In the first model the routes of the trains are fixed beforehand and in the second model the routes are determined by the model. The disadvantage of the last model is the larger computation time. If our models indicate that there is sufficient capacity, then it is rather sure that there is sufficient capacity in practice. If our models indicate that the capacity is insufficient for some shunting movements, a detailed plan has to be made at that moment in time and it could be that the rolling stock circulation has to be adjusted.

The rest of the paper is structured in the following way. In Section 2 we give a literature review of research related to shunting processes. Section 3 contains a detailed description of the problem and the goal is explained in more detail. The problem is formulated as a mixed integer program which is described in Section 4. Section 5 contains computational results for a few railway stations in the Netherlands. Some conclusions are given in Section 6.

2 Literature Review

A prototype model for a capacity test as described in the previous section has been developed in earlier research by Van den Broek [1]. This model assumes that the routes for all the shunting movements are fixed beforehand and verifies that each shunting movement can be scheduled at such a time instant that each element of the infrastructure is occupied by at most one movement at the same time. The model is a mixed integer program that is solved by CPLEX.

Research that was carried out by Duinkerken [8] deals with the storage capacity of a shunting area. This research provided a prototype of a tool to determine whether the storage capacity of a shunting area is sufficient for storing a certain set of train units. Duinkerken describes an integer linear program that is solved by CPLEX. This model does not only take into account the total number of rolling stock units that have to be stored concurrently at the shunting area, but also the arrival and departure times of these rolling stock units.

Other research related to shunting processes was carried out by Freling et al. [10]. Their research aims at the development of planning tools that support planners to generate detailed shunting plans from scratch. This is in contrast with our model, which focuses on the development of a global capacity test for the mid term planning. Freling et al. take into account many small details. They split the shunting problem into a matching problem for arriving and departing train units, a routing problem for routing train units to the shunting tracks, and a parking problem for storing the train units at the shunting tracks. The first step is solved by CPLEX, the second step by applying column generation to a set covering model, and the last step by applying A^* search.

Also Di Stefano and Koci [9] did research related to shunting processes. They looked at how to order trains on the available shunting tracks in order to minimize the number of required shunting movements on the next morning. They assume that each track is long enough to host the trains assigned to it. Their main objective is minimizing the number of required shunting tracks. They consider several variants of their shunting problems, distinguished from each other by the ends of the shunting tracks that can be used for entering or leaving these tracks. For example, in the SISO-variant (Single Input Single Output), all trains enter the shunting area from one end of the tracks and all trains leave the shunting area into one end of the tracks. For several variants of their problem they provide computational complexity results.

Next, Tomii et al. [12] and Tomii and Zhou [13] describe a genetic algorithm that handles both storage of train units and several related processes, such as cleaning and maintenance. However, the shunting part of their problem is of a less complex nature than the general shunting problem, since in their context at most one train unit can be parked on each shunting track at the same time.

Papers on shunting trams and buses in their storage depots have been written by Winter and Zimmermann [14], Blasum et al. [3], Di Miele and Gallo [7], and Hamdouni et al. [11]. Winter and Zimmermann [14] focus on storage areas in which the trams are stored one behind the other in dead-end sidings. They assume that the earliest departure takes place after the last arrival. They also describe real-time dispatch strategies. Their model assigns trams to depot positions, thereby minimizing the number of necessary shunting movements. Blasum et al. [3] study similar problems, especially focusing on a smooth start-up process of the tram system in the early morning. Gallo and Di Miele [7] describe a model for parking buses in a storage area based on Minimal Non-Crossing Matching and Generalized Assignment. This model also takes into account the fact that the vehicles have different lengths. Moreover, they present an approach for dealing with mixed arrivals and departures. That is, the earliest departure takes place before the last arrival. Another application of bus dispatching is described by Hamdouni et al. [11]. Here robust solutions are emphasized by having as little different bus types as possible in each lane of the depot, and by grouping in each lane the buses of the same type as much as possible together.

The shunting problem has some similarity with the problem of routing trains through railway stations that is described by Zwaneveld et al. [15], [16]. In this problem, the arrival and departure times of a set of trains at a certain railway station are given, and the question is whether the trains can be routed through the railway station in such a way that trains do not conflict with each other.

Zwaneveld et al. try to assign trains to platforms and to minimize the number of necessary shunting movements. The routing problem is proved to be NP-hard and modeled as a weighted node packing problem. A Branch-and-Cut algorithm has been developed to solve the problem. Main difference with our capacity test is that they don't verify whether the capacity of the infrastructure between the shunting area and the platform area is sufficient to carry out the shunting movements. A similar problem is studied by Billionnet [2].

