ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 10 (2015) 524 – 533

Transportation Research Procedia

www.elsevier.com/locate/procedia

18th Euro Working Group on Transportation, EWGT 2015, 14-16 July 2015, Delft, The Netherlands

Maintenance scheduling in rolling stock circulations in rapid transit networks

Javier Andrés[a], Luis Cadarso[b], Ángel Marín[a]*

[a] Polyt. Univ. of Madrid. ETS Aeronautical and Space Engineering. Pl. Cardenal Cisneros, 3, Madrid 28040, Spain
[b]Rey Juan Carlos University, Departamental III Building, Camino del Molino s/n, Fuenlabrada 28943, Spain

Abstract

The railway routing problem determines specific paths for each individual train, given its type and composition and considering possible maintenance locations and durations. The objective is to minimize operating costs and penalties related to waiting times and maintenance all while considering train scheduling and maintenance constraints. The model is solved using Branch and Bound and Column Generation approaches. In the paper the different approaches are compared for different planning horizons and model parameter settings. The computational tests have been run in a real RENFE network.

Keywords: routing; maintenance; branch and bound: column generation.

1. Introduction

Before solving the rolling stock routing problem, network design, line service, timetable and rolling stock assignment have been addressed. Designing a rail network is vital to reduce traffic congestion, passenger travel time and pollution in any major city. The main goal of this design is to decide the least costly station locations that provide maximum coverage of the demand for the new network.

The next logical step is train scheduling. Traditionally, train scheduling has been decomposed into sequential steps. The first one is line planning, at tactical planning, in which planners determine the appropriate service frequency for each line, such that all travel demands are satisfied and certain objectives are met, e.g. maximization of quality of service for passengers and minimization of operating costs of the railway system. The second one is

* Corresponding author. Tel.:+34-91-3366323, ext 105; fax:+34-91-3366324.
E-mail address: angel.marin@upm.es

doi:10.1016/j.trpro.2015.09.006

timetable development, in which planners place the purposed train services throughout the day, subject to network considerations and other constraints. The result of timetable development is a series of train trips, which will be determined by the departure and arrival times from/at depot stations. Next, in the Rolling Stock (RS) assignment problem, train types and compositions of a given fleet are assigned to these trips ensuring that train capacity matches closely with demand and fleet cost is minimized.

This paper deals with the next step, where the railway network is redefined according to the previously established rolling stock assignment. Each train trip has a particular type and composition assigned to it; the goal of this step is to assign individual rolling stock units (trains) to those trips. This assignment is performed considering capacity, cost and maintenance requirements.

This problem determines the actual routes taken by the individual rolling stock units. It is usually called Rolling Stock Routing or, when maintenance is taken into account, Rolling Stock Maintenance Routing or simply, Maintenance Routing (MR). At this level, maintenance is key, as its requirements (and of course, network flow constraints) usually predominate over the rest of the constraints when defining the optimal route for each train. In this routing process, operating and maintenance costs are minimized, while each scheduled trip must be covered by exactly one train. Also, train flow conservation and depot capacity constraints must be satisfied at each network node. As a result, the final route performed by each train is composed by trips, storage at depots and maintenance checks, in a way that all the aforementioned requirements are met.

1.1. State of the Art

The paper of Cordeau et al. (1998) is an excellent survey of existing locomotive planning models and algorithms for the Train Scheduling. Cadarso and Marín (2010) study robust RS and routing of rapid transit rolling stock, but they do not consider maintenance restrictions. A comprehensive locomotive planning model is due to Ahuja et al. (2005).

