

# **CONNECTION DISPATCHING – AN ALGORITHMIC AND VISUAL SUPPORT FOR THE DISPATCHER**

Anselmo Stelzer, [stelzer@verkehr.tu-darmstadt.de](mailto:stelzer@verkehr.tu-darmstadt.de)

Andreas Oetting, [oetting@verkehr.tu-darmstadt.de](mailto:oetting@verkehr.tu-darmstadt.de)

Friederike Chu, [chu@verkehr.tu-darmstadt.de](mailto:chu@verkehr.tu-darmstadt.de)

*Technische Universität Darmstadt, Germany, Chair of Railway Engineering*

*Tel.: +49/6151/16-65911*

## **ABSTRACT**

Securing connections in public transport is a mean to raise travellers' satisfaction and make their trips more convenient. Nevertheless, securing connections affects the trips of other travellers and therefore is not always the best solution regarding the complex train network. Thus, dispatching connecting trains is a complex task that demands special knowledge about the railway network, travellers, local situations and conditions, the schedule, rolling stock circulations, employees' duty rosters and rules concerning the dispatching process. Furthermore, the topic becomes more relevant recently as the traffic on European railway networks grows more complex whilst the demand for a proper connection dispatching on the part of travellers as well as the authorities rises.

To further improve this complex task, a decision support for the dispatcher is desirable. We focus on methods to (visually) support the dispatcher in his daily work. A mean to achieve a better connection dispatching is the detection and display of conflicts. Detected conflicts are evaluated and prioritized. With the results of these processes, the connection conflicts are displayed in a proper and easy-to-catch way.

Furthermore, we are investigating ways to automate conflict resolution routine processes. Here, many side conditions regarding the German railway's system of rules have to be taken into account. Also, we consider the whole dispatching process including the rejection of a connection and referencing to alternative possibilities for a connection request.

The developed solution is tested in the Railway Operations Centre Darmstadt. Possible improvements are verified in cooperation with the (connection) dispatcher to assure an acceptance of the developed tools by the future user.

The results of our research are designed such that algorithms can be directly implemented in the productive system of the German railway dispatching environment.

*Keywords: connection, dispatch, disposition, visualization, display*

## **1. INTRODUCTION**

Connection dispatching becomes more relevant for all transportation companies. On one hand this is due to raising traveller expectations regarding securing connections. On the other hand transport associations as contracting authority more and more often demand fixed rules how to handle connection conflicts including penalty payments if proceeded differently [11].

This implies a significantly rising number of connection conflicts to be actively handled by a dispatcher. The staff available for this task today generally does not cover the workload of future emergence of connections to be dispatched. Either more staff is needed or the existing staff needs a better software support.

Also important is the jurisdiction of a dispatching action regarding the railway domain. Here, generally at least two actors are responsible. For example, the transportation company is responsible for the decision if a distributing train waits or not. But it cannot solely decide about how long the train will wait if there is an impact on the infrastructural usage. This demands a close interaction with the responsible authority.

It is evident that a support for the dispatcher is needed. The support has to reflect certain needs: visualization, dispatching methods and automatic dispatching rules, where applicable. A solution also has to respect the different jurisdictions.

In this paper we present a possibility to support a connection dispatcher in his routine tasks. In the next chapter we talk about work that is related to ours and point out the difference between our approach and the ones presented. Then we give some general terms needed in this paper. The next chapter explains the context and restrictions our work is situated in. In chapter 5 we point out how connection conflicts are detected. In the following chapter we introduce a different approach of visualizing connections and their conflicts. Based on the visualization, we present reality based conflict solving methods and an approach to automation of the conflict solving part in chapter six. The results have been evaluated which is shown in the chapter seven and we give a brief summary in the end.

## **2. RELATED WORK**

In this chapter we will give a short overview about existing work in the domain of connection dispatching.

A common approach to face the problem of connection dispatching is to create a graph model. Apart from the schedule the model can also contain dependencies of which trains precede others to simulate track occupancies. To optimize calculation time on the graph and to find optimal solutions the problem is reduced to timing constraints. It is difficult to model all dependencies in railway operations in a graph model. Also, results of the optimization on the

graph can appear incomprehensible to the local connection dispatcher. A representative of this approach is [13]. The aim is to reduce the overall passenger delay what benefits the traveller. The operating company, however, might be interested in a different optimization strategy.

The approach presented in [1] uses a model that creates different predictions on the current situation for different dispatching possibilities. The predictions are created by a different tool and used as an input for the presented work. A cost function is used to evaluate the different possibilities and choose the best. Also future connections are taken into consideration. The authors reduce calculation time and the impact of imprecise predictions in the further future by limiting the simulation of future conflicts to an upper boundary of stops and use static information for evaluation afterwards. The approach is infrastructure based. The optimization is performed across the whole network. This requires data of all transportation companies and the infrastructure. Transportation and infrastructure are not separated in this approach.

[9] and [10] use a graph model to optimize calculate passenger cost in case of breaking and saving connections. For this the amounts of interchanging, passing through and boarding passengers are summarised. The dispatching effects in the network are included in this approach. The cost function is strictly passenger oriented.

In [8] possible disutility functions to evaluate the travellers' discomfort are presented. The authors simulate the passenger behaviour in case of delays and broken or secured connections. They use tabu search to find an optimal solution. Again, this approach is strictly passenger oriented.

One approach to automate dispatching is a rule or knowledge based approach. The knowledge has to be derived from real actions or rules. An example for research area is the work of [5] and [6]. In this work they author derived the rules by asking and discussing with dispatchers. Problems are that often more than one rule from the base fire or rules are contradictory. Furthermore it appears that not all input data which is needed to apply a rule is known or it might be interpretable in a wider range. To solve this, Fay uses fuzzy logic. The problem for a dispatcher here is that the solution can produce incomprehensive results to the user. Also the separation of infrastructure and transportation companies is difficult to model in this approach.

Most of the presented approaches are mainly passenger oriented. For a transportation company it is helpful to also consider existing rules or factors that are not solely based on the passengers. Furthermore no approach of the screened literature dealt with the visualisation of connection conflicts.

### **3. TERMINOLOGY**

In this chapter terms that are needed in the context of connection dispatching will be introduced.

## **Transit Passenger**

The transit passenger uses a certain stop to change from one means of transport to another. His travel does neither start nor end in the particular stop.

Information about transit passengers can be elevated with different means which at the same time have different usage. A common way is to use models of travel demand that also include interchanges. Numbers derived from these models cannot be reliably used for a dispatching process, though, as they are averaged across several journeys. Other possibilities aim at collecting actual data regarding the current journey. Current research will make this information available in the future [14]. With these numbers more reliable connection dispatching can be performed.

Generally spoken, the more information is known about a passenger, the better and more precise dispatching regarding his travel can be provided.

