Markov Models for the Performance Analysis of Railway Networks

Christoph Schmitz [a], Norman Weik [b], Stephan Zieger [b],
Nils Nießen [b], Anke Schmeink [a]

[a] RWTH Aachen University, Institute for Theoretical Information Technology,
Kopernikusstraße 16/Aachen, Germany
{schmitzc,schmeink}@ti.rwth-aachen.de

[b] RWTH Aachen University, Chair of Railway Engineering and Transport Economics,
Mies-van-der-Rohe-Straße 1/Aachen, Germany
{niessen,weik,zieger}@via.rwth-aachen.de

Abstract

In traffic networks like railway lines or roads, capacity is often limited by certain critical areas like junctions or single-tracked sections, which can be seen as servers with multiple queues for jobs arriving from different directions. The behaviour of these servers can be quite complex. For example, it might be possible to process jobs from several queues simultaneously, but not for all possible combinations. Service times can be deterministic or random, and in both cases they might depend on which queue is served, and on some internal state of the server. This work presents an approach to model such a system by Markov chains or Markov processes, with a multi-dimensional state space that is composed of the number of jobs in each queue, and possibly a finite number of internal states. Apart from the given constraints of the system (e.g. which queues can be served simultaneously), the transitions of the Markov model also describe the chosen service strategy. Thus, it is possible to compare the performance of different service strategies, as well as the performance of different systems (e.g., an existing at-grade junction and a grade-separated new version of that junction). The model is tunable to meet different service characteristics and is tested by comparison to simulation.

Keywords

Performance analysis, Markov chain, Markov process, Railway network

1 Introduction

1.1 Motivation

Accommodating the expected growth of European rail freight traffic (Commissie (2011)) poses a major challenge in operating railway infrastructure. Many railway lines and nodes are already heavily loaded and approaching the limits of capacity. A precise determination of the performance of railway facilities, possible bottlenecks as well as the optimization of operations therefore are of great importance.

The most common approaches currently used in capacity modelling either rely on the infrastructure's utilization (e.g., schedule compression according to UIC Code 406 (UIC

(2013)) or stochastic and queueing-based methods (see e.g. Fischer and Hertel (1990); Nießen (2014)). While utilization-based descriptions are easy to understand and implement they are highly dependent on schedule structure and hard to relate to the quality of operations. This can be achieved by the second class of stochastic and queuing based models by defining a level of service based on punctuality, queue lengths or waiting times. However, queueing-based models only allow for a generic consideration of operation and dispatching strategies.

Besides those two modeling types, a large number of scheduling-based papers investigating timetable robustness, e.g. (Andersson et al. (2013); Liebchen et al. (2009, 2010)), and the effectiveness of dispatching decisions (Cacchiani et al. (2014); Corman (2010); Narayanaswami and Rangaraj (2011); Dariano et al. (2007)) has been published. Here, the focus is on tactical or operational planning when schedules are known or basic properties such as schedule periodicity are known (Sparing and Goverde (2013)). Optimization-based planning tools are complemented by microscopic (Radtke and Hauptmann (2004); Janecek and Weymann (2010); Weymann and Nieen (2015)) or macroscopic simulation approaches (Büker and Seybold (2012); Goverde (2007)) for investigating the evolution of timetable stability. In long term capacity planning, where schedules are unknown or may still evolve in time or infrastructure adjustments are assessed, the usability of MIP- and simulation-based approaches is limited. In order to cope with data input uncertainty a large number of problem instances has to be calculated which makes these approaches very time consuming.

The aim of this paper is to present an efficient Markovian model which can be used to determine and evaluate the performance of railway systems. Unlike existing stochastic and queueing-based models the new approach allows to incorporate heuristically motivated strategies commonly used by dispatchers and can easily be adapted to meet different system requirements or operation guidelines. Besides being applicable for assessing existing railway infrastructure and allowing to evaluate different operating strategies, the approach is also provides insights into the implications of infrastructure adjustments.

1.2 Related Work

Stochastic models have successfully been applied in the capacity analysis of railway lines, junctions and stations for several decades. They can be subdivided into approaches where line segments or stations are analyzed individually (see e.g. Nießen (2014) for an overview) and models, where subnetworks consisting of multiple lines can be analyzed (Huisman et al. (2002)). Whereas the first class of models is suited to analyze and compare different infrastucture variants, the second class focuses on network routing effects.