Mellouli (2001) studies the routing and maintenance of trains and aircraft including the vehicle and crew scheduling. Maróti and Kroon (2005) consider the problem of routing locomotive units that require maintenance in the next one to three days and propose a “transition” multi-commodity flow model to solve this problem. The same authors develop the interchange model in Maróti and Kroon (2007) dedicated to regular maintenance routing of a few days, but considering preventive maintenance of about a month. However, small and frequent maintenance checks of only one or two days are not considered. They try to avoid empty trains forced by urgent maintenance tasks and are mainly interested in feasible solutions, given that in practice plans must be consulted with the local shunting crew. Other authors, Hong et al (2009), consider maintenance routing of uniform trains within a weekly train timetable in the High-Speed Railway (HSR) of Korea, covering the timetable with the minimum number of trains. As a second objective, the total working days of the train fleet is minimized. Typically, the MR is formulated in the literature as a multicommodity network flow problem or a set partitioning problem (Klabjan, 2005). The multicommodity flow network has a polynomial space complexity tractable by commercial solvers. The set partitioning formulation yields a tighter representation but involves an implicit enumeration of exponentially many paths and thus requires more complex solution techniques.

The idea behind this initial paper is developing a model suitable for the integration of maintenance routing with previous planning steps, consisting of timetable planning and rolling stock assignment. This future integrated model, with the necessary adaptations, can be used in a recovery planning scheme. In this way, some interesting reports may be mentioned: Borndörfer et al. (2012a, 2012b), Giacco et al. (2014), Wagenaar et al. (2015). The same ideas are being developed by Haahr et al. (2015) considering the integration to study the rescheduling during disruptions.

In this paper a detailed maintenance routing model is defined to deal with rapid transit network problems. The model approach is adapted to solve those problems, but considering the integration of the maintenance routing problem train scheduling planning stages. The paper is focused on maintenance routing but the intention to use it in an integrated recovery scheme, with the necessary adaptations. In contrast to the flow-based approaches to the same problems, our approach is based in the solution of the maintenance routing by an efficient Bellman-Ford's multilevel algorithm for each train type, in the context of a Branch and Price decomposition. In order to ease integration with the rolling stock assignment problem, we use a service based network where stations are nodes that connect services in order to satisfy flow conservation constraints.

The paper is organized as follows: In Section 2 the maintenance routing problem is introduced. In Section 3 the maintenance routing model is defined. In Section 4 the methodology used is described. In Section 5 some computational tests are described. The conclusions and extensions may be read in the last Section 6.

2. Problem description

Given a rail network with a defined schedule, a railway operator wishes to determine the rolling stock routing which covers all the scheduled trips and minimizes the operating costs, considering the regulation on maximum travelling time and distance before maintenance checks. Other considerations include maintenance depot capacity constraints, along with train trip coverage, or the use of soft or hard maintenance time windows, or the use of planning horizon to include in the planning period, which will permit flexibility to include the maintenance levels.

Considering that each train trip has been assigned an optimal type and composition to maximize demand coverage in the RS assignment problem, the Maintenance Routing problem seeks to determine the individual route taken by each train, over a certain planning period, such that the aforementioned maintenance constraints are checked.

2.1. Maintenance

We assume that every train must undergo maintenance checks after a certain number of hours or kilometers of travelling time or distance. Maintenance procedures are divided depending on their level, defined by the maximum times and distances established by the regulations. For high level maintenance procedures (taking place every several months or years), trains are unavailable for routing purposes. This maintenance procedures are usually performed at dedicated maintenance workshops, separated from the network.

For lower level maintenance checks (every several days or weeks), maintenance checks usually take place at the corresponding depots which have maintenance facilities within the network. According to this criterion, depots are classified into storage and maintenance depots and storage-only depots.

Low level maintenance may be performed at maintenance depots during a predefined maintenance time window (MTW). The MTW of each level of maintenance limits on how much time and/or distance a train can operate before the next mandatory maintenance. These MTW may be considered as soft or hard level. In the model formulation they are considered in its hard version.

The maintenance levels are the followings: daily inspection (DI), monthly inspection (MI), large inspection (LI), and overhaul inspection (OI). The first two are low-level maintenance, and the last two are the high-level maintenance. Low-level maintenance is composed by DI and MI. A train must go through a MI after operating for two or three weeks (Time MI, TMI) or for several thousand km (Distance MI, DMI) before it can circulate again. A train must go through DI after operating for two/three days. The inspections have an average duration of 1 or 2 hours for the DI and of 4-6 hours for the MI and they are performed at the specific maintenance depots. Therefore, planners have to keep track of the accumulative operating time and distance since the last maintenance check for each maintenance type and individual train. The specific values above are MR parameters and they may be easily changed for each specific problem (different railway regulations).

