Train rescheduling in the event of a full blockade

Bachelor thesis
June 2013

Author	**Joris-Jan Kraak**
Student number	**333602**
Institution	**Erasmus University Rotterdam**
Supervisor	**Dr. D. Huisman**

Abstract

Large railway companies are confronted with unexpected disruptions on their network on a daily basis. When a disruption occurs a new timetable should be found, able to fulfill the need for transportation as effectively as possible. In this thesis, we will look into the case of a full blockade on a railway line on the Dutch railway network. Full blockades prevent trains to pass on a certain track in both directions. This leads to trains having to turn around at the stations on each side of the blockade. To keep the train schedule feasible some trains will have to be delayed or even cancelled. A model to find a solution for this scenario has been demonstrated in Louwerse and Huisman (2012). This solution, however, only presents the replacing schedule which should be operated continuously during the period of the disruption. I.e. it does not take into account that we should implement a so-called transition period, in which we shift from the original schedule to the new one. There is also a high probability that some trains are stuck before the blockade. These trains should be returned to a station that is on their route and then fitted into the new schedule. In this thesis, we use integer programming formulations to find a new schedule which will cover both the transition period and the period until the blockade is gone. We test the model on two real life situations of which data was provided by Netherlands Railways. The new schedules will be compared to the schedules that Netherlands Railways used when the problems occurred. It is important that our model allows for a quick solution, so that the new timetable can be communicated to the trains and the passengers in a timely fashion.

Contents

1. Introduction

1.1 Netherlands Railways

In this thesis, we will test our model on situations that happen on the network of Netherlands Railways (NS). NS is settled in one of the most densely populated countries in the world. The train is also a relatively popular mode of transportation in the Netherlands. Therefore, NS has got to make full use of the capacity of their railway network. The downside to having such a tight schedule is that any unexpected event can have a major impact. Punctuality has always been a big issue for NS; therefore they will need to act fast in case of such an event. The sooner a new schedule is found, the less trains will get in trouble and the less passengers will complain about the lack of information.

Due to the complexity of NS' network, often a solution should be sought by using a simplified model focusing just on the trains on the affected route. After that, it should be checked if this solution is compatible with the availability of train staff and the schedule of the other trains. Real life data of two disruptions in the network of NS will be used to test the effectiveness of the model presented in this thesis. This data also contains implicit information about the schedules that NS used to deal with the situation.

1.2 Disruptions

The disruptions that we examine in this thesis are two complete railway blockades. A complete railway blockade does not allow for any train traffic in both directions. As a result, trains will need to turn around at the last station before the blockade, performing their regular route or another train's route in the opposite direction. This could cause an infeasibility in NS' original schedule, so that it will be necessary to delay or cancel some of the trains on the affected route.

1.3 Goal

The goal of this thesis is to find a model that can determine a new train schedule in case of a complete blockade. When looking for such a schedule we want to minimize the total delay and try to cancel as few trains as possible. We also try to minimize the maximum time existing between all two consecutive trains. We will try to find an optimal balance between these unwanted effects and see if we can find a better solution than the one used by NS.

The model should include a solution for the transition period as well. I.e. it should tell us how we can transition from the original schedule to the schedule used during the disruption. Note that this also covers returning trains that are stuck near the blockade and fitting them in the new schedule. By including the transition period in the scheduling horizon immediately, there is a higher probability that the result will be feasible. However, if this leads to a significant increase in computation time, this might ultimately not be considered desirable. As mentioned before, time is a crucial element in this process.

1.4 Method

Integer Linear programming (ILP) is a frequently used method for solving logistical problems. The same problem, without the transition period, has been addressed In Louwerse and Huisman (2012) using ILP. A big advantage of using ILP is that it is relatively easy to make adjustments to the model. In this thesis we will adjust the model presented in Louwerse and Huisman (2012), so that it will find a schedule that includes the transition period.

1.5 Structure of the thesis

The thesis will be structured as follows. Firstly, in chapter 2, we give a general description of the problem we are dealing with. Also the approach we take to tackle the problem is presented and the assumptions we make to justify our model. Chapter 3 contains a detailed description of the implemented model and all included parameters and variables. Next, in chapter 4, we go into detail about the provided data of each problem and how we implement this into our model. In chapter 5 we present the gathered results and examine the extent to which these satisfy our goals. Lastly, in chapter 6, we examine if our overall goal is met. We also discuss the implications of our results.

2. Problem

2.1 Description

We only inspect the area in which trains run that are directly influenced by the disruption; we will call this the problem area. As a consequence of a disruption, trains will have to turn around at the last station before the blockade that they would normally stop. Note that such a last station may vary across different types of train services. NS has two types of train services; local train services and intercity services. A train service is a collection of all itineraries that run on the same route (and stop at the same stations) in both directions. Local trains can be assigned to any local train itineraries and the same holds for intercity trains. In other words, when a train ends an itinerary at a border station, it can turn on any itinerary of the same type of train service leaving from the same border station. A border station is a station where an itinerary starts or ends in the disrupted situation.

Some trains turn on a different train service at a border station. If one of these train services runs entirely outside of the problem area, then we still want the trains to be able to turn on this train service. However, itineraries of this train service are not allowed to be delayed or cancelled to not cause any issues outside of the problem area. We do allow all itineraries inside the problem area to be delayed or cancelled to a certain amount. By doing this we ensure that a new feasible train schedule can be found. A schedule is only feasible when enough headway time exists between all trains running on the same track and when all itineraries that are not cancelled can be assigned to a train.

2.2 Approach

By looking at the original timetable, we can determine where all the trains are located at the moment of a disruption. We will use this state as an input for the model.

We will build on the ILP model presented in Louwerse and Huisman (2012). This model can determine a new regular train schedule for the duration of the disruption. The model ensures hourly regularity by addressing a group of trains as train series when they travel the same route at the same time of the hour. Because of the inclusion of the transition period, we will have to lose the restriction of hourly regularity in this thesis. Louwerse and Huisman scheduled the moment of all train events of which the itinerary is not cancelled by changing them relatively to the original time of each particular event. This allows for easily finding the total amount of delay.

