

COMPAT: a decision support tool for determining the necessity of rail infrastructure

M. J. Wolbers

Holland Railconsult, The Netherlands

Abstract

The functionality of a railway yard is for the largest part determined by the existing simultaneities. Simultaneities allow the simultaneous arrival, departure or drive through of trains.

At medium or large sized yards the process of manually determining all simultaneities is very time-consuming and error-prone, which limits the type and number of questions to be answered within a reasonable time, such as the question as to whether all existing infrastructure is really needed given the requirement specifications for compatible routes.

COMPAT is a new tool developed by Holland Railconsult, which uses a mathematical algorithm to determine all simultaneities on a railway yard. The results can be visualised graphically. The tool enables the user to quickly determine which points or links are and are not necessary to meet the functional requirements and to determine the added functionality when a certain point or link is added.

This means current railway yards or new designs for railway yards can be tested by COMPAT for redundant infrastructure. Therefore, COMPAT provides the railway infrastructure manager with a powerful tool to determine a cost effective layout of a railway yard. By presenting the results graphically, COMPAT contributes to the user's insight into the effects of adding or removing points and links in railway yards, thus reducing discussions among specialists.

Keywords: reachabilities, simultaneities, compatible routes, redundant infrastructure.



1 Introduction

The functionality of a railway yard is for the largest part determined by the existing simultaneities. Simultaneities allow the simultaneous arrival, departure or drive through of trains.

At medium or large sized yards the process of determining all simultaneities is very time-consuming, which limits the type and number of questions to be answered within reasonable time, such as the question whether all existing infrastructure is really needed given the required specifications for simultaneities. This is a very interesting question because of the tendency to critically assess the necessity of points.

Research revealed that there was a need within Holland Railconsult for a tool that could help answering questions regarding simultaneities [1]. Holland Railconsult has developed such a tool [2]. This paper describes the functionalities of the tool and the mathematical algorithms it uses.

2 Mathematical problem

In order to describe the mathematical problem, we need to define two concepts: reachability and simultaneity. A reachability $b(s,t)$ is defined as the ability to reach point t starting from point s over any available route in the railway yard. A simultaneity $g(b(s_1,t_1), b(s_2,t_2))$ is defined as the ability to reach t_1 from s_1 and t_2 from s_2 at the same time without using any common point.

Determining if a set of reachabilities form a simultaneity equals the ‘vertex-disjoint paths problem’. This problem is defined as follows [3]:

Given:

A (directed or undirected) graph $G=(V,E)$, pairs $(s_1,t_1), \dots, (s_k,t_k)$ of vertices of G

Asked:

Pairwise vertex-disjoint paths P_1, \dots, P_k , with P_i being an s_i-t_i -path ($i=1 \dots k$).

Here the pairs $(s_1,t_1), \dots, (s_k,t_k)$ are the reachabilities of the graph. If this has a solution then the reachabilities form a simultaneity.

This “vertex-disjoint-paths-problem” however is NP-complete even when having only two pairs [4], so there is no efficient algorithm finding all simultaneities.

The problem can also be seen as a multi-commodity network flow problem, with the reachabilities in the graph representing the commodities.

3 Determining all simultaneities

Determining all simultaneities equals the following question:

Given:

A railway yard, represented by a directed graph $G=(V,A)$ with given starting points (V_s) and ending points (V_e) .

Asked:

All simultaneities on the railway yard.



Because the number of simultaneities can go into millions on very large railway yards like Utrecht Central Station in the Netherlands, it is impossible to determine for every combination of reachabilities if it forms a simultaneity. That is why all simultaneities have been determined by enumeration.