High-level maintenance, such as LI and OI, is scheduled ahead of time for each train unit during the maintenance planning process since they require longer inspection times compared to DI and MI and the main workshops have limited capacity. Therefore, we will only consider low-level maintenance (DI and MI) in the tests conducted for this model, although higher level maintenance could be included very easily with no modifications to the model. In addition, the maintenance procedures of a higher level include all the maintenance procedures in lower level maintenance. Therefore, after conducting one class of maintenance process, all accumulative operating times and kilometers associated with that class and the corresponding lower-level classes of maintenance will be set to zero.

The MR is treated in this paper considering a planning horizon that is extended over the planning period. Lai et al. (2013). The planning period is defined as the period in which the model solution is applied, whereas the planning horizon covers a more extended time window considered in the model. This essentially means that the solution for one particular day is calculated taking into account an extended period of two or three days after the day we intend to plan the routing for. We study the effect that this kind of approach has on the routing solutions for rapid transit networks and compare it with a more standard approach (taking into account just the planning period**.**

2.2. Network definition

The MR problem is usually modelled within the framework of an aggregated space-time network in which each node is a depot station, with an associated time corresponding to the time when a train arrives or departs from said depot station. Incoming trains are only connected to outgoing trains if the connection time is sufficient.

The arcs in the network represent train trips, storage at the depot or maintenance checks. All the arcs are directed, and the only arcs that allow travelling between depots are the train trip arcs. Storage arcs are defined between every two adjacent nodes belonging to the same depot. Lastly, maintenance arcs are defined within the time window corresponding to each maintenance depot, and connect every node in the window with the last one that also belongs to the window and allows for a maintenance check of the specified time. Defining maintenance arcs of this way allows that each node in the window has a maximum of one reaching maintenance arc. The maintenance routes are defined by the service and empty route assignment (RS output).

3. Maintenance routing model

The MR model has as input the fleet assignment solution of the RS assignment problem, where train passenger capacity, fleet size, and other constraints have been considered. In the MR model, the operative routing, storage and maintenance costs are minimized, subject to train trip coverage, flow conservation, and maintenance time and distance constraints. The MR outputs are the paths of each individual train during the considered planning period. The MR model is defined by the following items:

Sets:

K: node set, indexed by k. Each defined by a depot and a departure/arrival time.

M: maintenance type set, indexed by m.

R: route train trip set, indexed by r.

R_t: routes used by train t.

P : train type set, indexed by p.

T : train set, indexed by t.

T_p : train subset for type p.

R_k^m route subset using the depot k for maintenance m.

$R_k^{dep(arr)}$ route subset departing (dep) or arriving (arr) to the depot k for maintenance m.

S: service set, indexed by s, output of RS assignment model.

Parameters

c_r^t : operating cost for route train trip r and train t.

$mc_{k,m}^t$: train maintenance cost of train t at node k with maintenance facilities of type m.

w_k : storage cost at node k.

$\left[ld_m, ud_m\right]$: travel maintenance time window of a maintenance session of type m.

$\left[ll_m, ul_m\right]$: length maintenance time window of a maintenance session of type m.

$cp_{k,m}$: capacity of trains in maintenance m depot k.

β_s^p= 1, if the train type p was assigned to service s, output of the RS model.

Variables

o_k^t = 1, if train t starts its route at station k.

$x_r^t = 1$, if train t covers train trip r, = 0 otherwise

$y_{k,m}^t = 1$, if train t uses the maintenance facilities m at depot k, = 0 otherwise

$g_k^t = 1$, if train t is stored at depot k, = 0 otherwise.

$d_{k,m}^t \geq 0$: time travelled by train t at node k since the last maintenance session of type m was performed.

$l_{k,m}^t \geq 0$: length travelled by train t at node k since the last maintenance session of type m was performed.