In this thesis we will adjust their model to ensure that it will be possible to transition from the original to the new schedule in an efficient manner. The model should also take into account that some trains could be stuck at the location of the disruption and that these trains should be returned to their new border station before being assigned to an itinerary leaving from that station.

2.3 Assumptions

To justify the model, presented in the next chapter, we have to make some assumptions that simplify the real situation. We assume that:

- The calculated train schedule is compatible with the schedule of the train staff.
- Trains that are waiting at a border station for longer than the normal turnaround time are being 'stored' at a location where they do not interfere with other trains and are directly available when needed again.
- The required time for a train activity is equal to the time needed in the original timetable. I.e. the supplement time is always included in this model. We do this, because we do not want any extra risk in our new schedule.
- No additional disruptions happen during the scheduling. Also, no trains have a non-scheduled delay.
- Railway switches are located directly in front of all stations, so that no extra headway time should be included.
- When the disruption happens, the following actions all happen instantly: the collection of the required data and the implementation of it in the model, the computation of the new schedule and the communication of the new schedule to the trains.
- Trains can only start and end an itinerary at a border station. Or, when they are stuck between a border station and the blockade, they can end their itinerary at the first station at which they can turn around.
- Trains of a certain service type, i.e. local train or intercity, can only take over other services of the same type.
- The schedule will not cause any problems for the period after the lifting of the blockade.
- Trains do not interfere with trains that are not included in the model.
- Trains of different service types do not travel significantly faster than each other between two consecutive stations, i.e. the original schedule does not contain any data to contradict this. By significantly we mean that a train should speed wise not be able to pass a train of another service level during one driving activity. Otherwise the model could implement one train passing another, even though there is only one track they can use.
- The cancellation of each itinerary is equally unwanted. The same goes for each minute that any event is delayed. This enables us to simply punish the total of cancellations and the total delay of all events in the objective function.

3. Model

In this chapter, we will present the ILP model that can find a feasible solution for a scheduling problem of the aforementioned kind. Apart from a few adjustments, the model is very similar to the one presented in Louwerse and Huisman (2012). Another example of an event-activity network model has been described extensively in Schachtebeck and Schöbel (2010). The output of this model will contain the updated times of all the departures and arrivals in the problem area during the affected period. It will also determine if train itineraries must be cancelled and what a train's next itinerary shall be after finishing one.

3.1 Sets and main variables

We can formulate the problem as a directed graph $N = (E,A)$, where E is the set of nodes representing all the train events and A is the set of arcs representing all defined activities between the events. The set of events includes all the original departure, arrival, short stop and through-times events $e \in E$ of trains planned during the scheduling period. All activities $a \in A$ can be defined as one of the following three types:

- Train activities $a \in A_{train}$ are all activities between events that are performed on the same itinerary, and thus by the same train. These can be driving activities between two consecutive stations or dwell activities at the same station.
- Headway activities $a \in A_{head}$ exist so that enough time is planned between trains using the same track or platform in either the same or the opposite direction. More specifically, a headway activity between two events is defined when those events happen on the same track and within 35 minutes of each other. Where 35 minutes is the summation of the maximum allowed delay (d = 30 min.) of an event and the highest minimum headway time necessary between two trains (5 min.).
- Inventory activities $a \in A_{inv}$ link a train arriving at a border station to itineraries departing from that station. At a border station an inventory activity is only defined between a final arrival event and a first departure event of another itinerary if the following two conditions hold. Firstly, the itineraries must be performed by the same train type and secondly, the departure must be originally planned within 28 minutes before or 90 minutes after the arrival event. 28 minutes, since the departure event could be delayed 30 minutes, but we should still allow for 2 minutes (L_a) minimum inventory time. The 90 minutes is derived by the maximum allowed delay of the first train (30 min.), plus one hour, since we use a restriction that every itinerary should be performed at least once per hour.

We have data available containing all the original planned times q_e for each event $e \in E$. These are coupled to decision variables x_e, which represent the event times for the updated schedule. Besides events, we also want to make decisions regarding itineraries. We therefore declare the set of itineraries $v \in V$, sorted in the order in which they depart from their border station. We define an itinerary as a full run from one border station to another. It should be noted that in this our model is different from the model in Louwerse and Huisman (2012). In that paper hourly regularity is required, which necessitates for the use of train series covering multiple itineraries.

Each event e belongs to an itinerary $\pi_e \in V$. Furthermore, we declare $\eta^{in}(e)$ and $\eta^{out}(e)$ as the sets of arcs into and out of node e respectively. We can divide the itineraries into two directions A and B, so that subsets $V^A \cup V^B = V$. Likewise we define subsets for both types of train services: local trains (V^L) and intercity trains (V^I), making $V^L \cup V^I = V$. We want to be able to determine if an itinerary should be cancelled, this calls for the introduction of the binary decision variable

$$y_v = \begin{cases} 1 & \text{if itinerary v is cancelled} \\ 0 & \text{otherwise} \end{cases}$$

Since some itineraries are not allowed to be cancelled. We define subsets of $V = V^C \cup V^N$, where V^C consists all itineraries v that may be cancelled and V^N consists of all itineraries v that may not be cancelled.

We state decision variable μ as the maximum time existing between the departures of any two consecutive trains of the same service from the same border station. From now on we will call this the maximum interdeparture time.