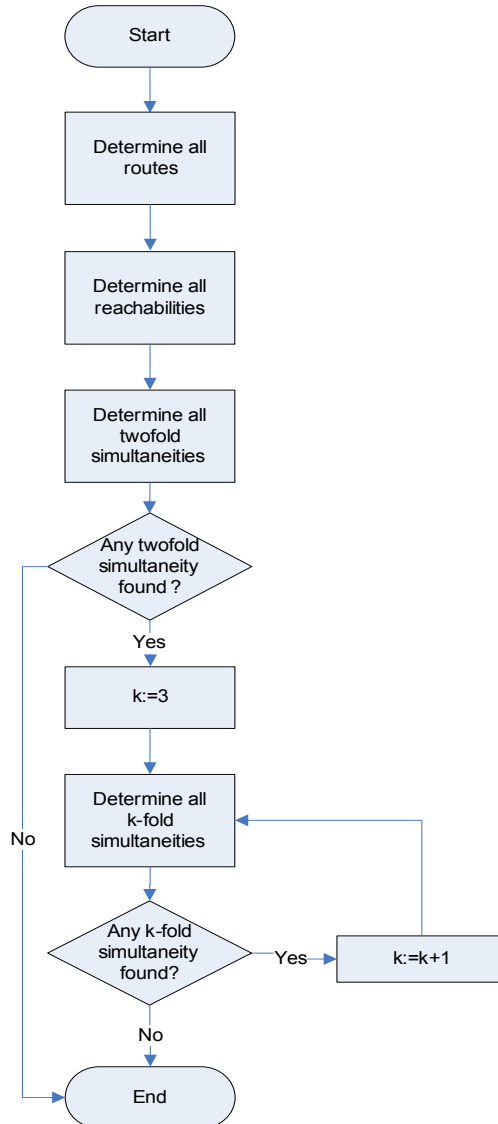


Figure 1: Determining all compatible routes.

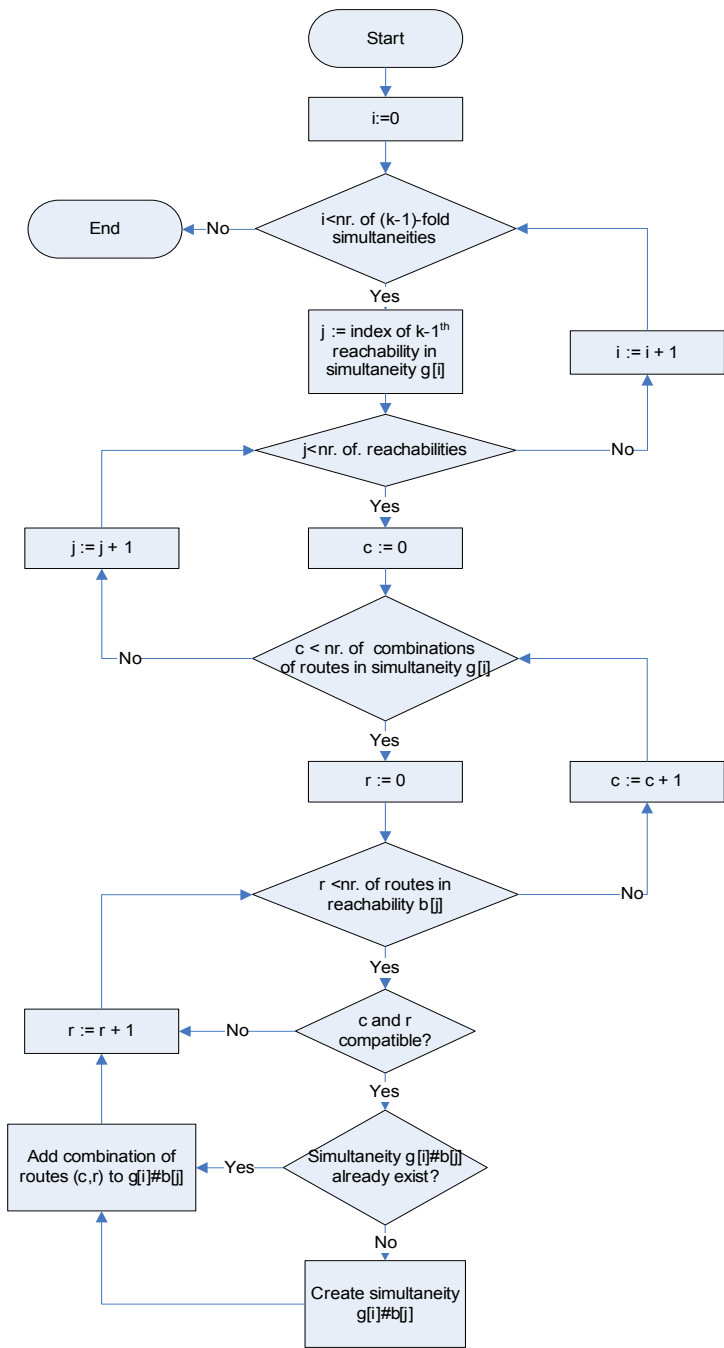


Figure 2: Determining all k -fold ($k \geq 3$) simultaneities.