Cornelsen and Di Stefano [6] also look at assigning trains to platforms given a timetable. They model the platform assignment problem as a graph coloring problem on a conflict graph. The vertices of the graph represent the trains and two vertices are adjacent if the corresponding trains cannot be assigned to the same platform due to their arrival and departure times. Cornelsen and Di Stefano consider the platform assignment problem both on a linear time axis and on a cyclic time axis. The main difference with the capacity test and the model of Zwaneveld et al. [16] is that Cornelsen and Di Stefano don't take into account any shunting movement nor the capacity of the switch zone. They distinguish between variants with and without the so-called midnight constraint. The midnight constraint means that the earliest departure takes place after the last arrival. They present several complexity results and approximation methods.

Note that, apart from research on shunting processes for vehicles for passenger transport (trains, trams, and buses), there is a lot of research going on related to shunting processes for cargo trains. Since shunting processes for cargo trains are usually carried out at dedicated locations, they fall outside the scope of the current paper. A recent overview of the use of Operations Research in railway systems focusing on train routing and scheduling problems is provided by Cordeau et al [5], also focusing on shunting problems related to cargo trains. Bussieck, Winter and Zimmermann [4] give a survey of the application of discrete optimization techniques in public rail transport.

3 Problem Description

To verify whether the capacity of the infrastructure is sufficient to facilitate all the shunting movements between the platform area of a certain station and the corresponding shunting area, a global capacity test is needed. The capacity of the infrastructure between those areas has to be shared by passenger trains, cargo trains and shunting movements. The test has to check whether it is still possible to schedule and route the shunting movements between the passenger and cargo trains. This section gives a formal description of this capacity test.

In the *planning* process, each train movement has a unique time instant which corresponds with an event on the platform tracks, and which is called the *plan time* of the movement. The plan time of an arriving passenger train corresponds with the arrival time and the plan time of a departing passenger train is equal to the departure time of that train. For a cargo train, the plan time is the time instant at which the train passes a platform track. The plan time of a shunting movement corresponds with the arrival or departure on a platform track.

In the planning process, each shunting movement has a given departure and arrival track as input. These tracks can be a shunting area, a shunting track or a platform track. Depending on the infrastructure of a railway station, a shunting movement can be routed along several possible routes to get from its departure track to its arrival track. These routes differ in the used tracks, switches and crosses. The possible routes between a pair of tracks consist of one *priority route* and a maximum of nine possible *alternative routes*.

Moreover, each shunting movement has a feasible *time window* which contains the allowed plan times of the shunting movement. This time window is given by an earliest and a latest possible plan time which are based on the availability of railway tracks, platform tracks, and shunting tracks. For example, a shunting movement bringing empty rolling stock from a platform track to a shunting area cannot start before the passengers got out after the arrival of the rolling stock on the platform track and has to be completed before the next train arrives at the same platform track.

Passenger and cargo trains are *planned in detail* far before the operations. Planned in detail means that the arrival- and departure tracks, the route and the plan time of a movement are fixed. Shunting movements are train movements with a relatively low priority. Therefore they are preferably planned in detail only briefly before the operations. Indeed, otherwise there is the risk that they have to be replanned several times, e.g. due to a modified rolling stock circulation or due to the fact that additional passenger or cargo trains have to be facilitated on the infrastructure. However, the current practice is that the shunting movements are also planned far before the operations. In fact, the shunting plans themselves serve as a capacity check for the capacity of the infrastructure of the stations. This current practice is due to the fact that creating shunting plans is a difficult problem, and that intelligent support is currently lacking.

Planners build some robustness into the plans by taking into account a certain *headway time* between each pair of train movements. Therefore at least a certain minimum number of minutes has to be scheduled between the plan times of two consecutive movements that use a common element of the infrastructure. This minimum amount of time is given by the planning norms. These norms depend for example on whether trains are cargo trains or passenger trains, or whether trains are arriving or departing trains.

Saw movements are movements that arise when the arrival track of a shunting movement can not be reached from the departure track by one forward movement, see Figure 1. Most of these saw movements arise when the arrival and departure track are parallel to each other or when at least one of the two tracks is a track which can only be approached from one side. Rolling stock that carries out a saw movement has to change direction on a track in between. In Figure 1 such a track can be found at ②. Such a track is called a *saw track*. At a saw track the driver has to walk from one side of the train to the other side. This means that the plan times of the two parts of a saw movement have to be separated in time to give the driver the opportunity to walk to the other side of the train and results in the occupation of the saw track for a number of minutes. Usually, there are several saw tracks to choose from. Which saw track is chosen depends on the other train movements that use the possible saw tracks. However, in the current paper it is assumed that for every saw movement, the saw track is given a priori.

