Railway Capacity in Stations
Evaluating the robustness of timetables
and optimizing the utilization of infrastructure.

E.A. Boogaard

November 1, 2005
Preface

This report is the result of my graduate project at ProRail, department ‘Strategie & Innovatie’. Since February 2005, I have been working on the evaluation of the robustness of timetables at station level. With the tool PETER, it is possible to calculate performance indicators of a timetable concerning its stability and the sensitivity for delays. The objective of this report is to apply PETER to timetables at station level. During the last two months, I have been working on optimizing the utilization of the infrastructure in a station.

In the past nine months, I enjoyed working on a subject with a practical context. Also the possibility to become acquainted with the processes in a large organisation and to experience the helpfulness of so many people, was very inspiring to me. Therefore, I like to thank all colleagues for the nice time and support during these months.

My supervisor from the TU Delft, Jacob van der Woude, I would like to thank for the support on the mathematical aspects of my project and for the comments on my report. Special thanks to my supervisor from ProRail, Dick Middelkoop, for the support and help during my project and the suggestions to improve my report.

With much pleasure I am looking back on this period.

Liesbeth Boogaard
Utrecht, November 2005.
Summary

The organization ProRail, manager of the Dutch railway network, is responsible for the control and maintenance of the present infrastructure and the planning and construction of new infrastructure. Moreover, the allocation of capacity over different transport operators is arranged by ProRail and it is responsible for the operation of the railway traffic.

The department ‘Strategie & Innovatie’ works on innovative methods to improve products and processes of ProRail. The main task is to develop and implement methods to analyze the quality and capacity of the infrastructure of the railway network.

This report mainly concerns the tool PETER, which can be used to evaluate the robustness of a given timetable. It also includes some chapters about the program STATIONS, which can be extended to optimize the utilization of infrastructure in a station.

**PETER**

To evaluate the robustness of a timetable at network level, the tool PETER (Performance Evaluation of Timed Events in Railways) is available. The robustness of a timetable concerns its stability and sensitivity for delays. As a result, PETER calculates performance indicators of the timetable, concerning critical circuits, recovery times and delay propagation.

PETER can be applied during the design of a new timetable. For example, it is possible to evaluate the consequences of introducing or neglecting a passenger connection. PETER can also be used to study the effect of changes in the infrastructure, which causes an increase or decrease of capacity. The advantage is that PETER evaluates a timetable very quickly and that it can work with different input formats.

However, PETER is only applicable at network level and does not consider the train movements inside a station. The main subject of this report is to investigate the possibility of using PETER to study a timetable at station level and to analyze the resulting performance indicators.

A timetable at network level is determined by departures from stations, running times between stations and dwell times at stations. Also dependencies between train lines, as headway times and passenger connections, are considered.

A station can also be seen as a network. The infrastructure inside a station consists of sections, which are separated by joints. Then, the timetable at station level is determined by departures from joints, running times over sections and dwell times alongside a platform. The dependencies between train lines from network level are still valid. If a station is modeled in this way, it can be analyzed with PETER too.
Chapter 0. Summary

To test the theory about modeling a timetable at station level, two case studies are investigated: station Roosendaal and station 's Hertogenbosch. The results are compared with expert opinions and realization data. The following results are obtained with PETER.

The critical circuit in a station determines the minimal time needed to operate all train movements in the station that are planned in the timetable. This minimal time possibly includes constraints too. Because different constraint groups can be included or excluded, the influence of these constraint groups can be analyzed separately.

The recovery time presents the buffer time between two given departures, this is the maximal delay of the first departure such that the second departure is not delayed. Analyzing the recovery times between a sequence of train lines shows which delay of the first train line influences all succeeding train lines. Besides, the recovery time between an arrival and departure of the same train line can be analyzed. Then, a small recovery time may suggest a decreasing punctuality between arrival and departure time.

To investigate the consequence of an initial delay for other train lines in the station, the delay propagation can be calculated. It shows which train lines are affected and how much. PETER can not handle a change in the order of train lines in case of delay, but can only handle the planned order. With the delay propagation can be seen what the impact is of a delayed train and when the consequences become unacceptable, such that changing the order of train lines is necessary.

STATIONS The second objective of this report is optimizing the utilization of the infrastructure in a station. The aim is to spread the intense utilization to reduce the maintenance costs. Besides, it can help to avoid bottlenecks, because the probability of causing a disturbance decreases as the level of utilization reduces.

The tool STATIONS generates a routing of train lines through a station, given the arrival and departure time of these train lines and their entrance and exit point in the station. The resulting routing is generated with respect to some preferences, namely minimizing the number of shunting movements, minimizing the number of deviations in arrival and departure times and maximizing the number of preferred routes and platforms. These preferences do not concern the utilization of infrastructure in the station, but this can be added to the model. The utilization of the infrastructure is measured by the number of times that each switch is passed per hour. The maximal utilization of all switches can then be minimized or it can be bounded to reduce the maximal utilization to a certain level.

The theoretical extension of STATIONS is tested on a small fictitious station. It shows that the weight of the utilization of the infrastructure with respect to the other preferences influences the resulting routing of train lines. Spreading the utilization of the infrastructure and maximizing the number of preferred routes and platforms are conflicting preferences, such that a balance must be found. The fictitious station also shows that measuring the utilization by the number of times that each switch is passed does not automatically optimize the utilization of tracks in the station. Then, also the maximal number of times that all tracks are passed must be minimized.
## Contents

Preface iii  
Summary v  
1 Introduction 1  
2 Problem description 3  
3 PETER 5  
  3.1 Objective ................................................. 5  
  3.2 Timetable design ........................................ 5  
  3.3 Input ...................................................... 6  
  3.4 Modeling method ......................................... 6  
  3.5 Performance indicators ................................. 7  
    3.5.1 Critical circuit ...................................... 7  
    3.5.2 Recovery time ....................................... 9  
    3.5.3 Delay propagation ................................. 9  
4 Mathematical model 11  
  4.1 Timed marked graphs ...................................... 11  
    4.1.1 Petri-nets ............................................ 11  
    4.1.2 Timed marked graphs ................................ 12  
    4.1.3 Construction ......................................... 13  
    4.1.4 Application .......................................... 15  
  4.2 Max-plus algebra .......................................... 16  
    4.2.1 Introduction ......................................... 16  
    4.2.2 Matrix aspects ...................................... 18  
    4.2.3 Recurrence relation ............................... 19  
  4.3 Performance indicators ................................. 23  
    4.3.1 Critical circuit ...................................... 23  
    4.3.2 Recovery time ....................................... 27  
    4.3.3 Delay propagation ................................... 28  
5 Station level 31  
  5.1 STATIONS .................................................. 31  
  5.2 Modeling method ......................................... 32  
    5.2.1 Timed marked graph ................................ 33  
    5.2.2 Adaptations ......................................... 34  
    5.2.3 Input file ............................................ 38
CONTENTS

5.3 Limitations ........................................ 43
   5.3.1 Fixed order of trains ............................ 43
   5.3.2 Increased running time ......................... 44
   5.3.3 Simulation .................................. 46

6 Results of case studies 49
   6.1 Interpretation of the results ...................... 49
   6.2 Roosendaal ...................................... 50
      6.2.1 Results .................................. 50
      6.2.2 Comparison ................................ 52
   6.3 ’s Hertogenbosch ................................. 52
      6.3.1 Results - Critical circuit ................. 54
      6.3.2 Results - Recovery time .................... 58
      6.3.3 Results - Delay propagation ............... 59
      6.3.4 Realization data .......................... 60

7 STATIONS 65
   7.1 Objective ....................................... 65
   7.2 Description of the model ......................... 65
      7.2.1 Processes in a station ....................... 66
      7.2.2 Assumptions ................................ 66
      7.2.3 Preferences ................................ 67
      7.2.4 Notation .................................. 68
      7.2.5 Model ..................................... 70

8 Optimizing the utilization of infrastructure 73
   8.1 Optimization purpose ............................. 73
   8.2 Measurement and optimization systems ............ 73
   8.3 Model extension ................................ 74
   8.4 Fictitious station ................................ 76
      8.4.1 Layout and timetable ......................... 76
      8.4.2 OPL-studio: original model ................. 77
      8.4.3 OPL studio: extended model ................. 78
      8.4.4 Evaluation with PETER ....................... 80

9 Conclusions and recommendations 83
   9.1 Conclusions ..................................... 83
      9.1.1 PETER at network level .................... 83
      9.1.2 Modeling a timetable at station level ........ 84
      9.1.3 Performance indicators at station level ........ 85
      9.1.4 Extending STATIONS ......................... 86
   9.2 Recommendations ................................ 87
      9.2.1 Evaluating a timetable at station level with PETER .... 87
      9.2.2 Optimizing the utilization of the infrastructure .... 88

A Abbreviations of station names 91

B Roosendaal 93
   B.1 Map of Roosendaal .............................. 93
### CONTENTS

#### C ’s Hertogenbosch

- C.1 Input file ’s Hertogenbosch ........................................... 95
- C.2 Map of ’s Hertogenbosch .............................................. 98
- C.3 Output of critical circuit analysis ................................. 99
- C.4 Recovery times ’s Hertogenbosch ................................. 101
- C.5 Output of delay propagation ...................................... 103
Chapter 1

Introduction

Since January 2003, the three organizations 'Railinfrabeheer', 'Railned' and 'Railverkeersleiding' are combined in the organization ProRail. ProRail, the manager of the Dutch railway network, is responsible for the control and maintenance of the present infrastructure and the planning and construction of new infrastructure. Moreover, the allocation of capacity over different transport operators is arranged by ProRail and it is responsible for the operation of the railway traffic every day.

At the moment, around 1 million people travel by train every day and 230 freight trains travel through the Netherlands. To assure a good performance, ProRail aims at a reliable and safe railway network. In the future an increase of passengers and freight trains is expected. The present infrastructure will not be capable to carry this increase. The construction of new infrastructure is no solution, because it is very expensive, it is difficult to place in the environment and because it takes a long time before the development is completed. Therefore, different railway organizations worked together at the vision 'Benutten en Bouwen' (Utilization and Construction), reported in [22]. This vision states that the increase of railway traffic can be realized by a better utilization of the present infrastructure. Its capacity can be enlarged by a better reliability.

The department 'Strategie & Innovatie' (Strategy & Innovation) of ProRail works at innovative methods to realize the vision 'Benutten en Bouwen'. It aims at enlargement of the reliability of the infrastructure and a better utilization of these infrastructure, to achieve a better price-performance ratio. The main task of 'Strategie & Innovatie' is to develop and improve processes and products for capacity management. The increasing demand for capacity by transport operators must be tested to the actual available capacity and quality. Methods are designed to support the decision making, such that a better utilization of the infrastructure can be realized, what leads to a higher reliability. One of these methods is the tool PETER, see references [4], [6] and [8], which helps to determine the quality of a given timetable at network level. Because it is also interesting to analyze the quality of a given timetable at station level, this tool is further investigated in this report.
Chapter 2

Problem description

In ‘Benutten en Bouwen’, [22], it is stated that the aim of more capacity and a high reliability will be realized by a better availability and utilization of the present infrastructure. The capacity is utilized by trains, which travel according to a timetable. A more robust timetable results in more reliability, because a robust timetable reduces the influence of delays. The need for a robust timetable also demands a method to test the robustness of a timetable.

The robustness of a timetable is related to its stability and sensitivity to delays. To test the robustness and find the bottlenecks of a timetable the tool PETER (Performance Evaluation of Timed Events in Railways) has been constructed, see [4], [6] and [8]. This tool is able to calculate network performance indicators of a given timetable, presented by critical circuits, recovery times and delay propagation.

PETER evaluates the timetable at network level, what means that stations are seen as black boxes. But also inside stations bottlenecks are possible, which can cause an extra delay. It is interesting to view stations more accurately to analyze the robustness of a timetable at station level. The infrastructure of a station can also be seen as a network, what gives the idea that PETER can also be applied at station level.

The main subject of this report is to investigate the possibility of using PETER to study a timetable at station level. To this end, first some detailed knowledge how PETER operates at network level is required. Then, the application at station level can be investigated. Both parts can be realized by answering the following questions.

• How does PETER operate at network level?
• What is the mathematical background of PETER and how does it produce the network performance indicators?
• How can these mathematical methods be applied at station level?
• What are the results of calculating performance indicators at station level and are they practical applicable?

This report answers these questions, as is described in the outline below.
Chapter 2. Problem description

The second subject of this report is to optimize the utilization of infrastructure in a station. Because of a preference for certain routes and platforms in a station, most train lines use the same routes. This causes a high utilization of just a few elements of the infrastructure, what can cause a lot of maintenance costs and a higher probability of disturbances. Therefore, this report analyzes whether and how it is possible to optimize the utilization of the infrastructure in a station. This can be considered by answering the following questions.

- What methods are available to measure the utilization of the infrastructure?
- Can one of these methods be extended to optimize the utilization of the infrastructure in a station?

These questions are answered in the last chapters of this report, which is described in the outline below.

The outline of this report is as follows. Chapter 3 concerns the operation of PETER at network level. It answers the first question by describing the input and assumptions of the program and its main results, the performance indicators. The next chapter, chapter 4, deals with the mathematical background of PETER and answers the second question. It also explains how to derive the performance indicators. In chapter 5 is investigated which information is available about the infrastructure and timetable at station level. Then, this information is modeled and written in the right input format, such that PETER can be applied. This answers the third question. To validate the theory in the previous chapter, chapter 6 describes the results of two case studies and answers question four. These results are compared with realization data and expert opinions. Chapter 7 describes how a routing of train lines through a station is generated with STATI ONS. How this model can be extended to optimize the utilization of infrastructure is described in chapter 8. These chapters together answer the two questions of the second subject. Chapter 9 contains the conclusions of this report and some recommendations for further research. This report ends with a bibliography and appendices.
Chapter 3

PETER

This chapter starts with an introduction of PETER: a description of the program, which results can be obtained and why it is practical to use. Then, a section follows about the required input and a short explanation about the modeling method. The chapter ends with a section about the results, these are the performance indicators.

The next chapter explains the modeling method in detail. It includes the mathematical background of the program and a derivation of the performance indicators.

3.1 Objective

PETER is an analytical tool to evaluate the robustness of a timetable, by calculating network performance indicators. The robustness of a timetable is determined by its stability and sensitivity for delays. The stability involves the amount of buffer time in the timetable and the time needed to settle a delay. The sensitivity for delays considers the dependency between train lines, as which train lines are affected by a certain delay and how much. The performance indicators represent characteristics about critical circuits, recovery times and delay propagation. The advantage of PETER is that it evaluates a timetable very quickly and that it can work with different input formats. A timetable can be generated with DONS (Designer Of Network Schedules), which is described in [14] and [19]. Specially, the output files of DONS can be imported in PETER, because PETER is compatible with DONS. This makes it very easy to evaluate a timetable that is constructed with DONS.

3.2 Timetable design

The basis of evaluating the robustness of a timetable is the timetable itself and possibly some disturbances. It consists of arrival times, departure times and of dependencies between train lines to make passenger changes possible and to avoid collisions. To construct a timetable, all train lines and dependencies between them must be determined. A train line is a train service, characterized by a route from an origin station to a destination station, the served intermediate stations along the route and the service frequency. A train line is operated by a
series of trains, commonly running at a regular interval. To generate a timetable with DONS, for every train line also its type of rolling stock must be defined. Dependencies between train lines are represented by constraints, which are defined as infra constraints, concerning headway times and requirements to avoid train collisions, or service-constraints, concerning synchronization and connection constraints. When these specifications are imported in DONS (Designer Of Network Schedules), its solver CADANS, see [16], tries to generate a cyclic timetable at a predefined infrastructure.

DONS is a tool to investigate whether and how it is possible to make a feasible timetable for the specified train lines, according to the given constraints. A feasible timetable is always a cyclic schedule, with a given period of 60 or 120 minutes, and defined for the whole network. It can be presented in a space-time diagram, showing all scheduled train lines at a certain part of the infrastructure. If there is no feasible timetable, DONS returns the conflicting constraints. This makes it easier to find which constraints must be relaxed and which train lines must be adapted.

3.3 Input

The basic information for PETER is a timetable at network level. This means, the arrivals and departures from all train lines at every station in the network. It also includes the running times and dwell times of train lines and the constraints for dependencies between train lines. This information is specified in three parts: coordinates, timetable and constraints, where the coordinates are used to draw a picture of the network and to visualize results. It is assumed that the timetable imported in PETER is feasible; it can theoretically be operated without delays and does not contain conflicting constraints. A timetable generated with DONS, can directly be imported in PETER, because it is compatible with DONS. PETER can also handle a generic input file or an earlier generated PETER file.

Starting with a feasible timetable, PETER is able to find the critical circuit and to calculate the recovery times. To evaluate the delay propagation of the network extra input is necessary, namely an initial delay of one or more train lines. This is imported in a separate file.

3.4 Modeling method

The imported information in PETER consists of coordinates, a timetable, constraints and possibly an initial delay. The timetable and constraints are used to construct a timed marked graph. A timed marked graph is a directed graph with two kinds of nodes, transitions and places. The transitions represent events and the places represent processes. Because train departures are discrete events and running and waiting are processes, the train line network is a discrete event dynamic system, which can be represented as a timed marked graph. How to construct a timed marked graph for a railway network is explained in subsection 4.1.3.

Next, the timed marked graph can be transformed into a recurrence relation in max-plus algebra. Every transition in the timed marked graph belongs to
3.5 Performance indicators

After importing the input information, PETER draws a picture on the screen of that part of the network that is used by the train lines in the timetable. And it builds the underlying timed marked graph, such that it is possible to calculate the performance indicators corresponding to critical circuits, recovery times and delay propagation. The results are presented in the picture and in tabular form. Below a description follows of each of these performance indicators.

3.5.1 Critical circuit

In the timed marked graph, every train line is represented by a circuit that models its way from origin to destination and back. The constraints between train lines are represented by arcs connecting the circuits in the graph. This means that there are also circuits in the graph formed by arcs of different train lines and arcs representing constraints.

A timetable includes buffer times to compensate for delays. In the first place, buffer times are added to the running and dwell time between two departures from different stations. If, for example, a train line is scheduled to depart from A at 9.15 and from B at 9.35 and if the running time is 15 minutes and the dwell time 2 minutes. Then, without delays, this train line arrives at B at 9.30 and is ready to depart at 9.32. The buffer time is 3 minutes. Secondly, buffer times are added to headway times between departures of different train lines. The difference between the scheduled departure times can exceed the headway time. The extra time is also called buffer time. If, for example, a train line is scheduled to depart from C at 9.45 and another train line is scheduled to depart from C at 9.49 and if the headway time between these departures is 3 minutes, then the buffer time is 1 minute.

A circuit of just one train line consists of a series of tracks between stations. Summing the buffer times from these tracks gives the total buffer time of the circuit. If a circuit includes parts of different train lines and constraints, the total buffer time is the sum of buffer times added to running and dwell times between departures from stations and of buffer times added to headway times.

To define the critical circuit of a timed marked graph, first consider a circuit of just one train line. The minimal time needed to operate this circuit is the

an equation in max-plus algebra, depending on its upstream transitions and places. The relations for all transitions together form a system of equations, a recurrence relation. The derivation of this relation can be found in subsection 4.2.3.

With this recurrence relation, it is possible to derive characteristics about the corresponding timed marked graph. Because this graph is a representation of the timetable, these characteristics give information about the timetable; more specifically, these are the performance indicators. Section 4.3 describes in detail how to derive the indicators, what they tell about the timed marked graph and what it means for the timetable.

The advantage of the max-plus algebra is that the recurrence relation becomes a linear dynamic system, such that the performance indicators can be computed by efficient polynomial algorithms.
sum of all running and dwell times, so without buffer times. The cycle mean of the circuit is the minimal time to operate it once, divided by the number of trains on this line. Assume for example a train line from A to B and back with a total running and dwell time of 100 minutes and a total buffer time of 20 minutes. Assume also that 10 minutes from this buffer time are added to running times between intermediate stations along the route and 10 minutes are added after arrival in station A again. This means that if a train departs from station A at 9.15, it arrives at station A again 110 minutes later, at 11.05. This train can depart again at 11.15 on the same route. To travel conform an hourly pattern, two trains are running along this line. The cycle mean of this circuit is $100/2 = 50$ minutes. This means that the minimal period needed to operate this circuit is 50 minutes. Because the actual period is 60 minutes, this train line is feasible.

The cycle mean can be calculated for every circuit in the timed marked graph, also for circuits including parts of more train lines and some constraints. Then, the minimal time needed to operate a circuit is the sum of running times, dwell times and headway times. The maximum of all cycle means of the graph is called the maximal cycle mean.

Every circuit has a minimal period needed to operate, what is specified by its cycle mean. In the same way, there also is a minimal time needed to operate the whole network, what is called the minimal cycle time. So every circuit in the graph is able to operate in this time. Therefore, the minimal cycle time of the network is the maximal cycle mean of all circuits in this graph.

The minimal cycle time is defined by the circuit with the maximal cycle mean, this circuit is called the critical circuit. It is possible that more circuits have the same maximal cycle mean, then each of them is called critical circuit.

The performance indicators of a critical circuit are: minimum cycle time, throughput and stability margin. The minimal cycle time of a network must be less than or equal to the actual cycle time of 60 minutes, otherwise it is not operable. Therefore, the minimum cycle time is an indicator of stability. If the minimum cycle time is less than the actual cycle time the timetable is called stable, and if it is equal to the actual cycle time it is called critical. Otherwise the timetable is called unstable. In case of equality, the circuit can be operated but has no buffer time. This means that there is no time to compensate a delay and it continues in the network. Only when the minimal cycle time is less than the actual cycle time buffer time can be used to settle a delay.

The throughput of the critical circuit denotes a trade-off between the maximum performance and robustness. It represents the minimum cycle time relative to the actual cycle time. This implies that the timetable is stable if the throughput is less than one.

The stability margin is a performance indicator of the robustness of the network. It shows how stable it is. The stability margin is defined as the maximum simultaneous increase of all process times such that the train line network can still be operated with the same actual cycle time. The process times are all running times, dwell times and headway times in the graph. The stability margin corresponds to the circuit with the least average buffer time per circuit segment.
3.5.2 Recovery time

To be able to study the propagation of delays through the network, it is important to know the recovery times between different departure times. The recovery time between departures A and B is defined as the maximum delay of departure A such that departure B can occur without delay. The departures A and B can be in two stations of the same train line, but can also be of different train lines, for example in case of a passenger change. The recovery time is the sum of all buffer times added to processes connecting departures A and B in the timed marked graph. If there are different paths from A to B, every path has its own recovery time. The recovery time between departures A and B is then defined as the minimum of them. The recovery time analysis involves the following performance indicators: delay impact, delay sensitivity and circuit recovery time. The first two indicators are calculated for a specific train departure, while the last one is for a whole circuit.

The delay impact from departure A to another departure is the maximum delay of departure A that does not reach the other departure. The delay sensitivity from departure A with respect to another departure is just the other way around, this is the maximum delay of the other departure that does not reach departure A. If one departure is specified, the delay impact and delay sensitivity with respect to all other departures in the model are calculated. The circuit recovery time of a departure A is the minimum total recovery time on any circuit back to A.

3.5.3 Delay propagation

To study the impact of delays on the network in space and time, there has to be an initial delay to start with. The initial delay at a reference time can consist of several initial delays at different stations. PETER specifies the delay propagation separately for each departure. It results in the following performance indicators.

- The total initial delay is the sum of all initial delays at different stations.
- The secondary delay is the sum of all delays to other trains caused by a constraint between two train lines. If an initial delay affects several stations of the same train, then only the first delay is added to the secondary delay, because the following delays are not caused by a constraint but by a stop from the same train.
- Delays in the next stations of the same train, which are caused by a stop of this train, are called consecutive delays. This also holds for the train that caused the initial delay.
- The cumulative secondary delay is defined as the sum of the secondary delay and all consecutive delays.

PETER also counts the number of stations and the number of trains that are affected by an initial, secondary or consecutive delay, such that also average delays can be calculated.

- The average primary delay is defined as the total initial delay divided by the number of trains with an initial delay.
The average secondary delay is the secondary delay divided by the number of trains affected by a secondary delay.

And the average total delay is the sum of the initial and secondary delay divided by the number of trains affected by an initial or secondary delay.

The last performance indicator of the delay propagation is the settling period.

The settling time is the period after which all delays have been compensated by the recovery times. This situation occurs if the departure times equal the scheduled departure times again.
Chapter 4

Mathematical model

The previous chapter is a description of the operation of the analytical tool PETER, its input and its resulting performance indicators.

This chapter is a further study into the mathematical background of PETER, with the purpose to explain how the performance indicators of a timetable are calculated. It shows how to represent the timetable as a timed marked graph, how to transform this graph into a recurrence relation in max-plus algebra and how this leads to the performance indicators of the timetable.

This knowledge is used in chapter 5 to study how the robustness and sensitivity of the timetable at station level can be evaluated.

4.1 Timed marked graphs

The timetable of a railway network can be modeled as a timed marked graph. This section explains how a timed marked graph is defined and how it is applied in modeling a railway network.

