Deliverable D 2.2

Smart planning – summary of methods dealing with incomplete data

Project acronym:	PLASA 2
Starting date:	01/09/2018
Duration (in months):	26
Call (part) identifier:	H2020-S2R-CFM/OC-IP/CCA-201X-0X
Grant agreement no:	826151
Due date of deliverable:	M12
Actual submission date:	29/08/2019
Responsible/Author:	Ingrid Johansson
Dissemination level:	PU
Status:	Issued

Reviewed: Yes

Document history		
Revision	*Date*	*Description*
0.1	12/06/2019	Draft 1
0.2	12/08/2019	Draft 2
1.0	29/08/2019	First Issue

Report contributors		
Name	Beneficiary Short Name	Details of contribution
Markus Zinser Ying Löschel	DB	Provided input about PRISM; reviewed document
Ingrid Johansson, KTH Jennifer Warg, KTH Oskar Fröidh, KTH Emma Solinen, TRV Carl-William Palmqvist, LU Magnus Wahlborg, TRV Pär Johansson, TRV	TRV	Wrote deliverable; reviewed document

Table of Contents

1. Executive summary

The aim of this document is to provide an assessment of possibilities for simulation approaches, both microscopic and macroscopic, that do not require detailed data on all aspects of the simulated scenario, for example for incompletely specified railway freight traffic.

In a macroscopic simulation, only a choice of aspects is included, e.g. stations and stops, but no detailed track plan between these network nodes. In that case, a stochastic approach is required because failures of different infrastructure elements of the same type on a specific macroscopic network-edge can have different consequences for train travelling times. This approach ignores the causal connection between disturbance, re-routing, possible waiting times and the resulting delay, and simply assigns a probability to different delay times given a type of disturbance.

The simulation tool PRISM uses a macroscopic model. It makes it less important to have a high level of detail in the data. Missing such data is thus less problematic than for microscopic simulation tools such as RailSys. In either case, it is prudent to start by establishing a well-calibrated base scenario, before scenarios with missing observations are run.

However, missing data makes the comparison to real data more difficult. Especially if many assumptions are used or the input is generalised to a large amount, the results will not be comparable for enabling an accurate analysis. In that case, other ways of validation have to be chosen.

The conclusions reached in this report are that there are certain situations where the available information is not sufficient for performing a simulation, but that there are ways to overcome these shortcomings.

2. Abbreviations and acronyms

Abbreviation / Acronym	Description
KTH	KTH Royal Institute of Technology, Stockholm
LU	Lund University, Lund
DB	Deutsche Bahn
TRV	Swedish Transport Administration (Trafikverket)
ETCS	European Train Control System
PRISM	Plasa Railway Interaction Simulation Model

Shift2Rail | Horizon 2020 European Union Funding for Research & Innovation | PLASA 2

3. Background

The present document constitutes the Deliverable D2.2 "Smart planning – summary of methods dealing with incomplete data" in the framework of the WA 4.1 of CCA.

In PLASA simulations have been done in parallel with RailSys micro simulation and PLASA macro simulation. The results have been compared in simplified simulations. In PLASA 2, simulation studies are continued, the model is further developed, and a systematic comparison will be performed. Further use of the PLASA model will be analysed. A concept will be developed for combining the PLASA model with RailSys.

PLASA has analysed the need for and use of simulation models in railway planning. The state of the art was described in deliverable 2.1 (Betz, et al., 2017). Then, the development of a simulation model meeting the outpointed needs started. It was later called PRISM (Plasa Railway Interaction Simulation Model) and is now the core of PLASA and PLASA 2. This chapter gives a short overview over that model and describes the differences between micro- and macroscopic models.

Simulating the impact of disturbances and infrastructure non-availability on train travelling times is one approach to evaluate timetable robustness. Such a simulation can be performed at different levels of detail. In a microscopic simulation, exact train paths through the network are simulated and infrastructure can, and in most models has to, be modelled at the level of individual switches or signals. Those models are commonly used, and often recommended, for example by (Borndörfer, et al., 2018) who state that, for example, dynamic, synchronous, microscopic, stochastic simulation represents a system in the best way. Examples are RailSys, see (Bendfeldt, Mohr, & Müller, 2000) and (Radtke & Hauptmann, 2004), as well as LUKS (Janecek & Weymann, 2010) and OpenTrack (Nash & Hürlimann, 2004). In PLASA and PLASA 2, RailSys is used as a reference model under the development of PRISM.