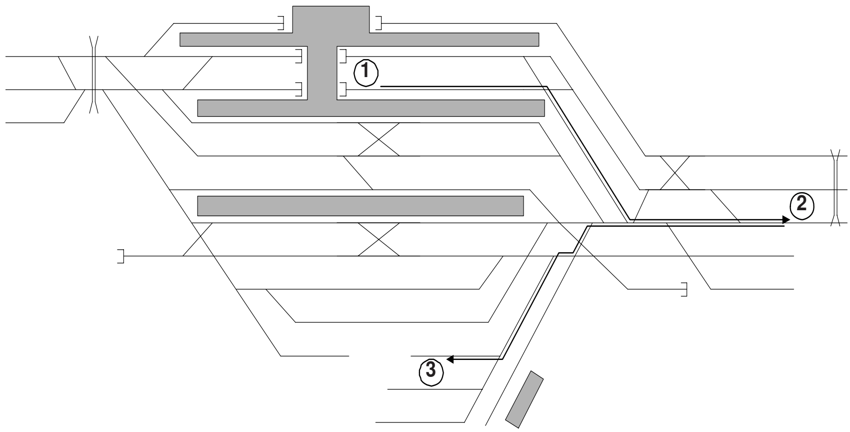

Fig. 1. Example of a saw movement

As was indicated before, shunting movements depend on the timetable and on the rolling stock circulation. So the timetable and the rolling stock circulation are assumed to be known when the capacity test is applied. This means also that the length of each train is known, which makes it possible to verify whether the rolling stock fits on a certain platform track. This is done before applying the capacity test. The minimum amount of time between the plan times of the two parts of a saw movement depends on the length of the train. Since the length of each train is assumed to be known, also this minimum amount of time between those plan times can be determined a priori.

The global capacity test has to indicate whether it is still possible to find for each shunting movement an appropriate route from its given departure track to its given arrival track within its given time window which does not conflict with the train movements which are already planned in detail nor with each other. The test is not allowed to change the arrival tracks, the departure tracks, and the plan times of the passenger and cargo trains. The test has to be used at the moment that the shunting movements between the platform area and the shunting area of the station still have to be scheduled and routed, which is *after* the timetabled trains have been scheduled and routed through the station.

During day time a large part of the rolling stock serves as a passenger train and will not be parked at a shunting area. This means that the capacity of the shunting areas will be sufficient during day time and only has to be verified during night. Therefore, it is assumed that the capacity and the detailed layout of a shunting area are not relevant for the capacity test. The capacity test focuses on the capacity of the infrastructure *between* the platform area of a station and the shunting area, not on the shunting area itself. Because the rolling stock circulation is known, it has been verified already whether the capacity of the shunting area is sufficient. This can be done e.g. by applying the model described by Duinkerken [8]. As a consequence, when our capacity test is applied, the shunting area can be seen as a set of tracks with sufficient capacity.

It is also not necessary to take into account the crew planning. In comparison with the infrastructure, crew is a relatively cheap resource and there is usually a sufficient number of train drivers available to carry out the shunting movements. Therefore, the details of the crew planning are skipped in this paper.

Note that the capacity test is *not* intended to be a detailed planning tool. As will be explained later, the constraints taken into account by the capacity test are somewhat stricter than the constraints taken into account in practice. As a consequence, if the result of the test is that the capacity of the infrastructure is sufficient, then it can be expected that the capacity is indeed sufficient in practice. If there is still a sufficient amount of time and space for each shunting movement, then no specific action of the planners is required: detailed planning of the shunting movements can be postponed. On the other hand, if the capacity test gives as a result that the capacity is not sufficient, then appropriate actions of the planners are required. In such cases, some shunting movements are critical and have to be planned in detail.

4 The Mathematical Programming Model

In this section we describe two mixed integer programming models that have been developed for the global capacity test. The models check whether it is possible to plan all the shunting movements, which have not been planned in detail yet, between the movements already planned in detail. In the first model, the route of each shunting movement is given and the plantime is to be selected. The second model is an extension of the first one. In the second model it is also allowed to select the route of each train movement from a pre-specified set.

4.1 Parameters of Model with Fixed Routes (MFR)

In both models, a train movement has to arrive at or depart from a platform track or a track parallel to the platform tracks. For example, a cargo train that doesn't stop at a station is split into two movements. The first movement arrives at a platform track and the second movement departs from this track and has an arrival track that is equal to the arrival track of the original movement. The set of movements is described by S.

At the moment of testing, some movements have been planned in detail and some have not been planned in detail. As a consequence, a set S_p of planned movements and a set S_n of not yet planned movements is introduced. The set S_p contains all the timetabled trains, the cargo trains and possibly the already planned shunting movements. The set S_n contains the shunting movements which have not been planned in detail yet.

Trains can arrive and depart from more than one track. For example: a long passenger train can arrive at platform tracks 7A, 7B and 7C. Also, several movements use more tracks than just their arrival and departure tracks. These other tracks are part of the route and have to be empty when the movement is carried out. The set T_j is defined as the set of tracks which are used by movement j along the given route of movement j. This set contains at least the arrival and departure tracks of movement j.

The plan time of a movement is defined as the number of minutes after the starting time of the planning horizon, thereby assuming that time zero corresponds with this starting time. For every movement j, the release date r_j is the first possible time instant that is allowed to be the plan time of movement j. The due date d_j of movement j is the last possible time instant that is allowed to be the plan time of movement j. For a movement $j \in S_p$, the release date and the due date are equal to the already determined plan time. Movement $j \in S_n$ has a time window $[r_j, d_j]$, which contains all the possible time instants that are allowed to be the plan time of movement j.