3.2 Formulation

We have now introduced all decision variables used in the objective function, which is presented below, followed by all the constraints used in this ILP model.

$$\min \alpha_1 \sum_{v \in V} y_v + \alpha_2 \sum_{e \in E} (x_e - q_e) + \alpha_3 \mu \tag{1}$$

$$x_e - q_e \geq 0 \qquad \forall e \in E_{delay} \tag{2}$$

$$x_e - q_e \leq d \qquad \forall e \in E_{delay} \tag{3}$$

$$x_e - q_e = 0 \qquad \forall e \in E_{fixed} \tag{4}$$

$$x_f - x_e \geq L_a \qquad \forall a = (e, f) \in A_{train} \tag{5}$$

$$x_f - x_e + M(1 - \lambda_{ef}) \geq L_a \qquad \forall a = (e, f) \in A_{head} \tag{6}$$

$$\lambda_{ef} + \lambda_{fe} + y_{\pi_e} + y_{\pi_f} \geq 1 \qquad \forall (e, f) \in A_{head} \tag{7}$$

$$x_{\varepsilon_w} - x_{\varepsilon_v} - M(y_v + y_w + \sum_{\substack{u \in V^A \cap V^I \\ v<u<w}} (1 - y_u)) \leq \mu \qquad \forall v, w \in V^A \cap V^I \tag{8}$$

$$x_{\varepsilon_w} - x_{\varepsilon_v} - M(y_v + y_w + \sum_{\substack{u \in V^A \cap V^L \\ v<u<w}} (1 - y_u)) \leq \mu \qquad \forall v, w \in V^A \cap V^L \tag{9}$$

$$x_{\varepsilon_w} - x_{\varepsilon_v} - M(y_v + y_w + \sum_{\substack{u \in V^B \cap V^I \\ v<u<w}} (1 - y_u)) \leq \mu \qquad \forall v, w \in V^B \cap V^I \tag{10}$$

$$x_{\varepsilon_w} - x_{\varepsilon_v} - M(y_v + y_w + \sum_{\substack{u \in V^B \cap V^L \\ v<u<w}} (1 - y_u)) \leq \mu \qquad \forall v, w \in V^B \cap V^L \tag{11}$$

$$\sum_{v \in V^A} (1 - y_v) \geq 1 \qquad \forall (k,h) | T^A(k,h) \geq 2 \tag{12}$$

$$\sum_{v \in V^B} (1 - y_v) \geq 1 \qquad \forall (k,h) | T^B(k,h) \geq 2 \tag{13}$$

$$\sum_{a \in \eta^{in}(e)} (1 - z_a) = 1 - y_{\mu_e} \qquad \forall e \in E_{orig} \tag{14}$$

$$\sum_{a \in \eta^{out}(e)} (1 - z_a) \leq 1 - y_{\mu_e} \qquad \forall e \in E_{dest} \tag{15}$$

$$x_f - x_e + M z_a \geq L_a \qquad \forall a = (e,f) \in A_{inv} \tag{16}$$

$$y_v = 0 \qquad \forall v \in V^N \tag{17}$$

$$y_v \in \{0,1\} \qquad \forall v \in V^C \tag{18}$$

$$\lambda_{ef} \in \{0,1\} \qquad \forall (e,f) \in A_{head} \tag{19}$$

$$z_a \in \{0,1\} \qquad \forall a \in A_{inv} \tag{20}$$

$$x_e \in \mathbb{N} \qquad \forall e \in E \tag{21}$$

$$\mu \in \mathbb{N} \tag{22}$$

The objective function consists of three parts. The first part tries to limit the amount of itineraries that are being cancelled. The second part tries to limit the total delay of all events. And the last part makes sure that the maximum interdeparture time is kept low. The multipliers α_1, α_2 and α_3 should be chosen such that a proper balance exists between the influences of the three parts.

Constraint (2) and (3) state that all events that are allowed to be delayed ($e \in E_{delay}$), can be delayed between 0 and d minutes, where d is equal to 30 minutes in our case. Constraint (4) says that all events that are not allowed to be delayed, i.e. fixed ($e \in E_{fixed}$), should be planned at the same time as in the original schedule. These fixed events are the first departure or last arrival events at border stations of itineraries that run outside our problem area. We add these events, because we want trains ending an itinerary inside the problem area to be able to turn on these itineraries and vice versa. By making these events fixed we make sure that no conflicts will arise outside the problem area.

Constraint (5) ensures that the minimum duration L_a of train activity $a=(e,f)\in A_{train}$ is respected by planning event f at least L_a minutes after event e. Constraint (6) does the same for headway activities $a\in A_{head}$. However, we only want the minimum headway time L_a for headway activity $a=(e,f)$ to be respected if event f happens after event e. Therefore, another decision variable must be introduced:

$$\lambda_{ef} = \begin{cases} 1 & if\ event\ e\ is\ scheduled\ before\ f \\ 0 & otherwise \end{cases}$$

Constraint (7) is introduced so that, as long as neither the itineraries performing event e or f are cancelled, one event is scheduled after the other.

Constraints (8), (9), (10) and (11) produce the maximum interdeparture time in the new schedule. We count these interdeparture times only between first departure events ε_v from the itinerary v's starting border station. We make sure that only consecutive itineraries are examined by relaxing the constraint if either itinerary v or w is cancelled or if an itinerary u departing between itineraries v and w is not cancelled.

Constraint (12) and (13) ensure that each hour $h\in H$ at least one train of each type $k\in K=(I,S)$ leaves into each direction A and B. For this we use matrices $T^A(k,h)$ and $T^B(k,h)$, which declare the number of trains of type k that head into direction A or B respectively during hour h. Also itineraries which started before and are still being performed when the blockade happens and their corresponding starting hours are included. We only apply the constraint to hours of which both originally scheduled itineraries are included in the used data.

In constraint (14) and (15) the binary decision variable z_a is introduced for all inventory activities $a\in A_{inv}$. We define this variable as:

$$z_a = \begin{cases} 1 & if\ inventory\ activity\ a\ is\ canceled \\ 0 & otherwise \end{cases}$$

Constraint (14) couples a first departure event of a non-cancelled itinerary to exactly one last arrival event at the same border station through a non-cancelled inventory activity. Whereas constraint (15) can couple the last arrival event of a non-cancelled itinerary to at most one first departure event from that station through a non-cancelled inventory activity. Lastly, constraint (16) makes sure that for each inventory activity a that is not cancelled, the minimum required time L_a between the corresponding arrival and departure events should be respected.