For dimensioning the number of tracks in station, loss probabilities in $GI/D/n/0$ (Potthoff (1965)) and waiting probabilities in $GI/GI/n/\infty$ queueing systems have been used (Hertel (1984)). Railway lines and station threads have also been assessed based on stationary waiting times and queue lengths obtained in queueing based system descriptions (Schwanhäußer (1974, 1994); Wendler (2007); Weik et al. (2016)). Here, the system is mapped to (effective) single channel queueing systems (Schwanhäußer (1974, 1994)), where service times may be adjusted to account for mutually non-exclusive train routes (cf. Nießen (2013)).

A more detailled account of correlation effects between different infrastructure segments in railway networks is given by queueing network descriptions. In (Huisman et al. (2002)), a railway subnetwork has been modelled by a Jackson queueing network (Huisman et al. (2002)). However, the solvability properties of Markovian queueing networks are not preserved if more general arrival and service processes are to be considered.

The concept of Markov chains and Markov processes is well-known, the same holds for the method of modelling a $M/GI/n/m$ queueing systems using Markov chains (Bolch et al. (2006b); Gelenbe et al. (1998); Kleinrock (1975)). Apart from (rail) traffic networks, it has also been used in the performance analysis of computer systems (Gelenbe and Mitrani (2010); Lazowska et al. (1984)) and communication networks (Bertsekas et al. (1992); Clark (1991)). In the latter, results derived decades ago for fixed line networks have recently become relevant again, in the advent of mobile networks with scarce resources.

1.3 Our Contribution

The approach discussed in this paper is aimed at complementing existing queuing-based approaches in individual modeling of infrastructure segments. It exceeds existing approaches in being able to accommodate for different operating strategies apart from FCFS-based service discipline or specific priority-based dispatching rules (Schwanhäußer (1974)) discussed so far. The consideration of phase-type distributed service times ensures a great deal of flexibility and allows to realistically model occupation times for a wide range of infrastructure layouts and train characteristics. At the same time all the advantages of a Markovian approach including the exact calculation of queue length distributions are kept.

The model is thought to be relevant in the planning of new railway infrastructure projects, evaluation of potentially dispensable infrastructure and improving the match between infrastructure variants and operation strategies. As an exemplory application, the model is used to compare the delay built-up at grade-separated and at-grad line junctions. While a MIP-based analytical model to model railway nodes and junctions has been presented in (Mussone and Calvo (2013)), it does not seem to account for guiding system specifications. An alternative approach allowing to estimate knock-on delays in at grade junctions in (Yuan (2006)) focuses on stochastic modelling of delay propagation. To the best of the authors knowledge, however, a direct analysis focussing on different infrastructure variants at railway junctions has not been given so far and is to be provided based on our new Markovian modelling approach.

2 Preliminaries

In the context of railway junctions two main classes of infrastructure variants exist: at-grade and grade-separated junctions. Figures 1 and 2 represent the two variants for a double track junction between the three stations A, B and C. The colored arrows show the possible train paths: Whereas train paths of trains starting or ending in B and C are unique, trains starting in A can proceed either to B or to C at the junction.

In both cases trains on route $C \rightarrow A$ and $B \rightarrow A$ compete for the right of way at the junction. In the at-grade case (portrayed in Figure 1) routes $A \rightarrow C$ and $B \rightarrow A$ conflict in addition. These conflicts (red arrows) are eliminated in grade-separated junctions (Figure 2) with a tunnel or bridge.

This relatively simple looking example of a junction is nevertheless quite illustrative since it shows the capability of the model and raises several questions.

One question arising for a junction is whether and how much the grade-level influences the capacity, i.e., the number of trains that can operate at a given time and of a given level of

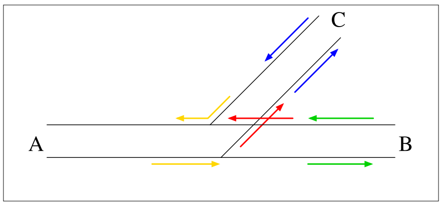

Figure 1: An at-grade junction

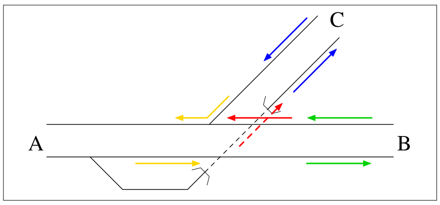

Figure 2: A grade-separated junction

quality on the segment. It can be answered by measuring either the expected length of each waiting queue or the expected time a train has to wait for the right to pass the junction.