We will consider the transition passengers as set  $P$  which contains real passengers or an estimation derived from some model. A set is used to be able to put different types of elements into the set. Depending on the type of the element the information can be handled differently, e.g. a special transition period and a highlighted visualization to the dispatcher for people with reduced mobility could be realized.

## **Transition Period**

The minimum transition period  $t_u$  defines the necessary time to change at a stop. It can be a station-wide value or a value for the link from platform to platform. A transition time can also be defined as a changing time from one stop to a nearby equivalent stop. The latter defined precise times are rarely used in practice yet. One reason for this is a high effort to be undertaken for collecting and maintaining the relevant data. Still we recommend using this more exact definition as it enables a much more precise conflict detection.

## **Connection**

A connection is the possibility for transit passengers to change from an incoming vehicle (feeder) to an outgoing vehicle (distributor) in compliance with the minimum transition period in a certain location. The vehicle can be a bus, tram train or other means of public transport. Also part of a connection is a set of transition passengers  $P$  which can be empty or unknown. Usually the minimum transition period not only has a lower boundary but also an upper boundary to avoid long stays in a stop place. While the lower boundary is generally definite the upper boundary can vary depending on the current configuration of a transportation company or the traveller's profile.

A connection therefore is defined as a tuple:

$$con := (feeder, distributor, feeder\ location, distributor\ location, P)$$

Each element of the tuple or the combination of elements contain further information such as an arrival time for the feeder, a departure time for the distributor and a minimum transition period depending either just on the location or on the feeder's platform and the distributor's platform.

A connection can be derived by different means. A first possibility is the manual maintenance, e.g. by a marketing department, but also by the operation when it comes to the maintenance of standard waiting times. A second possibility is the calculation of connections using the planned timetable. A special group is formed by ad-hoc connections which is calculated using the disposition timetable and can contain connections which are impossible referring to the planned timetable.

### **Standard Waiting Times**

Some transportation companies work with defined times that can be used by the distributors to secure connections that are conflicted. The time is generally defined depending on feeder and a distributor or a group of such and is given in minutes. The advantage of predefined waiting times is that appearing conflicts e.g. on the infrastructure have already been taken into consideration and should not be a problem.

Standard waiting times can be expressed as a function depending on the feeder, the distributor and the connection locations:

$$w := f(\text{feeder}, \text{distributor}, \text{feeder location}, \text{distributor location})$$

Waiting times are also more precisely described in [7].

### **Dispatching Times**

Depending on the action the dispatcher wants to perform, a different period of time is needed. That could be due to distinct communication procedures or various actors to deal with. The dispatching time is linked to the dispatching action. The dispatching time is needed, to find possible dispatching alternatives regarding the available time till a decision is due.

## **4. PROBLEM DESCRIPTION**

Aim of the research was to create a visual support for the connection dispatching as well as a decision support for a dispatcher of a transportation company. The research intent could be divided into several parts which are also reflected in the chapter structure.

As the whole research activity concerns connection conflicts, at first, a method to detect conflicts had to be defined (chapter 5). The method uses planned data, prognosis and real time data.

Secondly, an alternative way of displaying connections in general and conflicted in particular had to be found and defined. For this we followed the approach of using a matrix (chapter 6).

The user of the application should be relieved from routine work to gain time and focus for more complex problems. Therefore a possible conflict resolution was analysed that holds in the context of different jurisdictions and is implementable on the available data (chapter 7). The conflict resolution is based on rules derived from directives concerning the dispatching process. The application always has to step back if a dispatcher takes any dispositive action.

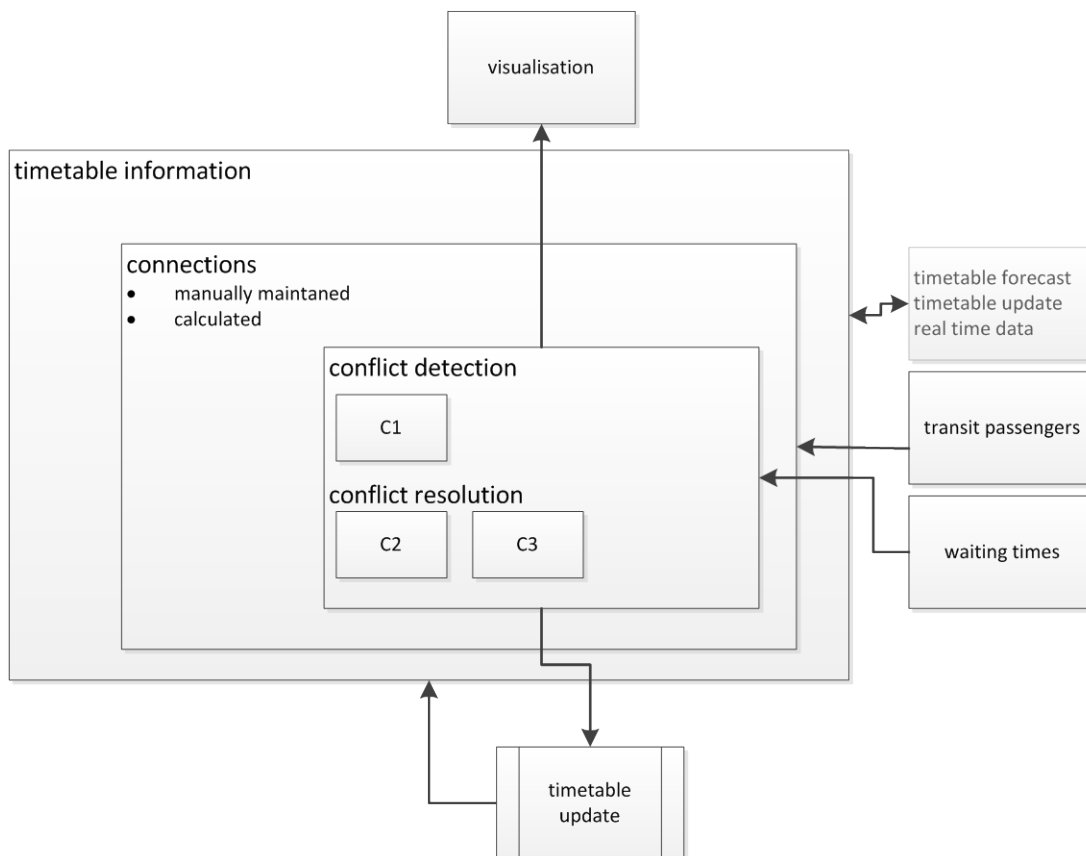


Figure 1 – Overview of the main components of the problem

Part of the research was to be very close to operations to be able to find a solution that is also implementable in a productive environment. Therefore we omit assumptions to reduce the problem complexity. This also includes a strict separation of train operating companies and the infrastructure which already limits available conflict resolutions. Jurisdictional rules and company rules are to be taken into decision. This also means that we, for now, do not try to optimize the global network. We analyse the process at the location where it is settled today and improve it from there. For this, we treat the connection as the dispatcher does.