3.3 Balanced objective function

Like previously mentioned, a proper balance should be found for the multipliers α_1, α_2 and α_3 in the objective function. We have not been presented with any information about the preferences of NS towards the three undesirable effects (cancelled itineraries, total delay and maximum interdeparture time). Therefore, we will have to make an educated guess to find a proper balance.

First we try to relate the total delay to the cancellation of itineraries to find relative values for the multipliers α_1 and α_2. If an itinerary is cancelled then travelers will have to wait for the next train of the same type. So from the traveler's point of view, the cancellation of an itinerary is equally as bad as a delay of the same itinerary by the normal interdeparture time (assuming that the traveler does not mind a fuller train). On the other hand, NS probably prefers cancelling an itinerary over letting it run right before the following itinerary, since the latter is a lot less energy efficient. Taking both preferences into account, we assume that the delay of an itinerary by 75% of the normal interdeparture time, should get the same penalty in the objective function as the cancellation of an itinerary gets. Note, though, that we do not have variables representing the delays of itineraries; we only measure the delays of events. Therefore, we will have to weigh the total delay down by the average number of events in an itinerary. Note, that the average number of events per itinerary is different for each problem, so that eventually the values of the multipliers will vary between problems.

The model should include a penalty for a high maximum interdeparture time to even out the delays of the trains. It should not punish too much though, otherwise the model would get an unfavourable high tendency towards cancelling itineraries. We settle at the point where one additional minute of interdeparture time gets punished slightly more than a one minute increase for a number of events equal to the number of events in the longest itinerary. Now, the relation between the three multipliers can be written as:

$$\alpha_1 = \alpha_2 * \ time\ between\ itineraries\ \ * average\ number\ of\ events * 0.75$$

$$\alpha_3 = \alpha_2 * (number\ of\ events\ in\ longest\ itinerary + 1)$$

We use these values unless stated differently.

3.4 Implementing NS' solutions

Since we want to compare our solutions to those used by NS, we need a way to calculate the delays and cancellations that will result from the schedules provided by NS. This can easily be done by simply narrowing the set of inventory activities A_{inv} down to just the ones used by NS. This way, trains can only turn on the itinerary that NS scheduled it to do. The only thing left for the model to do is determine which itineraries will be cancelled and/or which events will be delayed to find the schedule that NS probably used.

4. Data

In this chapter, we summarize the data that is used as input for the model. Like mentioned before, the two disruptions in Rosmalen between 8:00 and 11:00 and in Culemborg between 14:00 and 16:00 can be divided into two sub problems each. Which results into the problem areas Tilburg-Den Bosch, Oss-Nijmegen, Geldermalsen-Den Bosch and Utrecht-Houten.

4.1 Tilburg-Den Bosch

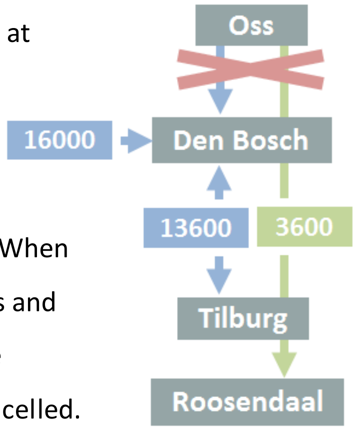

Figure 1: Schematic representation of observed train services between Oss and Roosendaal

In figure 1 a simplification of the problem area southwest of the blockade at Rosmalen is shown. Blue arrows represent local train services and green arrows represent intercity services. A head of an arrow means that this train service normally ends or starts at the station it is pointing at. All shown train services run once every half hour into both directions. When the disruption occurs, a train performing a local train service between Oss and Den Bosch gets stuck at Den Bosch. It shall remain there for as long as the blockade is present, since all services between Oss and Den Bosch are cancelled. The intercity service 3600 is interrupted as well at Rosmalen. However, this service will still be able to run between Den Bosch and Roosendaal. To increase the chance of ending up with an overall feasible solution local train service 13600 between Den Bosch and Tilburg is included in the model as well. This service normally turns on local train service 16000 and vice versa. Therefore we also include the (fixed) arrival and departure events of this service at Den Bosch station.

4.2 Oss-Nijmegen

On the other side of the blockade at Rosmalen we only have to deal with local train service 4400 and intercity service 3600 (figure 2). Both run once every half hour into both directions. All trains that are not between the blockade at Rosmalen and Den Bosch at the moment of the disruption will be used to perform the planned events between Oss and Zwolle. An extra characteristic about this problem is that there is a bridge with a single track located between Nijmegen and Oss. Trains that cross this track in opposing directions should do this with at least a certain safety time in between. The single track covers about 2 km, resulting in a crossing time of almost 1 minute. Now, assuming that the event times of the trains crossing the bridge are measured exactly halfway, we count two half minutes for opposing trains to get to/from the switch at the end/beginning of the single track. Trains passing the same switch in

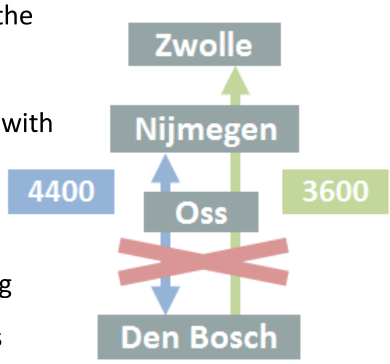

Figure 2: Schematic representation of observed train services between Den Bosch and Zwolle

opposing directions require a minimum headway time of 4 minutes, so that the minimum required headway time between the events of the trains crossing the bridge in opposing directions is 5 minutes in total.

4.3 Geldermalsen-Den Bosch

In the case of the blockade at Culemborg four train services are interrupted (figure 3). Two local train services, 6000 and 16000, run the same track between Houten and Geldermalsen, from where 6000 heads towards Tiel and 16000 towards Den Bosch. Intercity services 3500 and 800 share their route between Houten and Eindhoven, from whereon only 800 continues to Maastricht. Both services do not stop at Geldermalsen and will need to turn around at Den Bosch.