To determine all simultaneities, first all routes on the railway yard from a starting point to an ending point have been determined. From these routes, all reachabilities can easily be extracted. With each reachability, all routes that start at the starting point and end at the ending point of the reachability are saved. Then all twofold simultaneities are determined from these reachabilities. If two reachabilities have both a route which is compatible, these reachabilities form a simultaneity. In each simultaneity all sets of routes that are compatible are stored. From these twofold simultaneities, the k -fold ($k \geq 3$) simultaneities etc. can be determined. This is done by trying to add one reachability to a $(k-1)$ -fold simultaneity. The process of determining all simultaneities is shown in figure 1, a closer view on determining all k -fold ($k \geq 3$) simultaneities is shown in figure 2.

4 Reducing complexity

Because the number of simultaneities on the railway yard grows exponentially when the number of starting vertices and ending vertices increases, a functionality has been added to decrease the number of routes used during the process shown in figure 1.

The complexity of the problem can clearly be reduced by removing all redundant routes from the computations. Up to 90% of all routes can be redundant in complex railway yards. A route 'r1' is redundant if it blocks exactly the same or more other routes than another route 'r2' and both routes belong to the same reachability. Put otherwise, if $BI[i]$ is the set of routes blocked by route i , then route 'r1' is redundant if there is another route 'r2' with $BI[r2]$ being a subset of $BI[r1]$.

An example of a redundant route is shown in figure 3. The upper route is redundant because it blocks the same and more other routes than the lower route.

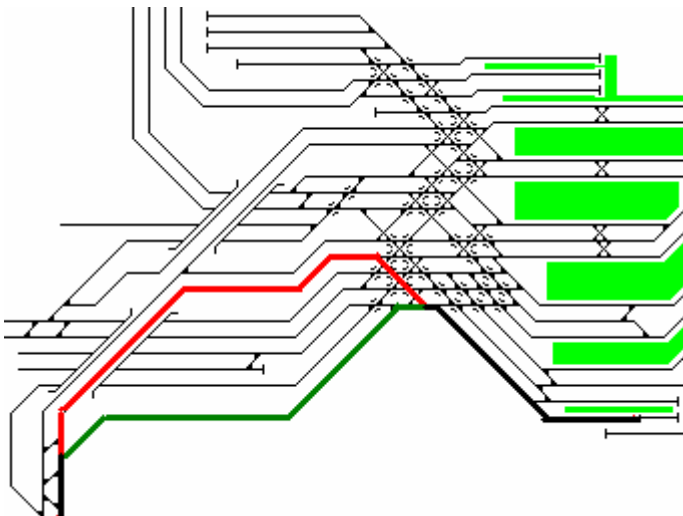


Figure 3: Example of a redundant route.

5 Main functions

5.1 Determine the difference between two alternative lay-outs

The tool allows the user to determine the difference between two alternative lay-outs, in the sense of available reachabilities and simultaneities. This means the user can get a quick view of the consequences when adding or removing a point or link, but also the consequences of more complex changes of the lay-out can be determined.

To determine the difference, the two lay-outs are merged into one new lay-out, with all points and links marked whether they exist in layout 1, 2 or both. Next, the same procedure for determining all simultaneities is followed. Every time a route has been found it is checked if that route consists of any infrastructure that was not in layout 1 and in layout 2 or vice versa. In that way it is clear if that route was in layout 1 or in layout 2 or in both.

A reachability is in layout 1 if it contains any route that is in layout 1. A simultaneity is in layout 1 if it contains a set of compatible routes with all these routes being in layout 1.

After having determined the existence of all simultaneities in layout 1 and/or layout 2, the difference in simultaneities between the two layouts is known.

5.2 Check whether layout meets the functional requirements

The tool can check whether a layout meets the functional requirements, specified by the user in terms of reachabilities and simultaneities. It is also possible to require all reachabilities and simultaneities currently available in the layout.