In Figure 1 a scheme of our system is drawn. We rely on external timetable data (shown greyed out). The other elements are actively used in our approach. It is visible that on one hand we rely on timetable updates that are provided by another system, e.g. a forecast mechanism. Our system receives timetable updates directly from the servers of German Railways. This enables us to work with realistic data of the complete German railway

network. On the other hand we apply timetable updates in those cases a dispatching action leads to a delay for a distributor.

It was desirable to find possibilities to evaluate the solution with dispatchers which implied the implementation of a prototype.

We did not consider problems of prognosis or update mechanisms although we are able to insert relevant data to our system. We assume as well that connections are defined and delivered outside our application. Nevertheless, if missing, we are able to compute reasonable connections based on the delivered timetable. Automation of dispatching methods in this paper deals with the application of waiting times to a train and automated communication towards travellers, train drivers and train attendants. Other dispatching actions can be performed using existing tools whilst our prototype can be integrated into such a solution. As mentioned before, our main objective is not to globally optimize delays in the network but to reduce the workload on the dispatcher.

## **5. DETECTION OF CONNECTION CONFLICTS**

For our work, we operate on a given timetable and rely on external mechanisms that update it with new planned data, prognosis or real time data, unlike for example [2] where the update of the timetable is part of the process. Creating prognosis is a central process that is also relevant for many other systems and processes. That is why it needs to be separated from connection dispatching. Updates of the timetable are created by our approach, but only punctual (see also section 7). The updates will then also act as an input for the prognosis module. We attached our system to servers of German Railways which provide apart from planned data also prognosis and real time data. This data provides a very realistic status of the operational situation. Based on this data, we apply components for conflict detection, conflict evaluation and automatic dispatching methods (see also Figure 1).

As we are using a connection oriented approach (see chapter 4), the first step in our model is to monitor relevant connections. In this step we apply already a first filter method to reduce the number of processed connections. This was necessary as we used all train connections within Germany on one day as a base for our research. Only connections meet current time constraints as well as local constraints regarding the dispatcher's responsibility are selected for further observation. These selected connections form a subset  $S$  of all available connections. The filtering leads to a significant reduction in runtime.

The connections in the subset  $S$  are monitored using the updated times for feeder and distributor trains from the German Railway's servers. As the time can be either a real time, a prediction or a planned time, it has to be determined, which of the available times can be used in the current context as the different times might be inconsistent. Furthermore transition periods are needed to recognize a connection conflict to respect the travellers' behaviour in a station. Referring to the arrival time of the feeder as  $t_A$  and the departing time of the distributor as  $t_D$ , a connection is in conflict as soon as a passenger does not have enough time to reach his distributor regarding the minimum transition period  $t_u$ :

$$t_D - t_A - t_u < 0$$

The result of the term also provides information about the time the feeder can delay without causing a conflict when the result is positive or the time the distributor needs to wait to secure the connection when the result is negative. We will further refer to this variable as connection buffer:

$$t_P := t_D - t_A - t_u$$

## Evaluation of Connection Conflicts

Having detected a conflict, an evaluation is needed to present the dispatcher what further action needs to be taken. The first choice to deal with a connection conflict is most of the times to the decision to wait or not to wait. This question is also addressed in [13], [2] and [1]. Apart from this, also other dispatching possibilities exist and could be used instead of delaying a distributor. A distinct possibility could for example be ordering a taxi for passengers that missed their last connection. For this existing dispatching software is used. The dispatcher needs a support to measure to quickly recognize if a conflicted connection is worth to be secured by delaying another vehicle or if that effort is too high and the connection should be broken. For this we implemented three categories for connections:

$$C := \{C_1; C_2; C_3\}$$

The first category  $C_1$  is for connections that are not in conflict. Subsequently a connection will be referred to as *con*, a category as *cat*.

$$1. C_1 \ni \{con \mid t_P > 0\}$$

The second category  $C_2$  is for connections that are in conflict but securing the connection by delaying the distributor might be a qualified solution. A third category  $C_3$  is for conflicted connections for which the dispatcher should not consider delaying the distributor. Nevertheless, dispatching is needed for these conflicted connections as well. The assignment to a category thus provides a proposal regarding the dispatching action. This will also be reused by the automatic conflict resolution (see chapter 7). The categories are disjoint. Each connection that is an element of  $S$  will be assigned to a category:

$$\forall con \in S \exists cat \in C$$

German railways use standard waiting times to provide a mechanism that allows a quick decision whether to secure a connection or not. The waiting times described in [3] are up to 30 minutes which is only feasible for regional areas with little allocation of infrastructure. These waiting times will be referred to as a set  $W^+$ . In general the dispatcher is free to exceed the standard waiting time (subject to the additional agreement of the infrastructure). However, a waiting time can also have the non-numerical value “no waiting time” which prohibits applying any waiting time to a distributor. They form the set  $W^-$ .  $W^+$  and  $W^-$  are disjoint.



$$W^+ \cap W^- = \{\}$$

$W$  is the set of all waiting times.

$$W := \{W^+ \cup W^-\}$$

The possibility to apply a waiting time to a distributor that secures a connection implicitly restrains it from being assigned to  $C_3$ .

$$2. C_2 \ni \{con \mid t_p < 0 \wedge con \in W^+\}$$

$$3. C_3 \ni \{con \mid t_p < 0 \wedge con \in W^-\}$$

Points 1. to 3. do not cover all connections yet as there are connections which are conflicted ( $con \mid t_p < 0$ , violates 1.) and do not have any waiting times defined ( $con \notin W$ , violates 2. and 3.). For those connections we define a fix threshold  $w_{max}$  that defines the maximum waiting time for which we consider a delay of the distributor as reasonable.  $w_{max}$  can be globally defined as well as depending on the train type (regional, high speed, ...). Note that  $w_{max}$  is not the general maximum waiting time:  $\exists w \in W \mid w > w_{max}$ . The maximum waiting time defines the threshold until which it seems reasonable for a distributor to wait in case the defined waiting time is lower than  $w_{max}$  and the connection does not have a “no waiting time” constraint or no waiting time is defined.

$$4. C_2 \ni \{con \mid t_p < 0 \wedge -t_p < w_{max} \wedge con \notin W\}$$

$$5. C_3 \ni \{con \mid t_p < 0 \wedge -t_p > w_{max} \wedge con \notin W\}$$

It follows

$$6. C_1 := \{con \mid t_p > 0\}$$

$$7. C_2 := \{con \mid t_p < 0 \wedge con \in W^+\} \cup \{con \mid t_p < 0 \wedge -t_p < w_{max} \wedge con \notin W\}$$

$$8. C_3 := \{con \mid t_p < 0 \wedge con \in W^-\} \cup \{con \mid t_p < 0 \wedge -t_p > w_{max} \wedge con \notin W\}$$

Each of the categories represents a recommendation of action to be taken by the dispatcher. Connections that are element of the first category do not need any action to be taken apart from communication towards the traveller. For members of the second category, it is recommended to take a closer look on the conflict and then take a decision about securing it. For members of the third category, it is recommended not to secure the connection as the possible delay to the distributor is considered to be too high. Still other dispatching actions need to be undertaken bring the traveller to his desired destination. The three categories represent a pre-evaluation of all connections to the dispatcher.