To limit the number of events in our model we merge all events and activities of each itinerary of intercity service 800 between Eindhoven and Maastricht into one train activity that connects the departure event from Eindhoven to the arrival event in Maastricht and vice versa. Three conditions are present so that we are allowed to do so. Firstly, no events of other train services are considered between those cities. Secondly, the delay of an departure event at Eindhoven will result in the same delay of the arrival event at Maastricht, therefore not disrespecting any headway constraints with trains heading in the same direction. Lastly, this entire route consists of at least double-track railway, so that no headway constraints of trains heading in opposing directions are ignored.

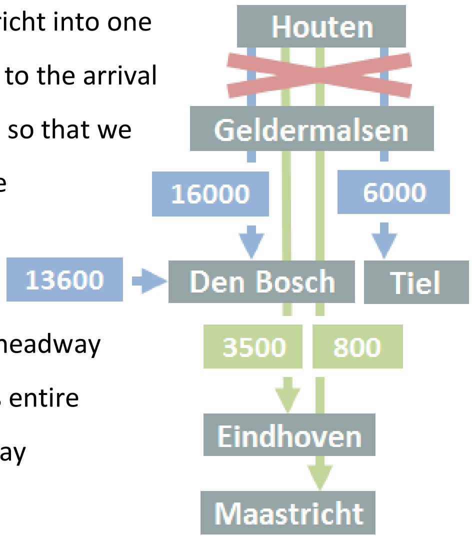

Figure 3: Schematic representation of observed train services between Houten and Tiel/Maastricht

Fixed departure and arrival events are integrated for local train service 13600 at Den Bosch, since this train service normally turns on local train service 16000 at this station and vice versa.

At the time of the disruption, one train of intercity service 800 is running from Den Bosch to Geldermalsen. We need to add events and activities for the train to be able to return to Den Bosch. To do this we should also introduce the corresponding 'original times' q_e of these events by taking the times that it will be first possible for the train to perform them. E.g. when the train arrives in Geldermalsen at 14:05, we introduce a departure event from Geldermalsen in the opposite direction L_a=2 minutes later.

4.4 Utrecht-Houten

In this problem, shown in figure 4, we are dealing with the interruption of the same four train services as in the Geldermalsen-Den Bosch problem. Local train series 16000 and 6000 run every half hour, so that between Utrecht and Houten actually four local trains run every hour. The two interrupted intercity services, 3500 and 800, run the same track between Geldermalsen and

Amsterdam Bijlmer Arena, again twice per hour each. After Amsterdam Bijlmer Arena 3500 continues to Schiphol and 800 to Alkmaar. Since both intercity services do not stop in Houten in their original schedule, they turn around in Utrecht. For the same reason as in the previous problem we merge all events and activities between Amsterdam Bijlmer Arena and Schiphol of each itinerary of train service 3500 into one driving activity. The same holds for the itineraries of train service 800 between Amsterdam Bijlmer Arena and Alkmaar.

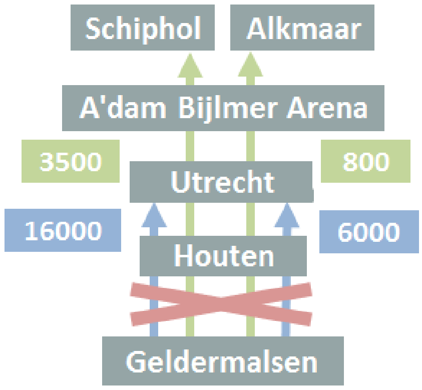

Figure 4: Schematic representation of observed train services between Schiphol/Alkmaar and Geldermalsen

At the moment of the disruption, one train of intercity service 3500 is running between Utrecht and Culemborg. Therefore, we need to introduce extra events and activities to enable this train to return to Utrecht, where it can be assigned to a new itinerary.

5. Results

For solving our problem we analyze the data using Microsoft Office Excel, from which we implement it into modeling software AIMMS. In AIMMS we use the solver CPLEX 12.5 to solve our ILP model. We run AIMMS on a computer equiped with a quad-core 3.6 GHz processor.

5.1 Tilburg-Den Bosch

Statistics of the solution found for the problem Tilburg-Den Bosch are displayed in table 1. It shows that 24 itineraries could be cancelled, but none are. Only 5 out of 32 delayable itineraries are delayed resulting in an average delay of all events of 1.6 minutes. All delayed itineraries have got the same delay for every corresponding event. The maximum interdeparture time is a respectable 38 minutes. All intercity services 3600 can easily turn on themselves without any delay. The local train services mostly turn on themselves, but sometimes turn on the other local train service departing from Den Bosch. I.e. service 16000 sometimes turns on service 13600 and vice versa. CPLEX 12.1 does not seem to have any trouble solving the problem and does this in only 2.7 seconds.

Cancelled itineraries	0/24
Delayed Itineraries	5/32
Average delay events	1.6 min
Max interdeparture time	38 min
Total solving time	2.7 sec

Table 1: Statistics of our solution for problem Tilburg-Den Bosch

In table 9 in the appendix we can see a how NS dealt with the situation in 2011. Naturally all intercity services 3600 turn on themselves in Den Bosch, which is what happens in our solution as well. Unfortunately, we did not take into account that local train service 9600 normally turns on the disrupted local train service 4400. Therefore, we did not include service 9600 into our model, but, since service 9600 does not share any track with the included train services, it would probably not have made the problem any more interesting. NS does not explicitly provide any solution for the local train services 16000 and 13600, which they maybe should, since our solution requires some of them to be delayed. Due to these differences we can not compare our solution to NS' solution for this particular problem.

5.2 Oss-Nijmegen

Table 2 shows that again no itineraries need to be cancelled, unfortunately 14 itineraries are delayed though. All of these delays are caused by the bottleneck at the bridge between Oss and Nijmegen. We can draw this conclusion from the fact that all delayed trains are only delayed from the bridge onwards. It also implies that no trains are delayed when they depart from their border station, so that the maximum interdeparture time remains 30 minutes.