Objective function:

$$\sum_{t \in T} \sum_{r \in R_t} c_r^t x_r^t + \sum_{t \in T} \sum_{k \in K} w_k g_k^t + \sum_{m \in M} \sum_{t \in T} \sum_{k \in K} mc_{k,m}^t y_{k,m}^t$$

The first term is the operative cost, the second the wait cost and the third the maintenance cost.

Trip coverage constraints:

$$\sum_{r \in R} \sum_{t \in T_p} x_r^t = \beta_s^p, \forall s \in S, \forall p \in P \qquad (\mu_s^p) \qquad (1)$$

, where $\mu_s{}^p$ is the Lagrange multiplier of the constraints (1). Every service s is covered by a train type t.

Train flow conservation constraints

$$o_k^t + \sum_{r \in R_k^{arr}} x_r^t + g_{k-}^t + \sum_{m \in M} y_{k-,m}^t = \sum_{r \in R_k^{dep}} x_r^t + g_k^t + \sum_{m \in M} y_{k,m}^t, \forall k \in K, t \in T \qquad (2)$$

Trains originating in k plus trains arriving at node k plus trains waiting from node k-1 plus trains in maintenance from node k-1 must be equal to trains departing from node k plus trains waiting at k plus trains in maintenance at k.

Maintenance windows:

$$d_{k,m}^t \in \left[ld_m, ud_m\right], \forall k \in K, \forall t \in T, \forall m \in M \qquad (3)$$

$$l_{k,m}^t \in \left[ll_m, ul_m\right], \forall k \in K, \forall t \in T, \forall m \in M \qquad (4)$$

The constraints (3) assure the travel time maintenance limitation; meanwhile the constraints (4) assure the length maintenance limitation, for each train t, depot k and maintenance type m, by the accumulated travelling time and distance in maintenance time windows.

Depot capacity constraints

$$\sum_t y_{k,m}^t \le cp_{k,m}, \forall k \in K, \forall m \in M \qquad (\eta_{k,m} \ge 0) \qquad (5)$$

, where $\eta_{k,m}$ is the Lagrange multiplier of the constraints (5). Trains in maintenance inventories must not exceed maximum capacity at each depot k of maintenance type m.

4. Column generation

The assignment of train types and compositions to the trips is known as output of the RS plan. The MR model has been formulated as a set partitioning model with side maintenance constraints. The MR can be solved as an independent problem for each train type if the train coverage and depot capacity constraints are relaxed. This approach, also called Branch-and-Price (B&P), has found wide application in solving vehicle routing and crew scheduling problems. In order to be successful, we need to generate efficiently columns and also to devise an efficient branching scheme: this is an ongoing work.

Column Generation (CG) is based on the interchange of prices (Lagrange multipliers) between a relaxed master model with the difficult constraints and the primal solutions of a linear model with the easy constraints of the original model (Barnhart et al., 1998). The CG approach decomposes the MR model in a linear submodel (SM) with the network structure and the maintenance constraints (2-4) and a relaxed master model (RMM) defined by the capacity constraints, the train trip coverage constraints (1, 5) and the convex hull of a subset of the extreme points generated by the SM, assuming that the SM feasible set is bounded and nonempty. The RMM elaborates the dual variables (prices) of the difficult constraints, to be used by the SM to generate new extreme points. The CG converges to an integer relaxed MR optimal solution in a finite number of iterations with absence of degeneracy and assuming feasibility and some concrete assumptions (Marín, 1995; García et al., 2003 and 2011).

The SM with the train conservation and maintenance constraints is separable by trains and it is solved considering the updated reduced cost. This model generates feasible paths considering the maintenance constraints as resource constraints in a shortest path algorithm. Feasible train paths with negative reduced costs are returned to the RMM as new columns. The convex hull of the active columns generated is included in the RMM objective function with the set partitioning and maintenance depot capacity constraints.

In the MR model only maintenance of low level constraints have been considered. The maintenance depots have a given capacity that must be kept, these maintenance capacity limitations are considered in the MR. The train rolling stock together with the maintenance inventory of the trains at the start of the planning horizon determines the number of trains at each depot station. Concretely, applying CG to the MR model, we can easily identify one block of coupling constraints (1 and 5), and |T| independent blocks of constraints (corresponding with constraints 2-4). Therefore, we can reformulate our problem for CG dividing it into a RMM and |T| submodels.