Another question comes from the operational point of view whether to build a bridge or tunnel, e.g. a grade-separated junction. For the associated capacity it is clearly an advantage, but on the other hand the construction is more expensive and consumes more space which can in some cases be rare. A supportive evaluation of the situation is as seen important.

3 Analytical Model

To compute analytical results, the stochastic process $X(t)$, which represents the behaviour of the system, has to be defined by a well-known model like a Markov chain or a Markov process. In a system with K queues for jobs coming from K different directions, the state space $\mathcal{S} = \{(k_1, \ldots, k_K, f) \mid k_j \in \mathbb{N}_0, f \in F\}$ typically consists of the lengths of the queues $k_1, \ldots, k_K$, and possibly an internal state f from a finite set F.

For practical reasons the length of the queues has to be finite since the equations for the stationary state have to be solved by mathematical software. Therefore the original (unlimited) system is approximated by the maximum length of a queue $k_{max} < \infty$.

3.1 Markov Chain

A Markov chain (Gelenbe et al. (1998)) describes the system state at discrete points in time. Typically, these *events* are determined for example by a job leaving the system after its service has ended. While arbitrary distributions are possible for the service times, the arrival processes have to be Poisson processes with memoryless interarrival times, to ensure that the sequence of system states at the events forms a Markov chain.

Let D_n be the time the n-th service time ends, and $X_n = X(D_n+)$ the system state at that time. Depending on that state $X_n = (k_1, \ldots, k_K, f)$ and possibly some random influence, the distribution of the next service time Y is chosen, as well the subset $\mathcal{K} \subseteq \{1, \ldots, K\}$ of queues that will be served in the next service time (with $k_j > 0$ for all $j \in \mathcal{K}$), and the next internal state $f' \in F$ (if present). This choice is made depending on the constraints of the system and on the service strategy. Then, the next system state is

$$X_{n+1} = (k_1 - \mathbb{I}_{\mathcal{K}}(1) + Z_1, \ldots, k_K - \mathbb{I}_{\mathcal{K}}(K) + Z_K, f'),$$

where Z_j is the number of jobs that arrive at the j-th queue during the service time Y. Given the intensity λ_j of the arrival process at that queue, and the distribution of Y, the distribution of Z_j can be calculated as

$$\mathrm{P}(Z_j = i) = \int_0^\infty \frac{(\lambda_j t)^i}{i!}\, e^{-\lambda_j t} dF_Y(t).$$

In case of deterministic service time of length t_0, the Z_j are Poisson distributed with parameter $\lambda_j t_0$.

The model can be extended to cover the case that jobs are served in parallel, but asynchronously. In that case, the time between two events is not the complete service time of one job, but only a part of it. In each state, the Markov chain has to store information about which jobs are still in service at this time, and for how long. In fact, the time is divided into short time slots of a positive length, to ensure that the state space remains countable. To know for how long some job has still to be served, it is necessary to know when service started, unless the distribution of service times is memoryless. Altogether, the model gets very complicated in this case and realistic service times are hard to consider.

3.2 Markov Process

A Markov process (Lazowska et al. (1984)) is a continuous time stochastic process. In contrast to a Markov chain it describes the behaviour of the system at any time, not only at discrete points in time. Arrivals and departures are now described by separate state transitions, thus they are independent. Naturally, the times between state changes of a Markov process are exponentially distributed. By the introduction of intermediate states, it is possible to cover Cox distributions, too. It is known that every distribution with positive support can be approximated by Cox distributions. Thus, arbitrary service time distributions can be considered in this model, although the number of phases (and thus the number of intermediate states) sometimes has to be large for a good approximation. Normally, jobs served in parallel are represented by independent transitions, so they can be served asynchronously anyway. If desired, Cox distributions can also be used for the interarrival times.

The state space can still be expressed as $\mathcal{S} = \{(k_1, \ldots, k_K, f) \mid k_j \in \mathbb{N}_0, f \in F\}$, where the variables $k_1, \ldots, k_K$ represent the queue lengths, and F is a finite set of internal states. The latter typically stores information about which types of jobs are currently served, and for how long they are already in service. In a system where at most R jobs can be served at the same time, the state space might be described as

$$\mathcal{S} = \{(k_1, \ldots, k_K, p_1, t_1, \ldots, p_R, t_R) \mid k_j \in \mathbb{N}_0\}$$

in more detail, in which p_i is the number of phases that have already passed for the i-th job, and t_i designates the type of that job. The value $p_i = 0$ indicates that there is currently no job being served by the i-th resource. Of course, there might as well be some more internal information to be stored.