In a macroscopic simulation, only a choice of aspects is included, e.g. stations and stops, but no detailed track plan between these network nodes. In that case, a stochastic approach is required because failures of different infrastructure elements of the same type on a specific macroscopic network-edge can have different consequences for train travelling times.

The required accuracy of a model highly depends on the task. In many cases, macroscopic simulation is sufficient, while many others require proper modelling of the system. In a model presented in Cui et al. (2018), the level of details can be adjusted according to the user´s and project´s needs, with the possibility of micro-, meso- and macroscopic simulation. In PRISM, that is to some extend also the case. The minimum level of detail must be reached, but beyond that, further details can be added if they are available and make a useful contribution. If the input is from RailSys it may be quite detailed, otherwise, for example, average speeds can be sufficient for the link between two station nodes.

However, it can be the case that not all data is available. How this affects the models and can be handled is the content of this deliverable.

4. Objective/Aim

The aim of this document is to provide an assessment of possibilities for simulation approaches, both microscopic and macroscopic, that do not require detailed data on all aspects of the simulated scenario, for example for incompletely specified railway freight traffic. The simulation process is described and cases for incomplete data shown. An evaluation of how simulation can be used as a method in case of incomplete input data for both macroscopic and microscopic simulation is also included.

The document is connected to task 2.1 – *Dealing with incomplete data – assessment*. With the background of information about required input from simulation models, the following chapters describe the term *incomplete data*, how existing methods deal with that and how the shortcomings can be overcome. In task 3.1 – *Smart planning: Approaches for simulation with incomplete data* developing work for overcoming the difficulties with incomplete data is ongoing and will be finalised in deliverable 3.3 – "Approaches for simulation with incomplete data" in august 2020. This document is intended to provide a background to that work.

5. Main areas for simulation and necessary data

This chapter describes railway simulation in general and, more specifically, how timetables are handled. It goes further into required input data on the example of the microscopic tool RailSys and the macroscopic simulation tool PRISM.

5.1. Simulation

Simulation aims to create a representative model of the reality, which can be used to perform analyses in a realistic environment. The ongoing improvements of computer performance allow, for example, to widen the analysed networks, speed up simulations or increase the level of details.

Simulation is usually performed in two ways: deterministic or stochastic.

Deterministic simulation is performed without random factors. Examples include the calculation of running times for a certain vehicle on a certain infrastructure, or timetable analysis in order to detect conflicts in the planned timetable. In stochastic simulation, random components are added and a representative number of simulation runs are performed to evaluate the performance of a system, mostly concerning timetable robustness or stability. In PLASA, focus is on stochastic simulation with randomly added disturbances.

Usually, a simulation model requires an infrastructure model, train models and a timetable. The required input data for RailSys and PRISM will be described in more detail in the sections 5.3.1 and 5.3.2, respectively.

5.2. Timetable creation

A railway simulation is dependent on a timetable. The following conditions can be considered:

- Conflict-free timetable: If simulated without disturbances or other randomness the result is deterministic
- Timetable with conflicts: Has to be conflict-solved (before or in the simulation)
- Stochastically added disturbances: Might lead to conflicts between the trains and conflict solving is necessary. Result is stochastic

While the first case has a clear solution, the second and third need some kind of conflict-solution to create a timetable that can be operated. This can be solved in advance, but is usually handled by dispatching during the simulation according to certain rules for prioritisation. That means that simulation can be used for both the creation of a timetable, and the evaluation:

- Timetable creation through dispatching – simulation without added disturbances is used to solve conflicts
- Evaluation of a certain (given) timetable concerning robustness and stability. For example, for comparison of different timetables that are simulated with the same added

disturbance distributions, for analysing how a certain timetable reacts, or for evaluating the operation with different levels of added disturbances

Figure 1 summarises the simulation process. A timetable has to be created from departure times and run times for each train, preferably calculated in a microscopic model. As a timetable created from pure departure and running times is likely to contain conflicts, a conflict solution might be desirable before the timetable is simulated with disturbances in order to evaluate its performance. The simulation is controlled by simulation parameters whose values are calibrated in an iterative process until the simulation output is validated to be reliable results. Usually, alternatives are compared, for example to evaluate different kinds of vehicle types, stopping patterns, etc. In the case considered here, the missing input data might allow for different timetable alternatives (based on the way of conflict solving) which may lead to different results. It might be reasonable to compare different solutions or use an average value. This is not illustrated in the figure below.

In the following sections, focus is on the input data, which of course is of large importance.