4.1.1 Petri-nets

Timed marked graphs are a subclass of marked graphs, which is a subclass of Petri-nets. Petri-nets are directed graphs with two kinds of nodes; transitions and places. Arcs connect these places with transitions and transitions with places. Arcs connecting transitions or places with themselves do not exist. The places have a special characteristic, they can contain one or more tokens. These tokens represent the dynamic behavior of the network, because they move through the graph in time. The transitions, places, arcs and tokens are represented by bars, circles, arrows and dots respectively. An example of a Petri-net is shown in figure 4.1.

In a Petri-net, transitions represent events and places represent conditions. An event has one or more conditions, which must be fulfilled before the event can occur. The conditions belonging to an event correspond to the upstream places of its associated transition. If a place contains a token, the condition is fulfilled. When all upstream places of a transition have a token, all conditions are fulfilled and the event can take place. This situation is also called: the transition can fire. When firing, one token is removed from every upstream
4.1.2 Timed marked graphs

The class of marked graphs consists of Petri-nets where every place has exactly one upstream and one downstream transition. The incoming and outgoing arc of a place and the place itself can be seen as one arc, connecting the upstream and downstream transition. A marked graph has the property that if each transition has fired once, all places have their initial number of tokens again. Because a place has exactly one upstream and one downstream transition, exactly one token is added and one removed from this place. This makes the number of tokens in the place equal to the number before all transitions fired once.

Figure 4.2 shows a representation of a timed marked graph. This subclass consists of marked graphs with a holding time for every place. This holding time represents the process time of the condition, indicated by this place. If, for example, two transitions represent the arrival and departure of a train in a station, then the holding time of the place between these transitions represents the dwell time in this station. Before the train is allowed to depart, the condition must be fulfilled. This means that the dwell time must be expired. This principle is the basic idea for modeling a railway network. Because all process times in a railway network, running times, dwell times and headway times, are preceded and succeeded by exactly one event, an arrival or departure, it can be modeled as a timed marked graph. This is further explained in subsection 4.1.3. The formal definition of a timed marked graph, as stated in [9], is as follows.

Definition 4.1 A timed Petri-net $\mathcal{G}$ is characterized by $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{D}$, $\mathcal{M}_0$ and $T$, where $\mathcal{P}$ is the set of places, $\mathcal{Q}$ is the set of transitions, $\mathcal{D}$ is the set of arcs, from transitions to places and vice-versa, $\mathcal{M}_0$ is the initial marking and $T$ is
the vector of holding times. If each place has exactly one upstream and one
downstream transition, then the timed Petri-net is called a timed marked graph.

**Average circuit weight** A graph \( A \) is defined by its set of nodes, \( \mathcal{N} \), and set of arcs, \( \mathcal{D} \). Every arc is defined by a pair of nodes \((i, j)\). A circuit in a graph is defined as a sequence of arcs \( S = ((i_k, j_k) \in \mathcal{D}(A) : 1 \leq k \leq n) \), such that \( j_k = i_{k+1} \) for \( k < n \) and \( i_1 = j_n \). Moreover, a circuit is called elementary if every node in the circuit has exactly one incoming and one outgoing arc of the circuit. This also holds for a timed marked graph.

The average weight of a circuit in a normal graph, with just one kind of nodes, is the sum of all arc weights divided by the number of arcs in the circuit. This is also called the mean weight of a circuit.

The mean weight of a circuit in a timed marked graph is defined a little different. The weight of every arc is the holding time of the place. The sum of the weights in a circuit is the sum of the holding times, thus the time to complete the circuit once. The number of tokens in a circuit shows with how many trains the total process time is completed. A timetable usually has a period of 60 minutes, so there must be enough trains traveling along the circuit to fire every transition at least once an hour. Therefore the average weight is computed by dividing the total circuit weight by the number of trains and not by the number of arcs in the circuit. The average circuit weight represents the minimal period needed to operate this circuit. This is also called the circuit mean or cycle mean. An example of the cycle mean is illustrated in figure 4.3. Assume that the sum of all holding times in a circuit is 90 minutes and that there are two trains running along the circuit. Then the cycle mean is defined as \( 90/2 = 45 \) minutes.

![Cycle mean example](image)

Figure 4.3: The cycle mean of this circuit with total weight of 90 minutes and with 2 tokens is 45 minutes.

### 4.1.3 Construction

Constructing a timed marked graph for a given timetable starts with the construction of the train lines in the network. Trains running along a train line from city A to city B make a round trip, returning along the same route. This round trip can be recognized as a circuit in the graph, consisting of a chain of transitions and places where the first and last transition are the same. The transitions represent departure times from stations and the places represent the minimal process times between two transitions. The construction of a train line is a repeating process consisting of the following steps.

---

13
Chapter 4. Mathematical model

1. Define a transition for the starting station of the train line.

2. If the next station is the starting station, go to step 3. Else, add a place and a new transition for the next station. Add an arc from the current transition to the new place and an arc from the new place to the new transition. The holding time of the new place is the sum of the running time to the next station and the dwell time there. Repeat step 2.

3. Add a place and one arc from the current transition to the new place and one arc from the new place to the first transition, which represents the starting station. The holding time of the new place is the sum of the running time to the first station and the dwell time there.

The result of this process is a circuit of transitions and places, modeling the train line. Repeating this process for all train lines in the timetable results in a set of disjunct circuits.

All dependencies between train lines are modeled as constraints, which define a condition between two events. The first event belongs to the leading train line and the second to the following train line. For passenger constraints these are called feeder and connecting train line respectively. Constraints are modeled in a timed marked graph by performing the following step for every constraint. This connects to the first part of constructing a timed marked graph, described in the three steps above.

4. Search for the two transitions representing the departures mentioned in the constraint. Define a place between these transitions and add an arc from the leading transition to this place and an arc from this place to the following transition. The holding time of this place is the headway time or connecting time as defined in the constraint.

In a timed marked graph only departures are modeled, which is further explained in subsection 4.1.4. Therefore, if one of the events of the constraint, or both, depend on an arrival time instead of a departure time, the constraint is connected to its previous departure transition. The holding time is increased or decreased with the running time between this departure and the original arrival. It is increased if it concerns the leading train line and decreased if it concerns the following train line. An example of a constraint is depicted in figure 4.4. It shows a constraint, which forces the lower train to wait for the upper train. This can be due to a passenger change, or due to common use of a railway track.

Finally, if the structure of nodes and arcs is completed, an initial marking of tokens must be specified. If the departure times according to the timetable and the period of the schedule, denoted by $T$, are known, the initial marking can be calculated with the following theorem, which is proved in [7], [9]. The notation $|Q|$ represents the number of elements in set $Q$. Here, $|Q|$ represents the number of transitions in the timed marked graph. In equation (4.1), the notation $⌈x⌉$ is used, referring to the ceiling function. Ceiling($x$), denoted by $⌈x⌉$, is the smallest integer larger than or equal to $x$.

**Theorem 4.2** Let $(P, Q, D, T)$ be the timed marked graph. Let $d ∈ [0, T)|Q|$ be the scheduled departure times (modulo $T$) associated with the transitions $q_i$. An initial marking is denoted by the number $m_{ji}$, referring to the number of
4.1. Timed marked graphs

Figure 4.4: An example of a timed marked graph, with a constraint between the upper and lower circuit.

tokens in place $p_{ji}$ with upstream transition $q_i \in Q$ and downstream transition $q_j \in Q$, and is determined by

$$m_{ji} = \left\lceil \frac{\tau_{ji} + d_i - d_j}{T} \right\rceil.$$  \hfill (4.1)

By adding the tokens to the timed marked graph, the construction is finished.

4.1.4 Application

As is already shown, a timed marked graph can be used to model a railway network. It describes the train lines and their dependencies because of passenger changes, synchronization and infrastructure. The transitions in such a timed marked graph represent departures and the places represent process times, such as running times, dwell times, transfer times, headway times or a combination of them. Usually, a timetable does not only consist of departure times, but also of arrival times. Arrival times can be modeled too, but this is not necessary, because they can be calculated from the last departure time and the running time over the track.

The timetable is a periodic schedule, with period 60 minutes. In subsection 4.1.3 is explained how to model a timetable and constraints. But it only models process times, not the planned departure times in the timetable. This means that a train travels along the circuit as soon as possible. The circuit contains no buffer times and trains can depart before the planned departure time. Therefore, it is possible that the minimal cycle time of the graph is less than 60 minutes. To operate the timed marked graph with a period of 60 minutes, add one transition. Define a loop from this transition to itself, with a place with holding time 60 minutes. Then, add a pair of arcs with a place in between from this extra transition to every other transition in the timed marked graph. Take the holding times of the added places equal to the planned departure times in the timetable. Because the extra transition fires only once every 60 minutes, the minimal cycle time of the graph is now increased to 60 minutes. By adding this extra transition and some places and arcs, the timed marked graph acts like the timetable, see also 3.

This timed marked graph can be transformed into a recurrence relation in max-plus algebra to calculate the performance indicators. However, it is also
possible to transform the version without extra transition into max-plus algebra. Then the scheduled departure times can be easily added by adding a vector to the recurrence relation. This is explained in subsection 4.2.3.

4.2 Max-plus algebra

If a timetable is modeled as a timed marked graph, this model can be transformed into a recurrence relation in max-plus algebra. The following two subsections give some general information about max-plus algebra and subsection 4.2.3 explains how to apply this theory to the timed marked graph to construct the recurrence relation.

4.2.1 Introduction

**Definition and notation** The max-plus algebra is based on two operators, maximization and addition, respectively ⊕ and ⊗, and given by
\[ a \oplus b = \max(a, b) \]
and
\[ a \otimes b = a + b. \]

The domain is the set of real numbers, extended with \(-\infty\), and is defined as \(\mathbb{R}_{\text{max}}\). Two elements of this set have a special notation: \(e\) denotes 0 and \(\varepsilon\) denotes \(-\infty\). The set \(\mathbb{R}_{\text{max}}\) and the two operators form the max-plus algebra, denoted as \(\mathbb{R}_{\text{max}} = (\mathbb{R}_{\text{max}}, \oplus, \otimes, e)\). The operation \(\otimes\) precedes over the operation \(\oplus\), such that the notation can be simplified as follows.

\[(a \otimes b) \oplus (c \otimes d) = a \otimes b \oplus c \otimes d\]

Powers are defined in the max-plus algebra in the following way:
\[x^{\otimes n} = \underbrace{x \otimes x \otimes \ldots \otimes x}_{n \text{ times}}\]

where \(x \in \mathbb{R}_{\text{max}}\) and \(n \in \mathbb{N} = \{1, 2, 3, \ldots\}\), while \(x^{\otimes 0} = e(=0)\) for \(n = 0\).

The operators of max-plus algebra are illustrated with some examples, including the meaning in conventional algebra.

**Example 4.3**
\[
\begin{align*}
2 \otimes 4 &= 2 + 4 = 6 \\
4 \otimes \varepsilon &= 4 + (-\infty) = -\infty = \varepsilon \\
2 \oplus 4 &= \max(2, 4) = 4 \\
4 \oplus \varepsilon &= \max(4, \varepsilon) = 4 \\
4 \oplus e &= \max(4, 0) = 4 \\
2^{\otimes 1} &= 2 \otimes 2 \otimes 2 \otimes 2 = 2 + 2 + 2 + 2 = 4 \times 2 = 8
\end{align*}
\]
4.2. Max-plus algebra

Vectors and matrices  The max-plus operators can also be applied to vectors and matrices, with elements in the set \( \mathbb{R}^{\text{max}} \). The \( m \times n \)-matrices are denoted as \( A \in \mathbb{R}^{m \times n}_{\text{max}} \) and the \( n \)-vectors as \( v \in \mathbb{R}^n_{\text{max}} \). In the max-plus algebra vectors are matrices consisting of one column, just like in conventional algebra.

The addition of compatible matrices in max-plus algebra is defined componentwise:

\[
[A \oplus B]_{ij} = a_{ij} \oplus b_{ij} = \max(a_{ij}, b_{ij}).
\]

(4.2)

The multiplication of a matrix and a scalar is also calculated componentwise:

\[
[c \otimes A]_{ij} = c \otimes a_{ij} = c + a_{ij}.
\]

(4.3)

The multiplication of two matrices is defined by multiplying the related row and column. For \( A \in \mathbb{R}^{m \times r}_{\text{max}} \) and \( B \in \mathbb{R}^{r \times n}_{\text{max}} \)

\[
[A \otimes B]_{ij} = \bigoplus_{k=1}^r a_{ik} \otimes b_{kj} = \max_{k=1, \ldots, r} (a_{ik} + b_{kj})
\]

(4.4)

for \( i = 1, \ldots, m \) and \( j = 1, \ldots, n \).

There is a special notation for the following matrices. The \( m \times n \)-matrix with all elements equal to \( \varepsilon \) has the notation \( E(m, n) \) and represents the zero element. The \( m \times n \)-matrix defined as

\[
[E(m, n)]_{ij} = \begin{cases} 
\varepsilon, & \text{for } i = j \\
\varepsilon, & \text{otherwise}
\end{cases}
\]

has the notation \( E(m, n) \). The last one is called the identity matrix if \( m = n \).

Powers in max-plus algebra can also be extended to square matrices and are denoted as \( A^\otimes n \), for any \( A \in \mathbb{R}^{m \times m}_{\text{max}} \). The definition is as follows:

\[
A^\otimes n = A \otimes A \otimes \ldots \otimes A,
\]

with \( n \geq 1 \), while \( A^\otimes 0 = E(m, m) \) for \( n = 0 \).

The next example shows the addition, multiplication and powers of matrices.

Example 4.4  If \( A = \begin{pmatrix} 1 & 5 \\ 3 & 7 \end{pmatrix} \) and \( B = \begin{pmatrix} 2 & 6 \\ 3 & 4 \end{pmatrix} \),

then

\[
[A \oplus B]_{11} = 1 \oplus 2 = \max(1, 2) = 2
\]

and the addition of \( A \) and \( B \) is

\[
A \oplus B = \begin{pmatrix} 2 & 6 \\ 3 & 7 \end{pmatrix}.
\]

Next,

\[
[A \otimes B]_{11} = (1 \otimes 2) \oplus (5 \otimes 3) = \max(1 + 2, 5 + 3) = 8
\]

and the multiplication of \( A \) and \( B \) is

\[
A \otimes B = \begin{pmatrix} 8 & 9 \\ 10 & 11 \end{pmatrix}.
\]

The third power of matrix \( A \) is

\[
A^\otimes 3 = A \otimes A \otimes A = \begin{pmatrix} 15 & 19 \\ 17 & 21 \end{pmatrix}.
\]
Chapter 4. Mathematical model

4.2.2 Matrix aspects

Communication graph To every square matrix $A \in \mathbb{R}^{n \times n}$ corresponds a communication graph, denoted as $G(A)$. This graph then has $n$ nodes. There exists an arc from node $i$ to node $j$ if element $a_{ji} \neq \varepsilon$ and the arc then has weight $a_{ji}$. Conversely, a matrix can be constructed for every graph such that the graph is its communication graph.

Regular matrix A matrix $A \in \mathbb{R}^{n \times m}$ is called regular if $A$ contains at least one element unequal to $\varepsilon$ in each row. If $A$ fails to be regular, it contains redundant rows. These rows can be left out to get a reduced regular version of $A$.

(Ir-)reducible matrix A matrix $A \in \mathbb{R}^{n \times n}$ is called irreducible if its communication graph is strongly connected. This means that for every pair nodes $(i, j)$, node $j$ is reachable from node $i$, this is that there exists a path from $i$ to $j$. If a matrix is irreducible, then every row contains at least one element $\neq \varepsilon$, so the matrix is regular.

Eigenvalue and eigenvector The eigenvalue and eigenvector are in max-plus algebra defined as follows.

Definition 4.5 Let $A \in \mathbb{R}^{n \times n}$ be a square matrix. If $\mu \in \mathbb{R}^\max$ is a scalar and $v \in \mathbb{R}^n$ is a vector that contains at least one finite element such that $A \otimes v = \mu \otimes v$, then $\mu$ is called an eigenvalue of $A$ and $v$ an eigenvector of $A$ associated with eigenvalue $\mu$.

According to this definition, the eigenvalue of a matrix can be $\varepsilon = -\infty$ and is not necessarily unique. An eigenvector can also have elements equal to $\varepsilon$, but must have at least one finite element. Eigenvectors are certainly not unique, because if a scalar is added to all components of an eigenvector, the newly obtained vector is again an eigenvector.

The next theorem shows that a finite eigenvalue of matrix $A$ is equal to the average weight of some circuit in the corresponding communication graph $G(A)$. The notation $|c|_w$ refers to the weight of circuit $c$, which is defined by the sum of the weights of all arcs in this circuit, and the notation $|c|_l$ refers to the length of circuit $c$, which is defined by the number of arcs in this circuit. This theorem is proved in [9].

Theorem 4.6 Let $A \in \mathbb{R}^{n \times n}$ have a finite eigenvalue $\mu$. Then there exists a circuit $\gamma$ in $G(A)$ such that

$$\mu = \frac{|\gamma|_w}{|\gamma|_l}.$$ 

This theorem also holds if $G(A)$ is a timed marked graph, but then the average weight is calculated different, as is already explained in subsection 4.1.2.

Theorem 4.6 only declares that there exists such a circuit, but it does not specify the circuit. If $A$ is an irreducible matrix, it can be specified by the next theorem, derived in [9].

18
4.2. Max-plus algebra

Theorem 4.7 Any irreducible matrix $A \in \mathbb{R}_{\max}^{n \times n}$ possesses one and only one eigenvalue. This eigenvalue, denoted by $\lambda(A)$, is a finite number and equal to the maximum average weight of circuits in $G(A)$, this is

$$\lambda(A) = \max_{\gamma \in C(A)} \frac{|\gamma|_w}{|\gamma|_l}.$$ 

4.2.3 Recurrence relation

If a timetable is represented as a timed marked graph, the graph can be transformed into a recurrence relation in max-plus algebra. To derive a recurrence relation in max-plus algebra a vector $x \in \mathbb{R}^{\mid Q \mid}$ is defined which describes the state of the network. Each element in this vector represents the firing time of a transition $q_i \in Q$ in the graph. Because the tokens move through the graph, the transitions fire again and again and the state vector changes at discrete time events. This dynamic property is represented by the integer $k \geq 0$. The $k$th firing times of all transitions are indicated by the state vector $x(k-1)$ and the first departure times are given by $x(0)$.

A single element of the state vector, the state of one transition, depends on its incoming arcs. If the conditions are fulfilled, all upstream places contain a token and the transition can fire. How to model this dependency is shown in the following example.

Example 4.8 This example considers two train lines, which are related by a constraint, as in figure 4.5. The served stations are modeled by transitions $q_i$ and firing time $x_i(k)$ models the $k$th departure time from these stations, for $k \geq 0$. The firing time from transition $q_1$ depends on the firing times from transition $q_2$ and $q_3$. Place $p_{q_1q_2}$ defines a holding time of $\tau_{q_1q_2} = 15$ minutes and place $p_{q_1q_3}$ defines a holding time of $\tau_{q_1q_3} = 3$ minutes. Because there already is a token in place $p_{q_1q_3}$, a firing of transition $q_1$ is only allowed 15 minutes or more after the last firing of transition $q_2$ and 3 minutes or more after the last but one firing of transition $q_3$. This can be modeled by

$$x_1(k) = \max(x_2(k) + \tau_{q_1q_2}, x_3(k-1) + \tau_{q_1q_3}).$$

A transformation into max-plus algebra yields to

$$x_1(k) = (x_2(k) \otimes \tau_{q_1q_2}) \oplus (x_3(k-1) \otimes \tau_{q_1q_3}).$$

In a more complex graph, a transition can have more upstream transitions and there can be more than one token in a place. More tokens in one place represent the fact that more trains are running behind each other on the same track. If the upstream place of a transition contains two tokens, the next firing time of this transition does not depend on the last firing time of the upstream transition, because that is the departure time of the second train on the track. The firing time of this transition depends on the last but one firing time of the upstream transition, because that represents the departure of the first train on the track.
Chapter 4. Mathematical model

Figure 4.5: The firing time of transition $q_1$ depends on the firing times of transitions $q_2$ and $q_3$.

If for example place $p_{q_1,q_2}$ in example 4.8 contains two tokens, the situation is modeled by the equation

$$x_1(k) = (x_2(k-2) \otimes \tau_{q_1,q_2}) \oplus (x_3(k-1) \otimes \tau_{q_1,q_3}).$$

For every transition in the timed marked graph, such a relation can be formulated. All these relations together, written in vector notation, form the recurrence relation in max-plus algebra describing the railway system.

However, two different recurrence relations in max-plus algebra can be distinguished. It is possible that a graph contains some transitions without an incoming or outgoing arc. These transitions are called source transition and sink transition respectively, and form a contact with the outside world. A source transition cannot be enabled by an incoming arc and must therefore be enabled by the outside world. A sink transition will be enabled, but cannot add a token to an outgoing arc and therefore delivers a token to the outside world. If a graph contains no sources, it is called an autonomous timed marked graph. Otherwise it is a non-autonomous timed marked graph.

Both recurrence relations, for the autonomous and non-autonomous case, are discussed below.

**Autonomous case** In the autonomous case, there are no sources, thus every transition has at least one upstream transition, such that a relation can be constructed. The general expression of a recurrence relation for all transitions is

$$x(k) = (A_0 \otimes x(k)) \oplus (A_1 \otimes x(k-1)) \oplus \ldots \oplus (A_M \otimes x(k-M)), \quad (4.5)$$

for $k \geq 0$. The integer $M$ is defined as the maximum number of tokens in any place. The vector

$$x(k-m) = (x_1(k-m), x_2(k-m), \ldots, x_n(k-m))^\top,$$

for $m = 0, \ldots, M$, contains the time that firing $(k-m)$ takes place for every transition. The notation $x^\top$ refers to the transpose of vector $x$. The matrices $A_m \in \mathbb{R}_{\text{max}}^{|Q| \times |Q|}$ contain the holding times of the places and are defined as

$$[A_m]_{ij} = \begin{cases} a_{ij} & \text{if the number of tokens in place } p_{q_i,q_j} \text{ equals } m, \\ \epsilon & \text{otherwise.} \end{cases} \quad (4.6)$$
4.2. Max-plus algebra

These matrices are defined with respect to a certain token distribution, for example the initial marking as defined in theorem 4.2. When all transitions have fired once, the state vector changes from \( x(k) \) to \( x(k+1) \), and the marking is again the same.

The obtained relation (4.5) is a \( M \)th order recurrence relation. It is possible to rewrite this equation to obtain a first order recurrence relation. The transformation starts with rewriting (4.5) such that \( x(k) \) only occurs at the left-hand side of the equation. If the communication graph of \( A_0 \) only contains circuits with negative weight, \( A_0 \) is defined as

\[
A_0^* = \bigoplus_{i=0}^{\left|Q\right|-1} A_i^0.
\]

Then, the recurrence relation (4.5) can be written as

\[
x(k) = (A_0^* \otimes A_1 \otimes x(k-1)) \oplus \ldots \oplus (A_0^* \otimes A_M \otimes x(k-M)),
\]

for \( k \geq 0 \). Now \( x(k) \) only occurs at the left-hand side of the recurrence relation, it can be written as a first order recurrence relation. Therefore, a new state vector \( \tilde{x} \in \mathbb{R}_{\text{max}}^{\left|Q\right|-M} \) and a new matrix \( \tilde{A} \in \mathbb{R}_{\text{max}}^{\left|Q\right|-M \times \left|Q\right|-M} \) must be defined, given by

\[
\tilde{x}(k) = (x^T(k), x^T(k-1), \ldots, x^T(k-M+1))^T,
\]

and

\[
\tilde{A} = \begin{pmatrix}
A_0^* \otimes A_1 & A_0^* \otimes A_2 & \ldots & \ldots & A_0^* \otimes A_M \\
E & \mathcal{E} & \ldots & \ldots & \mathcal{E} \\
\vdots & \ddots & \mathcal{E} & \ldots & \mathcal{E} \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
E & \ldots & \mathcal{E} & E & \mathcal{E}
\end{pmatrix}.
\]

Then, the recurrence relation (4.8) can be written as

\[
\tilde{x}(k) = \tilde{A} \otimes \tilde{x}(k-1),
\]

for \( k \geq 0 \), which is called the standard autonomous equation.

**Non-autonomous case** In the non-autonomous case, the timed marked graph contains one or more sources or sinks. Here only sources are considered. These are transitions characterized by the absence of incoming arcs and are also called input transitions. This means that constructing a relation for the input transitions is impossible, they do not have any upstream transitions. In this case, only for the transitions that are not an input transition a recurrence relation is derived. The input transitions are enabled by the outside world, not by the system.