In such a system, there are generally three types of transitions in the Markov process. The first type is related to the arrival process, and it looks like

$$(k_1, \ldots, k_j, \ldots, k_K, p_1, t_1, \ldots) \underset{\lambda_j}{\longrightarrow} (k_1, \ldots, k_j + 1, \ldots, k_K, p_1, t_1, \ldots).$$

The second type relates to jobs that are already in service, but not yet finished. For some i with $p_i \geq 1$, let the service time of a job of type t_i be given by a Cox$(\alpha_1, \ldots, \alpha_{m-1}; \mu_1, \ldots, \mu_m)$ distribution.

Thus, the next phase of the service time has the intensity μ_{p_i+1}. In case we already have $p_i = m-1$, the service time will definitely terminate after this phase, thus the transition is

$$(\dots, p_1, t_1, \dots, p_i, t_i, \dots, p_R, t_R) \underset{\mu_{p_i+1}}{\longrightarrow} (\dots, p_1, t_1, \dots, 0, t_i, \dots, p_R, t_R).$$

Otherwise, the service time will terminate after the next phase with probability $1 - \alpha_{p_i+1}$, and it will continue after that with probability α_{p_i+1}. This induces the transitions

$$(\dots, p_1, t_1, \dots, p_i, t_i, \dots, p_R, t_R) \underset{(1-\alpha_{p_i+1})\mu_{p_i+1}}{\longrightarrow} (\dots, p_1, t_1, \dots, 0, t_i, \dots, p_R, t_R),$$
$$(\dots, p_1, t_1, \dots, p_i, t_i, \dots, p_R, t_R) \underset{\alpha_{p_i+1}\mu_{p_i+1}}{\longrightarrow} (\dots, p_1, t_1, \dots, p_i + 1, t_i, \dots, p_R, t_R).$$

The third type is related to the start of new services. When there is a free resource, and there are jobs waiting in some queues whose type is compatible to the services currently running, then one of these jobs can be started. Apart from the constraints of the system, the choice is normally also based on some service strategy, which might include some random influence. Assume that $p_i = 0$ and $k_j \geq 1$, and the first job waiting in the j-th queue can be served by a service of type $\tilde{t}$, whose service time is given by a Cox$(\alpha_1, \dots, \alpha_{m-1}; \mu_1, \dots, \mu_m)$ distribution. If this service is compatible with the other services currently running, then this job might be chosen with some probability $\gamma > 0$. With probability $1 - \alpha_1$, the service terminates after the first phase, and with probability α_1 it continues. Altogether, this induces the transitions

$$(k_1, \dots, k_j, \dots, 0, t_i, \dots, p_R, t_R) \underset{\gamma(1-\alpha_1)\mu_1}{\longrightarrow} (k_1, \dots, k_j - 1, \dots, 0, \tilde{t}, \dots, p_R, t_R),$$
$$(k_1, \dots, k_j, \dots, 0, t_i, \dots, p_R, t_R) \underset{\gamma\alpha_1\mu_1}{\longrightarrow} (k_1, \dots, k_j - 1, \dots, 1, \tilde{t}, \dots, p_R, t_R).$$

If we have $\gamma < 1$, which means that there are other choices that are taken with a positive probability, then there have to be similar transitions for those, too.

Of course, this short description is only a rough sketch of how a complicated system is modeled as a Markov process, and there are many more details. For example, it might be necessary to store the type of a *waiting* job somehow, in case the first job in some queue still has to wait because its type is not compatible with the running services. In some cases, the job type of an idle resource might be used for that, especially if it is already determined that the job will be served by that resource. In general, nothing has been said about the mapping of the resources to the queues by now, because it is not possible to give a description that covers arbitrary systems.

In the case of the junction shown in the Figures 2 and 1, where a double-track line coming from some place A branches towards two different destinations B and C, there are three queues and two resources. The first resource serves the jobs coming from A, and those jobs can have two different types, depending on their destination. Jobs coming from B and C are both served by the second resource, as they cannot be served in parallel. Their type designates where the job came from. In case of a grade-separated junction, both resources can always be working simultaneously, but in an at-grade junction there are some exclusions, depending on the types of the jobs. If the job coming from A has to wait because it is not compatible with a running service, its type can already be stored in the first resource which has to remain idle in this case.