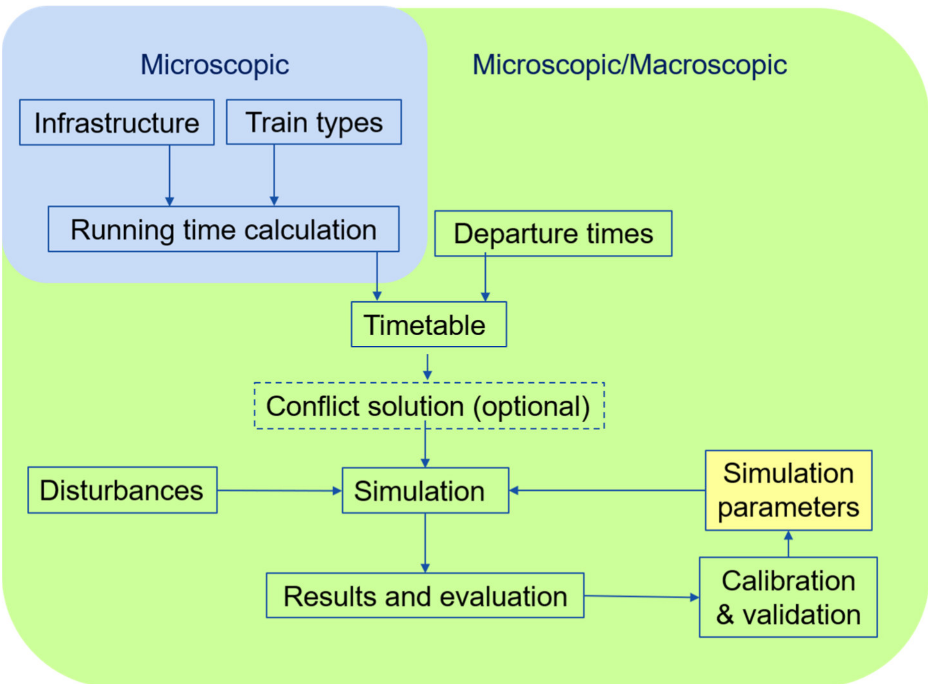

Figure 1: Illustration of the simulation process for microscopic and macroscopic simulation.

5.3. Input data

As previously discussed, the required input differs among the available simulation models, and microscopic models need more detailed input than macroscopic models, but microscopic models should in return give more precise results. Macroscopic models require less detailed input data, which is an advantage if there is a lack of data which would otherwise be necessary for a microscopic model. As a background for the treatment of incomplete data, the following sections describe the input data used in the two models that PLASA 2 focus on: the reference model RailSys and PRISM.

5.3.1. RailSys input data

5.3.1.1 Infrastructure

The infrastructure model used in RailSys is microscopic and hence detailed. More or less all aspects of the physical and functional design that can influence the traffic is possible to model. It contains information about distances, gradients, track layout, signal objects, electrification and signalled speed and much more. There are also possibilities to set rules and functionality of the interlocking systems (even though it's not possible to exactly replicate all aspects of the interlocking logic). The basic idea is to give the preconditions and let the program calculate the behaviour. Figure 2 shows an example for a station layout in RailSys.

On top of the objects and attributes that influence the functionality it is also possible to add a lot of information, like data regarding life cycle of the components, organizational division or aspects of the visual presentation.

Even though it is possible to make very detailed models, you don't need a perfect model in order to run the program. Depending on the objective of the study the level of details can be chosen accordingly. Most of the time you can, for example, use standard values for the release times of train routes rather than to individualize the interlocking systems. None the less, a relatively complex model is required in order to get meaningful results when creating a correct timetable and then running a simulation which takes complex signalling systems like ETCS L2 into account.