A detailed schematic of the new time schedule can be found in figure 5 in the appendix. It shows at what time all events are performed and by which trains. Note that all trains going in the same direction should keep a safety headway time of 3 minutes between them and that no trains may cross the bridge in opposing directions within 5 minutes after eachother. It looks like all these headway constraints are respected except for some at Wijchen station. E.g. it seems like at 8:32 intercity 6 is getting dangerously close to local train 1. However, this is just because the events of the intercity trains passing Wijchen are not included in the original timetable. Therefore, the points where the graphs of the intercity trains pass Wijchen are merely a visually interpolated representation. We can tell from the graphs of the local trains that the trains can, in fact, travel faster between Nijmegen and Wijchen than shown in the graphs of the intercity trains so that the intercity trains will pass Wijchen respecting the headway constraints with the local trains ahead of them. The total solving time is higher than that of the other problems, probably because of the bridge, but it is still a respectable 7.9 seconds.

Cancelled itineraries	0/23
Delayed Itineraries	14/34
Average delay events	0.7 min
Max interdeparture time	30 min
Total solving time	7.9 sec

Table 2: Statistics of our solution for problem Oss-Nijmegen

If we compare our solution to the one used by NS (table 9 in appendix), we see that they let intercity trains 3600 make the same turns in Oss as us. The local trains 4400, however, are scheduled differently. Our model lets the local trains arrive in Oss with a delay, as a consequence it is better to let them turn on the itineraries that depart half an hour after the one that NS lets them turn on. This difference is a result of the fact that after the moment of the disruption, two local trains arrive in Oss from opposite directions. But since no trains have to drive to Den Bosch anymore, only can directly be assigned to a departure event; the other train will be stored for a while and can be assigned to a next departure event without delay. Table 3 shows the implications of the schedule that NS used. Although less itineraries are delayed, the average delay of all events increased quite a bit. We should therefore conclude that our schedule is better than the one NS used in 2011.

Cancelled itineraries	0/23
Delayed Itineraries	12/34
Average delay events	1.2 min
Max interdeparture time	30 min
Total solving time	6.0 sec

Table 3: Statistics of NS' solution for problem Oss-Nijmegen

5.3 Geldermalsen-Den Bosch

The solution regarding the problem area Geldermalsen-Den Bosch requires one itinerary to be cancelled. This itinerary belongs to the local train service 16000 and it would have run from Den Bosch to Geldermalsen. Four itineraries are either fully or partially delayed, resulting in an average delay of 0.4 minutes. Even though there are delays, table 4 displays the maximum interdeparture time as 30 minutes. This is because all delayed itineraries are either not delayed from the start or are first itineraries (an explanation of this is presented in the results of the Utrecht-Houten problem). The data show that some trains turn from intercity service 800 on 3500 in Den Bosch and vice versa. All local train services 16000 turn on train service 6000 in Geldermalsen and the other way around except for the first itinerary of service 6000 which turns on itself. The same happens to train services 13600 and 16000 in Den Bosch.

Cancelled itineraries	1/29
Delayed Itineraries	4/44
Average delay events	0.4 min
Max interdeparture time	30 min
Total solving time	2.8 sec

Table 4: Statistics of our standard solution for problem Geldermalsen-Den Bosch

Since our standard solution requires an itinerary to be cancelled, it would be interesting to see what happens when we increase the punishment for any cancellations in our model to the point where no itineraries are cancelled. To achieve this, we doubled the value of the parameter α_1 that is linked to the number of cancellations. The statistics of the resulting alternative solution are displayed in table 5. By shifting the preference away from the cancellations we have caused one extra itinerary to be delayed, increased the average delay of all events by 0.8 minutes and increased the maximum interdeparture time to 34 minutes.

Cancelled itineraries	0/29
Delayed Itineraries	5/44
Average delay events	1.2 min
Max interdeparture time	34 min
Total solving time	3.3 sec

Table 5: Statistics of our alternative solution (doubled α_1) for problem Geldermalsen-Den Bosch

In the solution presented by NS (table 10 in appendix), train services 800, 16000 and 6000 all turn on themselves in Den Bosch and Geldermalsen, respectively. We can safely assume that they let train services 13600 and 3500 turn on itself in Den Bosch as well. When we implement the restriction that trains can only turn the way it is scheduled in NS' solution, we receive the output summarized in table 6. NS' solution performs notably worse in all four shown statistics. Especially the increase of cancelled itineraries to four and the increase of the maximum interdeparture time to 52 minutes are undesirable results. We can explain the

Cancelled itineraries	4/29
Delayed Itineraries	5/44
Average delay events	0.5 min
Max interdeparture time	52 min
Total solving time	3.0 sec

Table 6: Statistics of NS' solution for problem Geldermalsen-Den Bosch

cancellations by taking a deeper look at the data. At the moment of the disruption (08:00), a train just finished an itinerary of train service 16000 in Den Bosch and started an itinerary of train service 13600. NS' new schedule requires trains to keep running the same train service. As a consequence, no train can start the itinerary of train service 16000 scheduled to leave from Den Bosch at 08:03. In the meantime, a train finished an itinerary of train service 13600 in Den Bosch at 08:00, and is now idly stalled there. This train stays in Den Bosch for the total duration of the disruption, even though train service 16000 is one train short for this entire period. This shortage results in the cancellation of four itineraries on the route between Den Bosch and Geldermalsen. The increased maximum interdeparture time is a direct result of these cancellations.

Comparing all three presented solutions, we would say that the solution we found by using the standard multipliers is the best. It has the lowest value for both maximum interdeparture time and delay at the cost of only one cancelled itinerary.

5.4 Utrecht-Houten

In this problem four train services need to be rescheduled, which enables trains to turn on itineraries of another service in Utrecht or Houten. Table 7 shows that none of the itineraries need to be cancelled. Four consecutive itineraries leaving from Schiphol are fully delayed by 2 minutes and one other itinerary is delayed up to 3 minutes. Together these delays results in an average delay of events of 0.2 minutes.