If we solve the train routing problem for a strategic horizon, we can choose how far the optimization model will look ahead during the process. If we have the RS information on all the trains for a week, the actual train routing will be consider for one first week days. To obtain a complete week train routing will take several iterations, where the solutions of the first days are the input for the next week days. The SM is mentioned as a pricing model where try to find a sequence of trains satisfying the maintenance constraints considering the number of hours and kilometers permitted between maintenance checks, and price out the negative reduced costs. The pricing is solved by a Resource-constrained Shortest Path Algorithm (SPA) considering the reduced cost of each arc, so the length of each arc correspond to the reduced cost of the train sequence represents. Only trains that not violate some rules may be considered. The SPA algorithm used is an ad hoc Bellman-Ford-Moore (Ford and Fulkerson, 1962) multilevel for arbitrary arc weights and levels with maintenance time and distance accumulates.

The reduced cost of each route and train for each iteration l is defined as follow:

$$c_r^{t,l} = (c_r^t - \mu_r^{l-1}\sum_{r\in R_p}\beta_r^p)x_r^t + \sum_{k\in K}\sum_{m\in M}(mc_{k,m}^t - \eta_{k,m}^{l-1})y_{k,m}^t\delta_r^{km} + \sum_{k\in K}w_k g_k^t \delta_r^k$$

, where $\delta_r^{km}, \delta_r^k$ are arc-route indicators, equal 1 if node k or maintenance m are contained in route r, and 0, otherwise.

The SM(l) optimal solution for each train t at iteration l, and the optimal SM(l) objective function at iteration l are:

$$\left\{x_r^{t,l}, g_k^{t,l}, y_{k,m}^{t,l}\right\} \qquad f_{SM}^{opt,l} = f_{SM}^l\left(x_r^{t,l}, g_k^{t,l}, y_{k,m}^{t,l}\right)$$

The SM (l) solution is defined as a sequence of operations (train trips, waits and maintenance checks) performed by a train from their route origin to destination. These paths are fully determined by optimal variables. The SM(l) optimal objective function is a Lower Bound (LB) at each iteration of MR.

The RMM is defined by the cover and capacity constraints and the convex hull of the extreme points J(l) generated and active at iteration l. The RMM objective function is a convex combination of the extreme points:

$f_{RMM}^l(\alpha) = \min \sum_{j\in J(l)}\alpha_j \sum_{t\in T}\left(\sum_{r\in R} c_r^t x_r^{t,j} + \sum_{k\in K} w_k g_k^{t,j} + \sum_{m\in M}\sum_{k\in K} mc_{k,m}^t y_{k,m}^{t,j}\right); \alpha_j \geq 0, \forall j \in J(l); \sum_{j\in J(l)}\alpha_j = 1$, joint to the convex hull of the cover assignment and station capacity constraints define the RMM at iteration l. The optimal solution of RMM(l) is α_j^l and the optimal objective function $f_{RMM}^l(\alpha_j^l)$ is an MR upper bound at iteration l, UB^l.

Starting with an initial subset of extreme points given by the number of the cover and capacity constraints plus one, the CG algorithm determines a set of optimal RMM dual values. These dual values are used to price out; that is, to compute reduced costs of "nonbasic" variables. In a minimization, a SM solution with a negative reduced cost may improve the solution and therefore should be added to the RMM. Adding and removing columns to the constraint matrix is referred to as column generation.

5. Computational tests

Our preliminary experiments are based on the line C5 drawn from RENFE's regional network in Madrid, Spain. Line C5 has 23 stations and 4 depot stations. We have considered as data the results of the RS model proposed by Cadarso and Marín (2010, 2011). The RS output contains the composition assigned to commercial services. It will be considered as the input of our computational tests with the MR. In Line C5 only one train type is used, and the trains can be configured in simple (one convoy) or double (two convoys) compositions.