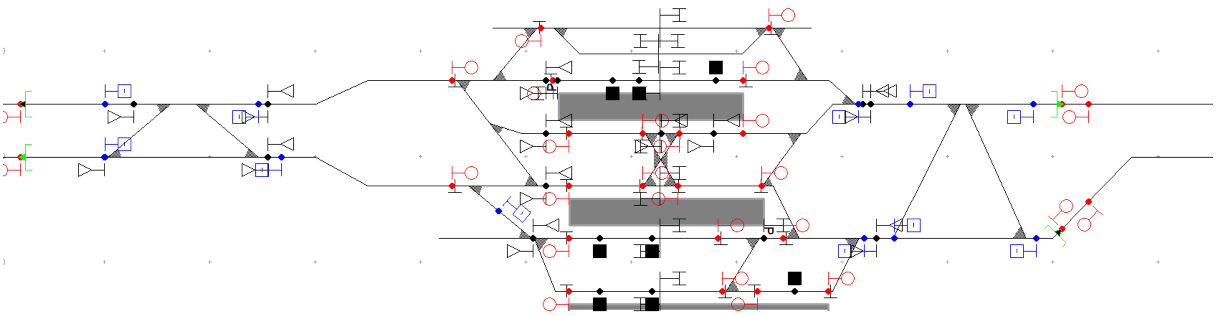

Figure 2: Västerås station (Sweden) modelled in RailSys. Visible is the track layout and switches, platforms (grey rectangles) main signals (red), shunting signals (blue), speed boards (black, wine glass shape), release contacts (black "T":s) balises (black squares), stop boards, and station borders (green).

5.3.1.2 Rolling stock

The rolling stock is also modelled in a relatively detailed way. At core is the traction force diagram and rolling resistance that together with weight, length, permitted speed and braking capabilities can give a good representation of how a train moves and interacts with the infrastructure, and it can also be further detailed with several additional attributes.

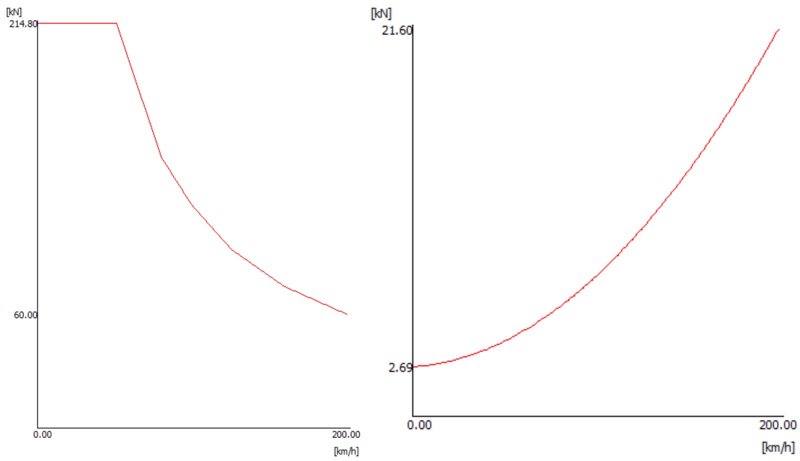

Figure 3: The left graph shows the traction force of the Swedish passenger electric multiple unit (EMU) X55, the right one the corresponding running resistance.

5.3.1.3 Timetable

When creating a timetable, RailSys calculates running times and headway times based on the previously mentioned details, including signalling systems. Settings can be done to include runtime margins of different kinds and additional buffer. Stops can be modelled conditionally based on the circumstances (like to omit a stop if the train is late). Specific stopping locations are used (RailSys will choose the main track if nothing is given) in the timetable and lists specifying alternative stopping locations in case of unavailability can be created for different train types. Associations of different kind between trains can be created, either of waiting time type where one train waits for passenger transfer from another, or associations that for example indicate that the train individual from an arriving train is turning around and becoming the leaving train.

Timetables can be imported/exported in a RailSys specific xml-based file format and RailML (at least to some degree). For the use in Sweden there is a converter module in order to import timetables from Trainplan. If the timetables are built in RailSys it is possible to bulk create reoccurring patterns.

5.3.1.4 Simulation

Simulations can be executed with stochastic disturbances of the timetable. Drawn from a defined distribution, delay time is added to the dwell time, running time, and departure time or as a train enters the simulations area.

There is an automatic dispatcher functionality in RailSys that makes choices in disrupted situations in the simulation based on a number of alternative and attributes that can be set. All trains can be given priority values and defined thresholds on how the priority is changed in case of lateness. A number of dispatching strategies can be activated and given ranges on when to be operated, the dispatcher can for example be given the authority to re-platform trains if the delay is more than 5 minutes. Whether the dispatcher will use this possibility depends on the availability of defined alternative stopping locations and the estimated gain in relation to the priority of the involved trains.

5.3.2. PRISM input data

5.3.1.5 Infrastructure

Unlike RailSys, for PRISM the representation of infrastructure is less important, and a much lower level of details is needed for being able to simulate. Basically, nodes representing stations or other important operational places, and links between these, are modelled and equipped with information. For the nodes the information is about additional tracks for overtaking and for the links about the train protection system (e.g., which system, block lengths, etc.), electrification, number of tracks and the maximum speed allowed. In previous research, it was shown that for passing stations, conflicts usually appear on the line close to stations but not in the stations themselves. For such stations, unlimited station capacity can be assumed in some cases (Zinser, et al., 2019).