Cancelled itineraries	0/30
Delayed Itineraries	5/44
Average delay events	0.2 min
Max interdeparture time	30 min
Total solving time	2.1 sec

Table 7: Statistics of our solution for problem Utrecht-Houten

A problem arises, when we get to the output of the maximum interdeparture time (30 min). This should be 32 minutes, since some itineraries are fully delayed by 2 minutes. However, due to the non-cyclical implementation of our model, it does not count the time between the first itinerary and the one that departed from Schiphol before the disruption happened. We should note, though, that since the delay is small, this will not influence the solution a lot.

To give insight into how all trains run we included tables 11 through 24 in the Appendix that show the routes that all the trains (3 local trains , 12 intercity trains) in the problem area should run during the disruption period. Note how tables 15 through 18 include the delays that start in Schiphol and that these are caused by the late previous arrival at this station. Table 17 displays how intercity 5 has to start its route in Culemborg, since this train is driving from Utrecht towards Culemborg at the moment of the disruption. Apparently, it arrives at Utrecht with some delay (14:20 instead of 14:17). This is done, so that enough headway time exists between intercity 5 and local train 2 (table 12), which is scheduled to arrive at Utrecht at 14:17.

Looking at the solution of NS presented in table 10, we can see that, again, all train services turn on themselves in Utrecht and Houten, at least that is what we have to assume since not all services are included in the table. When we tell our model to let all trains turn exactly the way it is done by NS, we get an output with only one delayed but with six cancelled itineraries (see table 8). As a consequence of the cancelled itineraries the maximum interdeparture time doubles to 60 minutes in NS' solution.

Cancelled itineraries	6/30
Delayed Itineraries	1/44
Average delay events	0.0 min
Max interdeparture time	60 min
Total solving time	2.1 sec

Table 8: Statistics of NS' solution for problem Utrecht-Houten

The difference between our solution and the solution of NS comes down to whether or not itineraries that are delayed by 2 minutes should be cancelled. In our view this would be far from desirable, since it stretches the waiting times for some passengers from 2 to 60 minutes.

6. Conclusion

Our model was able to produce feasible schedules for all our four problems. The three undesired effects by which we measure the optimality of the schedule (cancelled itineraries, total delay and maximum interdeparture time) were kept reasonably low. Certainly, if we compare our schedules to the ones used by NS in 2011. We only had to cancel one itinerary in one of the four problems, whereas NS had to cancel ten itineraries spread over the three problems we were able to inspect. In regard to the amount of delay, NS only slightly outperformed us in one out of three cases, but at the cost of many cancellations and a doubled maximum interdeparture time. The average delays per event never exceeded 1.6 minutes in our solutions and the maximum interdeparture was never more than 38 minutes.

Furthermore, we could say that the computation time needed to solve the problems is very respectable; it never exceeds 8 seconds. This time is probably not significant compared to the time needed to implement the data and to communicate the solution to the trains and public. It seems that by applying our presented model to these problems, travelers do not have to experience mayor inconveniences, aside from taking detours or busses to get past the blockade.

However, quite a few strong assumptions had to be made for our model, making it an unfair comparison to the schedules used by NS. E.g. it is impossible for NS to update their schedule instantly. Disregarding this fact probably gave us a huge advantage. We also assumed that the included trains do not interfere with any non-included trains, which could have easily resulted in an increase in delays or cancellations. Overall, we should state that, before being able to draw any conclusions about the usefulness of our model, tests of our model would have to be performed on real life situations.

7. References

Louwerse, I., and Dennis Huisman. *Adjusting a Railway Timetable in case of Partial or Complete Blockades*. No. EI 2012-23. Erasmus School of Economics (ESE), 2012.

Liebchen, Christian, et al. "Computing delay resistant railway timetables."*Computers & Operations Research* 37.5 (2010): 857-868.

Detailed map of the Dutch railway network: http://www.sporenplan.nl/

Details about train turnings in the regular NS train schedule: http://wiki.ovinnederland.nl/

8. Appendix

Station	Serie	Arrival time	Turns on serie	Departure time
Oss	3600	-.03	3600	-.26
Oss	3600	-.33	3600	-.56
Oss	4400	-.14	4400	-.14
Oss	4400	-.44	4400	-.44
Den Bosch	3600	-.10	3600	-.19
Den Bosch	3600	-.40	3600	-.49
Den Bosch	9600	-.46	9600	-.13
Den Bosch	9600	-.16	9600	-.43

Table 9: Solution used by NS regarding the disruption at Rosmalen in 2011

Station	Serie	Arrival time	Turns on serie	Departure time
Geldermalsen	6000	-.04	6000	-.23
Geldermalsen	6000	-.34	6000	-.53
Geldermalsen	16000	-.48	16000	-.09
Geldermalsen	16000	-.18	16000	-.39
Houten	16000	-.20	16000	-.38
Houten	16000	-.50	16000	-.08
Den Bosch	800	-.21	800	-.39
Den Bosch	800	-.51	800	-.09
Utrecht	800	-.05	800	-.25
Utrecht	800	-.35	800	-.55

Table 10: Solution used by NS regarding the disruption at Culemborg in 2011

Below are the tables displaying our solution to the Utrecht-Houten problem. Only arrival and departure events are shown with their corresponding time in minutes after the disruption occurred (14:00). Times of delayed events are displayed as original time + delay.