## **Prioritization of Connection Conflicts**

Apart from a pre-evaluation of a conflict, an order of all connections has to be defined which represents a priority of one connection above another. For this we propose the usage of a list

$L$  in which the position of the connection  $con \in L$  represents its order. There are several factors that can be taken into consideration to contribute to the priority of a connection.

One of the factors is the amount of transit passengers that want to use the considered connection. Another factor is the type of the connection, meaning the method it was derived from (see chapter 3). A planned connection that is communicated towards the travellers is generally more important than a simply calculated one. Relevant for the priority also is the remaining time for a dispatching action. As this time also depends on the dispatching action to be undertaken it is difficult to number. In our prototype we decided to choose the minimum dispatching time of all actions. Obviously the timings and types of the involved trains are relevant as well.

These factors form a function that allows ordering the list of connections. The priority list will be used for the visualization of connections. The ordered list can be cut after a configurable amount of conflicts to reduce the displayed workload for the dispatcher. The most important connection conflicts according to the priority function remain for being displayed.

## **6. VISUALIZATION**

This chapter describes how connections can be presented to the dispatcher. Current systems generally use lists or tables to display connections and their conflicts [12]. The disadvantage of such systems is the unclear correlation between connections. It is not directly visible which connections have common feeders or distributors.

We in contrast developed in close reconciliation with DB Regio AG a matrix that presents connections depending on feeder and distributor. For each connection additional information can be displayed. In contrast to simple lists containing journey number of the feeder and the distributor our approach provides easier-to-catch overview and gives more information for each connection. Additionally, for each feeder and each distributor the current delay is displayed and also how many connections are available for a particular vehicle. An example is shown in Figure 2.

The advantage of a matrix overview is that dependencies between different vehicles can be visually better accessed. For each feeder all distributing trains are available in the same row. The same applies for distributors; all feeders are displayed in the same column. That allows both, feeder oriented as well as distributor oriented dispatching. The difference between the approaches is that in feeder oriented connection dispatching, the responsible dispatcher for the feeder train takes action and coordinates the dispatching process for all its distributors. In contrary in the distributor oriented dispatching, the responsible dispatcher of the distributor actively monitors feeders and takes action if a feeder is delayed without being triggered.

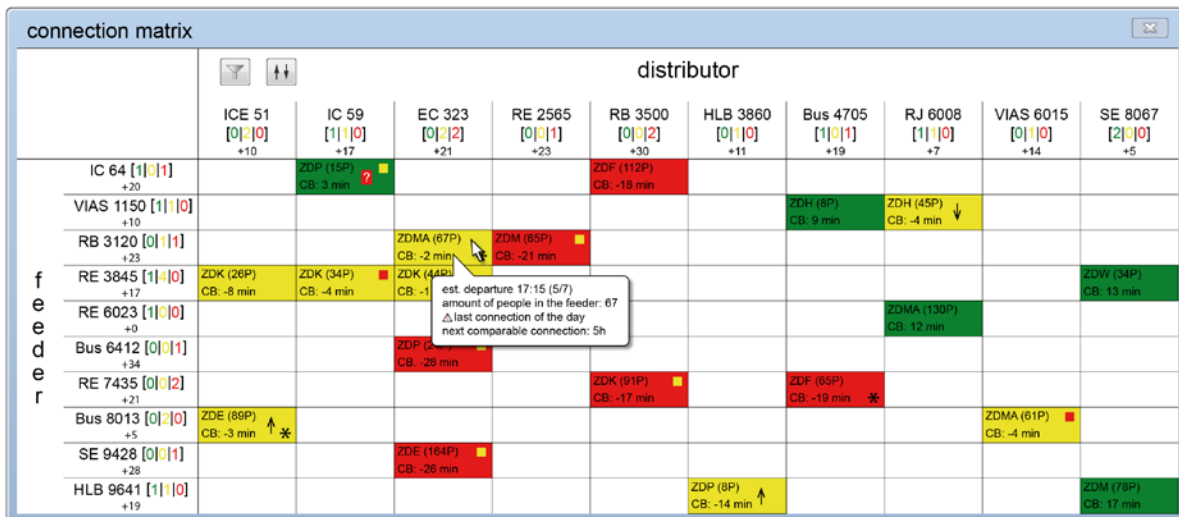


Figure 2 – connection matrix

A problem is a proper mapping of the priority of a connection. Because of the two dimensions of the matrix that depends on feeder and distributor rather than the connection itself, the priority of a connection can only be indirectly mapped. The connection with the highest priority will be placed in the upper left of the matrix. As soon as a feeder or a distributor for another connection is already set, this new connection has to be placed dependent on the already existing vehicle. Feeders and distributors that are not set yet will be attached to the matrix below or on the right side respectively. Also, sorting columns or rows can heavily interfere with prioritized visualisation of connections.

The effects of dispatching actions that result in delays for distributors are directly applied to the matrix. Implications on other connections are immediately visible. This permits also a “what-if” function in the matrix. One could assume a dispatching action and view the effects on other connections in the current situation (the operation time table). If the dispatcher agrees on the outlook he then can grant the dispatching action.

The main disadvantage of matrix visualization is the amount of space that is necessary. Cells will stay empty when there is no connection between a feeder/distributor pair.

### General visualization of connections

The connection is displayed within the matrix that way that a cell leading to a feeder and a distributor is filled with connection content as soon as a connection between these vehicles exists (Figure 3). The content consists of a background colour that indicates the connection status which is represented by the category a connection is assigned to (see section Evaluation of Connection Conflicts). Also displayed is essential information like the stop place where the connection will take place, the amount of transit passengers and the connection buffer (see section Detection of connection conflicts, chapter 5). Other information can be displayed in form of icons if necessary. An example is displaying an uncertainty state if the feeder’s delay cannot be exactly determined (indicated by the question mark in Figure 3).



	ICE 51 [0 2 0] +10	IC 59 [1 1 0] +17
IC 64 [1 0 1] +20		ZDP (15P)  CB: 3 min 
RE 3845 [1 4 0] +17	ZDK (26P) CB: -8 min	ZDK (34P)  CB: -4 min

Figure 3 – visualization of a connection

## Visualization of conflicts

Within the visualization of connections displaying conflicts is the main interest. Therefore conflicts need to be highlighted in a special way. In our approach this is done by choosing a different background colour depending on a category of a connection (see section Evaluation of Connection Conflicts). Connections of category  $C_1$  have a green,  $C_2$  a yellow and  $C_3$  a red background colour. This enables the dispatcher to grasp the information whether a conflict exists at a glance. He then can focus on the conflict and derive further information from the displayed conflict. He will also get displayed the time remaining for a dispatching decision.