Define the input transitions as a subset of all transitions \( Q_I \subset Q \). Then, the set \( Q\backslash Q_I \) contains all transitions in \( Q \) that are not in \( Q_I \). The dependencies between transitions in \( Q\backslash Q_I \) can be modeled as in the autonomous case. The vector \( x \in \mathbb{R}_{\text{max}}^{\left|Q\backslash Q_I\right|} \) is the state vector of these transitions and the matrices
Chapter 4. Mathematical model

\[ A_m \in S_{\max}^{\{Q \setminus Q_2\} \times \{Q \setminus Q_2\}} \text{ for } m = 0, \ldots, M, \text{ contain the holding times of the places between them. These matrices are defined in the same way as in the autonomous case. For the relation with the input transitions an extra part is added to this relation.} \]

Define the vector \( u(k) \in S_{\max}^{\{|Q_2|\}} \) as the state vector of all input transitions. The matrices \( B_m \in S_{\max}^{\{Q \setminus Q_2\} \times \{Q_2\}} \) are defined by

\[
[B_m]_{ij} = \begin{cases} b_{ij} & \text{if the number of tokens in place } p_{q_i} \text{ equals } m, \\ \varepsilon & \text{otherwise,} \end{cases}
\]

for \( m = 0, \ldots, M' \), where \( M' \) denotes the maximal number of tokens in all places downstream of any input transition. These matrices contain the holding times of all places between input transitions and other transitions. The recurrence relation for the non-autonomous case is then given by

\[
x(k) = (A_0 \otimes x(k)) \oplus (A_1 \otimes x(k-1)) \oplus \ldots \oplus (A_M \otimes x(k-M)) \\
+ (B_0 \otimes u(k)) \oplus (B_1 \otimes u(k-1)) \oplus \ldots \oplus (B_M \otimes u(k-M')).
\]

(4.13)

As in the autonomous case, this relation can be rewritten into a first order recurrence relation. For the non-autonomous graphs for which the communication graph of \( A_0 \) does not contain any circuit with non-negative weight, (4.13) is equivalent to

\[
x(k) = (A_0^* \otimes A_1 \otimes x(k-1)) \oplus \ldots \oplus (A_0^* \otimes A_M \otimes x(k-M)) \\
+ (B_0^* \otimes B_0 \otimes u(k)) \oplus \ldots \oplus (B_M^* \otimes B_M \otimes u(k-M')).
\]

(4.14)

for \( k \geq 0 \). Define the vector \( \tilde{x}(k) \), with length \( |Q \setminus Q_2| \cdot M \), and matrix \( \tilde{A} \), with dimension \( (|Q \setminus Q_2| \cdot M) \times (|Q \setminus Q_2| \cdot M) \), in the same way as for the autonomous case. And define the vector \( \tilde{u}(k) \), with length \( ((M' + 1) \cdot |Q_2|) \), and the matrix \( \tilde{B} \), with dimension \( (|Q \setminus Q_2| \cdot M) \times (|Q_2| \cdot (M' + 1)) \), as follows,

\[
\tilde{u}(k) = (u^T(k), u^T(k-1), \ldots, u^T(k-M'))^T
\]

(4.15)

and

\[
\tilde{B} = \begin{pmatrix} A_0^* \otimes B_0 & A_0^* \otimes B_1 & \ldots & A_0^* \otimes B_M' \\ \varepsilon & \varepsilon & \ldots & \varepsilon \\ \vdots & \vdots & \ldots & \varepsilon \\ \varepsilon & \varepsilon & \ldots & \varepsilon \end{pmatrix}.
\]

(4.16)

With these definitions, the recurrence relation (4.13) can be written as

\[
\tilde{x}(k) = \tilde{A} \otimes \tilde{x}(k-1) \oplus \tilde{B} \otimes \tilde{u}(k),
\]

(4.17)

for \( k \geq 0 \). This is called the standard non-autonomous equation.

**Modeling scheduled departure times** The derived recurrence relations are a description of the railway system, where trains depart if the dwell time is over and the required headway time, because of connections with other trains, is fulfilled. But usually, trains travel according to a timetable. This means
that there is also a scheduled departure time for every transition, which must be expired before the transition can fire. The departure time forms an extra condition, which can be modeled with max-plus algebra too. The departure times of train lines are the same every hour. If the vector $d(0) = d_0$ contains the initial departure times for every transition in the graph, then the departure times are modeled as $d(k) = T^\otimes k \otimes d_0$, for $k \geq 0$, with $d(k) \in \mathbb{R}^{n}_{\text{max}}$. In general, the recurrence relation is represented as

$$x(k) = A \otimes x(k-1),$$

for $k \geq 0$, with $A \in \mathbb{R}^{n \times n}_{\text{max}}$ and $x(k) \in \mathbb{R}^{n}_{\text{max}}$. The scheduled departure times are added to this relation, which gives the following recurrence relation

$$x(k) = A \otimes x(k-1) \oplus d(k),$$

(4.18)

for the departure times in accordance with the timetable.

### 4.3 Performance indicators

PETER calculates performance indicators to evaluate the robustness of a timetable. The purpose of this section is to show how these performance indicators can be derived. The input of PETER consists of three parts, the timetable of a network, the constraints and the coordinates of the stations. The previous sections showed how to model this as a timed marked graph and how to derive a recurrence relation. This section explains how this relation leads to the performance indicators, being the output of PETER.

#### 4.3.1 Critical circuit

The critical circuit of a timed marked graph is the circuit with the maximum cycle mean. This circuit determines the minimal cycle time of the network, which must be less than or equal to the actual cycle time of 60 minutes. The output of the critical circuit analysis in PETER includes for every critical circuit in the timed marked graph the performance indicators: cycle time, throughput and stability margin. It is possible to have more than one critical circuit in a timed marked graph. This can happen if different circuits have the same cycle mean or if the graph is not strongly connected, as is explained below.

A timed marked graph consists of paths and circuits, one for every train line, and of arcs between them, to model the constraints. A timed marked graph is called strongly connected if for each pair of nodes $(i, j)$ node $j$ is reachable from node $i$. It is possible that there exists a subset of nodes in the graph that are strongly connected with each other, but that the graph as a whole is not strongly connected. Then these nodes form a strongly connected subgraph. The following example shows that it is possible, but not necessary, that two strongly connected subgraphs have their own critical circuit and cycle time if the graph as whole is not strongly connected.

**Example 4.9** Assume that there are two circuits, circuit $A$ with cycle time 5 and circuit $B$ with cycle time 10. These circuits are connected by an arc with holding time zero. Both circuits are strongly connected subgraphs, but the graph as whole is not strongly connected, see also figure 4.6 and 4.7. In figure 4.6 it
Chapter 4. Mathematical model

Figure 4.6: The cycle time of circuit A is influenced by the cycle time of circuit B.

Figure 4.7: The cycle time of circuit A is not influenced by the cycle time of circuit B.

can be seen that circuit A cannot be operated with cycle time 5, because it is influenced by circuit B. This circuit has cycle time 10 and transition $q_2$ only fires once every 10 time units. Now look at figure 4.7, which is almost the same as figure 4.6, except for the direction of the connecting arrow. In this case circuit A is not bothered by circuit B and transition $q_1$ can fire every 5 time units. Circuit B can be operated with its own cycle time.

The example shows that different subgraphs have their own cycle times if they are not influenced by another circuit with a larger cycle mean. The critical circuit with the maximal cycle mean of all critical circuits, is the critical circuit of the whole timed marked graph and determines the minimal cycle time of the network.

The timed marked graph can be an autonomous or non-autonomous timed marked graph, as is explained in subsection 4.2.3. To analyze the critical circuits of the graph, only circuits are considered and no paths from sources or paths to sinks. If the graph is a non-autonomous one, it can be reduced to an autonomous graph by removing these paths. It does not influence the circuits in the graph and leads to an easier recurrence relation to work with, namely the standard autonomous equation. Therefore, the calculations of the performance indicators of critical circuits are based on the standard autonomous equation. The timed marked graph of a railway network already is an autonomous timed marked graph if all train lines form circuits and no single paths.

**Cycle time** Assume that the timetable is modeled by the following recurrence relation in max-plus algebra:

$$x(k + 1) = A \otimes x(k)$$

for $k \geq 0$ and $A \in \mathbb{R}_{\text{max}}^{n \times n}$ a square regular matrix.

If $A$ is an irreducible matrix, the corresponding timed marked graph is strongly connected. This means that there is just one cycle time for the whole graph, which can be derived from Theorem 4.7. This theorem states that if $A$ is an irreducible matrix, it has exactly one eigenvalue $\lambda(A)$, which is a finite number and equals the maximum average weight of all circuits in the corresponding timed marked graph. So by calculating the unique eigenvalue of matrix $A$, the maximum average weight and therefore the minimal cycle time is known.
4.3. Performance indicators

But if a matrix $A$ is reducible, the corresponding timed marked graph is not strongly connected, which means that different strongly connected subgraphs exist, with possibly their own cycle times. These cycle times can be found by calculating the cycle time vector, which is defined as follows.

**Definition 4.10** The cycle time vector of the sequence $x(k)$ is defined by $\eta = (\eta_1, \eta_2, \ldots, \eta_n)^\top$ if for all $j \in \mathbb{N} = \{1, 2, \ldots, n\}$, the quantity $\eta_j$ defined by

$$\eta_j = \lim_{k \to \infty} \frac{x_j(k)}{k}$$

exists.

To derive the cycle time vector, the generalized eigenvector of a regular matrix $A$ must be introduced. The symbols $+$ and $\times$ in this definition refer to vector addition and scalar multiplication, respectively, as in conventional algebra.

**Definition 4.11** The generalized eigenmode of a regular matrix $A$ is defined as a pair of vectors $(\eta, \nu) \in \mathbb{R}^n \times \mathbb{R}^n$ satisfying for all $k \geq 0$

$$A \otimes (k \times \eta + \nu) = (k + 1) \times \eta + \nu.$$  

For every regular matrix $A \in \mathbb{R}^{n \times n}_{\text{max}}$ the eigenmode exists. One way to compute these vectors, is to use Howard's algorithm, which is explained in detail in [9].

If the generalized eigenmode of a matrix is known, also the cycle time vector is known, because this vector coincides with vector $\eta$ from the eigenmode. This can be derived as follows. Assume that $(\eta, \nu)$ is a generalized eigenmode of the regular matrix $A$ and consider the recurrence relation (4.19) with $x_0 = \nu$. Then it can be shown that

$$x(k) = k \times \eta + \nu,$$

by using induction. For $k = 0$ equation (4.22) gives: $x(0) = 0 \times \eta + \nu = \nu$. If equation (4.22) is true for $k$, then for $k + 1$

$$x(k + 1) = A \otimes x(k) = A \otimes (k \times \eta + \nu) = (k + 1) \times \eta + \nu.$$

The generalized eigenmode can be seen as an extension of the eigenvalue-eigenvector pair. The cycle time vector is then an extension of the notion of eigenvalue, which represents the cycle time if matrix $A$ is irreducible.

It is thus possible to derive the cycle times of a reducible matrix via calculating the generalized eigenmode. Now the existence of a generalized eigenmode for a square reducible matrix $A \in \mathbb{R}^{n \times n}_{\text{max}}$ will be proved. Then also the existence of the cycle time vector is proved and the meaning of its elements is described.

Consider the recurrence relation as is written in (4.19) and the communication graph of matrix $A$. Renumbering the nodes in this communication graph $G(A)$, gives the normal form of matrix $A$. In this form the strongly connected components of the graph can be recognized. It is given by
Chapter 4. Mathematical model

\[
\begin{pmatrix}
A_{11} & A_{12} & \ldots & \ldots & A_{1q} \\
\mathcal{E} & A_{22} & \ldots & \ldots & A_{2q} \\
\mathcal{E} & \mathcal{E} & A_{33} & \vdots \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\mathcal{E} & \mathcal{E} & \ldots & \mathcal{E} & A_{qq}
\end{pmatrix}
\]

with, for all \(i \in q\), \(A_{ii}\) is an irreducible matrix such that the eigenvalue exists, or \(A_{ii} = \varepsilon\) such that the eigenvalue is equal to \(\varepsilon\). Every matrix \(A_{ii} \neq \varepsilon\) corresponds to a maximal strongly connected subgraph (m.s.c.s.) in the timed marked graph, denoted by [\(i\)]. The matrices \(A_{ij}\), with \(i < j\), represent the connections between these subgraphs. Every m.s.c.s. has its critical circuit with its own cycle time. But the cycle time of m.s.c.s. [\(i\)] can be dominated by the cycle time of m.s.c.s [\(j\)] if the matrix \(A_{ij}\) has a finite element. This is shown in example 4.9.

With this normal form, the recurrence relation (4.19) can be written as

\[
x_i(k+1) = A_{ii} \otimes x_i(k) \oplus \bigoplus_{j=i+1}^{q} A_{ij} \otimes x_j(k), i \in q.
\]

Note that \(x_i(k)\) is a vector, with length corresponding to matrix \(A_{ii}\). The cycle times of the different critical circuits are determined by the eigenvalues of the belonging diagonal matrices, denoted by \(\lambda_i\), or by the eigenvalue of the critical circuit of another m.s.c.s. This happens if this other eigenvalue exceeds the own eigenvalue and if the own m.s.c.s. is connected by an incoming arc to the other m.s.c.s. This is formally stated in the next theorem, which is proved in [9].

**Theorem 4.12** Consider the recurrence relations given in (4.23). Assume that \(A_{qq}\) is irreducible, and that for \(i \in q - 1 = \{1, \ldots, q - 1\}\), either \(A_{ii}\) is an irreducible matrix or \(A_{ii} = \varepsilon\). Then there exist finite vectors \(v_1, v_2, \ldots, v_q\) of suitable sizes and scalars \(\xi_1, \xi_2, \ldots, \xi_q \in \mathbb{R}\) such that the sequences

\[
x_i(k) = v_i \otimes \xi_i^\otimes k, i \in q,
\]

satisfy (4.23) for all \(k \geq 0\). The scalars \(\xi_1, \xi_2, \ldots, \xi_q\) are determined by

\[
\xi_i = \bigoplus_{j \in \mathcal{H}_i} \xi_j \oplus \lambda_i,
\]

where \(\mathcal{H}_i = \{j \in q : j > i, A_{ij} \neq \varepsilon\}\).

The scalars \(\xi_j\) represent the cycle times of maximal strongly connected subgraphs [\(i\)]. They are obtained by taking the maximum of the eigenvalue \(\lambda_i\) and the cycle times \(\xi_j\) of all m.s.c.s [\(j\)] from which an arc points to [\(i\)]. The vectors \(x_i\) in equations (4.24) can also be written as \(x_i = v_i + k \times u[\xi_i]\). Vector \(u[\xi_i]\) is a vector with all elements equal to \(\xi_i\) and with the same length as \(v_i\). By stacking the expressions for \(x_i\) on top of each other for all \(i \in q\), the following expression is derived: \(x(k) = k \times \eta + \nu\). The pair of vectors \((\eta, \nu)\) is then a generalized eigenmode as is defined in definition 4.11. The cycle time vector of matrix \(A\) in the normal form coincides the vector \(\eta\) from the eigenmode and is therefore given by

\[
\eta = (u^\top[\xi_1], u^\top[\xi_2], \ldots, u^\top[\xi_q])^\top.
\]
For a reducible matrix $A$, the cycle times of the strongly connected subgraphs are represented in the cycle time vector as in (4.26). The minimal cycle time to operate the whole network equals the maximum of all elements in the cycle time vector.

**Stability** The minimum cycle time of the whole network, denoted by $\lambda$, is the minimal time needed to complete one period. The actual cycle time of 60 minutes, denoted by $T$, should be more than the minimum cycle time. The difference between the actual cycle time and the minimal cycle time is buffer time; it can be used to compensate for delay. A timetable is stable, if a delay settles in a finite number of periods. Therefore, a timetable is stable if $\lambda < T$. How stable it is, can be presented by means of the performance indicator throughput, which is defined as:

$$\rho = \lambda / T.$$  \hspace{1cm} (4.27)

If the throughput is less than one the timetable is stable, if the throughput is equal to one it is called critical and else it is unstable. The closer $\rho$ comes to one, the less buffer time is included in the timetable and the more periods it takes to settle a delay.

Another performance indicator of the stability of the network is the stability margin, denoted by $\Delta$. This indicator defines the maximum simultaneous increase of all process times, these are the holding times in the graph, such that the train network can still be operated with cycle time $T$.

Matrix $A(T^{-1})$ is defined as

$$A(T^{-1}) = \bigoplus_{m=0}^{M} A_m T^{-m}.$$  \hspace{1cm} (4.28)

Consider a timed marked graph with holding times $\tau_{ij} - \mu_{ij} \cdot T$ and one token in each place. Then, the maximum cycle mean of this graph can be found by solving the eigenvalue problem $A(T^{-1}) \otimes v = \mu \otimes v$. The resulting eigenvalue $\mu$ equals the negative stability margin, so $\Delta = -\mu$. This result is derived in [7].

### 4.3.2 Recovery time

The recovery time between two train departures is defined as the maximal delay of the first departure, such that the second departure is not delayed. This means that the delay of the first departure must be compensated in the buffer time of the path between these two departures.

Train departures are represented in the timed marked graph by transitions. A path in the graph consists of arcs representing railway tracks between the departure points in the network, but also of arcs representing constraints between different train lines. This means that the recovery time between two departures of the same, but also of different train lines, can be calculated.

The recovery time between two train departures is defined through the accumulation of the buffer times of all tracks in the path between these departures. If there are different paths between two departures, the least accumulated buffer time defines the recovery time. The recovery times between each pair of train departures are notated in the recovery matrix $R$, which is defined in definition 4.13.
For $A \in \mathbb{R}^{n \times n}_{\text{max}}$, define
\[ A^+ = \bigoplus_{k=1}^{\infty} A \otimes k. \]
Then the element $[A^+]_{ji}$ is the maximal weight of any path in the graph $G(A)$ from $i$ to $j$. If no path exist from $i$ to $j$, then $[A^+]_{ji} = \varepsilon$.

**Definition 4.13** For system (4.19) the elements of the recovery matrix $R$ are given by
\[ r_{ji} = d_j - d_i - [A \otimes T^{\otimes -1}]_{ji}^+, \]
where the vector $d$ equals the vector $d(0) \in \mathbb{R}^n_{\text{max}}$, with scheduled departure times and where the notation $[A \otimes T^{\otimes -1}]_{ji}^+$ refers to the $ji$-th element of the matrix $[A \otimes T^{\otimes -1}]^+$. If in the graph of $A$ no path exists from node $i$ to node $j$, then $r_{ji} = +\infty$.

The recovery time between each pair of train departures is represented in $R$. By investigating all recovery times from departure $i$ to other train departures, the delay impact of train departure $i$ can be studied. This is the performance indicator that shows the maximal delay of train departure $i$ such that no other train departure is delayed. The delay impact of departure $i$ can be found in column $i$ of the recovery matrix.

Another performance indicator is the delay sensitivity of departure $i$. This studies all recovery times from other departures to departure $i$, presenting the maximal delay of the other departures without disturbing departure $i$. These recovery times can be found in row $i$ of the recovery matrix.

Finally the recovery time between train departure $i$ in different periods can be analyzed, what is called the circuit recovery time. This concerns the consequence of a delayed train departure to the same departure in a future period. This performance indicator is for every train departure $i$ defined by element $r_{ii}$ in the recovery matrix.

### 4.3.3 Delay propagation

The recovery matrix contains for every pair of train departures the maximal delay of a train departure that does not reach any other departure. However, these indicators do not tell what will happen if these times are exceeded and a delay propagates through the network. The delay propagation can be calculated by importing extra input, namely an initial delay of one or more train departures. PETER calculates the delay propagation in space and time, as result of interactions between trains.

Assume that the railway network, including the scheduled departure times, is modeled as
\[ x(k+1) = A \otimes x(k) \oplus d(k+1), \]
for $k \geq 0$, $A \in \mathbb{R}^{n \times n}_{\text{max}}$ a square regular matrix and $x(k), d(k) \in \mathbb{R}^n_{\text{max}}$. The state vector $x(k)$ contains the real departure times and vector $d(k)$ the scheduled departure times. Then vector $z(k) = x(k) - d(k)$ defines the delay of every departure. Also assume that at time $k = 0$ an initial delay is given of one or more departure times $x_i$, such that $x_i(0) > d_i(0)$. The delay propagation model is given by
4.3. Performance indicators

\[
\begin{align*}
    x(k+1) &= A \otimes x(k) \oplus d(k+1), \quad k = 0, 1, 2, \ldots \\
    d(k+1) &= T \otimes d(k), \quad k = 0, 1, 2, \ldots \\
    z(k) &= x(k) - d(k), \quad k = 1, 2, \ldots \\
    d(0) &= d_0 \\
    x(0) &\text{ given.}
\end{align*}
\]

The sum of all initial delays at different stations is the total initial delay. An initial delay of a train can continue in the next dwell stations of the train, these are called consecutive delays. Otherwise, because of connections between train lines, modeled by constraints, an initial delay can also continue to other trains. If a delay is caused by a constraint, it is called a secondary delay. The secondary delay can also continue in the next dwell stations of this train, these delays are called consecutive delays too. So, the secondary delay can be computed by summing all delays that are caused by a constraint and the consecutive delay can be computed by summing all delays that are caused by a stop in a station. Then, the sum of the secondary and consecutive delays results in the cumulative secondary delay.

By counting the number of trains that is affected by a delay, also average delays can be calculated. Dividing the total initial delay, the secondary delay and the sum of initial and secondary delay by the corresponding numbers of trains, the average primary delay, average secondary delay and average total delay, respectively, are obtained.

If the timetable is stable, the delay will be settled in a finite period, called the settling period. This performance indicator \( k_s \) is determined by

\[
k_s = \min\{k \in \mathbb{N} \mid x(k+1) = d(k+1)\}. \tag{4.30}
\]

The elements of the delay vectors show the effect of the initial delay for every distinct train departure. The other performance indicators give a more overall view of the effect of initial delays on the whole train network. This can be used to compare different versions of a timetable or to compare different strategies, as to cancel a possibility of a passenger transfer in case of delay or not.
Chapter 5

Station level

The previous chapter explains how to model a timetable as a timed marked graph and how to transform this into a recurrence relation in max-plus algebra. The performance indicators are derived from this relation.

The aim of this report is to apply PETER at station level. This makes it possible to evaluate a timetable at station level, by calculating its performance indicators. The infrastructure of a station can also be seen as a network, but is more detailed than the national network, and the headway times between train lines can be more specified at station level. This chapter describes how to model the timetable at station level and what specific information is needed. Because PETER is designed for network timetables, the model of the network at station level needs some adaptations. Unfortunately, some situations can not be modeled with max-plus algebra and can therefore not be analyzed with PETER. These limitations are discussed in the concluding section of this chapter.

If the timetable at station level is modeled, the results can be obtained. These are shown in the next chapter. It also explains the meaning and relevance of these performance indicators for the robustness of a timetable at station level.

5.1 STATIONS

This section describes STATIONS, the extension of DONS specially aimed at station level. DONS is a designer of schedules for railway networks. It models stations as black boxes, where trains can stop or run through. Only the arrival and departure times of train lines in the station are known and it is not known whether this leads to a conflicting routing inside the station or not. STATIONS is designed to make a detailed planning of the train movements inside the station. It aims at routing train lines through the station without conflicts and calculating their running times. STATIONS handles just one station at a time, so it must be performed again to generate routings for other stations in the network. A more detailed description of the model of STATIONS is written in chapter 7, see also [19].

The infrastructure of the station is given and the arrival and departure times at the platform are determined by DONS. Trains enter the station via an entrance point, travel to a platform track and leave the station via a exit point. Here, also tracks not directly alongside a platform, but parallel to a platform,
are called platform tracks. The route from entrance point to platform track is
called inbound route and the route from platform track to exit point is called
outbound route. A combination of two of these routes forms a complete route.
There are no shunting movements modeled in STATIONS, such that train lines
starting from the station just appear in the station and train lines finishing in
the station just disappear. They only have an inbound or outbound route.

Different routes can be specified from an entrance point to a platform track.
STATIONS tries to assign a feasible route to every train line, while taking into
account the conditions for safety, connections and service. A feasible solution is
found if every train line has an inbound and/or outbound route, such that there
are no conflicts and all conditions are fulfilled. If there is no feasible solution for
all train lines, a little deviation in the arrival and departure times is allowed.
However, too much deviation will make the network timetable unfeasible. If
some deviation does not help, only the non-conflicting train lines are routed
and the conflicting ones are returned.

To generate a routing of train lines through the station, STATIONS calcu-
lates the running times of the train lines inside the station. The infrastructure
is divided into sections, which are separated by joints. For every section, the
reservation time, ingoing and outgoing time are specified. The reservation time
is the time that the inbound or outbound route is reserved. This means that
no other train lines are allowed to use sections in this route and a train can
safely enter or leave the station. With these running and reservation times,
STATIONS determines whether a pair of routes is conflicting or not. Besides,
these running times also form a timetable at station level.

The results of STATIONS are represented in output files, representing informa-
tion about the infrastructure, train lines, constraints and timetable. These
files specify:

- sections, joints, entrance and exit points in the station,
- platform tracks, inbound and outbound routes through the station,
- train types and line numbers of passing or dwelling trains,
- routes and running times of train lines,
- passenger transfers,
- the planned timetable according to network level.