To check that all tracks on the route of a certain movement are not occupied, the concept of the *successor* of a movement is introduced as follows:

Definition 1. *The successor of movement j with respect to track t is the next movement after movement j that must use track t after movement j has arrived at track t or after it has left track t.*

Every other movement that uses track t has to be carried out before movement j or after the successor of movement j. Therefore no other movement is allowed to use track t after movement j as long as the successor of j has not been carried out. For example: if an ending train arrives at a platform, the arriving movement has as its successor the shunting movement that brings the rolling stock to a shunting area. As long as the shunting movement has not been carried out, no other movement is allowed to use the platform track. If a track is a shunting area or empty after movement j, then movement j has no successor with respect to this track. The set s_j of successors of movement j can be deduced from the input data. Set s_j^t is defined as the set of successors of movement j with respect to track t.

Trains that do not stop at a station are separated in an arriving movement and a departing movement. The departing movement is the successor of the arriving movement with respect to an appropriate track.

In the model, saw movements are split into two or even more movements. The first movement is the movement from the departure track to the saw track, and the second one departs from the saw track and arrives at the arrival track of the original movement. The second part is the successor of the first part.

Two movements are defined to be *route dependent* if they have an element of the infrastructure in common in their routes. Such an infra-element can be a track, a switch or a crossing. If two movements have no infra-element in common, then they do not directly influence each other's plan time. The plan times of two route dependent movements have to be separated by the headway time of which the value is given by the planning norms. The parameter b_{jk} is defined as the minimum amount of time between the plan times of the route dependent movements j and k if movement j is carried out before movement k. The transfer time of a driver and reversing the direction of the rolling stock are also covered by this parameter if movements j and k together form a saw movement.

Obviously, a movement which has a time window around 6:00 am does not influence a shunting movement which has a time window around 9:00 pm. If two movements have time windows that are in time far from each other, then the order of operation of the movements is known a priori. On the other hand, two movements are *time dependent* if their time windows differ less than the norm between those two movements. This means that movements j and k are time dependent if and only if $r_j < d_k + b_{kj}$ and $d_j > r_k - b_{jk}$. If two movements are route dependent and time dependent, then they are *dependent*. Summarizing, the definition of dependency of two movements is as follows:

Definition 2. *Two movements are dependent if they have an element of the infrastructure in common in their routes and the order of operation of the two movements is unknown a priori. Two movements are independent if one of the two conditions is not satisfied.*

Now assume that movement k is the successor of movement j and that movement m uses their common track. To avoid that movement m is scheduled over their common track between movements j and k, movement m is dependent of the movements j and k if $d_m + b_{mj} > r_j$ and $r_m < d_k + b_{km}$. If movements j, k and m are not fulfilling these constraints, then it is known a priori whether m is carried out before j or after k. Hence, the order of operation of the three movements is known a priori and hence the movements are independent.

In the model the following parameter is used to indicate whether two movements j and k are dependent:

$$a_{jk} := \begin{cases} 1 & \text{if movements } j \text{ and } k \text{ are dependent} \\ 0 & \text{if movements } j \text{ and } k \text{ are independent} \end{cases}$$

If two train units have to be combined, then they arrive at the track where they are combined from the same direction or from different directions. If they arrive from the same direction, then the order in which they enter the track is known and the train unit that arrives last is the successor of the one that arrives first. If the train units arrive from different directions and at least one of them has not yet been planned in detail, then they may arrive in either order. Because of technical and safety reasons they can not arrive at the same time, so the movements are dependent. By giving each movement the combined train as its successor, the order of operation with the combined train is known and hence both movements are independent of that train. Now each movement can arrive at the track after the other one has arrived.

If a train arrives on a track and is split into two parts, then we have a similar situation. If both parts leave the track into the same direction, then the order in which they leave the track is known and the train unit that leaves last is the successor of the other. If they leave the track in different directions and at least one of them has not yet been planned in detail, then they may leave the track in either order. Again, because of technical and safety reasons, they can not leave the track at the same time, so the departing movements are dependent. By giving the arriving train both departing movements as its successor with respect to the arrival track, the departing movements are independent of the arriving train. Now the departing movements can leave the track in either order.

4.2 Model with Fixed Routes (MFR)

The goal of the capacity test is to verify if it is possible to plan all the movements not yet planned in detail within their time windows. Therefore, the model minimizes the number of movements not yet planned in detail which can not be planned within their time window. If the objective value is zero, then it can be concluded that the capacity of the infrastructure is still sufficient. If the objective value is strictly positive, then not all the shunting movements of S_n can be planned within their time window.