## Filter

To reduce the amount of displayed connections and connection conflicts, several filters are applied by standard. Only connections that fit within a certain timeframe are selected for visualization. The timeframe respects planned data as well as prognosis and real time data to ensure to display all relevant connections to the user. Also, just the connections that are within the responsibility of a dispatcher are chosen. The remaining set of connections consists of disjoint sets with different categories (see chapter 5). A filter to display only the connections with a conflict ( $C_2$  and  $C_3$ ) is applied by standard.

More filters can be easily introduced. For example the user can switch to see only connections for which it is known that transition passengers exist. A very useful filter provides the possibility to hide all connections for which an automatic decision for securing the connection can be taken by the automatic dispatching component. The dispatcher can focus on conflicts which are more complicated to solve.

All filters are configurable and are adaptable to the users' needs. Especially the categorization filter can be easily overridden by the user. That enables him to also get details about connections that would be generally hidden. Still the evaluation has shown the importance of avoiding too many filter settings. The user can be overwhelmed and misconfigure the application.

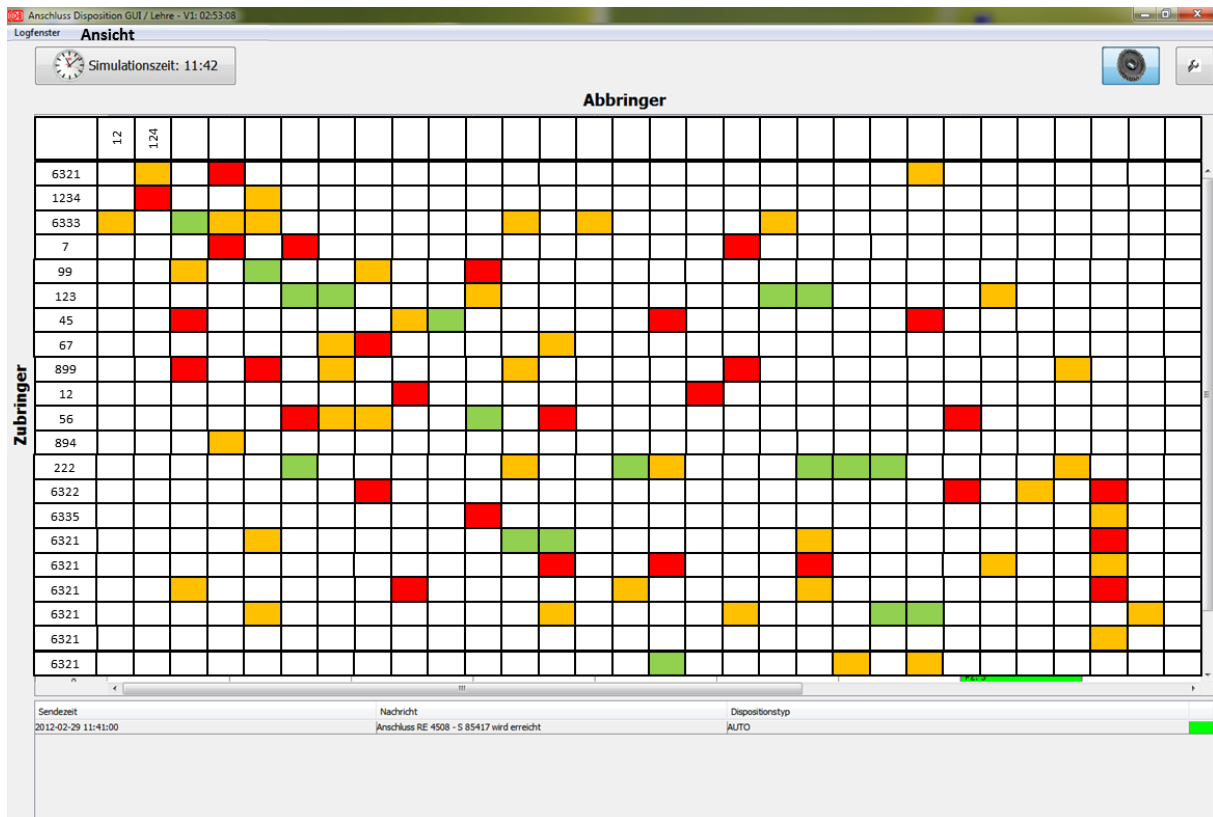


Figure 4 – General overview of the connection situation

Apart from the detailed view (Figure 2), a general overview is available that shows the dispatcher the connection in his area of responsibility. This gives a feeling for the general situation (Figure 4). This view is a special form of a filter as the displayed connections are differently filtered by extending the timeframe of displayed connections and removing the category filters. The result is then displayed in a special view which displays less information but more connections in the available space.

Another special view considers already dispatched connections. Dispatched connections do not need to be displayed in the conflict window as a solution already has been found. Still a dispatcher might need access to a dispatched connection. Thus, a view is implemented, that shows all dispatched connections (see also section Conflict ).

## 7. CONFLICT RESOLUTION

After the conflict detection the next step is a conflict resolution. Currently at German railways this is a completely manual process. An automation in the contemplated systems does not go beyond a suggestion of possible alternative trains in case of removing a connection [12].

In this section we will describe show that our solution is able to do practically usable connection oriented automatic dispatching.

## Automatic Conflict Resolution

Apart from detecting conflicts and displaying them to the dispatcher, our system is able to do a conflict solution within a defined set of rules. The assignment of categories to each connection is also used for the automatic conflict solving process.

Depending on the category, different rules apply for a connection. Apart from the category an important factor is the knowledge about transit passengers.

For the German railways the knowledge about transit passenger is the key factor that makes a dispatching decision obligatory. The information about transit passengers is derived from train attendants and electronically transmitted. A decision, positive or negative, for connections with knowledge about existing transit passengers is needed in any circumstances. Based on the decision, information to the travellers will be (automatically) provided.