3.3 Cox Distributions

As shown above, Cox distributions (Cox (1955); Bolch et al. (2006a)) can easily be considered in Markov processes, as they are composed of exponential distributions and some random decisions. While it is known that every distribution with positive support can be approximated by a Cox distribution when the number of phases goes to infinity, it is also interesting to examine the capabilities of Cox distributions with a fixed number of phases. As the state space of the Markov process grows with the number of phases, it is desirable to keep the latter small, if possible.

Let $\widetilde{Y} \sim \text{Cox}(\alpha_1, \ldots, \alpha_{m-1}; \mu_1, \ldots, \mu_m)$ be Cox distributed with m phases. It can be shown as an extension of (Cox (1955)) that for the coefficient of variation (CV) the inequality

$$\text{CV}(\widetilde{Y}) = \frac{\sqrt{\text{Var}(\widetilde{Y})}}{\text{E}(\widetilde{Y})} \geq \frac{1}{\sqrt{m}}$$

holds, and that this border is reached if the Cox distribution is an Erlang distribution, which means that $\alpha_1 = \ldots = \alpha_{m-1} = 1$ and $\mu_1 = \ldots = \mu_m$. On the other hand, this also implies that for a given coefficient of variation $\text{CV}(Y)$ of an arbitrary random variable Y, we can compute the required number of phases using the inequality

$$m \geq \left\lceil \frac{1}{\text{CV}(Y)^2} \right\rceil .$$

If the random variable Y, which is to be approximated, is characterized only by its mean $\text{E}(Y)$ and variance $\text{Var}(Y)$, there are generally some degrees of freedom remaining when selecting a Cox distribution with an appropriate number of phases. Our approach is to use a random variable

$$\widetilde{Y}_{m,\alpha} \sim \text{Cox}\left(\alpha, \ldots, \alpha; \frac{m}{\text{E}(Y)}, \frac{\alpha m}{\text{E}(Y)}, \ldots, \frac{\alpha^{m-1} m}{\text{E}(Y)}\right)$$

with $m \geq 2$ phases and $\alpha \in (0, 1]$. Its expected value is

$$\text{E}(\widetilde{Y}_{m,\alpha}) = \sum_{i=1}^{m} \alpha^{i-1} \frac{\text{E}(Y)}{\alpha^{i-1} m} = \text{E}(Y)$$

as desired, and it can be shown that the variance is strictly decreasing in α with

$$\text{Var}(\widetilde{Y}_{m,1}) = \frac{\text{E}(Y)^2}{m} \qquad \text{and} \qquad \lim_{\alpha \to 0} \text{Var}(\widetilde{Y}_{m,\alpha}) = \infty.$$

Thus, there exists a unique $\alpha \in (0, 1]$ for which $\text{Var}(\widetilde{Y}_{m,\alpha}) = \text{Var}(Y)$ holds. For $m = 2$, it can be calculated as

$$\alpha = \frac{\text{E}(Y)^2}{2\,\text{Var}(Y)} = \frac{1}{2\,\text{CV}(Y)^2}.$$

A closed form expression for an arbitrary number of phases is not known, in some cases it might be required to compute α by numerical methods.

4 Simulation

The analytical approach discussed in the previous section is tested by comparing the results with a direct Monte-Carlo simulation of the junction. In the simulation, independently Poi(λ_i)-distributed arrivals (with $i \in \{A, B, C\}$) of requests at the three queues are generated. Requests are served according to the same (heuristic) service discipline as in the Markov model. Service times of requests are generally independently distributed and can be matched to practical requirements. In the present work, exponentially distributed and Gaussian service times are considered.

The structure of the simulation is illustrated in Figure 3. We subsequently briefly discuss its constituents.

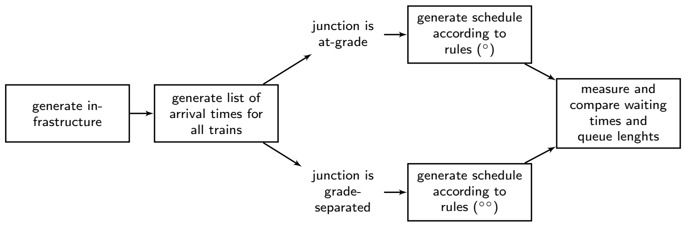

Figure 3: process of the schedule generation - the rules (°) and (°°) are included in Appendix B

First, infrastructure parameters including length, admissible velocity, block structure and position of the junction are defined. To facilitate comparison to the analytical model the expected service time is normalized to 1 such that the arrival rate corresponds to the utilization ratio of the queueing system in the grade-separated case. For the sake of simplicity in the analytical model, no blocks subdividing the incoming tracks are considered. This corresponds to the three starting points A, B and C being directly adjacent to the junction.