Depending on the length of the list of requirements, the tool will determine all simultaneities and check when finding one if it is in the requirements-list (long list) or it will consider the requirements one by one on the availability of compatible routes (short list).

5.3 Search for redundant infrastructure

Having more points usually means having more functionality, but points are expensive and are a large source of disturbances, so they are assessed more and more critically recently. With the tool the user can search for redundant infrastructure in a layout.

To determine which infrastructure is redundant, the tool needs a specification of the requirements on reachabilities and simultaneities from the user. Besides a list of requirements the user can also require that all current simultaneities in the railway yard must be preserved. With removing the redundant infrastructure, one can come to a railway yard that still has the same functionalities regarding reachabilities and simultaneities, but has less infrastructure. So lower costs and fewer disturbances.

The tool determines which infrastructure is necessary for the requirements. If a simultaneity always needs a certain part of the infrastructure to meet the requirements, that part is necessary. From all the infrastructure that has been



found unnecessary, any link can be removed. However, by removing a link, other simultaneities can now require parts of the infrastructure that were not necessary before, because it could use the removed link before. Therefore, after each removal of a link, the necessary infrastructure has to be recalculated. In this way the user can easily find a smaller layout that still meets the requirements.

Figure 4 shows a railway yard. Figure 5 shows the same railway yard, but with all redundant infrastructure removed. The simultaneities in both layouts are exactly the same (all current simultaneities were required by the user).

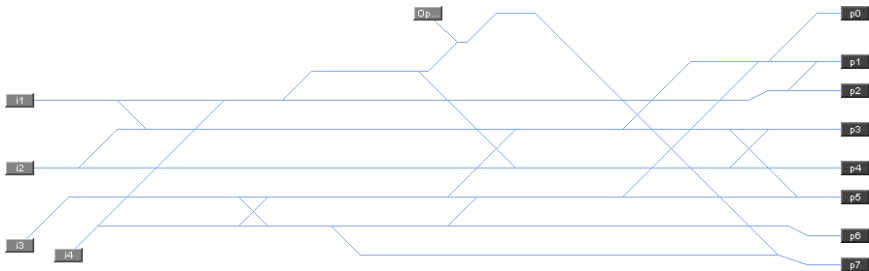


Figure 4: Example of a railway yard.

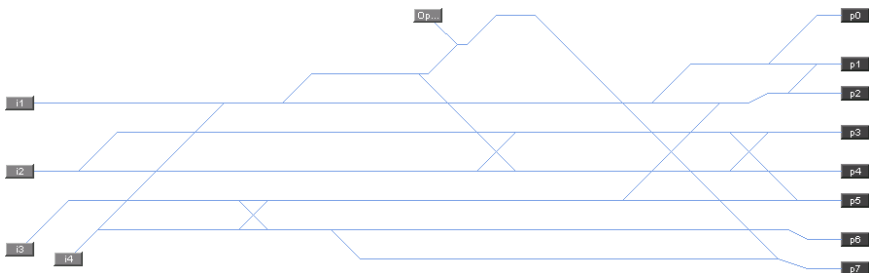


Figure 5: Railway yard of figure 4 without redundant infrastructure.

6 Conclusion

With the tool the possibilities to determine the benefits and necessity of a point have been improved. The tool enables the user to quickly determine which points or links are not necessary to meet the functional requirements and to determine the added functionality when a certain point or link is added.

This means current railway yards or new designs for railway yards can be tested by COMPAT for redundant infrastructure. Until now the results could only be achieved manually, at a much higher costs, longer lead times and lower quality. Therefore, COMPAT provides the railway infrastructure manager with a powerful tool to determine a cost effective layout of a railway yard. By presenting the results graphically, COMPAT contributes to the user's insight into the effects of adding or removing points and links in railway yards, thus reducing discussions among specialists.

References

- [1] Wolbers, M.J., *DSS Gelijktijdigheden*, January 2005 (*in Dutch*).
- [2] Wolbers, M.J., *Decompliceren wisselcomplexen*, June 2005 (*in Dutch*).
- [3] Schrijver, A., *A Course in Combinatorial Optimization*, pp. 149-174, 2004.
- [4] Fortune, S., Hopcroft, J. & Wyllie, J., *The directed subgraph homeomorphism problem*, Theoretical Computer Science 10, pp. 111-121, 1980.