Service	6000A	6000B	16000A	16000B	6000A	6000B	
Station	Before blockade	Houten	Utrecht	Houten	Utrecht	Houten	After blockade
Time		5 | 23	32 | 41	50 | 68	77 | 86	95 | 113	

Table 11: Route of local train 1

Service	16000B	6000A	6000B	16000A	16000B	6000A	
Station	Before blockade	Utrecht	Houten	Utrecht	Houten	Utrecht	After blockade
Time		17 | 26	35 | 53	62 | 71	86 | 98	107 | 116	

Table 12: Route of local train 2

Service	6000B	16000A	16000B	6000A	16000B	6000A		
Station	Before blockade	Utrecht	Houten	Utrecht	Houten	Utrecht	Houten	After blockade
Time		2 | 11	20 | 38	47 | 56	65 | 83	92 | 101	110	

Table 13: Route of local train 3

Service	3500A		800B		
Station	Before blockade	Utrecht		After blockade	
Time		17	55		

Table 14: Route of intercity 1

Service	3500B		3500A		
Station	Before blockade	Schiphol		Utrecht	After blockade
Time		14	14+2	47+2	

Table 15: Route of intercity 3

Service	3500B		3500A		3500B	
Station	Before blockade	Schiphol		Utrecht		After blockade
Time		44	44+2	77+2	103	

Table 16: Route of intercity 4

Service	3500A		3500B		800B		800A		800B	
Station	Before blockade	Culemborg		Utrecht		Schiphol		Utrecht		After blockade
Time		3	5	17+3	43	74	74+2	107+2	115	

Table 17: Route of intercity 5

Service	800A		3500B		3500A	
Station	Before blockade	Utrecht		Schiphol		After blockade
Time		65	73	104	104+2	

Table 18: Route of intercity 6

Service	800A		800B	
Station	Before blockade	Utrecht		After blockade
Time		5	25	

Table 19: Route of intercity 7

Service	800A		800B	
Station	Before blockade	Utrecht		After blockade
Time		35	85	

Table 20: Route of intercity 8

Service	800B		800A		
Station	Before blockade	Alkmaar		Utrecht	After blockade
Time		4	26	95	

Table 21: Route of intercity 9

Service	800B		800A	
Station	Before blockade	Alkmaar		After blockade
Time		34	56	

Table 22: Route of intercity 10

Service	800B		800A	
Station	Before blockade	Alkmaar		After blockade
Time		64	86	

Table 23: Route of intercity 11

Service	800B		800A	
Station	Before blockade	Alkmaar		After blockade
Time		94	116	

Table 24: Route of intercity 12

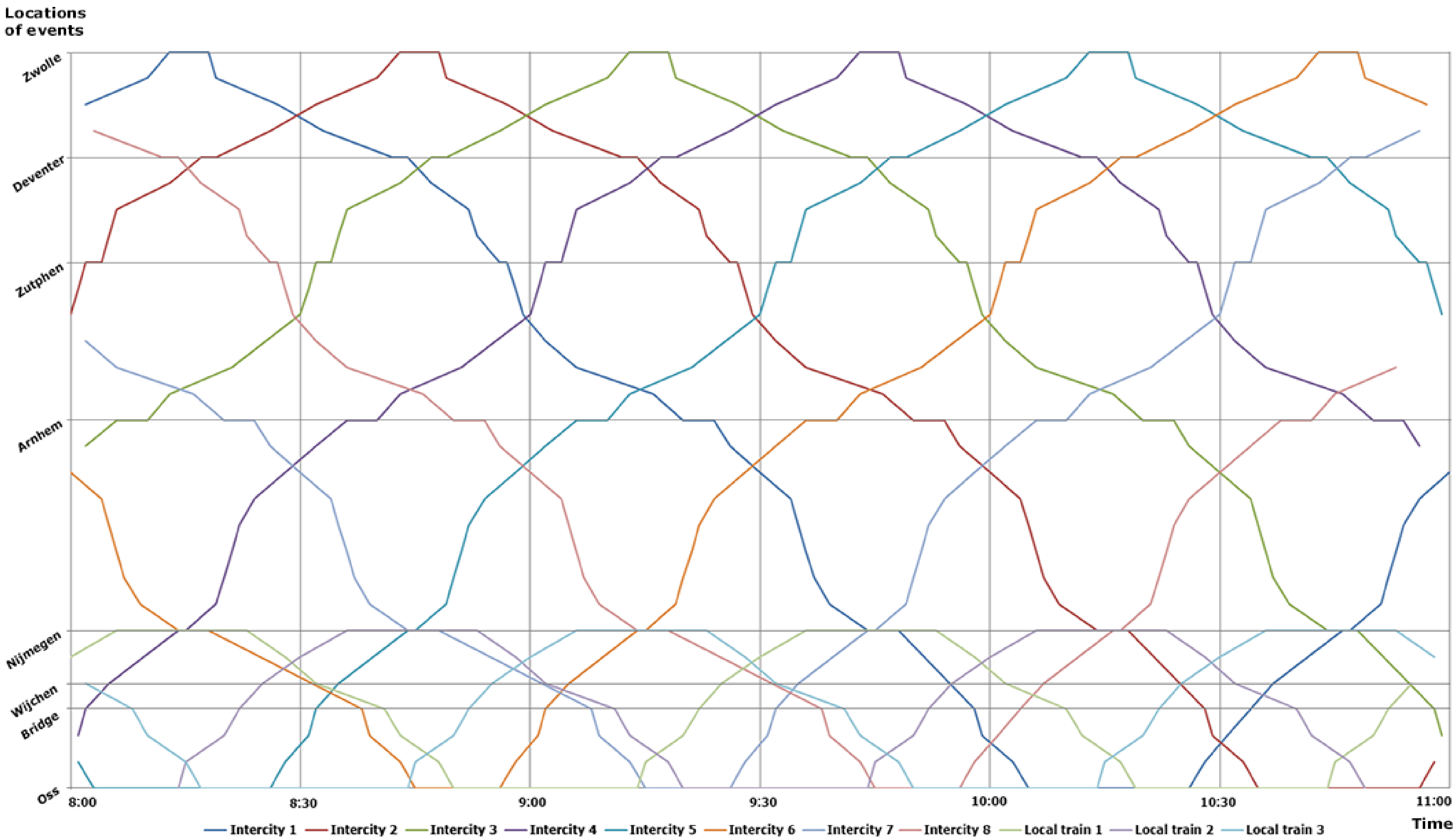

Figure 5: A schematic of all rescheduled trains during the affected period between Oss and Zwolle. The lines represent at which time (x-axis) the trains perform an event (y-axis). Only the interesting stations are shown.