The aim of the following tests is to measure the model's performance in different maintenance scenarios and combinations of planning periods and planning horizons. We have also tested the model for solving a real 5 day scenario. All the following results were obtained using GAMS 23.9.5/CPLEX 12.4.1, in a computer equipped with Windows 8.1 64-bit version, an Intel Core i5 2500k running at a clock frequency of 4.5 GHz and 8 GB of RAM. Also, all tests were solved to optimality (0% GAP) using CPLEX's MIP solver. Performance will be compared using total CPU time spent in the Branch & Bound algorithm, along with the number of nodes and the total number of linear programming iterations used in all nodes.

The initial tests were aimed to check the solution quality and the performance in a *small, controlled, environment*. We must take into account that there are over 300 train trips between depots per day in line C5, so maintenance durations and limits were scaled down (see maintenance durations and limits at the table below). With this scaling, maintenance constraints were allowed have an impact in planning periods of just a few hours. Also, given this short planning periods used in the initial testing, we only included one maintenance limit, DI:

Table 1 Initial performance tests

Model Size (trips)	Model Size (rows, columns)	DI duration (min)	TDI limit (min)	DDI limit (km)	Number of nodes	LP iterations	CPU time (s)
10	877, 1023	10	60	48	0	451	0.243
20	1476,2263	20	120	96	0	1074	0.248
50	3118,4898	50	300	240	0	2259	0.868

The initial tests in small scale networks offered very acceptable performance, as we can see in Table 1. In Table 2 we can see a solution sample for 3 trains extracted from the 10- trip model:

Table 2 Solution sample for a 10-trip model

Train	Operation Sequence
7	Trip 1 at 5:00 from Depot 3 -> Storage at 5:54 in Depot 2 -> DI +Storage from 6:14 to 6:31 in Depot 2
8	Storage at 5:07 in Depot 2-> Trip 4 at 5:26 from Depot 2-> DI +Storage from 6:25 to 6:45 in Depot 3
19	Storage at 5:00 in Depot 3 -> Trip 3 at 5:15 from Depot 3-> DI +Storage from 6:14 to 6:31 in Depot 2

An analysis on how parameter variation affects the optimal solution and the required computational time was also performed on a *1-day model*. The number of train trips for one day is usually around 300 in line C5 In this model the number of rows and columns ascends to 70291 and 78961, respectively. As expected, modifying trip coverage, storage and maintenance costs $(c_r^t, w_k, mc_{k,m}^t)$ resulted in enforcing the least costly action in the solution, with no measurable change in computational complexity. However, as seen in Table 3, some interesting results arose from modifying maintenance conditions:

Table 3 Performance variability due to maintenance conditions

DI duration (min)	TDI limit (min)	DDI limit (km)	Number of nodes	LP iterations	CPU time (s)
300	1440	1152	2	23784	89.306
120	1440	1152	2	26384	143.037
100	800	640	11	43870	183.495

Decreasing maintenance duration and limits requires additional CPU time to solve an otherwise identical problem. This is due to the fact that shorter maintenance checks can be scheduled easier at different times during the day and therefore increase the number of possible maintenance checks. In the case of decreasing maintenance limits, more checks are required to satisfy the maintenance restrictions, so finding a suitable solution takes more time. The inclusion of a maintenance window, i.e. a time frame that only allows maintenance checks to be performed within it.

In the following test, we proceeded to solve a problem based on a real *5 day planning* for line C5, extending from Monday to Friday. In this planning scheme, the total number of trips ascends to 1505, while rows and columns are in the 300000-400000 range for a problem of this size. We included again two maintenance restrictions, representing daily and monthly inspections. The maintenance limits we used were 24 hours of travelled time and 1152 km of travelled distance for the DI, and 168 hours and 8064 km for the MI. The durations of the maintenance inspections were 120 and 300 minutes, respectively.