5.3.1.6 Rolling stock

The rolling stock is categorized into train models, that means divided into groups with similar properties as for example maximum speed, safety systems, length, weight and acceleration/deceleration. Exact acceleration diagrams, etc. as, for example, in RailSys are not used.

5.3.1.7 Timetable

While train and infrastructure information in RailSys is used for running time calculation for the timetable, PRISM uses the scheduled (predefined) times instead. In RailSys, starting times at the first station, dwell times and the calculated running times are used to create the timetable. In PRISM, a full timetable with planned arrival and departure times at each node is needed. This can for example be calculated in RailSys or any other timetabling tool and be imported. As described in the previous chapter, that timetable does not need to be conflict-free.

5.3.1.8 Simulation

Simulation can be done similar to RailSys, but there are extended possibilities. For example, reduced run time on a whole train run (due to a vehicle failure) is possible to introduce as a disturbance in PRISM, while it is not in RailSys, where disturbances have to be defined for places and cannot follow a train run. In a similar way as in RailSys, different ways of dispatching and prioritizing can be chosen.

To calculate running times, technical travel times have to be given to the simulation. These could potentially come from RailSys. The representation of the timetable and the train types is used to calculate running time supplements, which means to estimate the time margin that can be used in case of disturbances. Simulation is then done in a similar way as described for RailSys in 5.3.1.4: Disturbances are stochastically added for a certain number of simulations and the resulting operation is calculated based on priorities and the program's dispatching algorithm. As in RailSys, the causal connection between disturbances, re-routing, possible waiting times and the resulting delay are ignored. Instead, a probability is assigned to different delay times given a type of disturbance. However, these disturbances may of course lead to secondary delays which are handled with dispatching and might be more realistic in a more detailed model.

Examples for simulations including results can be found in PLASA Deliverable 3.3 (Betz, et al., 2018).

6. Dealing with incomplete data

In an ideal case, all data is available when performing simulations and evaluating the results, i.e. the infrastructure, train types and the timetable are specified in detail. When this is not the case, we talk about incomplete data. It is not unusual that data is incomplete for the case one wants to simulate. In the following section, use cases with incomplete data are described, including the desired analysis (e.g., to create a timetable and/or to evaluate a timetable or an infrastructure).

In order to deal with incomplete data, this report assesses possibilities for simulation approaches, both microscopic and macroscopic, that do not require detailed data on all aspects of the simulated scenario, for example for incompletely specified freight traffic and information about maintenance activities. Firstly, it analyses the difficulties with the missing data, and secondly, it describes methods for handling the scenarios with incomplete data. The latter can be done by using other data, e.g., data on past freight traffic's use of train paths, past maintenance activities, or heuristics, to fill the gaps of missing information.

6.1. Use cases for incomplete data

The data is considered incomplete when at least one category of needed information is missing. Depending on the kind of model, that can be minor information, like missing speed information for a station track, but it can also be the more important data as, for example, departure times. Referring to chapter 5.3 where the required input is described for the two different models, it is obvious that the lack of minor data might not affect the results in a macroscopic model at all (for example if no station tracks are modelled at all). In microscopic simulation on the other hand that missing data could make the simulation impossible to run, for example if no information is available for an important line section. However, the more details available, the more has to be coded and be handled in the simulation.

Data can be considered as incomplete in several situations, for example:

- In the long-time planning
- When reality differs in large from the plans
- When certain information is not available to the planners (e.g. due of security reasons)
- Reduced input due to coding effort

As described in chapter 5.3, infrastructure, train types and timetable are the important input for a simulation. In the following, use cases are presented.

6.1.1. Unknown infrastructure

The infrastructure can be unknown in the sense that for railway lines in the planning or construction phase, all details regarding the infrastructure might not be known. Further, data might not be public, or the effort too large to code all details. It is still possible to get an overview of how the punctuality will be given a certain traffic level by using a very simplified and basic model.

Examples of unknown infrastructure:

- Signalling system
- Block lengths
- Incline
- Maximum allowed speed
- Length of overtaking station and station tracks
- Number of station tracks

6.1.2. Unknown train types

The train type might also be unknown. That can in a similar way as the infrastructure be due to the fact that the use of rolling stock has not been decided yet for a future project, or that the operator has different trains to choose of. If the latter is the case, the number of alternatives is not infinitive, but the differences might be large.