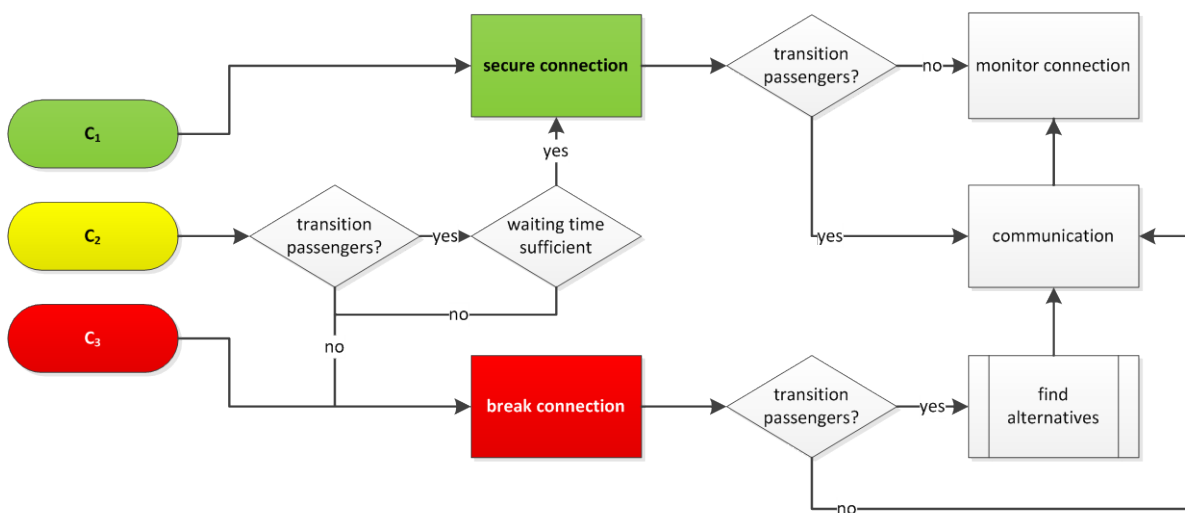


Figure 5 – automatic conflict solving process

The dispatching module uses the standard waiting times to decide whether a distributor will wait or not. For this we define two sets, one for secured connections  $S$  and one for broken connections  $B$ . Connections of the third category will never be secured because the resulting delay for the distributor is considered to be too high.

$$9. B \ni \{con \mid con \in C_3\}$$

Connections of the second category will be secured if there are transit passengers and if the waiting time rule (see chapter 3) applies.

$$10. S \ni \{con \mid con \in C_2 \wedge con \in W^+ \wedge -t_p < w \wedge P \in con \neq \{\}\}$$

Those connections in the second category for which no transit passenger are reported or it is unknown if there are any transit passengers will not be automatically secured.

$$11. B \ni \{con \mid con \in C_2 \wedge P \in con = \{\}\}$$

Connections without waiting time rules or with a delay for the distributor above the waiting time rule will not be automatically secured.

$$12. B \ni \{con \mid con \notin W^+ \vee (con \in W^+ \wedge -t_p > w)\}$$

Connections of the first category are only processed if they have transit passengers. The dispatching action taken will also be automatically communicated to train drivers and to the travellers including those that are not explicitly known as transit passengers. If a connection of the first category is reported as secured, travellers expect to be able to change from feeder to distributor. It should be only communicated that this connection is secured in case travellers really want to use it, i.e. if there is knowledge about transit passengers. To avoid expectations about securing these connections in case of short-term appearing delays they will not be marked as secured as long as there is no information about transit passengers.

$$13. S \ni \{con \mid con \in C_1 \wedge P \in con \neq \{\}\}$$

This leads to:

$$14. B := \{con \mid con \in C_3 \vee P \in con = \{\} \vee con \notin W^+ \vee (con \in W^+ \wedge -t_p > w)\}$$

$$15. S := \{con \mid P \in con \neq \{\} \wedge (con \in C_1 \vee (con \in C_2 \wedge con \in W^+ \wedge -t_p < w))\}$$

Note that  $B \cap S = \{\}$  but  $B \cup S \subseteq C$ .

The automatic conflict solution is a last minute action which is applied only if the dispatcher does not take any dispatching action before. In the current state of the conflict solving module we follow an approach with the least possible effects on the network. Using standard waiting times just in case we have information about existing transit passengers results in the least possible impact on the network while still following the guidelines of the German railways. We avoid any impact on the network as we are operating on the connection and not persuading a global optimum. Furthermore connection dispatching is in the jurisdiction of the transportation company whilst any decision with a network impact needs to be approved by the infrastructure company.

## Manual Conflict Resolution

The presented automatic dispatching process covers only standard dispatching actions. Thus, the usage of the automated solution is reserved for those conflicts which can be resolved using the described standard processes. For all other situations a manual dispatching process is still intended. Also, the automatic process can always be overridden by a manual decision. This may be necessary if the dispatcher possesses more information about the conflict situation. He knows the train crew schedule as well as the rolling stock constraints which are not taken into consideration for conflict resolution in the current prototype system. These constraints generally lead to more broken connections.

This divides the manual dispatching possibilities in two groups, those that correspond to the standard process, which can be automatically solved (but overridden if necessary), and those that involve more complex actions. The latter mentioned are not covered in our prototype and processed in existing tools. We will not further examine these functions in this paper. For the standard actions however we implemented buttons that are hidden in context menus which are easy to access with a right mouse click directly from the matrix (Figure 6).

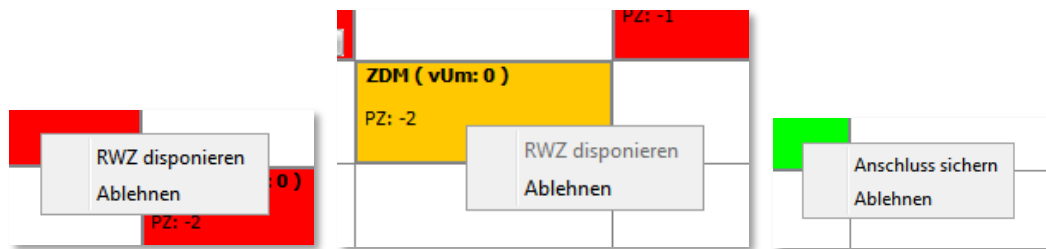


Figure 6 – Manual dispatching options

Only those options that are executable are available. In Figure 6 three different characteristics of dispatching dialog are shown. Depending on the connection category the user will see a button “secure connection” (in German “Anschluss sichern”) or “dispatch waiting time” (in German “RWZ disponieren”).

The option “secure connection” only appears for unconflicted connections and points out the dispatcher’s will to keep this connection unconflicted (right picture in Figure 6). Also, corresponding messages will be sent to the participants after choosing this option.

As soon as a connection is conflicted the option changes to “dispatch waiting time” as assigning the correct waiting time to the distributor will secure the connection (left picture in Figure 6). This button is only enabled if the standard waiting time is sufficient for the current conflict. In the middle of Figure 6 there is an example for the dialog if the standard waiting time is not sufficient. In this case the button is disabled and the dispatcher has to use other mechanisms if he yet decides to secure the connection. As in the automatic dispatching process, the assigned waiting time is added as a prognosis for the departure of the distributor.

In both conflict resolution modes, manual and automatic, securing a connection by delaying a feeder leads to an assignment of a prognosis for a delayed departure to the distributor.

## Alternatives

Connections that will not be secured need an assignment of possible alternatives for travellers that wanted to use these connections. Here we investigated several actions that can be automated. We identified ordering taxis and busses, booking a hotel for an overnight stay in case of missing the last connection, providing alternative trips or placing additional stops as dispatching actions that can be implemented in existing systems. Of course, the maintenance of corresponding data like prices and availability is mandatory. Also, for a



meaningful dispatching action more information about the traveller is needed as at least his destination. The alternatives can be part of an automatic dispatching process (see Figure 5).