This information can be used to model the timetable of a station as a timed
marked graph.

5.2 Modeling method

Subsection 4.1.3 explains how to construct a timed marked graph for a timetable
at network level. The timed marked graph is transformed into a recurrence
relation in max-plus algebra, which leads to the performance indicators of the
robustness of the timetable. If it is possible to construct a timed marked graph
for the timetable at station level, its performance indicators can be derived in
the same way. Therefore, the aim is to investigate whether and how a timetable
at station level can be modeled as a timed marked graph. This is explained
5.2. Modeling method

in subsection 5.2.1. Because PETER has been developed for network level, the model for station level needs some adaptations, otherwise PETER concludes it is inconsistent. These adaptations are described in subsection 5.2.2. Because the input file needs a special format, subsection 5.2.3 describes this format.

5.2.1 Timed marked graph

Subsection 4.1.3 describes in detail the construction of a timed marked graph for network level. The first part of construction models the train lines. This forms a chain of transitions and places, connected by arcs. The transitions model the events, departures from stations, and the places model the process times in between, running times and dwell times. The second part of construction models the constraints between train lines, by connecting transitions of different train lines in the graph. The timed marked graph is finished by defining an initial marking of tokens.

Figure 5.1: A part of the infrastructure in a station, divided in 5 sections, separated and bounded by 6 joints. The train occupies three sections.

At station level the transitions and places in the timed marked graph represent different information. The infrastructure at station level is divided into sections, separated by joints, see also figure 5.1. The ingoing and outgoing times of the sections are at the same time departure and arrival times at joints, which form a timetable at station level. This partition is also used to model the timetable in a graph. The transitions represent departures from joints in the infrastructure and that the places in between represent running times and dwell times over sections. Places between transitions of different train lines still represent constraints. All this information comes from output files of STATIONS and DONS. The construction of the timed marked graph is comparable with the construction at network level, consisting of three steps.

1. A station includes only parts of train lines, which means that a train line itself cannot form a circuit in the graph any more. A train line enters the station over a joint, which is modeled by a transition. For every successive joint the train line passes, a transition is added. A transition is connected to the next one by an arc, a place and an arc. The holding time of this place is the sum of the running time over the section between these joints and the dwell time at this section. Most sections in a station are just passed by a train line such that their dwell time is zero. However, if a train dwells in the station, it is possible that it occupies more than one section. Then the dwell time is included in the holding time of just one section, the section where the head of the train dwells, otherwise it is counted double. The joint over which the train line leaves the station models the last transition of the train line in the graph. So every train line, entering and/or leaving the station, is modeled by a chain of transitions and places, connected by arcs.
Chapter 5. Station level

2. Constraints model dependencies between train lines, due to safety and service requirements, just as at network level. They connect the train lines in the graph, represented by places and arcs. This makes a circuit in the graph possible. The holding time of the place represents the connection or headway time between the events. The following properties at station level are further explained in 5.2.2.

Trains enter and leave a station according to the 'route locking, sectional release' principle. This means that an inbound or outbound route is reserved as a whole before a train enters or leaves the station. And that the sections of the route are released one by one if the train has passed.

PETER is developed to evaluate timetables at network level and it checks the input for consistency. But some of the input at station level is not consistent if it is seen as a network. Therefore, the model must be adapted, such that it is consistent at network level. These adaptations are needed at crossings and passenger connections.

The constraints are not achieved in STATIONS, but in DONS. Fixed norms are used to determine headway times between train lines. However, STATIONS results in detailed information about the occupation and release of sections, such that the minimal headway times can be calculated. Dwell times and connection times are determined by norms as used in DONS.

3. At network level, the finishing part of constructing a timed marked graph, is defining the initial marking of tokens. Formula (4.2) can be used at station level too. But the timed marked graph at station level is a non-autonomous graph, it includes sources and sinks. This means, that some transitions have no incoming or outgoing arcs. To keep the graph alive, which means that every transition will once be enabled again, the sources must be enabled by the outside world every period and the sinks deliver tokens to the outside world. PETER realizes this automatically.

5.2.2 Adaptations

It is possible to model a timetable at station level as a timed marked graph. However, some modeling aspects are a bit different than at network level. Therefore, the model needs some adaptations. This subsection describes the subjects that need some extra attention.

Reserving routes. Before a train enters or leaves a station, its inbound or outbound route must be reserved, which is done by the 'route locking, sectional release' principal. Following this principal, a route is reserved as a whole and is released section by section. Reserving a route is only possible if all sections are free from other trains and are not reserved already. Then the train can enter or leave the station safely. If a section is released, it can be reserved for another route. A train is not allowed to enter its route until it is completely free and reserved. Also if the first sections are free, the train has to wait until the entire inbound or outbound route is reserved.

This principle must be modeled in the timed marked graph as well. This means that a constraint does not only connect the common sections of two
5.2. Modeling method

Figure 5.2: Infrastructure where different routes have two common sections.

routes, but also the sections before the common sections. Consider the infras-
structure represented in figure 5.2. If the first train line travels from section A
to G and the second one from section C to H, then they have sections E and F
in common. Figure 5.3 shows a model where just the common sections, which
are bounded by dotted lines, are bounded by the constraint. In this model, the
second train can enter its route although its not completely free. Figure 5.4
shows a model where the principle 'route locking, sectional release' is applied.
The second train has to wait at the start of its route until the complete route
is free and reserved. Then it can enter its route.

Figure 5.3: Model of two routes with some common sections, which are bounded
by dotted lines. The constraint just connects the common sections.

Figure 5.4: Model of an infra constraint using the 'route locking, sectional
release' principle. The numbers represent the holding times in seconds.

Common joint The consistency check of PETER checks at network level
whether the two segments of the train lines in a constraint have a common
station in the railway network. This, to assure that the two train lines in the
constraint arrive at or depart from the same station. If not, there is no need for
a constraint, because train lines are in different stations.

But at station level, a constraint is defined between two section. Then, this
check insists that a constraint between two train lines is only possible if their
sections in this constraint are linked by a joint. In practice, this is not always
the case. Assume for example, that there is a constraint between two routes that have one or more common sections. It specifies a headway time between the last released section of these common sections and the first section of the next route. In most cases, these sections are not directly connected and the consistency check of PETER rejects the constraint.

Because it is very complex to remove this check from PETER, the input must be adapted. For a feasible input, the constraints must be defined in such a way that the sections where the constraint is specified have a common joint. This can be modeled by shifting the arcs in the timed marked graph and adapting the holding time of the place. This is shown in the following example.

**Example 5.1** In figure 5.4 is shown how a constraint should be modeled to approach the real situation of the 'route locking, sectional release' principle as best as possible. It shows that 12 seconds after the departure from the last common joint the second train is allowed to enter its route. The problem is that the sections connected by these constraint, F and C, do not have a common joint. The constraint can be defined between other sections and the holding time is adapted. If the constraint is shifted and points from section B to C, the headway time increases. Not only the switch time must be spend, but also the time needed to travel sections E and F by the first train. This means that the headway time becomes 23 instead of 12 seconds. If the constraint is shifted again, to the situation in figure 5.5, the headway time increases again because the holding times of sections C and D are added. This situation does not model the real situation, but gives the same result in max-plus algebra.

![Diagram of a train system with constraints and sections](image)

**Figure 5.5:** The constraint is modeled at sections with a common joint.

**Crossings** In the paragraph above is explained that a constraint between two train lines must be defined upon two sections of their routes with a common joint. It is possible that two train lines have a common section, without common joints, namely at a crossing. Consider the infrastructure in figure 5.6 and two routes crossing each other. Then, these routes have no common joint, because one train line travels along joints 113 and 115 and the other train line along joints 111 and 114. This situation can be solved by adding an extra joint at the middle of the crossing. In this way, the section is divided into four new sections and the routes have sections with a common joint. This new situation is illustrated by figure 5.7, where the new joint has number 200. By changing the infrastructure in the model also the input must be adapted, because this also causes a change in the series of sections of the routes. Moreover, also new ingoing, outgoing and reserving times of these sections must be defined.
Passenger connections

Also for passenger constraints, PETER checks whether there is a common joint in the routes. But, there is no common joint for passenger changes, because trains dwell at different platform tracks. They do not even have a common section to specify the constraint at.

A passenger connection between two train lines can be defined by modeling a transfer train line. This is just a fictitious train line, representing the flow of passengers between these train lines. It travels along a fictitious section between the arrival joint of the feeder train line and the departure joint of the connecting train line. The passenger constraint is replaced by two new constraints, one defining a connection time between the feeder train line and the transfer train line and one defining a connection time between the transfer train line and the connecting train line. The sum of these connection times is the original connection time between the feeder and connecting train line.

The connection constraints have a different format in the input file from the infra and synchronization constraints. A connection constraint always just specifies one connection time between the arrival and departure of two train lines in the station. For other constraints two headway times are specified, one for the possibility that the first train line is the feeder line and the second one the connecting line and one possibility for the other way around. Besides, PETER expects two departure sections as input for a connection constraint, whereas other constraints can be defined for both arrival and departure at a section. The connecting train line of a passenger connection departs from the station. This makes it easy to define a departure section, namely the departure section alongside the platform. But for the feeder train line, a departure section must be found in the past. The last departure is the departure from the previous station, but this is beyond the boundary of the current station. Therefore, the entrance point in the station can be seen as the departure section of the feeder train line.

Headway times

Headway times define the minimal time between two successive events of different train lines. In the railway network, it is used to avoid collisions between ongoing trains or by overtaking. Headway times are mainly specified to avoid conflicts and to assure safety.

The constraints at network level, as used in DONS, are specified by norms. This means that for all situations where two train lines leave a station, a fixed norm determines the headway time. However, it is possible that the real head-
Chapter 5. Station level

way time differs per situation, station and train type.

The output files from STATIONS include a lot of detailed information about the running times of train lines in stations. Also the routes, and therefore the common sections, are given. This makes it possible to calculate the exact headway times between each pair of train lines. This section explains how to do this, which is illustrated by example 5.1.

The headway times are calculated as follows. Assume that a constraint is defined between the departure from Section1 by Line1 and the departure from Section2 by Line2, such that these sections have a common joint. Every headway time consist of three parts, based on the sections on which the constraint is specified. Headwaytime12 is obtained if Line1 is leading line and Line2 is following line. It must cover all time that is needed to release the common sections by Line1, to possibly switch a switch and to enter the second route until Section2 by Line2. Headwaytime21 is derived in the same way, but now with Line2 as leading line and Line1 as following line.

Because the output of STATIONS gives the ingoing and outgoing times of every section, every part of the headway time can be calculated in detail. The first part, the release by Line1, is the difference between the departure time from Section1 and the outgoing time from the last common section in the route of Line1. The second part is the switch time, defined at 12 seconds. The third part, entering the second route until Section2 by Line2, is the difference between the ingoing time of the first section of the route of Line2 and the departure from Section2. This calculation is illustrated by example 5.1 and figure 5.8.

Example 5.1 A constraint is defined between Line1 and Line2 to avoid a collision, see also figure 5.8. Their common sections are sections B up to and including E. Section1 and Section2, on which the constraint is specified, are both section E, such that they have a common joint. Headwaytime12 consists of: the time to release E by Line1, the switch time and the running time of Line2 over F till E. Headwaytime21 consists of: the time to release all common sections E up to and including B by Line2, the switch time and the running time of Line1 over A up to and including D, till section E.

![Figure 5.8: Headway times depend on the common sections of two routes.](image)

5.2.3 Input file

The input file specifies all information that PETER needs to calculate the performance indicators of a given timetable. PETER has been developed for network level, but for station level the format of the input file must be the same. This means that the appearance is the same, but the content is different. The input file, used for the case study of ’s Hertogenbosch is printed in appendix C.1.
5.2. Modeling method

Every input file starts with defining the cycle time and recovery threshold, this is the maximal recovery time that is presented in the output of PETER. In general they are both 60 minutes, at network and station level. Then three components follow, namely Coordinates, Timetable and Constraints, specifying all important information about the timetable. Every component is represented as a matrix. The three paragraphs below explain these components for station level.

The input files for network and station level correspond in format and in headings of columns and components. The differences between the network and station level can be found in the content of all three components. The first component, Coordinates, specifies all stations at network level, or all joints at station level. Every station has its own abbreviation in Dutch, but the joints of a station do not have a name, they are numbered. This difference appears in the whole input file. Another difference between the network and station level can be found in the running times in the timetable. At network level the running times are at least some minutes, while at station level they are mostly less than 10 seconds. Both running times and headway times are usually defined in minutes at network level, because DONS defines them in this way. However, the output from STATIONS is more detailed, such that at station level these times are defined in seconds.

Coordinates This component of the input file describes the joints in the infrastructure. These are represented by the transitions in the timed marked graph. PETER uses the coordinates to draw a picture of the network. Figure 5.9 shows an example of a line of the component Coordinates. For every joint, the following information is defined.

- Station: What stations are at network level, are joints at station level. Because joints do not have a name and abbreviation, they have a number to identify them.

- Type: PETER expects a station type here, like in the network input file. Possible types are IC (InterCity), IR (InterRegional), AR (Agglom-Regional), and GE (GoodsYard). The different types are represented by different symbols in the picture. This can be used at station level to distinguish points where a train stops or leaves the station, defined by the type IR and represented by PETER as a circle, from other joints, defined by the type AR and represented by PETER as a smaller circle.

- X- and y-coordinates: They specify the place in the infrastructure of each joint and are used to draw a picture of the infrastructure. The coordinates must be integer numbers, otherwise PETER does not recognize them.

An example of a line in Coordinates is shown in figure 5.9. It tells that joint 1 has coordinates (3632, -821) and that it is not a point where a train line stops.

Timetable The timetable describes the planned place and time of trains in the station. This is modeled by a chain of transitions and places in the timed marked graph. The holding times and planned departure times represent the time-aspect. A train travels through the station section by section, therefore
**Chapter 5. Station level**

Coordinates:

<table>
<thead>
<tr>
<th>station</th>
<th>type</th>
<th>x-coordinate</th>
<th>y-coordinate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AR</td>
<td>3632</td>
<td>-821</td>
</tr>
</tbody>
</table>

Timetable:

<table>
<thead>
<tr>
<th>line segment</th>
<th>origin</th>
<th>destination</th>
<th>departure</th>
<th>run time</th>
<th>dwell time</th>
<th>dwell type</th>
<th>marge</th>
<th>included</th>
</tr>
</thead>
<tbody>
<tr>
<td>036H01</td>
<td>55</td>
<td>211</td>
<td>8</td>
<td>36:00</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
</tr>
</tbody>
</table>

Connection constraints:

<table>
<thead>
<tr>
<th>feeder line</th>
<th>feeder segment</th>
<th>connecting line</th>
<th>connecting segment</th>
<th>connection time</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>501H01</td>
<td>501</td>
<td>350H02</td>
<td>16</td>
<td>2:00</td>
<td>P</td>
</tr>
</tbody>
</table>

Other constraints:

<table>
<thead>
<tr>
<th>line segment</th>
<th>line segment</th>
<th>headway12</th>
<th>headway21</th>
<th>event1</th>
<th>event2</th>
</tr>
</thead>
<tbody>
<tr>
<td>350H01</td>
<td>37</td>
<td>442H01</td>
<td>37</td>
<td>0:34</td>
<td>1:00</td>
</tr>
</tbody>
</table>

Figure 5.9: One line from each component of the input file.

The timetable is defined per section. For each section, ten attributes are defined, which are described below. One line of a timetable is illustrated in figure 5.9.

- **Line**: All sections in the complete route of a train line have the same line number, defined in this column Line. This number includes the serial number, direction and frequency of the line. In figure 5.9, from Line 036H01; 036 gives the serial number, H shows that it is an outward journey (T means the return) and 01 means that it is the first train on this train line in this period. In the input file, all sections with the same line number must succeed each other, see also Origin and Destination.

- **Segment**: The number of the section which is described by this row in the input file, which identifies this section.

- **Origin**: The joint across which a train of this line enters the section. This must be the same joint as the Destination of the previous section.

- **Destination**: The joint across which a train of this line leaves the section. This must be the same joint as the Origin of the next section.

- **Departure**: The departure time specifies at what time a train is planned to enter this section, the time it leaves the Origin.

- **Run time**: The time to travel from Origin to Destination is called the Run time. This can often be calculated as the difference between the departure times of the current and next section, because at most sections a train line does not stop. The Run time of the last section in a route is often difficult to define, because the next section does not exist. This can be estimated or approached with the occupation time. The occupation time of a section is the difference between the departure time of the trains head from the Origin and the departure time of the trains tail from the Destination. The running time is defined by mm:ss (minutes and seconds), which also holds for the dwell time.

- **Dwell time**: The dwell time is zero for most sections. If the difference between the departure times of two successive sections seems to be large,
there might be a stop in between. However, the exact running and dwell time of this section are not known. An estimation of the running time can be made with the running times of previous sections, such that the dwell time equals the other part of the difference between the departure times of the successive sections.

- **Dwell_type**: The dwell type is specified for the Destination and is defined by one of these possibilities: R (Run), S (Stop), T (Turn) and E (End). These are the types used to describe the dwell types of stations at network level. At station level, most joints have type R. If the dwell time is unequal to zero, the dwell type is S. The end of an outbound route is always marked by a joint with type E.

- **Marge**: This defines the percentage of the running time that is used as slack. At station level, the running times are calculated by STATIONS. These are minimal running times, such that there is no slack included and the marge is zero for every section.

- **Included**: In most cases this will be YES for every section. Only if one train line must be eliminated for a certain study, it is not necessary to delete the train line from the input file, it is enough to turn this option into NO. Note that all sections of a train line must always have the same value, otherwise just a part of the line is modeled and a part is eliminated. This is not possible.

An example of a line in Timetable is shown in figure 5.9, which means that train 036H01 departs at 36:00 minutes from joint 211 and travels 2 seconds over section 55 to joint 8. There is no stop at this section, no marge included and this line must be included.

**Constraints** Constraints are designed to model the dependencies between train lines, both infra constraints and service constraints. In the timed marked graph, the constraints connect transitions of different train lines. The connection constraints, both passenger and rolling stock connections, have a special format in the input file. These are described first. The other constraints are defined in another format. An example of a connection constraint in the input file is represented in figure 5.9.

Connection constraints:

- **Feeder_line**: A constraint is defined between two routes in the station. It connects a feeder train line on an inbound route to a connecting train line on an outbound route. This item defines the line number, Line, of the feeder train line. It coincides with the line number in the timetable.

- **Feeder_segment**: This defines the departure section of the feeder train line. In the paragraph ‘Passenger connections’ of subsection 5.2.2 is described that this is the entrance section in the station.

- **Connecting_line**: This item defines the line number of the connecting train line.
Chapter 5. Station level

- **Connecting segment**: This defines the departure of the connecting train line. This is the first section of its outbound route, the first departure after the connection is realized.

- **Connection time**: This gives the minimal connection time between the arrival of the feeder train line and the departure of the connecting train line.

- **Connection type**: The type of a connection constraint is represented by a P, for passenger connections, or a C, for rolling stock connections.

An example of a connection constraint is shown in figure 5.9. It defines a passenger transfer of 2 minutes between the feeder train 501H01, departing from section 501, and the connecting train 350H02, departing from section 16.

**Other constraints:**
The first four items represent the same information as the first four of the connection constraints.

- **Line1**: The line number of the leading train line.

- **Segment1**: This item defines the section of the leading train line on which the constraint is defined.

- **Line2**: The line number of the following train line.

- **Segment2**: This item defines the section of the following train line on which the constraint is defined.

- **Headway12**: This headway time is obtained when Line1 is the leading train line and Line2 is the following train line. It specifies the minimal time that must be spend after Event1, before Event2 is allowed to take place. In the paragraph 'Headway times' of subsection 5.2.2 is described how this can be calculated and that it is usually different from Headway21.

- **Headway21**: This specifies the minimal time that must be spend after Event2, before Event1 is allowed to take place. This headway time is obtained when Line2 is the leading train line and Line1 the following train line.

- **Event1**: A constraint defines a connection between two events, which are specified by a section and an event, namely a departure or arrival. This item defines the event of Line1 on Segment1.

- **Event2**: This item defines the event of Line2 on Segment2.

The last line in figure 5.9 shows how a constraint is defined: a headway time of 34 seconds is specified between the departure of leading train 350H01 at section 37 and the departure of the following train 442H01 at section 37. At the same time, this constraint also specifies a headway time of 1 minute between the departures of these trains when train 442H01 is the leading train and 350H01 is the following train.
5.3 Limitations

A model is an approximation of reality, not all complexities can be modeled. This also holds for the max-plus model used in PETER. At network level, PETER is limited because it can only handle a fixed order of trains. At station level, an extra limitation showed up, namely that the increased running times as result of a delayed train can not be modeled. This section discusses both limitations.

5.3.1 Fixed order of trains

The problem of the fixed order of trains was already distinguished at network level. It forms a problem in case of a delayed train, because the order of trains can not be changed such that the delayed train forces other trains to wait. This leads to delays for other trains too, such that the delay propagates through the network. In reality it is possible to change the order of trains. This means that trains, planned to enter the station after a delayed train, get right of way and enter the station without (much) delay. This can not be modeled with max-plus algebra, because this method bases each departure on one or more earlier departures. As long as a previous departure does not happen, the successive departures can not happen either. Consider the timed marked graph in figure 5.10. If the upper train has a delay, such that the holding time of $p_{q_1q_2}$ increases very much, transition $q_3$ is enabled long after the planned time. Although the holding time of $p_{q_5q_6}$ has passed, transition $q_6$ is not enabled until $q_3$ is enabled and the holding time of $p_{q_7q_8}$ has passed. This causes a delay for the lower train too.

The tool PETER is operational for some years and it has been a subject of study in many reports. The limitation of fixed order of trains, is already discussed in [17]. This PhD thesis describes a solution using an extension of the max-plus algebra, namely $(\min, \max, +)$ algebra.

This algebra makes it possible to model the first arriving train, also if this changes the order trains. However, the consequence of using this model, is that the routes are not fixed. If a train is delayed, such that the order is changed and its following train becomes first, this train continues its way following the route of the delayed train. And the delayed train continues along the second route. In [17] is shown that it is not possible to change the order of trains, while keeping the routes fixed. This is not acceptable in practice, because passengers have a fixed destination.

Figure 5.10: In case of a fixed order of trains, a delay of the upper train is probably propagated to the lower train.
To assure passengers they will reach their destination in a straight way, the trains have to stop in the station such that passengers can change trains. This can be modeled with a service constraint. However, this subsection started with the aim of minimizing delays by modeling changing of trains. The result is that modeling changing of trains is possible, but without obtaining the predefined aim. Because trains have to wait for each other, both trains depart with a delay. Besides, if the possibility of changing the order of trains is applied at more stations in the network, the number of service constraints increases and trains have to wait more often, such that the traveling time of passengers increases. This will make the system less stable, maybe even unstable, and is a negative consequence for the passenger. In [17] is concluded that this solution can not be used to model railway networks.

The problem of fixed order of trains cannot be modeled with (min,max,+) algebra. This means that also at station level this problem will stay. Trains have to wait at their entrance point or alongside the platform if a preceding train is delayed. This does not always agree with the real situations in station. As long as the delays are not too big, analysis with PETER gives a good indication, because in that case trains have to wait for real too. Differences appear when the delays increase.

The limitation of fixed order of trains can also be seen as a boundary condition of the model. It can be used to analyze for which delays the consequences are still acceptable and for which delays the consequences are not acceptable anymore and the order of trains must be changed.

### 5.3.2 Increased running time

Another limitation that shows up at station level, is the problem of increased running times. Running times increase when a train has to slow down or has to make an extra stop. Not the waiting time caused by this stop is aimed at here, but the extra time caused by slowing down and accelerating before and after the stop. The train needs more time to travel the same distance. This limitation is only important when the delay propagation is studied. Calculating performance indicators for the critical circuit and calculating the recovery times are both based on the planned timetable.

A train has to stop if the preceding train is more delayed than the train itself, such that its route is not released yet. The condition that the preceding train has a delay is not sufficient. If the train itself has a delay as well, its route can still be released in time. So the question whether running times increase or not, is dependent on the amount of delay of both the preceding train and the train itself. Increased running times can also occur at network level. But, because then the extra running time is very small in comparison with the planned running time, it is not seen as a limitation. At station level the extra running time is of the same size as the planned running time. Thus, an extra stop is a limitation at station level, because it can not be trivially modeled with max-plus algebra.

Research in reports about PETER did not deliver any information about the limitation of increased running time. Probably because applying PETER at station level is not tried before. Therefore, some time has been spend on this subject during this study. Two options to model this limitation has been investigated, one by adding a fictitious train and one by adding an extra constraint.
Before both options are explained, a more detailed description of the situation follows.