The decision variables used in the model with fixed routes are the following:

- y_j = the plan time of movement j in minutes
- $U_j = \begin{cases} 1 & \text{if movement } j \text{ can not be planned within its time window} \\ 0 & \text{if movement } j \text{ can be planned within its time window} \end{cases}$
- $x_{jk} = \begin{cases} 1 & \text{if movement } j \text{ has to be operated before movement } k \\ 0 & \text{if movement } j \text{ has to be operated after movement } k \end{cases}$

The decision variable y_j is a real variable which gives the plan time of movement j as the number of minutes after the starting time of testing. The decision variable U_j can be derived from the decision variable y_j. If $r_j \leq y_j \leq d_j$, then the variable U_j is zero, else it is one. No conflicts between planned movements gives $y_j = r_j = d_j$ for every movement $j \in S_p$, which gives $U_j = 0$. The decision variable x_{jk} is only defined if movements j and k are dependent. The variable gives the order in which the movements j and k have to be operated. It would be sufficient to define x_{jk} only for movements $j < k$, but then the model becomes hard to read and the computation time is not influenced, because CPLEX eliminates the 'extra' variables.

The problem with fixed routes is described with the following mixed integer programming model:

minimize $\sum_{j \in S_n} U_j$

subject to:

$$y_j = r_j \qquad \forall j \in S_p \tag{1}$$

$$U_j = 0 \qquad \forall j \in S_p \tag{2}$$

$$y_j \geq r_j \qquad \forall j \in S_n \tag{3}$$

$$y_j \leq d_j + U_j M \qquad \forall j \in S_n \tag{4}$$

$$x_{jk} + x_{kj} = 1 \qquad \forall j, k \in S \text{ with } a_{jk} = 1 \tag{5}$$

$$y_j + b_{jk} \leq y_k + (1 + U_j - x_{jk})M \quad \forall j,k \in S \text{ with } a_{jk} = 1 \tag{6}$$

$$y_j + b_{jk} \leq y_k \quad \forall j,k \in S_n \text{ with } k \in s_j \tag{7}$$

$$x_{jm} = x_{km} \quad \forall j,k,m \in S, \text{ with } a_{jm} = a_{km} = 1, \; \exists t \in T_m : s_j^t = k \tag{8}$$

$$x_{jk} \in \{0,1\} \quad \forall j,k \in S \text{ with } a_{jk} = 1 \tag{9}$$

$$y_j \in \mathbb{R}^+ \quad \forall j \in S \tag{10}$$

$$U_j \in \{0,1\} \quad \forall j \in S \tag{11}$$

The meaning of the first three constraints is obvious. Constraint (4) handles the fact that the plan time of a movement not yet planned in detail preferably does not exceed the movement's due date. Constraints (5) and (6) take care that there is enough time between the plan times of two dependent movements. Constraints (4) and (6) contain a big-M, a large constant integer value. Due to these constraints, there always exists a feasible solution for the model. Constraint (6) is binding only if $U_j = 0$ and $x_{jk} = 1$. If movement k is a successor of movement j, then its plan time has to be larger than the plan time of movement j plus the required minimum amount of time between movements j and k. Constraint (7) handles this. If movement k is a successor of movement j with respect to track t, then constraint (8) takes care that movement m, which uses track t and is dependent of movements j and k, is operated before movement j or after movement k.

4.3 Model with Variable Routes (MVR)

In order to increase the flexibility of the model (MFR), an extension (MVR) has been developed in which it is possible to select the routes for all train movements, not only for the shunting movements, but also for the passenger and cargo trains. In order to facilitate the additional flexibility, a few parameters and decision variables have been added. The added parameters are the following:

- R_j = the set of possible routes of movement j.
- T_{jr} = is the set of tracks used by movement j if route $r \in R_j$ is chosen.
- $a_{jrks} = \begin{cases} 1 \text{ if movements } j \text{ by route } r \text{ and } k \text{ by route } s \text{ are dependent} \\ 0 \text{ if movements } j \text{ by route } r \text{ and } k \text{ by route } s \text{ are independent} \end{cases}$

The definition of dependency is the same as in the model (MFR). The route of every movement has to be determined by the model. So a new decision variable is introduced, namely:

$$z_{jr} = \begin{cases} 1 \text{ if movement } j \text{ has to be routed along route } r \\ 0 \text{ otherwise} \end{cases}$$

The model minimizes the weighted number of not yet planned movements which can not be planned within their time window. The parameter w_n represents the penalty if a not yet planned movement can not be planned.