Unknown train types can be characterised by every attribute used in the description of the train type, e.g.:

- Maximum speed
- Acceleration
- Deceleration
- Length
- Weight

6.1.3. Unknown timetable

Defining the location of every train at every point in time, i.e., the timetable is the central part of a simulation, see Figure 1. If the timetable is supposed to be simulated deterministically based on the departure times, train types and infrastructure, missing data makes it impossible to create the timetable. If also disturbances are included, the signalling systems and headways are important. Another case, which again is a good example for a use case, is when the departure times are unknown. This can be the case if a new line is built, like a high-speed line which is planned to be used by a certain mix of services. Further, stopping patterns can be unknown.

Use cases

- Incomplete infrastructure
- Incomplete train information
- unknown timetable for all trains (e.g., forecasts of the traffic level with known number of trains per hour, but not their exact departure times and running times)
- unknown timetable for some trains

6.2. How to handle incomplete data

6.2.1. Methods to overcome incomplete data

The following strategies can make it possible to simulate a timetable even if not all data is available:

- An easier model can be used in order to make the model work (e.g., code decreased level of details, categorise trains into groups with similar characteristics as in PRISM instead of exact models, etc.)
- Use assumptions (e.g., use a common train type or speed limit)
- Introduce incomplete data by disturbances (e.g., in case of unknown departure times: design the departures with help of an entry delay distribution for distributing the trains over the day)
- Introducing relevant factors by randomisation for replacing missing data

As the input data in general is much more detailed in microscopic models than in macroscopic, missing data can be handled more easily in macroscopic models as the microscopic model requires more adjustments.

6.2.2. Description of chosen use cases

In the following section, several use cases with incomplete data are described in more detail, including why the data is incomplete and what the desired analysis is. The aim with the use cases can, for example, be to create a timetable and/or to evaluate a timetable or an infrastructure.

6.2.1.1 Unknown timetable (all trains)

In a long-term horizon, the service amount might be known, but hardly the exact timetable. Assume that the train paths are known, but not the train order or departure times. In that case, several timetable suggestions can be created. With the help of constraints as, for instance, intervals between certain departures, the number of alternatives can be limited. The aim is then to evaluate the traffic level to see if it is reasonable or results in too many disruptions. Simulation can then be used both to create timetables and to evaluate them. It can be desirable to create and evaluate several timetables to find the best one. Further, the train types and with that the train paths can be unknown. In that case assumptions about the future trains has to be used.

6.2.1.2 Unknown timetable (selection of trains)

It can also happen that the timetable is known for most trains, but that some trains have incomplete data. The missing data is, typically, unknown departure time and/or train type, which for freight trains results in their running time also being unknown, while for passenger trains their (future) stopping pattern becomes unknown. The case of unknown departure times for freight trains is highly relevant for Swedish conditions, where freight trains in practice can be dispatched when they are ready to depart and there is a free time slot, before or after the time stated in the timetable.

6.2.3. Dealing with the chosen use cases

In addition, stochastic approaches that do not require precisely specified timetables for all trains and maintenance activities will be considered. If, for example, the exact train route is unknown, there will not be any difficulties in PRISM if train routes are not modelled (e.g. inside stations). In RailSys, a default route is chosen, or rerouting to an alternative track is performed. That means it is still possible to evaluate the punctuality, etc. (see chapter 7). However, this rerouting process could be optimised. Unknown speed profiles can be evaluated by aggregating broad categories of trains, such as freight trains, where there is a large heterogeneity, whereas more well-defined ones can be compared by using less aggregated sets of trains in the comparison cases. While this corresponds to the input used in PRISM, RailSys usually uses more detailed data. However, simulation using these train models is possible without problems.