Whether the running time of a train increases depends on the delay of the previous train and the train itself. To study the consequences of a delay with PETER, an initial delay must be given, because there are no effects in the model that can cause a delay. This initial delay can cause secondary delays by constraints with other train lines. Therefore, the condition that the preceding train has more delay than the train itself, is not known on beforehand. After calculating the delay propagation, the delays for different trains are known and the condition can be tested. This makes the problem to model this limitation like an if-then-else statement: if the condition is fulfilled, then the running time increases, else the running time stays the same. Both options can happen and therefore both options must be modeled. The choice is made during the execution of the delay propagation.

**Fictitious train** In a timed marked graph both alternatives, planned and increased running times, can be modeled by adding a fictitious train. This train travels along the same route, but has an increased running time over one or more sections. Figure 5.11 shows an example, where place $p_{5,6}$ represents the planned running time and $p'_{5,6}$ the increased running time. It is also possible to model a fictitious train for the whole route. In transition $q_6$ must be chosen which arrival time is used further. This transition fires if all holding times of the preceding places are fulfilled, so when both the planned and increased running time are passed. But in practice, most of the time the planned running time is realized. This means that transition $q_6$ should already fire if only the planned running time has passed. This is the minimal instead of the maximal point in time. The minimum can be calculated by adding the min-operator to the max-plus algebra, together forming the (min,max,+)-algebra. Then, $x_6(k) = \min(x_5(k) + \tau_{56}, x_5(k) + \tau'_{56})$ leads to the planned running time and $x_6(k) = \max(x_5(k) + \tau_{56}, x_5(k) + \tau'_{56})$ leads to the increased running time. One of these equations succeeds, but which one is not known on beforehand. The choice for one or the other equation is based on the condition, on the delays of both the first and second train. Now the problem is to model the condition.
with \((\min, \max, +)\) algebra. The next paragraph tries to find a way to model this condition in a constraint.

**Extra constraint** The condition to be modeled, is that the preceding train, train1, has more delay than the train itself, train2. Because it depends on the departure times of two trains, it is difficult to represent it in a timed marked graph. Therefore, it is tried to model an extra constraint directly in a max-plus equation.

The constraint must only add some running time if the condition is true, so if the delay of train1 is larger than the delay of train2. Observe the difference between these delays; delay of train1 minus delay of train2. This will be positive if the condition is true and negative otherwise. So if this difference is positive, some running time must be added. This can be represented in the following form \(\max(0, \text{delay}_1 - \text{delay}_2)\). The delays can be calculated, because for every joint \(i\) in the infrastructure, the real departure time \(x_i\) is calculated and the scheduled departure time \(d_i\) is known. The difference between these departure times determines the delay of a train and the equation is as follows

\[
\text{max}(0, (x_1(k) - d_1(k)) - (x_2(k) - d_2(k))).
\]

This can be written in terms of \(\oplus\) and \(\otimes\) as follows:

\[
0 \oplus [(x_1(k) \otimes d_1^{-1}(k)) \otimes (x_2(k) \otimes d_2^{-1}(k))^{-1}].
\]

However, this expression just represents the condition, now it must be added to an equation of a transition of train2. Then, an expression with the max-operator inside a max-plus equation is obtained, which results in a non-linear equation in max-plus algebra. Therefore, it can not be used in the model.

Although it is possible to represent this condition with the given information, equations (5.1) and (5.2) are not quite correct. If there is some buffer time planned between the arrivals of train1 and train2 in the station, then this buffer time can be used to compensate (part of) the delay of train1. This means that some extra running time for train2 is possibly not necessary. This can be modeled by subtracting the buffer time from the second expression in equation (5.1). However, it is still not possible to write the resulting equations as a linear system in max-plus algebra.

**5.3.3 Simulation**

The condition as stated in subsection 5.3.2 cannot be modeled in a linear max-plus equation and can therefore not be handled in PETER. However, this situation can be modeled in a simulation tool. Simulation is a method based on imitating a system or network by repetitive calculating events. It can also handle equations like equation (5.1).

An advantage of simulation is that it can deal with very detailed and complicated systems, mostly easier than an analytical tool. An analytical tool is restricted to its modeling method, like PETER is restricted to the max-plus algebra. Another advantage is that a simulation tool can use different distributions to model disturbances, even different distributions in the same model, while an analytical tool often has restrictive assumptions about primary disturbance distributions.
5.3. Limitations

The advantage of analytical tools is that they calculate the results very fast, while a simulation tool needs a lot of time to run a single experiment. PETER, for example, presents the results within some minutes. A disadvantage of a simulation tool is that, because of the complex systems it can manage, it often brings complexity for the users.

The simulation tools SIMONE and FRISO are operational at ProRail. SIMONE (= SImulation MOdel of NEtworks), see [11], is developed for punctuality research. It is possible to compare different timetables and to study the effects of a new infrastructure. A disadvantage of SIMONE is that the stations are not modeled in detail. Therefore, FRISO (Flexibele Rail Infra Simulatie Omgeving), a flexible rail infrastructure simulation environment, is designed, which specially aims at station level. It contains a detailed description of the infrastructure and safety system, which makes a detailed study of the punctuality at station level possible. This tool is capable to handle the problem of increased running times.
Chapter 6

Results of case studies

The previous chapter explained how to analyze a timetable at station level. The most important part of modeling is the timed marked graph, because the max-plus algebra can then be applied in the same way as at network level. It also pays attention to the input file, because that needs a special format.

To test the theory of the previous chapter, two case studies have been investigated. The first case study is of station Roosendaal, because it is not too complicated. This was mainly a study into the way of modeling a station and a practical lesson about what problems and limitations can arise. With this information, it was possible to analyze a somewhat larger station in less time, namely ’s Hertogenbosch. During this case study, the modeling was already more familiar, such that the focus was more on the results of this study. With these two case studies the questions stated in the 'Problem description' can be answered. Both studies are discussed in the sections below. These results show that it is possible to analyze a timetable at station level with PETER.

6.1 Interpretation of the results

PETER represents the performance indicators as a number or in minutes and seconds. But just a number does not tell how good or bad the robustness of a timetable is. To valuate the number, a norm is necessary. It is not easy to determine such a norm, because it will be different for every station and maybe different every year. However, PETER is very useful to evaluate the robustness of timetables under construction. By adding, deleting and modifying aspects of the timetable and infrastructure, different versions of the timetable are obtained. The various versions can be compared by using PETER at station level. The resulting performance indicators show the advantages and disadvantages of the different versions. Then, a norm is not necessary, but it can be used to guarantee a minimal level of service and capacity.

In the case studies just one version of each timetable is analyzed, because no adapted versions were available. Station Roosendaal is valuated with expert opinions and station ’s Hertogenbosch with realization data.

The data used for these case studies is generated by STATIONS. This program only routes train lines that are scheduled by DONS. Shunting movements inside the station are not modeled and therefore not routed. This means that
it is possible that there are some more actual train movements than modeled train movements. This makes also a shift of critical circuits possible. Although shunting movements are not scheduled, planners do consider these movements during the planning. They obtain a timetable with enough time to perform the shunting movements. If the shunting movements are available, they can be added in PETER to obtain a more complete set of train movements.

6.2 Roosendaal

The data used for the case study of Roosendaal is from a model of 2010. The timetable of this model is already developed at station level, such that all information needed for this case study is available. Besides, Roosendaal is not a very large station, which makes the model not too complicated. The disadvantage of using data from this model is that there are no realization data available to compare the results of PETER. However, the results can be verified with expert opinions. A schematic representation of station Roosendaal is depicted in appendix B.1.

Because Roosendaal is the first case study in which the methods of the previous chapter are applied, the main results are the solutions to the problems and limitations obtained during the study. Most problems are solved by adaptations to the model, as is written in subsection 5.2.2. But because every modeling method has its limitations, not all problems can be solved. These limitations are discussed in section 5.3.

The final model of station Roosendaal, with adaptations, resulted in the performance indicators of its timetable. These are discussed in subsection 6.2.1 below. Then, subsection 6.2.2 includes a short verification of these results by expert opinions.

6.2.1 Results

The most critical circuit of Roosendaal consists of headway times between different train lines. It also includes one stop, but this is between the shunting movement and departure of the same train line. Figure 6.1 shows the output of PETER after the critical circuit analysis. Only that part of the infrastructure that is used by train lines in the timetable is represented. It is constructed by connecting the joints in a route with a straight line, which is not always the real situation. The train lines in the most critical circuit are colored red. They do not form a circuit themselves, but they do form a circuit in the timed marked graph, with the constraints between them. This circuit has a cycle time of 28:16 minutes and a stability margin of 3:58 minutes. This means that the timetable of this station can be performed with a period of 28:16 minutes, or that every headway time can increase with maximal 3:58 minutes to fit the current period of 60 minutes. The details are presented in table 6.1. It shows for every train line in this circuit which part of its route contributes to the critical circuit, after how long the next part of the critical circuit can be performed and by which kind of constraint it is connected to the next part of the critical circuit.

All train lines in this circuit are leaving or entering the station over the same crossing, which is depicted by fat lines in appendix B.1. This alternating behavior partly causes the large headway times. Furthermore, Roosendaal is
the final destination of the arriving trains in the critical circuit. This means that their occupation time of the sections along the platform is longer to let passengers leave the train before shunting.

Most arrival and departure times are spread over the hour, such that a fast settling of delays can be expected. But the recovery times show that this is not always true.

Two other critical circuits are mentioned in this station, not strongly connected to the first one or to each other. These circuits are less relevant, because they include just one or two trains from the same train line.

Studying the recovery times shows that some train lines already have impact on other train lines with a delay of less than one minute. This happens also in the critical circuit, where the arrival from the 213T01 starts just 1:27 minutes later than the departure of the 341H01. Because of a headway time of 55 seconds, there is just a recovery time of 32 seconds. It seems that the 213T01 can afford some delay, because the 291T01 thereafter departs more than 5 minutes later. But, there is a large headway time between these train lines, because of a long occupation time of the 213T01. Consequently there is again a short recovery time of just 48 seconds. This shows that the recovery times can give a more accurate description of the sensitivity of train lines for other lines than the critical circuit analysis.

A short recovery time means a fast propagation of delays. In the case of the three train lines above, a delay of 5 minutes of the 341H01 causes a delay of 4:28
Chapter 6. Results of case studies

<table>
<thead>
<tr>
<th>line number</th>
<th>origin and destination</th>
<th>route</th>
<th>departure</th>
<th>connection time</th>
<th>connection type</th>
</tr>
</thead>
<tbody>
<tr>
<td>213H01</td>
<td>Rd - Zl</td>
<td>outbound</td>
<td>04:00</td>
<td>0:45</td>
<td>headway</td>
</tr>
<tr>
<td>341H01</td>
<td>Rd - Dn</td>
<td>outbound</td>
<td>09:00</td>
<td>0:55</td>
<td>headway</td>
</tr>
<tr>
<td>213T01</td>
<td>Zl - Rd</td>
<td>inbound</td>
<td>16:00</td>
<td>3:00</td>
<td>stop</td>
</tr>
<tr>
<td>291T01</td>
<td>Rd - Hfd</td>
<td>platform</td>
<td>19:00</td>
<td>1:49</td>
<td>headway</td>
</tr>
<tr>
<td>428H01</td>
<td>Sloe - Valb</td>
<td>outbound</td>
<td>29:00</td>
<td>4:35</td>
<td>headway</td>
</tr>
<tr>
<td>341H02</td>
<td>Rd - Dn</td>
<td>outbound</td>
<td>39:00</td>
<td>0:55</td>
<td>headway</td>
</tr>
<tr>
<td>291H01</td>
<td>Hfd - Rd</td>
<td>inbound</td>
<td>40:27</td>
<td>7:32</td>
<td>headway</td>
</tr>
</tbody>
</table>

Table 6.1: Details of the critical circuit of Roosendaal. It shows for every train line which part of its route contributes to the critical circuit, after how long the next part of the critical circuit can be performed and by which kind of constraint it is connected to the next part of the critical circuit. Also the origin and destination of the train lines are added, the abbreviations are explained in appendix A.

6.2.2 Comparison

Because there are no realization data of this model, some expert opinions are used to valuate this model. These experts are from the departments Capacity Planning and Capacity Allocation at ProRail.

Primarily, the opinions stated that Roosendaal is a complicated station, mainly because of the freight trains. It is hard to fit all train lines in a timetable without conflicts. In addition, some freight-trains are risky for the environment. Although the last reason is based on the current situation, it will probably play a role in 2010 too. However, these arguments are not about the capacity of the infrastructure in the model. Restricting to the latter, experts do not expect complicated problems. Only the alternating arrivals and departures at platform 1, from the direction Etten-Leur and Tilburg could be a bottleneck. This opinion coincides with the results of PETER, because the critical circuit also presents these entrance and exit points.

6.3 ’s Hertogenbosch

The case study of ’s Hertogenbosch is performed with data from 2004. From this year, also realized departure and arrival times are available and some of them are also analyzed in [10]. These realizations are used for a comparison with the results of PETER. In the next subsections the results of PETER are presented and compared with the realization data. Appendix C.2 presents a picture of station ’s Hertogenbosch.
Choice of dwell type  During the evaluation of 's Hertogenbosch a little deviation in the way of modeling resulted in different critical circuits. Subsection 5.2.3 explains that one component of the input file specifies all data of the timetable, including the dwell type of every section. The dwell type defines the behavior of a train line at the end of the section, a stop, run or end are possible. Primarily, most sections became dwell type R, only sections where a train line really stops got type S and the last section of train lines in this station became type E.

After constructing the timed marked graph with the imported data, PETER tries to reduce the graph. A timed marked graph consists of many transitions and places, but some of them are unnecessary. The graph can be reduced, which simplifies the calculation of the performance indicators. The following two methods can be applied.

1. If a transition has only one upstream place and only one downstream place, then this transition can be removed. The upstream and downstream transitions are connected by a new place, replacing the old two places. The holding time of this new place is the sum of the holding times of the upstream and downstream place of the removed transition.

2. In subsection 5.2.2 and example 5.1 is shown how a constraint between two train lines can be shifted in the graph. This principle can also be applied to isolate a transition, such that it can be removed as written in the previous item.

PETER keeps all transitions with dwell type S in the graph and also the first transition of every train line. All other transitions are isolated and removed. The constraints are shifted to the remaining transitions. For the lines of freight-trains, this reduction has the most impact. These trains do not stop in the station, such that all connections with other train lines, also the connections concerning the outbound route, are shifted to their first transition, which represents the entrance of the station. This makes the reduced graph very different from the original one.

To keep the original graph, every section should have the dwell type S. Then all transitions are kept and the graph is not reduced by PETER. This is done for the timetable of 's Hertogenbosch and resulted in different critical circuits than in the first version.

For further investigation a third version is made, based on the first one, but with an extra stop for the three freight-trains. Each of them became a stop of zero minutes, such that the timetable stays the same. The only difference with the first version is that PETER does not shift all their connections with other train lines to their first transitions, but that connections concerning their outbound routes are shifted to the transition that represents the extra stop. This version results in the same critical circuits as the second version.

The advantage of the first version is that it seems to be close to the real situation, because a stop is only defined if a train line really stops. Unfortunately, this leads to a large reduction of the graph. The second version does not seem to be very realistic, because a train line does not stop at every section. But its graph is more realistic, because the constraints are represented in the graph as they are defined in the model instead of shifted to other transitions. These results seem to be more reliable, because the model behaves as it was intended.
Chapter 6. Results of case studies

<table>
<thead>
<tr>
<th>Critical circuit</th>
<th>cycle time</th>
<th>throughput</th>
<th>stability margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14:46</td>
<td>0.25</td>
<td>4:06</td>
</tr>
<tr>
<td>2</td>
<td>11:57</td>
<td>0.20</td>
<td>4:06</td>
</tr>
</tbody>
</table>

Table 6.2: Critical circuits in ’s Hertogenbosch.

to do and is not changed by PETER in a way it was not defined by the user. This third version has aspects from both other versions. It is also close to the real situation, except that the freight-trains have a non realistic stop. Apart from this, it presents the same reliable results as the second version.

Because the last version has reliable results and represents almost a realistic situation, this version is used for this case study.

6.3.1 Results - Critical circuit

The results of critical circuit analysis are presented in table 6.2 and are depicted in appendix C.3. There are two critical circuits with different cycle time, which means that the timed marked graph is not strongly connected. Because the maximal cycle time is 14:46 minutes, the timetable in this station can theoretically be realized with a period of 14:46 minutes instead of 60 minutes. Therefore, there is a lot of buffer time in the actual timetable such that delays will probably settle in one period. How the processes of this circuit are spread over the hour is shown in the detailed information of the circuit, represented in table 6.3.

In a station, the running times are short, just some seconds. The dwell times and connecting constraints need some more time, some minutes, and the headway times are varying between 30 seconds and some minutes. Summing the holding times in a circuit in the timed marked graph will not lead to large cycle times, only if some dwell times or connecting constraints are involved.

<table>
<thead>
<tr>
<th>line number</th>
<th>origin and destination</th>
<th>route</th>
<th>departure</th>
<th>connection time</th>
<th>connection type</th>
</tr>
</thead>
<tbody>
<tr>
<td>037T01</td>
<td>Rsd - Zl</td>
<td>inbound</td>
<td>23:05</td>
<td>1:26</td>
<td>stop</td>
</tr>
<tr>
<td>037T01</td>
<td>Rsd - Zl</td>
<td>outbound</td>
<td>25:00</td>
<td>1:15</td>
<td>headway</td>
</tr>
<tr>
<td>900T01</td>
<td>Hrl - Hlm</td>
<td>outbound</td>
<td>30:00</td>
<td>1:00</td>
<td>headway</td>
</tr>
<tr>
<td>442H01</td>
<td>Mt - Br</td>
<td>outbound</td>
<td>39:42</td>
<td>1:42</td>
<td>headway</td>
</tr>
<tr>
<td>350H01</td>
<td>Ehv - Asd</td>
<td>outbound</td>
<td>47:00</td>
<td>1:16</td>
<td>headway</td>
</tr>
<tr>
<td>044T02</td>
<td>Nm - Ht</td>
<td>inbound</td>
<td>54:17</td>
<td>0:19</td>
<td>passenger</td>
</tr>
<tr>
<td>516H01</td>
<td>fictitious</td>
<td>-</td>
<td>54:36</td>
<td>4:00</td>
<td>passenger</td>
</tr>
<tr>
<td>900H01</td>
<td>Hlm - Hrl</td>
<td>outbound</td>
<td>4:00</td>
<td>1:13</td>
<td>headway</td>
</tr>
<tr>
<td>350T01</td>
<td>Asd - Ehv</td>
<td>outbound</td>
<td>16:00</td>
<td>1:02</td>
<td>headway</td>
</tr>
<tr>
<td>417H01</td>
<td>Em - Kfh</td>
<td>outbound</td>
<td>18:42</td>
<td>1:33</td>
<td>headway</td>
</tr>
</tbody>
</table>

Table 6.3: Detailed information of critical circuit 1. For every train line is shown which part of its route contributes to the critical circuit, after how long the next part of the critical circuit can be performed and by which kind of constraint it is connected to the next part of the critical circuit. The abbreviations of the origin and destination of the train lines are explained in appendix A.
Figure 6.2: In this 'Basis Spoor Opstelling', basic track allocation, the train lines of critical circuit 1 are connected by the blue line. The connection type of the connection between two train lines is indicated by H for headway, P for passenger and S for stop.

This makes the small cycle time acceptable, although 's Hertogenbosch is not a small and quiet station.

The detailed information of the critical circuits of 's Hertogenbosch are represented in tables 6.3 and 6.4. An illustration of critical circuit 1 is depicted in figure 6.2. This figure represents the 'Basis Spoor Opstelling', the basic track allocation, of the train lines in 's Hertogenbosch. It shows which platform tracks are used by the train lines. By connecting the train lines of the critical circuit with the blue line, it shows which infrastructure is used by these train lines. The connection type of the connection between two train lines is indicated by H for headway, P for passenger and S for stop.

The critical circuit with the largest cycle time, critical circuit 1, consists of headways, a stop and passenger constraints. Because the connection between 044T02 and 900H01 is no cross platform connection, the connection time is 4 minutes in stead of 2 minutes. The 516H01 in between is a transfer train, a fictitious train to model the dependency caused by a passenger change. This is explained in the paragraph 'Passenger connections' of subsection 5.2.2. The stop connects the inbound and outbound route of the 037T01. The other connections are defined because of headway constraints at the right exit track to the north, direction Utrecht and Nijmegen, and because of headway constraints at the
Chapter 6. Results of case studies

middle exit track to the south, direction Roosendaal and Eindhoven, and the crossing in front of it. These parts of the infrastructure are represented by fat lines in appendix C.2. The train lines in this circuit are spread over 60 minutes, such that a delay will probably settle soon. Only train lines 350T01 and 417H01 succeed each other with less than 3 minutes in between. But the 417H01 represents a freight-train, which does not travel every hour, such that chance of a delayed train decreases.

The second critical circuit has cycle time 11:57 minutes and also consists of headways and passenger constraints. The headway constraints connect train lines using the right entrance track from the north, direction Utrecht and Nijmegen, and train lines using the right entrance track from the south, direction Roosendaal and Eindhoven. The train lines 503H01 and 505H01 are again transfer trains to model the dependency of a passenger change between two train lines. The time between every pair of train lines in this circuit is between 2.5 and 8 minutes such that a possible delay is probably settled soon.

The stability margin is 4.06 minutes for both circuits. This means that every headway, stop or passenger constraint in one circuit can increase with 4.06 minutes such that this timetable can still be operated in a period of 60 minutes.

<table>
<thead>
<tr>
<th>line number</th>
<th>origin and destination</th>
<th>route</th>
<th>departure</th>
<th>connection time</th>
<th>connection type</th>
</tr>
</thead>
<tbody>
<tr>
<td>037H01</td>
<td>Zl - Rsd</td>
<td>inbound</td>
<td>6:00</td>
<td>0:37</td>
<td>headway</td>
</tr>
<tr>
<td>620H02</td>
<td>Gdm - Ht</td>
<td>inbound</td>
<td>9:56</td>
<td>0:28</td>
<td>headway</td>
</tr>
<tr>
<td>350T01</td>
<td>Asd - Ehv</td>
<td>inbound</td>
<td>14:00</td>
<td>0:24</td>
<td>passenger</td>
</tr>
<tr>
<td>503H01</td>
<td>fictitious</td>
<td>-</td>
<td>14:27</td>
<td>2:00</td>
<td>passenger</td>
</tr>
<tr>
<td>096H01</td>
<td>Ht- Dn</td>
<td>outbound</td>
<td>17:00</td>
<td>0:59</td>
<td>headway</td>
</tr>
<tr>
<td>900T01</td>
<td>Hrl - Hlm</td>
<td>inbound</td>
<td>26:44</td>
<td>3:24</td>
<td>headway</td>
</tr>
<tr>
<td>350H01</td>
<td>Ehv - Asd</td>
<td>inbound</td>
<td>45:05</td>
<td>0:21</td>
<td>passenger</td>
</tr>
<tr>
<td>505H01</td>
<td>fictitious</td>
<td>-</td>
<td>45:29</td>
<td>2:00</td>
<td>passenger</td>
</tr>
<tr>
<td>620T02</td>
<td>Ht - Gdm</td>
<td>outbound</td>
<td>50:00</td>
<td>1:03</td>
<td>headway</td>
</tr>
<tr>
<td>900H01</td>
<td>Hlm - Hrl</td>
<td>inbound</td>
<td>0:56</td>
<td>0:41</td>
<td>headway</td>
</tr>
</tbody>
</table>

Table 6.4: Detailed information of critical circuit 2. For every train line is shown which part of its route contributes to the critical circuit, after how long the next part of the critical circuit can be performed and by which kind of constraint it is connected to the next part of the critical circuit. The abbreviations of the origin and destination of the train lines are explained in appendix A.

Include/exclude constraints PETER has the option to include or exclude groups of constraints after the input file is imported. Including or excluding constraints can change the performance indicators. The constraints are divided in four groups: layover times, passenger connections, coupling connections and infra constraints. The results above are obtained by including all constraints.

Including or excluding the layover times does not influence the critical circuits, because there are no layover times defined in the timetable. A layover time can be defined by a T (Turn) as dwell type at some joint. This means that the train line enters the station and leaves some time later in the same
direction as it came from, so the train line turns in the station. This happens in ’s Hertogenbosch with the line 044, traveling to Nijmegen and back. However, the turn in ’s Hertogenbosch is defined by connecting constraints and not with a layover constraint, such that including or excluding these constraints has no effect on the results.

If all constraints are excluded, there is no critical circuit. The train lines themselves form no circuit. With only passenger and coupling connections there is still no circuit. These form some connections between the train lines, but not enough to form a circuit. Therefore the infra constraints must always be included to analyze critical circuits.

With only infra constraints included, there are 5 critical circuits. The three most critical ones define the most critical points in the infrastructure. These are three entrance and exit points. The most critical of them is the right exit track to the north, with a cycle time of 10:37 minutes. ’s Hertogenbosch is a middle-large station, with just 2 tracks to the north and 3 tracks to the south. Inside the station are 9 tracks, four of them alongside a platform. Because all trains have to enter and leave the station over four tracks, these form the bottleneck. The two less critical circuits each consist of two train lines traveling the same track in opposite direction. Their cycle times are less than 2 minutes.