A second term has been added to the objective function. The goal of this term is to prevent that a lot of alternative routes are chosen, especially for the already planned movements. Alternative routes are less comfortable for the passengers, since they usually use more switches. The parameter w_{jr} represents the penalty if alternative route $r \in R_j$ is chosen for movement j. These penalties have to be much smaller than the penalty w_n, because the most important objective is to find a solution such that all movements can be planned. Parameter w_{j0} is the weighting factor if the priority route of a movement is chosen. Now the mixed integer programming model (MVR) can be described as follows:

minimize $$w_n \sum_{j \in S_n} U_j + \sum_{j \in S} \sum_{r \in R_j} w_{jr} z_{jr}$$

subject to:

$$y_j = r_j \quad \forall j \in S_p \tag{12}$$

$$U_j = 0 \quad \forall j \in S_p \tag{13}$$

$$y_j \geq r_j \quad \forall j \in S_n \tag{14}$$

$$y_j \leq d_j + U_j M \quad \forall j \in S_n \tag{15}$$

$$y_j + b_{jk} \leq y_k \quad \forall j, k \in S_n \text{ with } k \in s_j \tag{16}$$

$$\sum_{r \in R_j} z_{jr} = 1 \quad \forall j \in S \tag{17}$$

$$x_{jk} + x_{kj} = 1 \quad \forall j, k \in S : \exists r \in R_j, \; \exists s \in R_k \text{ with } a_{jrks} = 1 \tag{18}$$

$$y_j + b_{jk} \leq y_k + M(3 + U_j - x_{jk} - z_{jr} - z_{ks}) \quad \forall j, k \in S, \forall r \in R_j, \; \forall s \in R_k \text{ with } a_{jrks} = 1 \tag{19}$$

$$z_{mq} - 1 \leq x_{jm} - x_{km} \leq 1 - z_{mq}, \quad \forall j, k, m \in S, \text{ with } d_m + b_{mj} > r_j, r_m < d_k + b_{km}, \; \forall q \in R_m \exists t \in T_{mq} : s_j^t = k \tag{20}$$

$$x_{jk} \in \{0, 1\} \quad \forall j, k \in S \text{ with } \exists r \in R_j, \; \exists s \in R_k : a_{jrks} = 1 \tag{21}$$

$$z_{jr} \in \{0, 1\} \quad \forall j \in S, \forall r \in R_j \tag{22}$$

$$y_j \in \mathbb{R}^+ \quad \forall j \in S \tag{23}$$

$$U_j \in \{0, 1\} \quad \forall j \in S \tag{24}$$

Also for this model defining x_{jk} only for $j < k$ would be sufficient, but for sake of readability $j > k$ is included into the model. This doesn't influence the computation time of CPLEX. Several constraints of (MVR) are the same as those in (MFR). Therefore, we only describe the differences. For every movement only one route can be chosen. Constraint (17) looks after this. Constraints (18) and (19) take care that, if there are routes r and s for movements j and k such that these movements along these routes are dependent, then there will be sufficient time between the plan times of these movements if routes r and s are selected.

Obviously, constraint (19) has an effect on the plan times of movements j and k only if $U_j = 0$ and $x_{jk} = z_{jr} = z_{ks} = 1$. If track t of route q for movement m is occupied by successive movements j and k during part of the time window of movement m, then m is not allowed to use track t as long as this track is occupied. If route q is chosen for movement m, then m has to be carried out before movement j or after movement k. Constraint (20) handle this.

5 Application to Railway Stations in the Netherlands

To check whether the models can be solved quickly and effectively, they have been tested for three railway stations in the Netherlands. These stations are Groningen, Utrecht and Zwolle. A description of the stations is given in Section 5.1. Thereafter the results are presented in Sections 5.2 and 5.3.

5.1 Introduction Railway Stations Groningen, Utrecht and Zwolle

Railway station Groningen is a relatively small station in the northern part of the Netherlands with one shunting area (see Figure 1). Because the shunting area is located parallel to the platform tracks, many saw movements are required. There are even many saw movements which have to be split into three parts, because the shunting area can only be reached from one side. A typical 24-hour weekday for station Groningen has about 575 train movements where the saw movements have been split already into separate movements. Approximately 175 of these 575 movements are shunting movements.

A second railway station which is used to test the model is station Utrecht. This is the largest railway station in the Netherlands. Trains from this station depart to all directions of the Netherlands. Utrecht has two shunting areas, a large one at the southern part of the station (OZ) and a small one at the northern part of the station (Landstraat). The shunting area Landstraat can only be reached from a few platform tracks and can only be entered from one direction. The shunting area OZ is a large shunting area which is always entered or left via the same track. To get to this shunting area, a shunting movement needs to use the same infrastructure as a lot of trains that arrive from or depart to the southern and eastern part of the Netherlands.

Not much trains start or end at railway station Utrecht, so there is a relatively small number of shunting movements. A typical weekday for station Utrecht has approximately 1800 movements including 150 shunting movements. Saw movements only take place twice or three times a day. But if a saw movement takes place, then planners take a transfer time for the driver of at least 10 minutes. As a consequence, each saw movement is a serious bottleneck.

The third railway station that is used for testing the model is station Zwolle in the north-eastern part of the Netherlands. This station is chosen because it is known as one of the hardest stations of the Netherlands with respect to shunting. This is caused by the fact that it has several smaller shunting areas and because many shunting movements are related to the internal and external cleaning of rolling stock. A typical weekday at station Zwolle has approximately 900 movements including 175 shunting movements.