We investigated how a user interface for dispatching these standard actions can look like. The main idea is to provide those specific functions bundled in a connection context to the connection dispatcher to enable him to work strictly in his connection context. Figure 7 shows the overview of available actions depending on the destination. Integration into our prototype application has not been carried out.

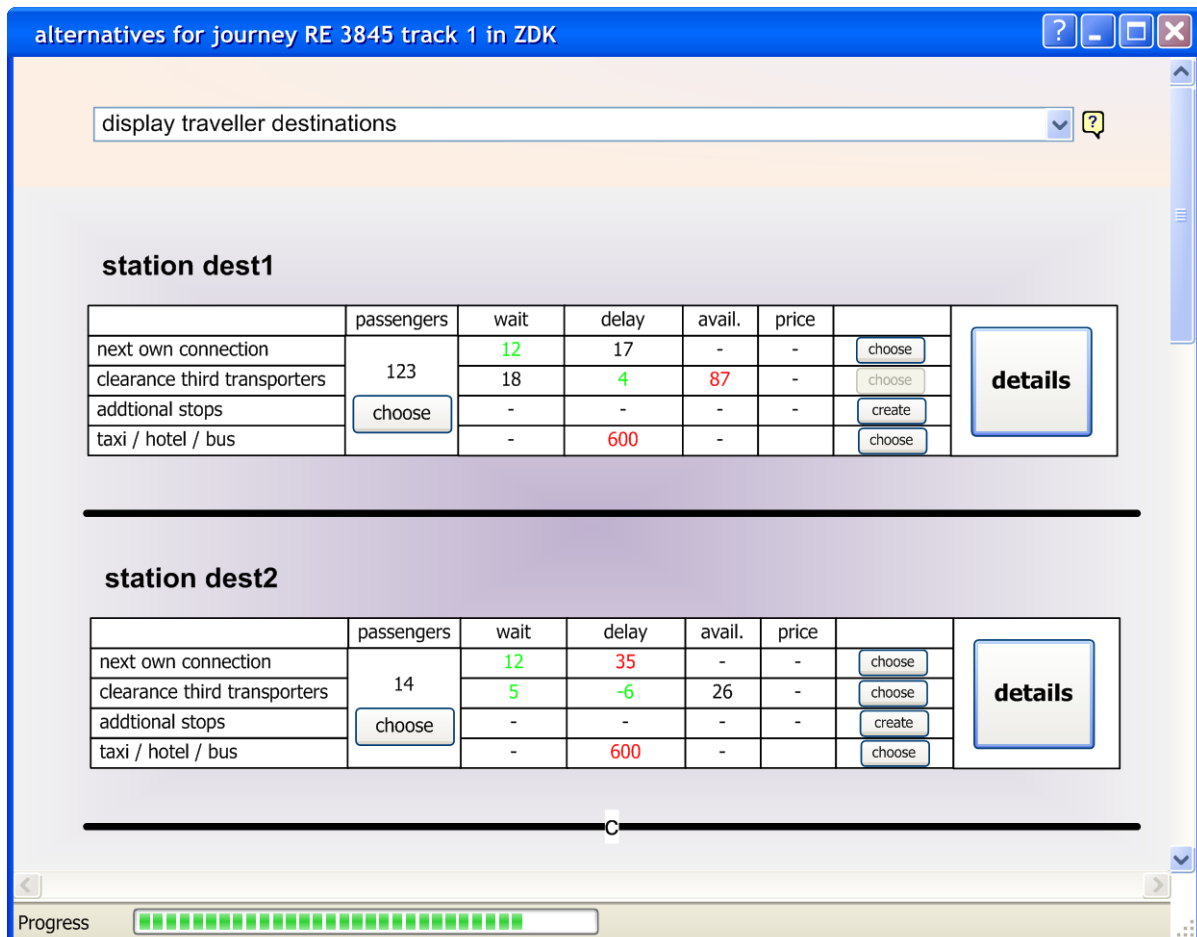


Figure 7 – dispatching dialogue for alternatives

## 8. EVALUATION

To get information about acceptance and usability of our work, we decided to implement a prototype system that permits to use and test the developed visualization as well as the automatic and manual dispatching methods. To have realistic backend data, we used an educational database including timetable schedules and connection parameters.

Apart from the implementation a realistic surrounding was needed to represent realistic train operations. The Railway Operations Research Centre Darmstadt satisfies these demands.

The next section describes the implementation in the research centre. In the following section we show how we integrated dispatchers in the development process.

## **Railway Operations Research Centre**

The Railway Operations Research Centre Eisenbahnbetriebsfeld Darmstadt (EBD) provides a range of services regarding research purposes in the railway domain. For our demands, the evaluation of a connection dispatching prototype, the offer of realistic interlocking technologies and thus the possibility of realistic railway operations regarding the infrastructural part was relevant. Also, software like staff and resource dispatching tools that are used by the transportation companies are available in the EBD. This forms a realistic surrounding. [15]

The prototype was adapted to work in our Railway Operations Centre. This includes the creation and adaption of datasets to run distinct scenarios. Data created by our application is reflected to the EBD and is displayed in different tools. For example the application of a waiting time to a distributor is available in transportation company's dispatching software as well as passenger information systems are triggered. The advantage in the EBD is that any situations could be provoked, including situations which lead to unwanted delays or deviations in reality, without any critical impact.

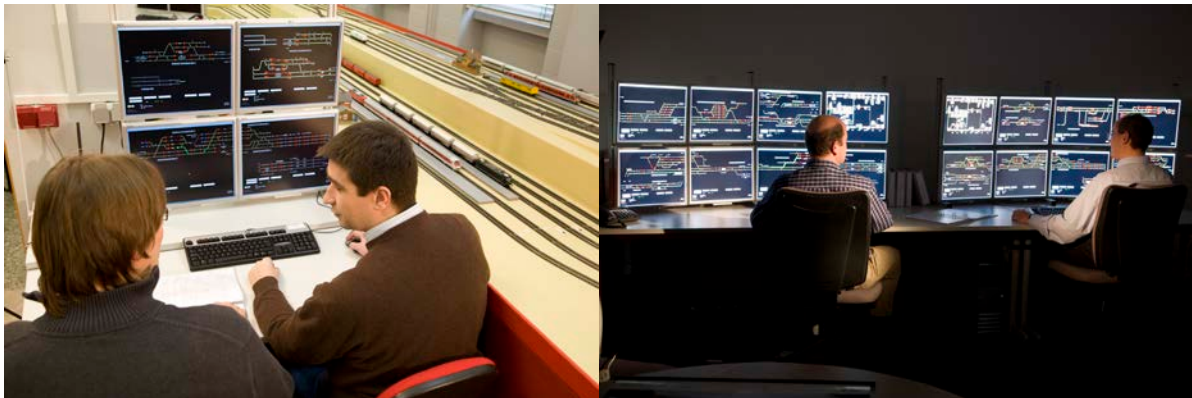


Figure 8 - railway operations centre Darmstadt [4]

Our solution for displaying connection conflicts and automating the resolution of conflicts can be thoroughly tested in the EBD. The functionality of the automation as well as manual dispatching can be evaluated in realistic railway operation. Realistic dispatching software is also available and can be used aside. This permits close to reality conclusions regarding the functionality of the application.