Including infra constraints and passenger connections results in the two critical circuits presented in tables 6.3 and 6.4. Besides two other critical circuits are shown, again the two with cycle time less than 2 minutes.

Excluding the passenger connections and including both infra constraints and coupling connections gives two new critical circuits. The first one has a cycle time of 13:22 minutes and consists of headways and couplings. These couplings define two train lines, one arriving and one departing, traveling the same route and using the same rolling stock. The rolling stock turns in ’s Hertogenbosch and the driver has to walk to the other end of the train. For this train line, the time to turn is 4:12 minutes, which increases the cycle time a lot. The second critical circuit only consists of headways, connecting train lines over the left exit track to the south. Its cycle time is 9:54 minutes.

**Critical infrastructure and train lines** The results of including and excluding different constraints show where the bottlenecks in the infrastructure are and which passenger or coupling constraints are most critical. The infra constraints show that the right exit track to the north, direction Utrecht and Nijmegen, is the main bottleneck. The other results of including and excluding constraints show that the coupling connections make the circuits more critical, but that including passenger changes increases the cycle time most.

The different critical circuits as result of including and excluding constraints also shows which train lines are most critical. The outbound route of train line 350H01 and the inbound route of line 044T02 appear in a critical circuit if all constraints are included, if only infra constraints are included and if infra and coupling constraints are included. The following train lines appear in a critical circuit if all constraints are included and if only infra constraints are included: inbound routes of 900H01, 037H01, 620H02 and 350T01.
Chapter 6. Results of case studies

6.3.2 Results - Recovery time

The recovery time between event A and B is defined as the maximum delay of departure A such that B can occur without delay. This is interesting to apply at train lines entering the station successively. Recovery times make it also possible to investigate with what maximum delay a train can enter the station to leave it in time.

In the Netherlands, the punctuality of arrivals is measured by the percentage of arrivals that has a maximal delay of 3 minutes. In 2004 the punctuality of all arrivals in the Netherlands was around 83%. This boundary of 3 minutes is also used to classify the recovery times. Large recovery times are not very interesting at station level, because they affect each other rarely. If it happens that one of them has such a delay, the order of trains is probably changed and some trains may follow another route. This situation differs too much from the model, so the results are not reliable. It is more interesting to analyze the short recovery times, for 0 to 3 minutes. This means that although a train has just a small delay, it is even punctual, its delay propagates to other trains. Middle long recovery times can be analyzed too, how interesting it is depends on the station and timetable.

Recovery times less than 3 minutes are shown in a table in appendix C.4. There is a sequence of four train lines, arriving in and departing from the station, with less than 3 minutes recovery time between each of them, see also table 6.5. Train line 442H01 enters the station and has a delay impact of 1:46 minutes to the arrival of line 096T02. This line has a delay impact of 2:25 minutes to the arrival of the 350H01, which has a delay impact of 1:21 minutes to the departure of the 96H02. Each of these train lines can cause a little delay to the next train line, although they arrived punctual at ’s Hertogenbosch, so with a maximal delay of 3 minutes. These small recovery times and tight following of four train lines will make the impact of a delay of the 442H01 worse. The delay impact from the 442H01 is depicted in figure 6.3. This train line enters the station over the right upper track, from direction Roosendaal and Eindhoven. The black line from the middle to the left is its departure from the station, which is also delayed. The delayed arrival of the 096T02 is represented by the dark brown line at the right. This train line is connected to the 350H01 by a passenger connection. The 350H01 arrives in the station over the same upper right track and travels along the upper red line. Then the 096H02 departs from the station over the same upper right track, which is depicted by the right red line. The figure shows also the recovery impact of the 442H01 to other train lines, depicted by the orange and yellow lines. The delay impact from the first to the fourth train line is just 5:32 minutes. But the 442H01 is a freight-train

<table>
<thead>
<tr>
<th>train line</th>
<th>recovery time</th>
<th>cumulative recovery time</th>
</tr>
</thead>
<tbody>
<tr>
<td>442H01</td>
<td>1:46</td>
<td>1:46</td>
</tr>
<tr>
<td>096T02</td>
<td>2:25</td>
<td>4:11</td>
</tr>
<tr>
<td>350H01</td>
<td>1:21</td>
<td>5:32</td>
</tr>
<tr>
<td>096H02</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.5: Recovery times and cumulative recovery times between four train lines.
and these do not travel through the station every hour. If the freight-train does not pass, just a sequence of three train lines is realized. When these train lines arrived and departed, there is a recovery time of more than 8 minutes to settle a possible delay.

In appendix C.4 can also be seen that the three freight-trains have the minimal recovery time of 0:00 minutes. They do not stop in the station, such that the recovery time between inbound and outbound route is zero minutes. The smallest recovery time between an inbound and outbound route of the same train line is 29 seconds of the 36T01 and 37T01. If they have an arrival delay of less than 29 seconds they can still depart in time, but otherwise their delay continues beyond ’s Hertogenbosch. The maximal recovery time between two routes of the same line is 1:36 minutes. This holds for the 36H01, 800H01 and 900H01. They have more buffer time in this station to settle a delay incurred before this station. It also means that if the train arrives in time, it has a long dwell time, because the train can not leave before its planned departure time.

### 6.3.3 Results - Delay propagation

The recovery time only represents how much buffer time is available to avoid secondary delay. If a secondary delay is obtained, the delay propagation shows how much it is and which train lines are affected. As is already written in subsection 5.3.2, the increased running time of a train as consequence of a delay of the previous train is not modeled in PETER. Also changing the order of

---

Figure 6.3: Results of the recovery impact of the 442H01 to other train lines. The darker the color of the line, the less the recovery time is.
Chapter 6. Results of case studies

trains is not modeled, which is explained in subsection 5.3.1. This can occur if one train has such a delay that trains, scheduled to arrive later, enter the station first.

If the delays are not too large and if it is assumed that the order of trains does not change, then the delay propagation gives an estimation of the secondary delay. Subsection 6.3.2 describes the recovery impact of train line 442H01. With the delay propagation can be seen what happens if its delay exceeds this recovery time. Assume that the 442H01 has a delay of 10 minutes. If the order of trains stays the same, the secondary delay is 41:31 minutes. The delayed trains are depicted in appendix C.5. The 096T02 obtains a delay of 8:14 minutes, the 350H01 of 5:49 minutes and the 096H02 is 4:28 minutes delayed. Besides these trains, also the 620T02 will be delayed, with 3:15 minutes. No other trains are delayed, because the recovery time in between is large enough to settle this delay.

The secondary delay of 41:31 minutes is obtained by including also the delay of the transfer trains and the departure delay of the 350H01. This means that the delay of the 350H01 is counted twice, which makes the secondary delay worse. The transfer trains are just fictitious, but in PETER they are seen as real trains.

Because the number of delayed trains is known, the average primary delay, average secondary delay and average total delay can be calculated. Again, PETER counts the inbound and outbound routes of 350H01 separately. Another indicator is the number of junctions with a delay, which is 6. This can be used to compare the distribution of initial delays inside the station. The last performance indicator is the settling period. This is interesting, because it gives the number of periods that is needed to settle the delays. In this example the settling period is 0, so one hour after the primary delay took place, the trains travel without delays again.

6.3.4 Realization data

The study of realization data of ’s Hertogenbosch, reported in [10], is based on data from March 2004. The data used for this study with PETER is also from 2004, such that the report [10] can be used to valuate the results of PETER. The report declares that the information is representative for the whole year 2004. It analyzes both station ’s Hertogenbosch and the railway section Utrecht - ’s Hertogenbosch. It focuses on the capacity and bottlenecks of this section and on its punctuality. Not all information can be used for a comparison with results of PETER, but the corresponding situations are discussed here.

Bottlenecks in infrastructure The report describes two bottlenecks in the infrastructure. First, the problem of realizing the entering and leaving train lines over the same track, the right exit track to the north, direction Utrecht and Nijmegen. And secondly, the successive departures of train lines over the right exit track to the south, direction Roosendaal and Eindhoven.

PETER makes it possible to signalize bottlenecks in the infrastructure with critical circuits. In subsection 6.3.1 is shown what the critical circuits in ’s Hertogenbosch are. The bottlenecks in the infrastructure are found by including only the infra constraints and excluding the other groups of constraints. It shows that the right exit track to the north is most critical. This circuit consists of
10 headway constraints and has a cycle time of 10:37 minutes. This bottleneck is one of the two studied in the report and includes alternating entering and leaving train lines over the same track. The second bottleneck studied in the report is not found with PETER, possibly because a circuit at this track is strongly connected to the most critical one. If there is a circuit of headway times on the right exit track to the south, its cycle time is less than 10:37 minutes. The second bottleneck found by PETER is the left exit track to the south, with cycle time 9:54 minutes.

With PETER, the most critical point in the infrastructure can be found without a detailed study of the punctuality and headway times of the corresponding train lines. Critical circuit analysis helps to find this point, but it does not define how critical it is, thus whether it is a real bottleneck or not. In section 6.1 is already discussed that a norm or comparison with other stations or situations is needed to confirm this.

**Punctuality** The report analyzes the punctuality over the whole railway section Utrecht - ’s Hertogenbosch. Because it presents punctuality separated for each arrival and departure, the punctuality data of ’s Hertogenbosch can be distinguished. The rush hours are analyzed separately, but these data are only used for comparison of extreme situations. The following train lines are marked.

- The most remarkable train lines are the 036 and 037 in both directions, to the north and the south. Their punctuality in ’s Hertogenbosch decreases dramatically. The worst situation is in the rush hours with direction to the north, where the punctuality decreases from 80% to 58%.
- The opposite is presented by the lines 800 and 900. They present in almost all situations an increasing punctuality in ’s Hertogenbosch. Only in rush hours there is a little decrease, from 91% to 86%.
- In the rush hours also the lines 350 to the south, 350T01 and 350T02, show bad results. Their punctuality decreases from 79% to 70%.

This increase or decrease in punctuality can be analyzed with the recovery times in PETER. It shows that the train lines 800 and 900 have a recovery time between their arrival at and departure from the station of 1:34 minutes on average, in both directions. The train lines 036, 037 and 350 have recovery times of about 30 seconds, except the 036H01 which has a recovery time of 1:36 minutes. This makes the statements above plausible. The series 800 and 900 have more buffer time of about 1 minute. It is also possible that the punctuality decreases because trains have to wait for a passenger transfer. All train lines above must wait for a passenger transfer, but there is no information available about the impact of these constraints on the punctuality.

**Dwell times** The report presents a table about the realized dwell times in ’s Hertogenbosch, for the train lines to the south. To eliminate extreme dwell times, the median is used instead of the average dwell time. This is presented in table 6.6. In the model used for the study with PETER, all planned dwell times are 1 minute. To compare the realized dwell times in table 6.6 with the dwell times in PETER, not the planned dwell time of 1 minute is used, but the difference between the departure and the arrival time in the timetable. This is
Chapter 6. Results of case studies

<table>
<thead>
<tr>
<th>train line</th>
<th>planned dwell time</th>
<th>median of realized dwell times</th>
</tr>
</thead>
<tbody>
<tr>
<td>800/900H01</td>
<td>2:00</td>
<td>1:54</td>
</tr>
<tr>
<td>350T01/02</td>
<td>1:00</td>
<td>1:24</td>
</tr>
<tr>
<td>037H01</td>
<td>1:00</td>
<td>2:03</td>
</tr>
<tr>
<td>036H01</td>
<td>2:00</td>
<td>2:35</td>
</tr>
</tbody>
</table>

Table 6.6: Realized dwell times in ’s Hertogenbosch for train lines to the south, as represented in [10].

<table>
<thead>
<tr>
<th>train line</th>
<th>maximal dwell times</th>
</tr>
</thead>
<tbody>
<tr>
<td>800/900H01</td>
<td>2:36</td>
</tr>
<tr>
<td>350T01/02</td>
<td>1:36</td>
</tr>
<tr>
<td>037H01</td>
<td>1:36</td>
</tr>
<tr>
<td>036H01</td>
<td>2:36</td>
</tr>
</tbody>
</table>

Table 6.7: Difference between departure and arrival times in ’s Hertogenbosch, based on PETER.

the sum of the planned dwell time and recovery time, thus the maximal dwell time, and is represented in table 6.7. If these maximal dwell times from PETER are compared with the planned dwell times in table 6.6, every train line has a recovery time of 36 seconds. Only the median of the 037H01 causes a problem, because this exceeds the maximal dwell time and will cause a delay. Probably, the planned maximal dwell time is too short, which can cause the decreasing punctuality as found in the paragraph above. For the 350T01/02 and 036H01 the median of realized dwell times approaches the maximal dwell times and also these train lines have a bad punctuality. This can be caused by short maximal dwell times again. If these train lines have a little delay, there is not enough buffer time to settle the delay. Only the 800H01 and 900H01 have 40 seconds more time to spend than the median of its realized dwell times. This can explain why these train lines have a better punctuality.

Relations The relation between delays of two trains is studied in the report with regression analysis, see [10]. These relations can be investigated with PETER with the recovery times and delay propagation. The report states that the main relations are between the arrival of the 800/900H01, the arrival and departure of the 036/037H01 and the arrival of the 350T01/02.

According to the report, the 036H01 will be delayed if the 800H01 has an arrival delay of more than 3:20 minutes. The recovery times of PETER show that this will happen if the 800H01 is 4:22 minutes delayed. This difference possibly appears because the theoretical headway times are defined by norms and the headway times used in PETER are calculated, which gives minimal headway times. Because these minimal headway times are less than 3 minutes, the corresponding recovery times are more. How much delay line 036H01 obtains when the delay of the 800H01 is more than 4:22 minutes, can be investigated with the delay propagation.

The report also says that the relation between the 036H01 and the 350T02 is more stronger than the previous one. This can not be valued with PETER,
which only shows the recovery time between these train lines, namely 6:56 minutes. This is not very short, such that a strong relation is not indicated in this way. The strength of the relation can be investigated by how often this situation occurs, but this can not be evaluated with PETER.

**Freight-trains** The report includes one chapter about freight-trains in ’s Hertogenbosch. It studies the number of freight-trains, their division over the hours of a day and their planned and realized arrival and departure times. This data can not be obtained with PETER, but the results of this chapter can help to get a better idea of the possible delays of freight-trains.

The number of freight-trains varies during a week, with an average of 33 trains per day on Monday till Friday and an average of 13 trains per day in the weekend. The number of freight-trains passing ’s Hertogenbosch can also be analyzed per part of the day. It shows that least trains pass during the rush hours, on average 0.74 per hour, and most freight-trains during the evening, on average 1.30 trains per hour. In the model, used in PETER, a freight-train is scheduled 3 times per hour. These planned freight-trains are not realized every hour, such that the chance of delay propagation by a freight-train decreases.

The realization data show that the freight-trains have a punctuality of 60%. From all freight-trains, 40% has a delay of more than 3 minutes and 35% arrives too early. Although not all scheduled freight-trains pass ’s Hertogenbosch, freight-trains that pass do not follow the timetable very well. This means that they are regularly delayed by other trains or cause a delay to other trains. This information can not be obtained with PETER, but can be used to interprete its results.

The report shows that the number of freight-trains passing ’s Hertogenbosch varies per hour and per day. It is less than the number of scheduled freight-trains, which is $24 \times 3 = 72$ trains per day. PETER shows that there are freight-trains with a delay impact to other trains and with a delay sensitivity from other trains of less than 3 minutes. This can also be seen in appendix C.4, where the freight-trains are represented by train lines 416, 417 and 442. The freight-trains are not very punctual, such that these small recovery times can lead to delays. So on one hand, there are less freight-trains than scheduled, but on the other hand they are less punctual than passenger trains. This can cause less or more delays respectively, depending on the situation.

In the report, the results about punctuality and number of freight-trains per day are not specified separately for every freight-train line. If this information is known, then it can be used to evaluate the consequences for other train lines with PETER.

If the planned time of a freight-train changes, PETER can be used to evaluate the consequences for other trains. Only the departure times of this freight-train should be adapted in the timetable and the new situation can be analyzed. Different schedules for this freight-train can be investigated, such that the possibility with least total delay can be realized.
Chapter 7

STATIONS

The second objective of this report is to investigate how to optimize the utilization of the infrastructure in a station. This can be done by extending STATIONS, a program that generates a routing of train lines through a station. STATIONS is described in [19] and is already introduced in this report in section 5.1. To extend STATIONS, some detailed knowledge of it is necessary. This chapter describes how STATIONS models different processes in a station and which assumptions are made to generate a routing. The next chapter shows how this model can be extended for optimizing the utilization of the infrastructure.

7.1 Objective

STATIONS is designed to route train lines through a station and to calculate their running times. This routing is part of generating a timetable for a whole network. DONS (Designer Of Network Schedules), which is already introduced in section 3.2, generates a timetable for a network, but models the stations in the network as black boxes. This means that the exact train movements in a station are not considered. To check whether a network timetable from DONS is also feasible inside the stations, STATIONS can be used. It considers just one station at the same time, so it has to be applied several times to investigate all stations in the network. According to a given infrastructure and the arrival and departure times from DONS, STATIONS tries to route all train lines in the timetable through the station. If a routing exists without conflicts and satisfying all constraints as defined in DONS, the routing is feasible. If there are more possible routings through the station, the optimal one is chosen by optimizing the preferences, which are described in subsection 7.2.3. If there is no feasible routing for all train lines, only the non-conflicting train lines are routed and the conflicting ones are returned.

7.2 Description of the model

This section starts with a description of the processes in a station. To model these processes, assumptions are made and preferences are specified. Then, after introducing the mathematical notation, the model is presented in subsection 7.2.5.
7.2.1 Processes in a station

The border of a station is marked by its entrance and exit points. These are fictitious points that separate the inside railway tracks from the outside railway tracks. The railway tracks outside these points are called open tracks and connect the station to the next station. Every train enters and leaves the station over one of these points. Of course, the entrance point for one train is possibly the exit point for another train and vice versa. A station includes a number of platform tracks. It is also possible that there are some tracks parallel to a platform track, but not alongside a platform. These are also called platform tracks and can be used as parking track or for overtaking.

A train line enters the station over its entrance point. The sequence of sections from this point to a platform track form its inbound route. Similarly, the sequence of sections from platform track to exit point form the outbound route. An inbound and outbound route using the same platform track form a complete route. From every train line in STATIONS its entrance and exit point are known from DONS, but the platform track, inbound and outbound route are determined by STATIONS. Also its planned arrival and departure time are defined by DONS. It is possible that the station is a starting or terminal station for one or more train lines. Then, the train line only has an inbound or outbound route and the train waits alongside the platform or is shunted away.

If two trains are coupled or uncoupled in a station, one of them is the leading train and one of them is the following train. The leading train has a complete route through the station, the following train just an inbound or outbound route. The route of the following train is compatible with the route of the leading train.

Entering or leaving a station is performed in accordance with the route locking sectional release principal. This means that an inbound or outbound route is reserved as whole for an arriving or departing train and is released section by section when the train passes the sections of the route. This principal assures that a train can safely enter or leave a station without collisions with other trains. Besides, also minimum headway times assure safety of trains in a station. They define a minimal time between two arrivals or departures in the station.

Besides different train movements in a station, there are also movements of passengers. To make a passenger transfer easy and fast, it is preferably cross-platform or between two platforms that are close to each other. To assure a passenger transfer between two trains, a minimum overlap time between their dwell times alongside the platform is needed.

7.2.2 Assumptions

The assumptions for this model are made with the intention to have a good representation of reality and to assure that the resulting routing of train lines is compatible with the timetable at network level.

Because the result of DONS is a cyclic timetable with a period of 60 minutes, all time calculations in STATIONS are also done modulo 60.

For every train line, a maximal train length is defined to avoid a train-platform combination where the train length exceeds the platform length. From the timetable generated by DONS, the origin, destination, arrival and departure time of every train line are known. If a station is the terminal station of the
train line and the line turns, the destination and departure time are known too. Also the velocity at the entrance and exit point is assumed to be given. Every train line that stops in the station has zero velocity at the platform. By considering a constant velocity or acceleration, STATIONS is able to calculate the running times from every train line per section.

Shunting movements inside a station are not modeled in STATIONS. Just whether shunting is possible or not is considered for every train line, because it determines whether a certain platform track can be used in between for other train lines or not. If a train line is planned to be shunted, a certain time interval alongside the platform after arrival is assured, such that there is enough time to leave the train. Also a certain time interval alongside the platform before departure is assured. After shunting, the departure platform is possibly different from the arrival platform. If a train turns in a station and waits alongside the platform until its departure, the departure platform is necessarily the same as its arrival platform.

Coupling and uncoupling of trains is modeled in STATIONS too. It is assumed that beforehand it is known which train line is the leading one and which train line is the following one. The platform track where coupling or uncoupling takes place must be divided in two parts, otherwise the platform can be occupied by only one train line at the same time.

7.2.3 Preferences

Preferences are used to choose the most optimal routing. The most important preference is to generate a feasible routing for all train lines. If there are different feasible routings, the next preferences are considered too.

The second preference concerns deviations in arrival and departure times, which are generated by DONS. It is possible that an unfeasible routing becomes feasible when some small deviations are allowed. For some train lines this is acceptable, but for other train lines the resulting arrival and departure times can be incompatible with the network timetable. Therefore, the maximal allowed deviations are defined for every train line individually and this second preference tries to minimize the number of deviations.

Next, if all train lines are routed with a minimum number of deviations, the number of shunting movements is minimized. Shunting movements use capacity of the station and they are expensive, because personnel is needed for the operation. Therefore, the least possible number of shunting movements is preferred.

The final preference is to use certain routes and platforms for certain train lines. For some train lines different inbound and outbound routes are possible to enter and leave the station. One route can be preferred above another because of the number of switches it contains, or the number of switches in non-preferred position that are in the route. The more switches a route contains, the more conflicts it may have with other routes, and the more time it needs to finish the route because of a lower speed. The preferred position of a switch is in general the position of the switch such that trains travel in a straight line. If a switch is used in non-preferred position, it is less comfortable for the passengers and it deteriorates more than if it is used in preferred position.
Chapter 7. STATIONS

7.2.4 Notation

Before specifying the model to route train lines through a station, the notation for train lines, routes and infrastructure in the station is introduced. This notation is already defined in [19].

A set of track sections, denoted by \( S \), defines the infrastructure of the station. All track sections that are included in a platform or passing track are defined by the subset \( P \subset S \).

With the set of track sections all possible routes \( R \) in a station can be defined. There are three kinds of routes in a station, each of them forming a subset of \( R \). The set of inbound routes \( R^i \subset R \) contains all routes from an entrance point to a platform track. The set of outbound routes \( R^o \subset R \) contains all routes from a platform track to an exit point. And the set of all platform routes \( R^p \subset R \) contains all routes representing platform tracks.

All train lines to be routed through the station are denoted by the set \( T \). For every train line \( t \in T \), the planned arrival time and planned departure time are known, which are denoted by \( a_t \) and \( d_t \), respectively. The set of train lines for which a small deviation of these arrival and departure times is allowed, is denoted by subset \( T^\Delta \subset T \). The subset \( T^S \subset T \) denotes the set of train lines that may be shunted.

To make a connection between train lines and routes in the station, the feasible routes for every train line are specified. The possible routes for a certain train line are based on its entrance and exit points and the length of the platforms. With these aspects a set of inbound routes \( R^i_t \) can be defined for every train line \( t \). Also a set of possible outbound routes, \( R^o_t \), and platform routes, \( R^p_t \), can be defined for every train line. The union of these three sets forms the set \( R_t \) of all possible routes in the station for train line \( t \).

The notations above for infrastructure, routes and train lines are summarized in table 7.1.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S )</td>
<td>Set of track sections.</td>
</tr>
<tr>
<td>( P )</td>
<td>Set of track sections included in a platform or passing track, ( P \subset S ).</td>
</tr>
<tr>
<td>( R )</td>
<td>Set of all possible routes in the station.</td>
</tr>
<tr>
<td>( R^i, R^o, R^p )</td>
<td>Subsets with all possible inbound, outbound and platform routes respectively.</td>
</tr>
<tr>
<td>( T )</td>
<td>Set of train lines to be routed.</td>
</tr>
<tr>
<td>( T^\Delta )</td>
<td>Subset of trains with allowed non-zero deviations, ( T^\Delta \subset T ).</td>
</tr>
<tr>
<td>( T^S )</td>
<td>Subset of trains which may be shunted, ( T^S \subset T ).</td>
</tr>
<tr>
<td>( a_t )</td>
<td>Planned arrival time for train line ( t ).</td>
</tr>
<tr>
<td>( d_t )</td>
<td>Planned departure time for train line ( t ).</td>
</tr>
<tr>
<td>( R_t )</td>
<td>Set of all possible routes in the station for train line ( t ).</td>
</tr>
<tr>
<td>( R^i_t, R^o_t, R^p_t )</td>
<td>Subsets with all possible inbound, outbound and platform routes, respectively, for train line ( t ).</td>
</tr>
</tbody>
</table>

Table 7.1: Notation of infrastructure, routes and train lines in STATIONS.
are defined. Because arrival and departure times are required to be integers, the deviations should be integers as well. The set of allowed arrival deviations is denoted by $\Delta^a_t$ and the set of allowed departure deviations is denoted by $\Delta^d_t$. A deviation of train line $t$ is then a combination of an arrival and departure deviation, denoted as $\delta_t = (\delta^a_t, \delta^d_t)$, where $\delta^a_t \in \Delta^a_t$ and $\delta^d_t \in \Delta^d_t$.