5.2 Results of MFR

The model introduced in Section 4.2 is solved by the standard MIP solver CPLEX 9.0. The computations were carried out on an Intel Pentium M, 1.8 GHz processor with 512 MB internal memory. The model is tested on the stations introduced in the previous section and with a time interval of testing from Tuesday 2:00 am to Wednesday 2:00 am during a normal week. The CPU times CPLEX needs to solve the model for the different stations are given in Table 1. For the three railway stations the model is solved quickly.

Table 1. Running times of CPLEX on MFR

	All preferred routes		Real plan	
	Objective	CPU Time (sec)	Objective	CPU Time (sec)
Groningen	9	2.81	0	2.79
Utrecht	1	1.09	1	1.14
Zwolle	5	4.58	3	4.72

The used data were derived from real shunting plans. The plan times of the already planned passenger and cargo trains were kept the same as in the real plans and for the shunting movements a time window was derived. In the first data set of each station, it was assumed that all movements use their priority route and in the second data set their routes were taken equal to the ones in the real plan.

Because real data should be conflict free, the model should not find movements that could not be planned. But for railway stations Utrecht and Zwolle the model found shunting movements that could not be planned. This can be explained by the fact that planners have the opportunity to violate the norms.

A first impression may be that it is a bit strange that Utrecht has the smallest CPU time. This can be explained by the fact that Utrecht has less shunting movements in comparison with the number of passenger and cargo trains and by the fact that it only has two or three saw movements a day. Especially saw movements are responsible for many dependent movements. The latter require a lot of variables and constraints. That the CPU time for railway station Utrecht is small can also be explained by the fact that the capacity of the infrastructure is so scarce that there is not much to decide for the model.

5.3 Results of MVR

The model introduced in Section 4.3 is also solved with the standard MIP solver CPLEX 9.0 and the computations were also carried out on an Intel Pentium M, 1.8 GHz processor with 512 MB internal memory. The model is tested with the same data sets as used for testing the model MFR.

Solving the model MFR with the routes as given in the real plans results in plan times for all the movements not yet planned in detail. The resulting solution can be used as a starting solution for the model MVR. This results in a good upper bound and a smaller running time. The values of the weighting factors are taken 1000 for w_n and r for w_{jr}. The latter implies that lower numbered routes are preferred, which represents the current practice.

The CPU times CPLEX needs to solve the model for the different stations and for varying numbers of allowed alternative routes are given in Table 2. The CPU times are given in seconds. Table 3 shows the results for the case that an objective difference of five is used. This means that CPLEX terminates as soon as the absolute difference between the lower bound and the upper bound is less than five. Especially for station Zwolle this reduces the running time of CPLEX on most instances significantly.

Table 2. Running times of CPLEX without objective difference on MVR

Number of alternative routes	Groningen		Utrecht		Zwolle	
	Objective	CPU time	Objective	CPU time	Objective	CPU time
0	9000	0.54	1000	0.29	5000	4.55
1	1008	31.53	1	15.76	1004	142.70
2	10	49.74	1	43.29	6	280.20
3	10	29.14	1	59.48	6	323.19
4	10	31.34	1	105.85	6	492.26
5	10	33.42	1	184.56	6	647.00
6	10	33.39	1	237.71	6	646.00
7	10	33.60	1	154.12	6	957.68
8	10	33.36	1	200.93	6	1127.91
9	10	33.38	1	146.63	6	1357.99

If for all movements only the priority route is allowed, then the number of shunting movements that can not be planned is obviously the same as in the previous section. For station Groningen already eight of the nine infeasible shunting movements can be planned if one alternative route is allowed. Allowing two or more routes for every movement makes it possible to plan all the movements. The objective value of 10 can be explained by the fact that eight shunting movements get a first alternative route and one movement gets its second alternative route. The running times for station Groningen are very small.

If only the priority route of the movements is allowed, then station Utrecht has only one shunting movement that can not be planned. This movement is part of the only saw movement in the data set. If one alternative route is allowed, then all movements can be planned. For station Utrecht the running time for solving the model becomes larger than for station Groningen if two or more alternative routes are allowed. This can be explained by the fact that the number of possible alternative routes in Utrecht is much larger, which results in much more possible solutions for station Utrecht in comparison with station Groningen. But the running time of CPLEX is still very small for station Utrecht.

Table 3. Running times of CPLEX with objective difference 5 on MVR

Number of alternative routes	Groningen			Utrecht			Zwolle		
	LB	UB	CPU time	LB	UB	CPU time	LB	UB	CPU time
0	9000	9000	0.53	1000	1000	0.22	5000	5000	4.97
1	1008	1011	10.80	1	1	15.00	1003	1004	119.04
2	10	10	57.23	1	1	40.26	6	7	233.41
3	10	10	36.25	1	1	58.89	5	6	264.03
4	10	10	32.55	1	1	88.07	6	6	426.19
5	10	10	39.67	1	1	179.75	5	6	597.23
6	10	10	39.36	1	1	223.72	6	6	573.88
7	10	10	39.18	1	1	153.10	4	6	804.32
8	10	10	38.97	1	1	198.34	5	6	1022.56
9	10	10	37.83	1	1	133.50	4	6	1208.04

For station Zwolle there are five shunting movements which can not be planned if all movements have to take their priority route. Allowing one alternative route gives only one shunting movement which can not be planned. If two alternative routes are allowed, then all movements can be planned. Allowing an extra possible route for a movement gives a large increase in the running time for station Zwolle. But also in these cases, the running times of CPLEX are still less than half an hour.