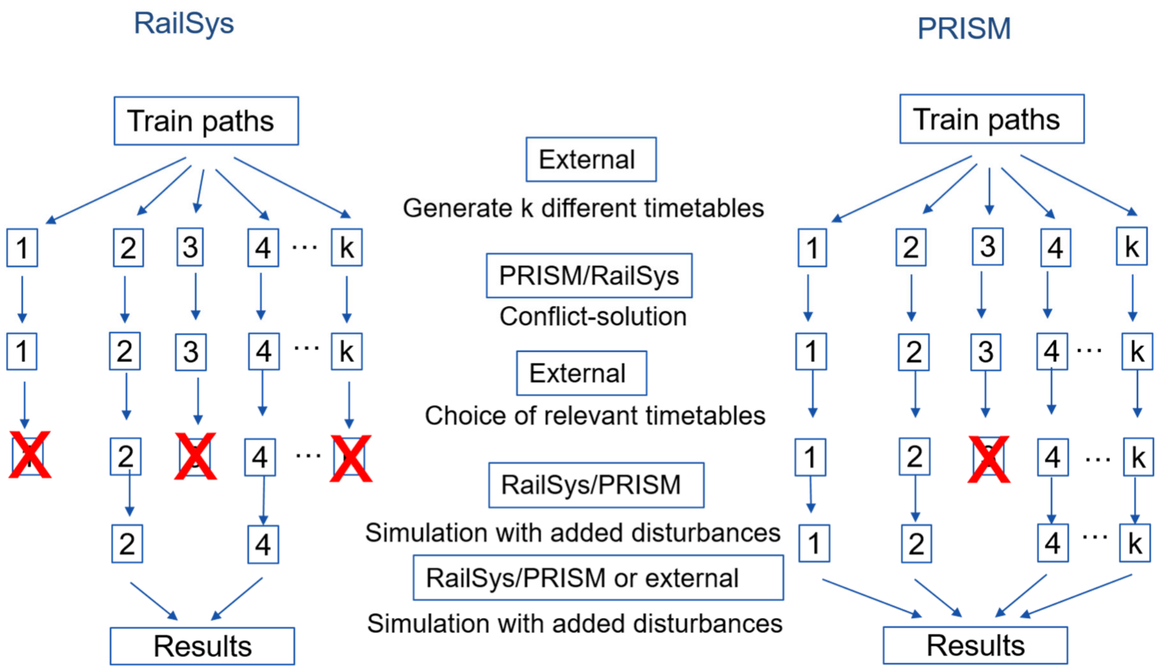

Figure 4: A possible method in case of incomplete timetable data

The analysis is limited to incomplete timetable data. Assuming that the running and dwell times of all trains are known (that means either all times for all trains between and on all nodes, or data about vehicles, infrastructure and stops), but no or not all departure times, it can be handled in the following way: With help of an external model, departure times for all trains with unknown departure time are randomised. To make this process more realistic, the randomisation can be based on distributions adjusted to real data. In the example with the freight trains, that means statistics about the departure times. Being based on a distribution, the randomisation should be done for a representative number of repetitions in order to make sure to reach relevant timetable. Certain restrictions can be introduced, for example to keep certain intervals between departures or to avoid certain combinations that are not likely to occur in properly planned timetables (for example that all trains of the same kind depart after each other). For creating a timetable based on the created departure times, conflict-solution as described in Figure 1 is recommended, for example by deterministic simulation in PRISM or RailSys. The created timetables can be analysed and one or several relevant ones be chosen to be simulated with disturbances. Figure 4 visualises that process. As simulation in RailSys is much more time-consuming, it is recommended to limit the cases to be simulated more than for simulation in PRISM. How the simulation results are treated is described in the next chapter.

7. Evaluation

After simulation, the results have to be evaluated. Usually, statistics summarising and averaging the results for all runs provide a good representation, but it might also be worth to look further into certain runs. If several different timetables are simulated (due to missing data as described in the previous chapter), a relevant method to evaluate the chosen traffic amount on that infrastructure has to be found. Usually, averaging can fit well here, too.

The simulation runs in PRISM and RailSys are evaluated against both empirical macro-level data and against microscopic simulation in RailSys. The purpose of the evaluation is to validate and calibrate the model, so that the results can be used and trusted in practice. If the results deviate far from the actual traffic or from microscopic simulation runs, the usability of the model decreases. On the other hand, if the outcomes are similar on a number of dimensions, the usability of the model is increased.

Traditionally, empirical data is collected and analysed on a macroscopic level of stations. It is on this level that PRISM has been designed and will be operated, and this makes comparisons to empirical data straightforward. Microscopic simulation tools like RailSys run on a more detailed model and geography, on the level of single objects like signals and switches. However, the typical output from RailSys in Sweden is already on the same macroscopic level as that found in both empirical data and in PRISM. Thus, the results of these models are easy to compare with one another, and against historical traffic records.

The key parameters to be evaluated are punctuality, average delays, and the delay distributions. These will be evaluated and discussed visually, and with summary statistics such as means, medians and standard deviations. A more formal analysis can be performed using ANOVA (analysis of variance) to check if the values deviate in a statistically significant manner, for both punctuality and average delays. The delay distributions provide more information on the size and frequency of delays and will be mainly be compared using visual methods. The more formal test for the delay distributions is the Kolmogorov-Smirnov test, which checks whether the underlying distributions of the delays in the simulations belongs to the same distribution underlying the empirical delays, which is unlikely and may require extensive calibration to achieve.