In terms of planning horizon and period, this *5-day model* was solved considering two additional days in a extended planning horizon scheme. We considered an initial maintenance travelled time and length for all available trains, represented by a uniform random variable, ranging from 0 to the maximum time limit for each maintenance type. As far as objective function parameters go, we have considered as penalized the storage and the maintenance performance: w_k ; $3c_r^t, mc_{k,m}^t$; $5c_r^t$, in order to enforce the utilization of trains instead of having them stored or being checked frequently. We include a comparison of CPU Time and LP iterations needed to solve the same routing problem, excluding maintenance restrictions:

Table 4 Main test results. Comparison non-maintenance versus maintenance restricted MR problem

Maintenance restricted	O.F Value (%)	Required Train Units	Number of nodes	CPU Time (s)	LP Iterations
Yes	100	28	31	996.42	140987
No	88.25	24	22	49.68	51884

As we can clearly appreciate in Table 4, the inclusion of maintenance restrictions carries the expected cost and required train units increase, along with a very large increase in the required computing time and LP iterations required solving the MR problem. In the context of formulating a routing model suitable for integrated approach, the decomposition method will likely allow for better solution times, thus enabling the model to solve larger problems.

The following testing environment consisted of a *100-movement*, approximately 6-hour planning period (10000-12000 rows/columns), which included the rush hour (8-9 am). The problem was initially solved considering a coinciding planning period and horizon of 6 hours, then extending the planning horizon to 12 hours and, finally, to the rest of the day for a total of 18 working hours. The idea behind this test was to measure the effect that the planning horizon has on solution optimality for rapid transit networks.

Table 5 Main test results. Costs vs. planning horizon variations

Planning horizon	O.F Value (%)	Number of nodes	CPU Time (s)	LP Iterations	CPU Time (s)
Coinciding with period	100	0	8.886	1799	8.886
+6 hours (+100% trips)	95	2	64.312	7568	64.312
+12 hours (+200% trips)	92	5	211.892	15759	211.892

We can see that there is indeed an associated cost improvement with the inclusion of an extended horizon period, but it comes at the cost of a much greater additional computational time. We can expect diminished returns when extending the horizon period even further, as a 100% extension yields a 5% cost improvement and a 200% extension yields an 8% cost improvement. Finally, we present preliminary tests for *the column generation algorithm compared with Branch and Bound*. In these tests, we iteratively solve a relaxed version of the master model and generate columns that improve the solution until it cannot longer be improved. As seen in table 6, CG provides better results when the model approaches the 300 movement mark, so we expect a great improvement in computing times from a Branch and Price decomposition method in our future integrated model, as the maintenance routing component is by far the most costly:

Table 6 Column generation tests. Comparison with B&B

Model Size (trips)	Number of nodes	LP iterations	B&B CPU time (s)	CG iterations	CG CPU time (s)
10	0	500	0.266	6	0.96
20	0	958	0.389	6	0.87
50	0	2347	1.365	12	2.50
300	4	15759	211,892	150	103.50*

*We must mention that the solution obtained for the 300 trip model is fractional. This is expected when solving large problems with Column Generation algorithms. A branching scheme to obtain integer solutions will be implemented in our future integrated model.

6. Conclusions and extensions

In this paper, we present a mixed integer linear programming model for the maintenance routing problem. We assume that the rolling stock assignment is fixed and known. We consider two types of maintenance constraints: time and length constraints. The maintenance constraints may be adapted to solve different planning horizons. Here the model is adapted for a maintenance horizon of days or weeks, but with small changes the formulation could be adapted to study larger maintenance requirements. The model considers the maintenance depot capacities and other maintenance considerations, e.g. the maintenance procedures of highest levels set zero the lower maintenance levels.

We solve the maintenance routing using Branch and Bound and Column Generation, where we consider a relaxed master model composed of train cover and workshop maintenance capacity constraints, joint with the convex hull of the active columns generated. The columns are generated in a sub-model, which receive the prices of the Relaxed Master Model. The sub-model is separable by train units and solved by resource-constrained shortest path algorithm.

Preliminary computational experiments drawn from RENFE are presented. The model is solved within reasonable computational times. The results are compared using the mentioned methodologies under different test cases. Other experiments consist in proving the maintenance parameters and planning horizons. We are planning to extend the model in the near future by considering the integration of the rolling stock and maintenance routing problems. Other extensions are also being considered, such as recovery problems.