The combination of a feasible inbound route and an allowed arrival deviation for train line $t$ specifies an inbound routing possibility for train line $t$. The set of all inbound routing possibilities of train line $t$ is denoted by $F^i_t \subset R^i_t \times \Delta^a_t$. Then, inbound routing possibility $f \in F^i_t$ is defined as $f = (r, \delta^a_t)$, with $r \in R^i_t$ and $\delta^a_t \in \Delta^a_t$. The set $F^o_t \subset R^o_t \times \Delta^d_t$ denotes a similar set of allowed combinations of an outbound route and a departure deviation $(r, \delta^d)$. Allowed combinations of a platform route and an arrival and departure deviation $(r, \delta)$, with $\delta = (\delta^a, \delta^d)$, are denoted by set $F^p_t \subset R^p_t \times \Delta^d_t$. The set of all routing possibilities for train line $t$ is defined as $F_t = F^i_t \cup F^o_t \cup F^p_t$. The complete set of all routing possibilities in the station is denoted by $F$.

Subsection 7.2.1 mentions two kinds of dependencies between trains, namely by minimal headway times and by minimal transfer times. These dependencies are modeled as constraints and denoted in STATIONS by the set $F_{t,t'}$, which contains pairs of allowed routing possibilities $(f; f')$ that are compatible with each other. The infra constraints with minimal headway times can be modeled with these pairs of routing possibilities because STATIONS calculates the running times of the train lines, as is explained in [19]. Then, for a given pair of routing possibilities, the claim and release times of every section are known, showing whether this pair is compatible or not. The passenger constraints with minimal transfer times can be modeled by excluding all pairs of routing possibilities that give a transfer time less than the minimal transfer time. Also the connection of inbound, outbound and platform routes of the same train line can be guaranteed with the set $F_{t,t'}$. In this case, the pair $(f; f')$ represents two routing possibilities of the same train line. For example, one possibility to specify an inbound route and one to specify a platform route. If these routes are connected and deviations belonging to these routes are compatible, then this pair is an element of $F_{t,t}$.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta^a_t, \Delta^d_t$</td>
<td>Set of allowed arrival and departure deviations, respectively, for train $t$.</td>
</tr>
<tr>
<td>$\delta_t$</td>
<td>A deviation for train line $t$, $\delta_t = (\delta^a_t, \delta^d_t)$, is a combination of an arrival deviation $\delta^a_t \in \Delta^a_t$ and a departure deviation $\delta^d_t \in \Delta^d_t$.</td>
</tr>
<tr>
<td>$F$</td>
<td>Set of all routing possibilities. A routing possibility $f$ is an allowed combination of a route $r \in R$ and a deviation $\delta$.</td>
</tr>
<tr>
<td>$F^i_t, F^o_t, F^p_t$</td>
<td>Sets of inbound, outbound, and platform routing possibilities for train $t$ respectively.</td>
</tr>
<tr>
<td>$F_{t,t'}$</td>
<td>Set of pairs of allowed routing possibilities $(f; f')$ for train $t$ and $t'$.</td>
</tr>
<tr>
<td>$\rho_{t,f}$</td>
<td>Preference of train $t$ for routing possibility $f$.</td>
</tr>
</tbody>
</table>

Table 7.2: Notation of deviations, routing possibilities and preferences in STATIONS.
Chapter 7. STATIONS

The preference of a train line $t$ for a certain routing possibility is represented by $\rho_{t,f}$. The different preferences, as described in subsection 7.2.3, are combined in this parameter as follows: $\rho_{t,f} = p_\delta + p_{t,f} + p_{t,r}$. The preference to minimize the number of deviations is represented by $p_\delta$, the preference to minimize the number of shunting movements is represented by $p_{t,f}$ and the preference to use certain routes and platforms is represented by $p_{t,r}$. How to choose the values for these preferences is explained in detail in [19].

Table 7.2 contains a summary of the notations above, concerning deviations, routing possibilities and preferences.

### 7.2.5 Model

Finally, this section presents the mathematical model for routing train lines through a station. It consists of an object function, to maximize the preferences specified in subsection 7.2.3, and a set of constraints. For every allowed combination of a train line and a routing possibility, a binary decision variable $X_{t,f}$ is defined as follows.

$$X_{t,f} = \begin{cases} 1 & \text{if train line } t \in T \text{ uses routing possibility } f \in F_t, \\ 0 & \text{otherwise}. \end{cases} \tag{7.1}$$

If $X_{t,f} = 1$, then routing possibility $f \in F_t$ is assigned to train $t \in T$ and represents an inbound, outbound or platform route $r \in R_t$ and a deviation in $\Delta_t$.

The routing problem is then modeled as follows:

$$\max \sum_{t \in T} \sum_{f \in F_t^i} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t^o} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t^p} \rho_{t,f} X_{t,f} \tag{7.2}$$

subject to

$$\sum_{f \in F_t^i} X_{t,f} \leq 1 \text{ for all } t \in T, \tag{7.3}$$

$$\sum_{f \in F_t^o} X_{t,f} \leq 1 \text{ for all } t \in T, \tag{7.4}$$

$$\sum_{f \in F_t^p} X_{t,f} \leq 1 \text{ for all } t \in T, \tag{7.5}$$

$$X_{t,f} + X_{t',f'} \leq 1 \text{ for all } t, t' \in T; \ f \in F_t; \ f' \in F_{t'}; \ (f, f') \notin F_{t,t'}, \tag{7.6}$$

$$X_{t,f} \in \{0, 1\} \text{ for all } t \in T; f \in F_t. \tag{7.7}$$

The objective function (7.2) maximizes the total preferences for inbound, outbound and platform routing possibilities. The constraints (7.3) ensure that at most one inbound route is assigned to every train line. In the same way, at most one outbound route and one platform route is assigned to every train line by constraints (7.4) and (7.5) respectively. Then, constraints (7.6) guarantee that only allowed combinations of routing possibilities are selected, which are defined by all combinations $(f, f') \in F_{t,t'}$. Finally, constraints (7.7) declare that the decision variables are binary.
7.2. Description of the model

More detailed information about how to choose the preferences $\rho_{t,f}$ in the object function and about extensions of the model concerning shunting movements and joining routing possibilities are explained in [19].

This model of the routing problem does not optimize the utilization of the infrastructure in a station. However, by extending the model, this preference can be considered too, which is explained in the next chapter.
Chapter 8

Optimizing the utilization of infrastructure

The previous chapter describes how STATIONS generates a routing of train lines through a station. Optimizing the utilization of the infrastructure in a station is not considered in this model. This chapter shows that it is possible to extend the model of STATIONS with this optimization. First, the theoretical extension is explained and then this is illustrated with an example of a small fictitious station.

8.1 Optimization purpose

STATIONS generates a routing of train lines through a station, considering the preferences that are specified in subsection 7.2.3. The preference concerning the infrastructure is to have routes with a minimal number of switches and a minimal number of switches in non-preferred position. Therefore, train lines often use the same routes in the station, such that some switches are used very often.

Optimizing the utilization of the infrastructure can spread the intense utilization of some switches over a larger group of switches in the station, such that the maximal utilization, and therefore the maintenance costs, reduces. Besides, spreading the intense utilization can also help to avoid bottlenecks. Reducing the maximal utilization of switches decreases the probability that they cause a disturbance.

8.2 Measurement and optimization systems

There are different ways to measure the utilization of infrastructure in a station. It is possible to measure the time that each track and switch is used. Another measure is the number of trains that passes each track and switch, or their weight or number of axles. And also the number of times that a switch is switched can be counted. Each of these measures makes it possible to compare the utilization of the infrastructure for different routings through the station. Different systems are available to perform these measurements.
Quo Vadis, presented in [15], is a measurement system that registers at certain points in the Dutch railway network the number of passing trains and their weight and number of axles. For every train, its collected data is stored in a database. Another system, TNV (Trein Nummer Volgsysteem), see also [15], registers all realized routes of trains through a station. With these results, the number of times that a switch is passed by a train can be counted. If all trains in the station are registered by Quo Vadis, for every switch in the station the total weight and total number of axles that have passed can be derived. So combining the results from the measurement systems Quo Vadis and TNV, gives a detailed overview of the utilization of the infrastructure of a station. However, it is based on realization data, so there is no information about other routings.

A system that measures the utilization of the capacity of the infrastructure is CAMERA (Capaciteit Meetmodel Railinfrastructuur), see [1]. It represents the utilization of the capacity in minutes per hour for tracks and junctions. The capacity is mainly used for running time, dwell time, headway times and disruption time and the remaining minutes in an hour are denoted as remaining capacity. Different from the detailed information from TNV and Quo Vadis, CAMERA calculates the capacity more roughly. However, CAMERA can obtain every given routing through a station, also not-realized routings. By comparing the results of different routings, the utilization of the infrastructure in the station can be optimized manually.

STATIONS, which is described in chapter 7, is an optimization program. It generates a routing of train lines through the station, optimizing the specified preferences. This means that the routing is not specified beforehand. It is not possible to optimize the utilization of infrastructure with STATIONS, but its model can be extended to do this. By using a simple measure, or by importing measurement data from other systems, it is possible to find the optimal utilization of the infrastructure with the extended model.

This report starts with a simple measure to study whether an extension of the model in STATIONS gives the desired results: the utilization of infrastructure is measured by the number of trains that pass the switches in a station.

### 8.3 Model extension

The utilization of the infrastructure is measured by the number of trains that pass a switch in a station per hour. This will be modeled as an extension of the model in STATIONS, which is already introduced in subsection 7.2.5.

Assume that the set $W$ contains all switches in a station. Then for every switch $w \in W$ and for every routing possibility $f \in F$, $Y_{w,f}$ defines whether routing possibility $f$ uses switch $w$.

$$Y_{w,f} = \begin{cases} 1 & \text{if routing possibility } f \text{ uses switch } w, \\ 0 & \text{otherwise.} \end{cases}$$  \hspace{1cm} (8.1)

With this definition it is possible to count the number of times that a switch is used per hour, this is the number of trains that pass a switch per hour, which is denoted by $B_w$. A switch is used if it is contained in a routing possibility that
8.3. Model extension

is realized by a certain train line, so \( B_w \) is defined by

\[
B_w = \sum_{t \in T} \sum_{f \in F} (X_{t,f} \times Y_{w,f}). \tag{8.2}
\]

To optimize the utilization of the infrastructure, the maximum use of all switches must be reduced. This means, that the maximum of \( B_w \), for all \( w \in W \), must be reduced. Define \( M \) as the smallest integer such that

\[
B_w \leq M \text{ for all } w \in W. \tag{8.3}
\]

Then, \( M \) is equal to the maximum utilization \( B_w \) of all switches in the station. The utilization of the infrastructure can now be optimized by minimizing \( M \) or by defining a constraint for \( M \).

\( M \text{ free} \) Minimizing \( M \) can be obtained by adding \(-M\) to the object function (7.2), because the object function is a maximization. To make the minimization of \( M \) more important, a weight \( n \) can be added to \( M \). Then the object function is as follows:

\[
\max \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} - n \times M. \tag{8.4}
\]

For every combination of \( w \in W \) and \( f \in F \) the value \( Y_{w,f} \) must be defined and the set of constraints (7.3) until (7.7) are extended with the expressions (8.1), (8.2) and (8.3). This model generates an optimal routing through a station with respect to the preferences mentioned in subsection 7.2.3 and the utilization of the infrastructure. The weight \( n \) determines the importance of optimizing the utilization of the infrastructure with respect to the other preferences.

\( M \text{ bounded} \) It is possible that beforehand it is known that the maximal utilization of all switches must be reduced to a certain level \( c \) anyway. This can be obtained by adding the following constraint:

\[
M \leq c, \tag{8.5}
\]

for a positive integer \( c \). The object function, as in (8.4), can be used again. Then, \( M \) is still minimized, such that \( M \) can also become smaller than \( c \). The set of constraints is the set as used for \( M \text{ free} \), extended with constraint (8.5).

Instead of defining one maximum for all switches, it is also possible to define the maximum number of times that a switch is used for every switch separately, denoted by \( M_w \). Then instead of constraints (8.3) and (8.5), the following constraints are added to the model:

\[
B_w \leq M_w \text{ for all } w \in W, \tag{8.6}
\]

\[
M_w \leq c_w \text{ for all } w \in W. \tag{8.7}
\]

Define also the weight \( n \) for every switch separately by \( n_w \), such that different switches can be more or less important in the optimization. The object function can then be written as follows:

\[
\max \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} + \sum_{t \in T} \sum_{f \in F_t} \rho_{t,f} X_{t,f} - \sum_{w \in W} n_w M_w. \tag{8.8}
\]
Chapter 8. Optimizing the utilization of infrastructure

8.4 Fictitious station

There are different options to test the theoretical extension. It is possible to make an extension to the already programmed part of the model. Alternatively, it is possible to design a new program, optimizing both the preferences which are described in subsection 7.2.3 and the utilization of the infrastructure. However, both options are very complex and time-consuming. Therefore, it is chosen here to investigate the extension of the model for one simple fictitious station. This can be easily modeled and solved with OPL-studio, a software system for solving optimization and constraint-based problems, see also [21].

The extended model of STATIONS is applied at a simple fictitious station. The objective of this example is to show whether and how the extension influences the resulting routing.

8.4.1 Layout and timetable

Consider the layout of the station as represented in figure 8.1. The infrastructure is divided into 14 sections, denoted by A to N. This station has only two entrance and exit points and contains six switches, denoted by 1 to 6. There are 14 routes: 2 platform routes, 6 inbound routes and 6 outbound routes, which are the same as the inbound routes but in opposite direction. The timetable for this station is represented in table 8.1. Three train lines pass the station from east to west, and three from west to east. Two times an hour, a passenger change is guaranteed, namely from train line L1 to L2 and from L3 to L4.

<table>
<thead>
<tr>
<th>serial number</th>
<th>origin</th>
<th>destination</th>
<th>arrival time</th>
<th>departure time</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>west</td>
<td>east</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td>L2</td>
<td>east</td>
<td>west</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>L3</td>
<td>west</td>
<td>east</td>
<td>45</td>
<td>48</td>
</tr>
<tr>
<td>L4</td>
<td>east</td>
<td>west</td>
<td>46</td>
<td>49</td>
</tr>
<tr>
<td>L5</td>
<td>west</td>
<td>east</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>L6</td>
<td>east</td>
<td>west</td>
<td>30</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 8.1: Timetable of the fictitious station.
8.4.2 OPL-studio: original model

First, the existing model of STATIONS is modeled with OPL-studio, and then the extension is added, such that the difference in results can be analyzed.

Modeling the fictitious station starts with defining the sets with train lines, routes, routing possibilities, preferences and conflicting routing possibilities.

The set with all train lines to be routed through the station, set \( T \), consists of the six train lines represented in table 8.1. For every train line its serial number, entrance and exit section and arrival and departure time are defined.

All possible routes through the station are defined in two files. One file specifies for every inbound, outbound and platform route its route number, first section and last section. The other file specifies for every route which sections its contains.

Assume that there are no deviations allowed in this fictitious station, then the routing possibilities for a certain train line are determined by the compatible routes for that train line. The compatible routes are inbound routes starting at the entrance section of the train line and outbound routes finishing at the exit section of the train line. Routes of the same train line are compatible if they are connected to each other. It is also assumed that the train length will not exceed the platform length. Because whether or not routes are compatible for certain train lines can be easily tested with a constraint, the set with routing possibilities is defined to be the same as the set with all routes.

The preferences of train lines for certain routes are based on the number of switches in a route: less switches means a higher preference. If different routes have the same number of switches, then the route with most switches in preferred position gets a higher preference, because traveling over a switch in a straight line is more comfortable for the passengers and can be performed with higher speed. For the inbound routes in the fictitious station, the routes from A to G and N to H are in a straight line and have the minimal number of switches, such that they become the highest preference. Thereafter, the routes from A to H via section D and from N to G via section K become the highest preference, because they also have just two switches but one in non-preferred position. The inbound routes via sections E and J become the lowest preference. The outbound routes, the same routes but in opposite direction, become the same order of preferences. Also for the two platform routes G and H, preferences are specified. Because the passenger building is defined at platform 1, the preference for platform route G is higher than for platform route H.

The last set to be specified is the set with pairs of non-conflicting routes. In this model for the fictitious station in OPL-studio not the pairs of non-conflicting routes, but the pairs of conflicting routes are defined. Then, for every pair of routes in this set, at most one of the routes may be assigned to a train line. A pair of routes is defined to be conflicting if they have at least one section, including the platform sections, in common.

The binary variable \( X_{t,f} \) is defined for all 6 train lines and for all 14 routes in the station. By solving the problem, OPL-studio assigns a one or zero value to each of these variables.

With these specifications, the model can be defined. The object function looks the same as equation (7.2) and also the constraints (7.3) till (7.5) are defined in the same way; one for all inbound routes, one for all outbound routes and one for all platform routes. Because there is no set with routing possibiliti-
ties defined, extra constraints must be added to assure that train lines are only assigned to inbound routes starting at their entrance section and to outbound routes finishing at their exit section. In the OPL-studio model, this is specified by two constraints. One for compatible inbound routes and one for compatible outbound routes. Also the compatibility of routes from the same train line is modeled by two constraints, one for the compatibility of inbound and platform routes and one for the compatibility of platform and outbound routes. Constraints (7.6) are modeled by three constraints in OPL-studio, with a headway time specified at 2 minutes. If two train lines arrive in the fictitious station with inter arrival time less than two minutes, then their inbound routes must not be conflicting. In the same way the outbound routes of train lines with inter departure times less than two minutes must not be conflicting. And the inbound and outbound routes of train lines with a difference between arrival and departure time less than two minutes must not be conflicting.

**Results original model** The model that is described in this subsection results in a routing through the fictitious station with four train lines passing platform section $G$ and two train lines passing platform section $H$. All train lines travel across the outside sections $D$ and $K$. Their routes are specified in table 8.2. Because of the preferences, the solver tries to route train lines as most as possible across platform section $G$. But because two times an hour, two train lines arrive in 2 minutes, two train lines have to be routed across platform section $H$.

<table>
<thead>
<tr>
<th>serial number</th>
<th>inbound route</th>
<th>outbound route</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>$A, B, C, G$</td>
<td>$G, I, K, N$</td>
</tr>
<tr>
<td>L2</td>
<td>$N, M, L, H$</td>
<td>$H, F, D, A$</td>
</tr>
<tr>
<td>L3</td>
<td>$A, B, C, G$</td>
<td>$G, I, K, N$</td>
</tr>
<tr>
<td>L4</td>
<td>$N, M, L, H$</td>
<td>$H, F, D, A$</td>
</tr>
<tr>
<td>L5</td>
<td>$A, B, C, G$</td>
<td>$G, I, K, N$</td>
</tr>
<tr>
<td>L6</td>
<td>$N, K, I, G$</td>
<td>$G, C, B, A$</td>
</tr>
</tbody>
</table>

Table 8.2: Resulting routing of train lines through the fictitious station without optimizing the utilization of infrastructure.

### 8.4.3 OPL studio: extended model

To model the utilization of the infrastructure in the fictitious station, the OPL-studio model in subsection 8.4.2 is extended. The utilization of infrastructure is measured by the number of times that a switch is passed by a train. Therefore, there must be a file that specifies which switches are contained in which routes. For every combination of a route and a switch, it is defined whether this is the case or not.

Figure 8.1 shows that switches 1 and 6 are always contained in the inbound and outbound route of a train line, such that they are always passed by six train lines, for every possible routing through the station. Therefore, these switches are not considered in equation (8.3) for the fictitious station.
The utilization of the infrastructure can now be optimized by minimizing the variable $M$, as is described in section 8.3. Both options, $M$ free and $M$ bounded, are applied to the fictitious station.

**Results $M$ free** The resulting routing through the fictitious station is exactly the same as without the extension. The switches 2 to 5 are passed 4, 4, 2 and 2 times per hour, respectively. This means that the preferences for certain routes and platform sections are stronger than the preferences for optimizing the utilization of the infrastructure.

In objective function (8.4) variable $M$ has weight $n$. If $n$ is set to 10, then the resulting routing is different because the utilization of the infrastructure has become more important. The number of times that a switch is passed by a train line is now spread over the four switches 2 to 5: each of them is traveled 3 times per hour such that $M = 3$. Three train lines, L1, L3 and L5, pass section $G$ and the other three train lines pass section $H$. However, all of them use the outside sections $D$ and $K$ and none of the train lines uses the inner sections $E$ and $J$.

**Results $M$ bounded** To assure that the maximal utilization does not exceed a certain level $c$, constraint (8.5) is added to the model. If $c$ equals 3 or 4, the same results are obtained as with $M$ a free variable and only the outside sections are used.

This extension of the model optimizes only the number of times that the switches in the model are used. The results from the fictitious station show that the utilization of the tracks in the station is not automatically optimized too. Therefore, also the number of times that a track is passed by a train line must be counted, such that it can be minimized in the same way as is done for the switches. However, in this small fictitious station it is also possible to derive this by adding a bonus preference to the preferences of sections $E$ and $J$.

To make sure that every possible value of $M$ is analyzed, constraint (8.5) is replaced by $M = c$ in the OPL-studio model. The routing is obtained again, with the bonus preferences and $c$ varying from 3 to 6.

If $c$ is equal to 6, there are no restrictions for the number of times a switch is used and all train lines in the routing use the inner sections $E$ and $J$. Because platform section $G$ still has a higher preference than platform section $H$, caused by the passenger building at platform 1, four train lines are routed over platform section $G$ and the other two lines over platform section $H$.

By decreasing $c$, the number of times that switches are used must be spread more and more over all switches. Then, the results show that the preferences are less satisfied and more and more train lines use the outside sections $D$ and $K$ again. If $c$ equals 5, switches 2 to 5 are used 5, 4, 2 and 5 times per hour respectively. The resulting routes are presented in table 8.3. Two train lines are now forced to use the outside sections, but it is still possible to route 4 train lines over platform section $G$ and 2 train lines over platform section $H$. Also if $c$ is decreased to 4, the distribution over the platforms stays the same, but then 4 train lines are forced to use the outside sections. Finally, if $c = 3$, the resulting routing is exactly the same as if $M$ was free and equal to 3, such that all train lines use the outside sections $D$ and $K$. 

79
Chapter 8. Optimizing the utilization of infrastructure

<table>
<thead>
<tr>
<th>serial number</th>
<th>inbound route</th>
<th>outbound route</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>A, B, C, G</td>
<td>G, I, J, M, N</td>
</tr>
<tr>
<td>L2</td>
<td>N, M, L, H</td>
<td>H, F, D, A</td>
</tr>
<tr>
<td>L3</td>
<td>A, B, C, G</td>
<td>G, I, J, M, N</td>
</tr>
<tr>
<td>L4</td>
<td>N, M, L, H</td>
<td>H, F, E, B, A</td>
</tr>
<tr>
<td>L5</td>
<td>A, B, C, G</td>
<td>G, I, K, N</td>
</tr>
<tr>
<td>L6</td>
<td>N, M, J, I, G</td>
<td>G, C, B, A</td>
</tr>
</tbody>
</table>

Table 8.3: Resulting routing of train lines through the fictitious station with \( M \) equal to 5.

**Results fictitious station** The fictitious station is just a small station with only two entrance and exit points. Therefore there is just a small number of routes in the station, and there are at most two trains in the station at the same time. However, the example of the fictitious station shows that the extension of the model leads to an improved spreading of the utilization of the infrastructure. The maximal utilization of the switches in the station is reduced, which was the objective of the extension of the model. The utilization of the tracks in the station is not automatically optimized too, as is shown by the example. Therefore, the number of times that a track is used must be counted too, or the preferences of certain routes can be increased. The choice of the preferences is very important and really influences the resulting routing. The importance of optimizing the utilization of the infrastructure with respect to the other preferences can be adapted in the extended model.

### 8.4.4 Evaluation with PETER

Chapter 6 shows with the results of case studies that it is possible to analyze a timetable at station level with PETER. PETER can be used to analyze different versions of a timetable. In section 8.4 different routings through a station are realized, with different timetables. The timetable contains the entrance and exit times of the sections in the infrastructure; these times change if the routing is changed. Therefore, different routings can be analyzed with PETER, to observe the effect of changing routes for a better utilization of the infrastructure on the critical circuits and the recovery times.

However, the layout of the fictitious station, presented in figure 8.1, shows that the train lines always have to enter and leave the station over two entrance and exit points. Probably, these points will cause the main bottleneck, which occurs in every routing. Therefore, it is expected that an evaluation with PETER does not give very different results for different routings in this station.