Based on our data we could show that the process of manual as well as automatic dispatching works properly. Also the visualization appeared to be well usable and displayed correct connection status. The application dispatches any connection if the dispatcher does not undertake a manual dispatching action and reduces the dispatcher's workload significantly.

## **Real Data Evaluation**

For the evaluation in the EBD we have constructed data sets that fit to the infrastructure that is available there. To be able to give statements for usage in a real environment we also evaluated our implementation on real data. For this we created a realistic data base with the data derived from a normal operating day. We imported all time table data including prognosis and real time data and all connections for one day throughout Germany. We also integrated knowledge about transfer passengers that is derived from reports of train attendants. The result was that also here our application behaved well. The connections were displayed correctly and the dispatching operations could be applied as expected. To simulate a real data flow a simulation tool was executed that applied the corresponding real data to the timestamps they occurred. This enabled us to display the connection matrix as if used in a real context.

Of course, in this scenario we could not manipulate data close to real operations like in the EBD and therefore could not provoke special scenarios. That is why the combination of both variants, real operations with a possibility of interfering into the planned operations in the EBD and in parallel the usage of real data gives us the necessary reliability on our results.

Again, it could be shown that using our application, the workload for the dispatcher could be significantly reduced as the application is automatically dispatching all connection conflicts the dispatcher cannot handle. Also, the evaluation has shown that using the matrix display is feasible to display connection conflicts in an easy-to-catch way. It permits to have a good overview on the current situation as well as actual conflicts that need to be considered.

## **Dispatchers' Evaluation**

Additionally to our tests we invited real dispatchers to test the prototype system. For this we invited them to the EBD and encouraged them to work with the known transportation company's dispatching tools aside our prototype application. We then simulated typical operational situations and applied several disturbances to the system. The dispatchers could test the behaviour of their known systems and additionally the behaviour of the connection matrix. The surrounding of the EBD represents real railway operations properly and is therefore suitable for such an evaluation [15].

The feedback was generally positive and the solution was believed to perform in a real environment. The possibility to have an overview about several connections was appreciated. Still, some aspects regarding usability and visualization could be improved. Especially sorting and filtering in a two dimensional user interface was reported to be difficult to handle. Based on this feedback we are now developing methods to overcome this drawback. Also we learned that an evaluation of filters that are to be integrated is necessary and to avoid filter settings which hinder or confuse the user.

The feedback gathered in the EBD is of high value for us due to the proximity to real operations. With this feedback we are able to improve our solution facing the aim to create a tool that is usable in real operations.

## **9. SUMMARY AND FURTHER RESEARCH**

In our research we have shown an alternative approach of displaying connection conflicts. We developed methods to evaluate the connections which a proper displaying could be based on. We further presented an approach for an automated conflict solving solution which does not override the dispatcher's decisions and works in the background in a supportive way. We could point out that this approach is usable in a real environment as designed without any further adaptations. The approach can be used in an intermodal context as it is indeed derived from but not solely fit to railway operations. Furthermore the approach is generally designed and therefore usable in any public transportation context.

This solution therefore can support connections dispatchers in their routine tasks and reduce significantly the amount of connection conflicts that have to be manually solved. The user can get information about conflicted connections within a glance.

We evaluated the solution in a close to reality environment on one hand and with real data on the other hand. The practical functioning of the approach could be proven.

The next step is a field test in a real environment. For this, two operation centres of German Railways have been chosen to test the visualization. In this phase, no dispatching actions will be stored into productive databases, as it is solely tested how the visualization supports the connections dispatcher. As a part of the field test also the existing functions will be evaluated and improved and new functions derived from the feedback of the users will be integrated subsequently during the field test.

Several additional functions mentioned in chapter 7 – alternatives have to be further analysed for implementation and automation. Furthermore a better priority function needs to be developed. Also, possibilities of displaying (hypothetic) dispatching actions in the user interface need to be evaluated.

## **10. LITERATURE**

- [1] Bär, M., S. Kurby, H. Meier and U. Steinborn (2006). Modell zur Anschlussdisposition (ANDI/L) im Rahmen von DisKon, *EI - Eisenbahningenieur*, 57, 96–104.
- [2] Berger, A., C. Blaar, A. Gebhardt, M. Müller-Hannemann and M. Schnee (2011). Passenger Flow-Oriented Train Disposition.
- [3] DB AG (2010). Konzernrichtlinie Bahnbetrieb.
- [4] EBD Fotoarchiv. professionelle Fotoaufnahmen EBD.
- [5] Fay, A. (1999). Wissensbasierte Entscheidungsunterstützung für die Disposition im Schienenverkehr: Eine Anwendung von Fuzzy-Petrinetzen, VDI Verlag, Dusseldorf.
- [6] Fay, A. (2000). Wissensbasierte Dispositionsunterstützung, *ETR*, 11, 740–744.

- [7] Goverde, R. (1998). Optimal scheduling of connections in railway systems, 8th WCTR, Antwerp.
- [8] Kanai, S., K. Shiina, S. Harada and N. Tomii (2011). An optimal delay management algorithm from passengers' viewpoints considering the whole railway network, *Journal of Rail Transport Planning & Management*, 1/2011, 25–37.
- [9] Klemenz, M.-A. and T. Siefer. Method for an optimized passenger orientated connection management during the planning and the operation process., <http://www.railway-research.org/IMG/pdf/o.3.4.4.4.pdf>, accessed 13 May 2013.
- [10] Kurby, S. (2012). Makroskopisches Echtzeitdispositionsmodell zur Lösung von Anschlusskonflikten im Eisenbahnbetrieb. Zugl. Dissertation an der TU Dresden 2012.
- [11] Ministerium für Infrastruktur und Raumordnung Brandenburg (2007). ÖPNV-Finanzierung im Land Brandenburg: Leitfaden für die kommunalen ÖPNV-Aufgabenträger, Potsdam.
- [12] Schnick, T. Informationssystem Transportleitung Personenverkehr: Benutzerhandbuch (BHB-ISTP), DB Fernverkehr AG, Frankfurt/Main.
- [13] Schöbel, A. (2011). Warten oder nicht warten? – Kunden-orientiertes Verspätungsmanagement und dessen Komplexität.
- [14] Stelzer, A., F. Englert, A. Oetting and R. Steinmetz. Information Exchange for Connection Dispatching: accepted at the 21st International Symposium EURO - ŽEL 2013, Žilina.
- [15] Streitig, C., A. Stelzer, S. Schön and F. Chu (2012). TU Darmstadt – Research, Training & More Besides, *EURAILmag*, Issue 26, 152–159.