These three parameters (punctuality, average delays and delay distributions) can and will be evaluated both on an aggregate, overall level, and on a more disaggregated level of train types and individual stations. The most relevant level to use will vary from case to case, depending on the specific purpose, but a good balance between detail and overfitting is likely to be reached by focusing on evaluating the outcomes at major hubs and junctions in the network. This is more detailed than considering merely the final destination but will be more generally applicable and robust than considering every station or timing point. When considering specific train services, rather than the network as a whole, it is appropriate to evaluate the stations where the trains stop, and to omit any intermediate timing points.

As PRISM uses a macroscopic model, it is less important to have very detailed data in the evaluation. Missing such data is thus less problematic than for microscopic simulation tools such as RailSys. As described above, it is still possible to simulate, and with that evaluate the punctuality, average delay and delay distribution at the final destination. Unknown speed profiles can be evaluated by aggregating broad categories of trains, such as freight trains, where there is a large heterogeneity, whereas more well-defined ones can be compared by using less aggregated sets of trains in the comparison cases. In either case, it is prudent to start by establishing a well-calibrated base scenario, before scenarios with missing observations are run.

Missing data makes the comparison to real data more difficult. Especially the maximum speed and the number of stops is expected to have large impacts on delay outcomes, while the exact routing or location of the stops is expected to be less important. If many assumptions are used or the input is generalised to a large degree, the results will not be comparable enough to enable a proper analysis. In that case, the evaluations must rely more heavily on the comparison to microscopic simulations in RailSys, for instance.

8. Conclusions and future work

The conclusions reached at this stage of the R&I and highlighted in this report are that there are certain situations where the available information is not sufficient for performing a simulation, but that there are ways to overcome these shortcomings.

The report describes simulation processes for micro- and macroscopic simulation tools with the example of RailSys and PRISM. It discusses which kind of missing data that can occur, and how it affects simulation. The effect differs between the models: While the microscopic simulation model is dependent on detailed input, less complete data is sufficient for the macroscopic model to deliver useful results. However, the macroscopic model needs input about the running times, while that information can be created in the microscopic tool itself.

Focussing on cases with incomplete timetable, it is shown how randomisation of train passes based on empirical data can be used to recreate adequate timetables. Further, it is discussed how the simulation results can be evaluated and compared to real data, and which difficulties occur when this is done for simulations based on incomplete data.

Research about simulation with incomplete data will continue and, in a case study, the proposed method in this deliverable will be applied and further developed.

9. References

Bendfeldt, J.-P., Mohr, U., & Müller, L. (2000). RailSys, a system to plan future railway needs. *WIT Transactions on the Built Environment, 50*, 249-255.

Betz, T., Schüle, I., Kordnejad, B., Wahlborg, M., Bohlin, M., & Kempf, S. (2017). *D2.1 - Summary of state-of-the-art in simulation.* PLASA – Smart Planning and Safety for a safer and more robust European railway sector.

Betz, T., Terschlüsen, C., Zinser, M., Solinen, E., Warg, J., Spiller, A., & Saeednia, M. (2018). *D 3.3 – Depiction of analysed case studys.* PLASA – Smart Planning and Safety for a safer and more robust European railway sector.

Borndörfer, R., Klug, T., Lamorgese, L., Mannino, C., Reuther, M., & Schlechte, T. (Eds.). (2018). *Handbook of Optimization in the Railway Industry.* Springer.

Cui, Y., Martin, U., & Liang, J. (2018). PULSim: user-based adaptable simulation tool for railway planning and operations. *Journal of Advanced Transportation, 2018*.

Janecek, D., & Weymann, F. (2010). LUKS-Analysis of lines and junctions. *Proceedings of the 12th World Conference on Transport Research (WCTR).* Lisbon.

Nash, A., & Hürlimann, D. (2004). Railroad Simulation using OpenTrack. *Computers in Railways IX*, 45-54.

Radtke, A., & Hauptmann, D. (2004). Automated planning of timetables in large railway networks using a microscopic data basis and railway simulation techniques. *WIT Transactions on The Built Environment, 74*.

Zinser, M., Betz, T., Becker, M., Geilke, M., Terschlüsen, C., Kaluza, A., Johansson, I., Warg, J. (2019). PRISM: A Macroscopic Monte Carlo Railway Simulation. *12th World Congress on Railway Research* (p. 6). Tokyo, Japan: Submitted to the 12th World Congress on Railway Research.