Three routings are evaluated with PETER; the routing obtained without extension of the model, the routing with all train lines using the outside sections and \( M \) equal to 3 and the routing with \( M \) equal to 5 where most train lines use the inside sections \( E \) and \( J \). The first two routings have a critical circuit with cycle time 16:32 minutes and the last one has a critical circuit with cycle time 16:31 minutes. These cycle times are calculated by PETER, which is explained in subsection 4.3.1. The sequence of train lines in the critical circuit is for all routings the same. Because the entrance and exit points of the station are the same for all routings and the running times and headway times do not differ
very much, also the recovery times are nearly the same for the different routings. This shows that the expected absence of differences between the routings is true.

This fictitious station is too small to cause differences in the performance indicators calculated by PETER. If there are more routes with different lengths in the station, then also running times of routes and headway times between routes will change. This will probably lead to different performance indicators for different routings too.
Chapter 9

Conclusions and recommendations

In the 'Problem Description' of this report, chapter 2, two subjects are mentioned. The main subject of this report concerns a new application of the tool PETER, which can evaluate the robustness of timetables at network level. The objective is to investigate the possibility to use PETER for evaluating a timetable at station level. The second subject of this report concerns the optimization of the infrastructure in a station. Therefore, the program STATIONS can be extended.

Both subjects are described by six research questions, which are studied in this report. This chapter presents the conclusions of this report and presents some recommendations for further research.

9.1 Conclusions

9.1.1 PETER at network level

The first two research questions of this report were: 'How does PETER operate at network level?' and 'What is the mathematical background of PETER and how does it produce the network performance indicators?'. These questions are studied in detail in this report, the concluding remarks are presented in the next two items.

- PETER is an analytical tool to evaluate the robustness of a timetable. This is done by calculating performance indicators concerning critical circuits, recovery times and delay propagation. The advantage of the tool is that it quickly presents the results and that it can work with different input formats, specially the timetables generated by DONS.

- The modeling method is based on timed marked graphs and max-plus algebra. A timed marked graph is a convenient tool to model discrete event dynamic systems, such as a railway network. The timed marked graph of the network can then be transformed into a recurrence relation in max-plus algebra, which is used to calculate the performance indicators.
Chapter 9. Conclusions and recommendations

9.1.2 Modeling a timetable at station level

The third research question was stated as follows: ‘How can these mathematical methods be applied at station level?’ This report shows that it is possible to evaluate a timetable at station level with PETER. It must be concluded that there were some problems, caused by a consistency check in PETER, but these problems can be solved. Also some limitations of the max-plus algebra were obtained during the study. These conclusions are explained below.

- This report shows that it is possible to model a timetable at station level as a timed marked graph. If the graph is constructed, then the recurrence relation in max-plus algebra and the performance indicators can be derived in the same way as for a timetable at network level. To construct the timed marked graph, a station can be seen as a network too, the infrastructure of a station is divided into sections, which are separated by joints. At station level, the transitions in the graph model the departures of train lines from joints in the station. The running times over sections and dwell times along the platforms define the holding times in the graph. The constraints, both infra and service constraints, are modeled in the same way as at network level.

- PETER is developed to evaluate timetables at network level. However, because of some differences between network and station level and because of a consistency check in PETER, importing constraints caused some problems. At station level, PETER only accepts constraints where the sections of the two train lines have a common joint in the station. This is not the case for most constraints at station level. Because it is very complex to remove this check from PETER, the input of constraints must be adapted. For headway times, the problem can be solved by defining the constraint on a common section and adjusting its holding time. However, it is also possible that two train lines have a common section, but do not have a common joint. Then the infrastructure in the model must be adapted by adding a joint at the middle of the crossing, such that the constraint can be defined. For constraints defining a passenger connection, the problem can be solved by adding a fictitious train, which represents the flow of passengers between these train lines. The old constraint is replaced by two new constraints between this fictitious train and the feeder and connecting train line.

The adjustments of constraints took some time but has shown to be very useful, because now also constraints at station level can be imported in PETER. However, for further applications it is recommended to computerize it, see also subsection 9.2.1.

- Because of limitations of the max-plus algebra, two situations at station level can not be modeled and therefore not be analyzed with PETER. In the first place, this leads to the limitation of fixed order of trains. If a certain train is delayed, such that other trains have to wait, it is not possible to change the order of trains and let the other trains enter the station first. The planned order of trains can be analyzed with PETER. Otherwise, for a given initial delay, it can be analyzed when the planned
order is not acceptable anymore. Because of secondary and consecutive delays, changing the order of trains can be necessary.

Secondly, there is a limitation concerning increased running times caused by a delayed train. If a train has to stop or slow down, its running time over a section increases. This can occur both at network and station level, but the impact is much more at station level. There, the extra running time can be equal to the planned running time, while at network level the extra running time is very small compared to the planned running time.

These limitations only have to be considered for calculating the delay propagation, because then the performance of trains is possibly not in accordance with the timetable.

9.1.3 Performance indicators at station level

The last research question about this subject was: 'What are the results of calculating performance indicators at station level and are they practical applicable?'. The results of the case studies show that the performance indicators deliver useful results. The concluding remarks of every indicator are described below.

- The cycle time of a critical circuit represents the minimal time needed to operate the planned train movements in this circuit. How the resulting buffer time is spread over the hour depends on how the processes in this circuit are spread over the hour. With this information, it can be investigated whether and how the buffer time can be used for extra train lines. If there are many critical circuits with small cycle times, it suggests that the timed marked graph is not strongly connected at all and that the train lines in the station are only slightly dependent.

A critical circuit at station level always includes constraints, because train lines themselves do not form a circuit. Therefore, the critical circuits are appropriate to investigate which kind of constraints cause the largest cycle times, by including and excluding constraint groups. Including only the infra constraints shows the main bottlenecks in the infrastructure of the station.

In both case studies, the main bottleneck found with this method was also indicated as bottleneck by experts and realization data. The main bottleneck in the infrastructure in 's Hertogenbosch was found to be the right exit track to the north, direction Utrecht and Nijmegen. The most critical train movements were found to be the outbound route of the 350H01, from Eindhoven to Amsterdam, and the inbound route of train line 044T02, from Nijmegen to 's Hertogenbosch. These train movements were presented most often in the critical circuits.

- The recovery time between two given departures shows the maximal delay of the first departure, such that the second one is not delayed. This report concentrated on recovery times of less than 3 minutes, because in this situation a train can propagate a delay to the next train, although it has a punctual arrival according to the actual definition of punctuality.
A sequence of train lines with recovery times of less than 3 minutes in between indicates a bottleneck concerning delay propagation.

In the case study of 's Hertogenbosch, a sequence of four train lines with recovery times of less than 3 minutes in between showed up. The recovery time between the first and last train lines of this sequence was only 5:32 minutes. Therefore, a delay of the first train quickly influences the succeeding train lines.

Also the recovery times between the arrival and departure of the same train line in a station can be analyzed. In the case study of 's Hertogenbosch, these were compared with the arrival and departure punctuality of the train lines in the realization data. Train lines with a small recovery time, of about 30 seconds, correspond to train lines with a decreasing punctuality in the realization data, and train lines with recovery times of about 1:30 minutes, correspond to train lines with an increasing punctuality. So, the realization data confirm the expectation that short recovery times increase the possibility for delay.

• The delay propagation shows the influence of a certain delay to other train lines. By interpreting these results, the limitations of the model, fixed order of trains and increased running time, must be taken into account. The consecutive and secondary delays show which train lines are affected and how much. If the consequences of an initial delay are not acceptable, then changing the order of trains is necessary.

In the case study of 's Hertogenbosch, the sequence of four train lines with recovery times of less than 3 minutes in between, was analyzed with delay propagation. An initial delay of 10 minutes was specified for the first train, which resulted in a delay of 4:28 minutes for the last one. The delay propagation can quickly show the consequences for all affected trains.

Because of these useful results, PETER can be used to evaluate timetables at station level for different applications. It can be used during the planning of a new timetable. Different versions of the timetable can be compared with PETER to analyze the bottlenecks in each version. Otherwise, PETER can also be used to study the consequences of adding or removing some elements of the infrastructure. The performance indicators show the consequences for the robustness of the timetable. And else, in case of delays, PETER can be used to analyze the consequences of decisions of the traffic control, such that the best option can be performed.

9.1.4 Extending STATIONS

The second object of this report was to optimize the utilization of the infrastructure in a station. The two research questions about this subject were: 'What methods are available to measure the utilization of the infrastructure?' and 'Can one of these methods be extended to optimize the utilization of the infrastructure in a station?'. STATIONS is used to optimize the utilization of the infrastructure, because it already is an optimization program and because its model can easily be extended. Although the fictitious station in the example was a small station, it shows that the theoretical extension of STATIONS does
reduce the intense utilization of some switches. These conclusions are explained below.

- From the different measurement and optimization systems and programs described in this report, STATIONS is the most suitable one. This program already is an optimization program and it can be extended to optimize the utilization of the infrastructure in a station. STATIONS generates a routing of train lines through a station and tries to satisfy the preferences as good as possible. The utilization of the infrastructure in a station is measured by the number of times that a switch in the station is used. Then, the maximal utilization of all switches is minimized, such that the intense utilization of some switches is spread over more switches.

- The example of a fictitious station shows that the extended program really reduces the maximal utilization of the switches. However, this does not automatically optimize the utilization of the tracks too. Therefore, another extension of the model is required. The choice of the preferences is very important and really influences the resulting routing. The importance of optimizing the utilization of the infrastructure with respect to the other preferences in the model can be adapted in the model.

9.2 Recommendations

The research questions of this report are answered, but some new questions about the subjects have risen. Therefore, the following recommendations for further research are specified.

9.2.1 Evaluating a timetable at station level with PETER

- This report shows that applying PETER at station level gives useful results. For further applications it is necessary to make a program that defines the right input format for PETER from the output of STATIONS. During this study, this conversion has been done manually, but this takes too much time.

- The running times of every train line are presented per section in the timetable. These times are based on the running times of the head of the train. However, it is possible that the tail of the train has a longer or shorter running time over the same section. It is interesting to find a suitable running time, maybe dependent on both running times. The last section of a route is extra complicated to calculate. In this report an estimation is made, but there must be other possibilities to obtain a more exact running time for this section.

- Also the calculation of headway times can be more refined. Partly depending on the previous item and also to assure that the right sections are involved. Sections that are occupied by a dwelling train must not be included in the calculation of headway times. This can occur, depending on the composition of its route. In this study, these sections are manually removed afterward. It is recommended to investigate whether and how this can be changed in the model.
• For this study, the input data is adapted, because it was too complicated to remove the consistency check from PETER. For further applications of PETER at station level, it is recommended to investigate the possibility of removing the consistency check again.

• It is known that PETER simplifies the model after constructing the timed marked graph. It is interesting to investigate whether this can be done beforehand. Probably it makes the input file more readable and it can reduce the amount of work too.

• Sometimes bottlenecks are not found in a station, but just outside the station boundary. If railway tracks from different directions join at one track to the station, as in ’s Hertogenbosch, this can also form a bottleneck. By enlarging the area of analysis, possible bottlenecks outside the station can be found too.

• If the whole process of defining an input file can be programmed, it is interesting to analyze smaller stations too, because they can have unknown bottlenecks. However, mainly larger stations are interesting. PETER can help to understand the dependencies between processes in the station and to find the bottlenecks.

• An idealistic view of the developments of PETER at station level is that it can be integrated with PETER at network level. Then, the consequences of a certain delay do not have to be considered separately for the network and each station. However, this is a huge step in the size of the recurrence relation and in the size of the input file, which must be possible to perform too.

9.2.2 Optimizing the utilization of the infrastructure

• The example of the fictitious station showed that the utilization of tracks in the station is not automatically optimized too. This can be achieved by extending the model again. The maximal utilization of tracks can be reduced in the same way as is done for switches, the number of times that each track is used must be counted.

• The fictitious station was just a small station, with only two entrance and exit points and at most two trains in the station at the same time. A larger station has more possible routings, which makes the optimization more complex. For further research, it is recommended to analyze larger stations too.

• In this report only a simple measure is used for the utilization of a switch, this is the number of times that a switch is used. It is also possible to use another measure, for example the number of times that a switch is switched. With different measures, a more accurate representation of the utilization can be obtained.

• Mainly for larger stations, it is recommended to write a program that can be easily adapted and can quickly optimize the utilization of the infrastructure. This can be done by making an extension of the already
9.2. Recommendations

programmed part of the model in STATIONS, or by designing a new program, optimizing both the already existing preferences and the utilization of the infrastructure.
Appendix A

Abbreviations of station names

The abbreviations of station names that are mentioned in this report are explained in the table below.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Station name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asd</td>
<td>Amsterdam</td>
</tr>
<tr>
<td>Bv</td>
<td>Beverwijk</td>
</tr>
<tr>
<td>Dn</td>
<td>Deurne</td>
</tr>
<tr>
<td>Ehv</td>
<td>Eindhoven</td>
</tr>
<tr>
<td>Em</td>
<td>Emmerich</td>
</tr>
<tr>
<td>Gdm</td>
<td>Geldermalsen</td>
</tr>
<tr>
<td>Hfdo</td>
<td>Hoofddorp Opstelterrein</td>
</tr>
<tr>
<td>Hlm</td>
<td>Haarlem</td>
</tr>
<tr>
<td>Hrl</td>
<td>Heerlen</td>
</tr>
<tr>
<td>Ht</td>
<td>'s Hertogenbosch</td>
</tr>
<tr>
<td>Kfh</td>
<td>Kijfhoek</td>
</tr>
<tr>
<td>Mt</td>
<td>Maastricht</td>
</tr>
<tr>
<td>Nm</td>
<td>Nijmegen</td>
</tr>
<tr>
<td>Rsd</td>
<td>Roosendaal</td>
</tr>
<tr>
<td>Sloe</td>
<td>Sloehaven</td>
</tr>
<tr>
<td>Valb</td>
<td>Valburg</td>
</tr>
<tr>
<td>Zl</td>
<td>Zwolle</td>
</tr>
</tbody>
</table>

Table A.1: Abbreviations of station names.
Appendix B

Roosendaal

B.1 Map of Roosendaal

Schematic representation of station Roosendaal. Roosendaal has a lot of railway tracks that are not depicted in the figure below. Only the main tracks, which are used for the train lines in this model, are shown.
Appendix C

’s Hertogenbosch

C.1 Input file ’s Hertogenbosch

The input file of ’s Hertogenbosch. Lines starting with # are comment lines.

```
# casestudie sHertogenbosch
# juli 2005

cycle_time    recovery_threshold
     60           60

# coodinaten
# lasnr.; obj_type; x-coord; y-coord
station type  x-coordinate  y-coordinate
  1    AR    3632    -821
  2    AR    3640    -767
  3    AR    3674    -741
  4    AR    3431    -878
  5    AR    3525    -936
  6    AR    3404    -821
  7    AR    3501    -821
  8    AR    3262    -821
  9    AR    3515    -971
  210   IR    2903    -983
  211   IR    3120    -821
  212   IR    9323    -962
  213   IR    9051    -819
  214   IR    8904    -685
```
Chapter C. ’s Hertogenbosch

# dienstregeling
# lijn(nr,H/T,freq); sectie; van_las; naar_las; inrijtijd; rijtijd; doorkomst/stop; marge; included

<table>
<thead>
<tr>
<th>line</th>
<th>segment</th>
<th>origin</th>
<th>destination</th>
<th>departure</th>
<th>run_time</th>
<th>dwell_time</th>
<th>dwell_type</th>
<th>marge</th>
<th>included</th>
</tr>
</thead>
<tbody>
<tr>
<td>036H01</td>
<td>55 211 8</td>
<td>36:00</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>57 8 6</td>
<td>36:02</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>88 6 7</td>
<td>36:04</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>22 7 2</td>
<td>36:05</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>105 2 3</td>
<td>36:07</td>
<td>0:00</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>59 3 44</td>
<td>36:07</td>
<td>0:03</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>60 44 43</td>
<td>36:10</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>61 43 42</td>
<td>36:12</td>
<td>0:04</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>63 42 41</td>
<td>36:16</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>6 41 39</td>
<td>36:17</td>
<td>0:04</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>64 39 40</td>
<td>36:21</td>
<td>0:00</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>44 40 138</td>
<td>36:21</td>
<td>0:03</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>3 138 139</td>
<td>36:24</td>
<td>0:00</td>
<td>1:00</td>
<td>S</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>91 139 140</td>
<td>39:00</td>
<td>0:06</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>93 140 141</td>
<td>39:06</td>
<td>0:05</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>94 141 142</td>
<td>39:11</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>96 142 143</td>
<td>39:13</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>95 143 109</td>
<td>39:15</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>99 109 108</td>
<td>39:17</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>80 108 107</td>
<td>39:18</td>
<td>0:00</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>306 107 201</td>
<td>39:18</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>305 201 112</td>
<td>39:19</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>51 112 123</td>
<td>39:20</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>36 123 214</td>
<td>39:21</td>
<td>0:37</td>
<td>0:00</td>
<td>E</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>36 214 123</td>
<td>53:05</td>
<td>0:07</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>51 123 122</td>
<td>53:12</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>036H01</td>
<td>310 122 202</td>
<td>53:13</td>
<td>0:00</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>307 96 201</td>
<td>17:14</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>304 201 110</td>
<td>17:15</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>311 110 202</td>
<td>17:16</td>
<td>0:00</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>309 202 111</td>
<td>17:16</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>314 111 203</td>
<td>17:17</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>312 203 114</td>
<td>17:18</td>
<td>0:01</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>34 114 118</td>
<td>17:19</td>
<td>0:02</td>
<td>0:00</td>
<td>R</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>096H01</td>
<td>37 118 212</td>
<td>17:21</td>
<td>0:29</td>
<td>0:00</td>
<td>E</td>
<td>0</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

# fictieve treinen voor overstappen

<table>
<thead>
<tr>
<th>fictieve treinen voor overstappen</th>
</tr>
</thead>
<tbody>
<tr>
<td>501H01</td>
</tr>
<tr>
<td>502H01</td>
</tr>
<tr>
<td>503H01</td>
</tr>
</tbody>
</table>
## overstapconstraints

<table>
<thead>
<tr>
<th>feeder_line</th>
<th>feeder_segment</th>
<th>connecting_line</th>
<th>connecting_segment</th>
<th>connection_time</th>
</tr>
</thead>
<tbody>
<tr>
<td>044T01</td>
<td>53</td>
<td>044H02</td>
<td>69</td>
<td>4:12 C</td>
</tr>
<tr>
<td>044T02</td>
<td>53</td>
<td>044H01</td>
<td>69</td>
<td>4:12 C</td>
</tr>
<tr>
<td>096T01</td>
<td>37</td>
<td>620T01</td>
<td>38</td>
<td>1:00 C</td>
</tr>
<tr>
<td>096T02</td>
<td>37</td>
<td>620T02</td>
<td>38</td>
<td>1:00 C</td>
</tr>
<tr>
<td>620H01</td>
<td>55</td>
<td>096H02</td>
<td>5</td>
<td>1:00 C</td>
</tr>
<tr>
<td>620H02</td>
<td>55</td>
<td>096H01</td>
<td>5</td>
<td>1:00 C</td>
</tr>
<tr>
<td>096T01</td>
<td>37</td>
<td>501H01</td>
<td>501</td>
<td>0:00 P</td>
</tr>
<tr>
<td>501H01</td>
<td>501</td>
<td>350H02</td>
<td>16</td>
<td>2:00 P</td>
</tr>
<tr>
<td>096T02</td>
<td>37</td>
<td>502H01</td>
<td>502</td>
<td>0:00 P</td>
</tr>
<tr>
<td>502H01</td>
<td>502</td>
<td>350H01</td>
<td>16</td>
<td>2:00 P</td>
</tr>
<tr>
<td>518H01</td>
<td>518</td>
<td>900T01</td>
<td>52</td>
<td>2:00 P</td>
</tr>
<tr>
<td>800H01</td>
<td>55</td>
<td>519H01</td>
<td>519</td>
<td>0:00 P</td>
</tr>
<tr>
<td>519H01</td>
<td>519</td>
<td>036H01</td>
<td>91</td>
<td>2:00 P</td>
</tr>
<tr>
<td>900H01</td>
<td>55</td>
<td>520H01</td>
<td>520</td>
<td>0:00 P</td>
</tr>
<tr>
<td>520H01</td>
<td>520</td>
<td>037H01</td>
<td>91</td>
<td>2:00 P</td>
</tr>
</tbody>
</table>

## constraints

<table>
<thead>
<tr>
<th>line1 segment1</th>
<th>line2 segment2</th>
<th>headway12</th>
<th>headway21</th>
<th>event1</th>
<th>event2</th>
</tr>
</thead>
<tbody>
<tr>
<td>350H01</td>
<td>37</td>
<td>442H01</td>
<td>37</td>
<td>0:34</td>
<td>1:00</td>
</tr>
<tr>
<td>350H02</td>
<td>37</td>
<td>442H01</td>
<td>37</td>
<td>0:34</td>
<td>1:00</td>
</tr>
<tr>
<td>350T01</td>
<td>95</td>
<td>416H01</td>
<td>95</td>
<td>1:21</td>
<td>1:14</td>
</tr>
<tr>
<td>350T01</td>
<td>95</td>
<td>417H01</td>
<td>95</td>
<td>1:21</td>
<td>1:14</td>
</tr>
<tr>
<td>350T02</td>
<td>95</td>
<td>416H01</td>
<td>95</td>
<td>1:21</td>
<td>1:14</td>
</tr>
<tr>
<td>350T02</td>
<td>95</td>
<td>417H01</td>
<td>95</td>
<td>1:21</td>
<td>1:14</td>
</tr>
<tr>
<td>442H01</td>
<td>37</td>
<td>800T01</td>
<td>37</td>
<td>1:00</td>
<td>0:40</td>
</tr>
<tr>
<td>442H01</td>
<td>37</td>
<td>900T01</td>
<td>37</td>
<td>1:00</td>
<td>0:40</td>
</tr>
<tr>
<td>036H01</td>
<td>22</td>
<td>620T01</td>
<td>22</td>
<td>0:54</td>
<td>0:42</td>
</tr>
<tr>
<td>036H01</td>
<td>22</td>
<td>620T02</td>
<td>22</td>
<td>0:54</td>
<td>0:42</td>
</tr>
<tr>
<td>036H01</td>
<td>36</td>
<td>036T01</td>
<td>36</td>
<td>0:43</td>
<td>0:55</td>
</tr>
<tr>
<td>036H01</td>
<td>36</td>
<td>037T01</td>
<td>36</td>
<td>0:43</td>
<td>0:55</td>
</tr>
<tr>
<td>036T01</td>
<td>51</td>
<td>037H01</td>
<td>51</td>
<td>0:47</td>
<td>0:50</td>
</tr>
<tr>
<td>036T01</td>
<td>53</td>
<td>044T01</td>
<td>53</td>
<td>0:43</td>
<td>1:02</td>
</tr>
<tr>
<td>036T01</td>
<td>53</td>
<td>044T02</td>
<td>53</td>
<td>0:43</td>
<td>1:02</td>
</tr>
<tr>
<td>037H01</td>
<td>22</td>
<td>620T01</td>
<td>22</td>
<td>0:54</td>
<td>0:42</td>
</tr>
<tr>
<td>037H01</td>
<td>22</td>
<td>620T02</td>
<td>22</td>
<td>0:54</td>
<td>0:42</td>
</tr>
<tr>
<td>037H01</td>
<td>36</td>
<td>037T01</td>
<td>36</td>
<td>0:43</td>
<td>0:55</td>
</tr>
<tr>
<td>037T01</td>
<td>53</td>
<td>044T01</td>
<td>53</td>
<td>0:43</td>
<td>1:02</td>
</tr>
<tr>
<td>037T01</td>
<td>53</td>
<td>044T02</td>
<td>53</td>
<td>0:43</td>
<td>1:02</td>
</tr>
<tr>
<td>044H01</td>
<td>53</td>
<td>044T01</td>
<td>53</td>
<td>0:36</td>
<td>0:52</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter C. 's Hertogenbosch

Schematic representation of station 's Hertogenbosch.

track 1
track 2
track 3
track 4
track 5
track 6
track 9
track 10
track 11

direction Geldermalsen, Utrecht, Nijmegen

platform

direction Vught, Eindhoven, Roosendaal

North
C.3 Output of critical circuit analysis

Picture of the output of PETER after critical circuit analysis of ’s Hertogenbosch.

Figure C.1: Results of critical circuit analysis with PETER.
Chapter C. ’s Hertogenbosch

help
C.4 Recovery times ’s Hertogenbosch

This matrix presents the recovery times of less than 3 minutes in ’s Hertogenbosch. The letters I and O at the end of the line numbers denote the inbound and outbound routes of the train lines respectively. Element \((i, j)\) represents the delay impact from train line \(i\) to train line \(j\), or the delay sensitivity of train line \(j\) for train line \(i\).
help
C.5 Output of delay propagation

Picture of the output of PETER after calculation of delay propagation in ’s Hertogenbosch.

Figure C.2: Results of calculating the delay propagation in PETER for a delay of 10 minutes of train 442H01.
Bibliography