Acknowledgements

Research supported by project grant TRA2011-27791-C03-01 and TRA2014-52530-C3-1-P by the "Ministerio de Economía y Competitividad" Spain.

References

Ahuja, R.K., Cunha, C.B., Sahin, G., 2005. Network models in railroad planning and scheduling. *Tutorials in Operations Research* pp. 54-101.

Barnhart, C., Johnson, EL., Nemhauser, GL, Savelsbergh MWP, Vance, PH., 1998. Branch-and-price: column generation for solving huge integer programs. *Operation Research* v46, pp. 316-329.

Borndörfer et al., 2012a. Vehicle Rotation Planning for Intercity Railways. *ZIB-Report* 12-11. ISSN 1438-0064.

Borndörfer et al. 2012b. Integrated Optimization of Rolling Stock Rotations for Intercity Railways. *ZIB-Report* 12-39. ISSN 1438-0064.

Cadarso, L., Marín, Á., 2010. Robust routing of rapid transit rolling stock. *Public Transport*, v2, pp. 51–68.

Cadarso, L., Marín, Á., 2011. Robust Rolling Stock in Rapid Transit Network. *Computers & Operations Research*, v38, n8, pp. 1131-1142.

Cordeau, J. F., Toth, P., Vigo, D., 1998. A Survey of Optimization Models for Train Routing and Scheduling. *Transporation Science*, v32, pp. 380–404.

García, R., Marín A., Patricksson, M., 2003. Column generation algorithms for nonlinear optimization, I: Convergence analysis. *Optimization*, v52, No. 2, pp. 171–200.

García, R., Marín, Á, Patricksson, M., 2011. Column Generation Algorithms for Nonlinear Optimization II: Numerical Investigations. *Computer and Operation Research*, 38, 591-604. DOI 10.1016/j.cor.2010.07.021,.

Giacco G.L., A. D'Ariano, D. Pacciarelli, 2014. Rolling stock rostering optimization under maintenance constraints. *2nd International Conference on Models and Technologies for Intelligent Transportation Systems* 22-24 June, 2011, Leuven, Belgium. Accepted in JITS.

Hong, S.P., Kim, K.M, Lee, K. and Park, B.H., 2009. A pragmatic algorithm for train-set routing: Case of Korea HSR. *Omega* 37, pp. 637-645.

Klabjan, D., 2005. Large-scale models in the airline industry. In Desaulniers G., Desrossiers, J., Solomon, MM editors. *Column generation*, Kluwer Sci. Pub., pp. 163-196.

Lai, Y.Ch., Zeng, W.W., Liu, K.Ch. and Wang, S.W., 2013. Development of Rolling Stock Assignment System for Taiwan High Speed Rail. 5th Seminar on Railway Operations Modelling and Analysis Rail Copenhagen, 13-15 May, *Tech. Univ. Denmark*.

Marín, Á., 1995. Restricted simplicial decomposition with side constraints. *Networks*, vol. 26, pp. 199-215.

Maróti, G., Kroon, L.G., 2005. Maintenance routing for train units: the transition model. *Transportation Science*, v39, Issue 4, pp. 518–525.

Maróti, G., Kroon, L.G., 2007. Maintenance routing for train units: the interchange model. *Computers and Oper. Research*, v34, pp. 1121-1140.

Mellouli, T., 2001. A Network Flow Approach to Crew Scheduling Based on an Analogy to an Aircraft/Train Maintenance Routing Problem in Computer-Aided Scheduling of Public Transport. *Lecture Notes in Economics and Mathematical Systems* Volume 505, 2001, pp 91-120.

Haahr et al. (2014) A Branch-and-Price Framework for Railway Rolling Stock Rescheduling During Disruptions. *Tech Univ. Denmark*.

Wagenaar, Kroon, 2015. Maintenance in Railway Rolling Stock Rescheduling for Passenger Railway. http://ssrn.com/abstract=2566835.