6 Conclusions and Future Research

In this paper a global capacity test for the infrastructure between shunting areas and platform areas of railway stations is described. Two mixed integer programming models are introduced. Both models verify whether there is still enough capacity of the infrastructure to plan all the not yet planned shunting movements in between the already planned train movements. In the first model the routes of the movements are given and can not be changed. This model can be solved very quickly by the MIP solver CPLEX 9.0. The second model allows selecting the route of a movement, which results in much more feasible solutions. Nevertheless, for stations Groningen and Utrecht, the model can be solved very quickly. For station Zwolle, the solution process takes somewhat more time. It can be concluded that the computation times are small enough for practical use.

Currently, the capacity test is being implemented in practice. This will facilitate the postponement of planning the details of the non-critical shunting movements until briefly before the operations. The latter will result in a reduction of the amount of replanning of the shunting movements. In the end, this will also lead to a reduction of the throughput time of the complete logistic planning process.

In our future research we will focus on relaxing the assumption that the saw track of each saw movement is known a priori. This makes the problem more complex. However, we intend to enable this increased complexity by using the model on a rolling horizon with shorter time horizons of at most a couple of hours per run. Note that the experiments so far covered a time horizon of a complete day.

References

1. Van den Broek, J.J.J.: Toets op Inplanbaarheid van Rangeerbewegingen (in Dutch), M.Sc. thesis, Eindhoven University of Technology (2002)
2. Billionnet, A.: Using Integer Programming to Solve the Train-platforming Problem. Transportation Science 37, 213–222 (2003)
3. Blasum, U., Bussieck, M.R., Hochstättler, W., Moll, C., Scheel, H.-H., Winter, T.: Scheduling Trams in the Morning. Mathematical Methods of Operations Research 49, 137–148 (2000)
4. Bussieck, M.R., Winter, T., Zimmermann, U.T.: Discrete Optimization in Public Rail Transport. Mathematical Programming 79, 415–444 (1997)
5. Cordeau, J.F., Toth, P., Vigo, D.: A Survey of Optimization Models for Train Routing and Scheduling. Transportation Science 32(4), 380–404 (1998)
6. Cornelsen, S., Di Stefano, G.: Platform Assignment (to appear)
7. Di Miele, F., Gallo, G.: Dispatching Buses in Parking Depots. Transportation Science 35(3), 322–330 (2001)
8. Duinkerken, E.: Toets op Opstelcapaciteit, M.Sc. thesis (in Dutch), University of Amsterdam (2003)
9. Di Stefano, G., Koci, M.L.: A Graph Theoretical Approach to the Shunting Problem. In: Proceedings of the 3rd Workshop on Algorithmic Methods and Models for Optimization for Railways (ATMOS 2003) Electronic Notes in Theoretical Computer Science, vol. 92 (2004)
10. Freling, R., Lentink, R.M., Kroon, L.G., Huisman, D.: Shunting of Passenger Train Units in a Railway Station. Transportation Science 39, 261–272 (2005)
11. Hamdouni, M., Desaulniers, G., Marcotte, O., Soumis, F., Van Putten, M.: Dispatching Buses in a Depot Using Block Patterns. Technical report G-2004-051. GÉRAD, Montréal (2004)
12. Tomii, N., Zhou, L.J., Fukumara, N.: An Algorithm for Station Shunting Scheduling Problems Combining Probabilistic Local Search and PERT. In: Imam, I., et al. (eds.) Multiple approaches to intelligent systems: 12^{th} international conference on industrial and engineering applications of artificial intelligence and expert systems, pp. 788–797. Springer, Heidelberg (1999)
13. Tomii, N., Zhou, L.J.: Depot Shunting Scheduling Using Combined Genetic Algorithm and PERT. In: Allen, J., et al. (eds.) Computers in Railways VII, pp. 437–446. WIT Press, Southampton (2000)
14. Winter, Th., Zimmermann, U.T.: Real-time Dispatch of Trams in Storage Yards, Annals of Operations Research 96, 287–315 (2000)
15. Zwaneveld, P.J.: Railway Planning, Routing of Trains and Allocation of Passenger Lines, Ph.D. thesis, Erasmus University Rotterdam (1997)
16. Zwaneveld, P.J., Dauzère-Pérès, S., van Hoesel, C.P.M., Kroon, L.G., Romeijn, H.E., Salomon, M., Ambergen, H.W.: Routing Trains Through Railway Stations: Model Formulation and Algorithms. Transportation Science 30, 181–194 (1996)