In a second step, the lists of arrival times of trains at the three stations are generated from independently exponentially distributed inter-arrival times with parameters corresponding to the arrival rate at the respective station.

Depending on the given infrastructure a conflict free schedule is generated. Trains depart their station of origin at the moment the intersection is empty, i.e. no conflicting train run is in service. At this point, the information on the infrastructure layout of the intersection enters, defining which train routes are mutually exclusive, resulting in different conflict-resolution routines for the grade-separated and at-grade case.

The queue length distribution as well as statistical parameters such as the expectation value and variance of queue length are determined by summing the length of inter-event time windows the system is in a given state. By dividing by the total simulation time the share of time the system exhibits queue length i, and hence the queue length distribution is obtained.

5 Results and Evaluation

The discussion of the results is subdivided in two sections. The first deals with the capabilities and the statistical significance of the simulation approach. In the second part the evaluation and comparison of the simulation and analytical results are presented.

Quality and computational complexity

In the following the performance of the simulation approach is discussed. All computations were performed using MATLAB (Matlab2016b (2016)) on a PC with i5-6500 (3.20 GHz) kernel and 8 GB of RAM. We start by investigating the accuracy of the simulated results. To this end, the standard deviation of the entries of the queue length distributions is analyzed for different numbers of runs and trains within one simulation run. In the associated rows the standard deviation of the marginal distribution for Stations A, B and C can be extracted. The corresponding results are presented in Tables 1, 2 and Appendix C.

		Queue length				
(# runs/# trains)	m	0	1	2	3	4
(5/10)	A	0.06616	0.09373	0.01056	-	-
	B	0.11256	0.09235	-	-	-
	C	0.09045	0.05792	0.05824	-	-
(10/10)	A	0.12143	0.08142	0.02015	0.03967	0.01905
	B	0.07266	0.06628	-	-	-
	C	0.08224	0.08397	-	-	-
(100/100)	A	0.02614	0.01967	0.01023	0.00426	0.00176
	B	0.02286	0.01838	0.00757	0.00199	0.00014
	C	0.02099	0.01782	0.00571	0.00097	0.00039
(100/1,000)	A	0.00800	0.00581	0.00314	0.00125	0.00047
	B	0.00660	0.00575	0.00207	0.00059	0.00016
	C	0.00577	0.00499	0.00162	0.00046	0.00011
(100/10,000)	A	0.00276	0.00210	0.00111	0.00040	0.00013
	B	0.00236	0.00198	0.00072	0.00021	0.00005
	C	0.00193	0.00172	0.00054	0.00014	0.00003
(100/100,000)	A	0.00068	0.00061	0.00029	0.00012	0.00004
	B	0.00064	0.00052	0.00018	0.00006	0.00001
	C	0.00065	0.00055	0.00017	0.00004	0.00001

Table 1: Standard deviation of the marginal distributions for different (number of runs/number of trains)-combinations of the at-grade junction

The first observation is that a large number of trains is necessary to encounter longer queues in the stations. Due to the junction layout station A is more vulnerable to longer queues. In general one needs many train runs to encounter rare cases when subsequent delays push each other to a level where a lot of trains wait.

The next observation is that with small number of trains and runs the standard deviation is not only higher, but more likely to produce random results. With bigger combinations a decrease of the standard deviation and thus a convergence to the expected queue length is achieved. As expected it is observed that at the grade separated junction the standard

(# runs/# trains)	m	Queue length 0	1	2	3	4
(5/10)	A	0.03291	0.05231	0.02480	0.00228	-
	B	0.06472	0.06385	-	-	-
	C	0.10427	0.05599	-	-	-
(10/10)	A	0.08117	0.07428	0.03766	0.00107	-
	B	0.07997	0.06542	0.01998	-	-
	C	0.09651	0.05382	0.01828	-	-
(100/100)	A	0.02797	0.02031	0.01021	0.00462	0.00186
	B	0.01976	0.01682	0.00485	0.00124	0.00031
	C	0.02116	0.01774	0.00512	0.00109	0.00071
(100/1,000)	A	0.00773	0.00639	0.00293	0.00110	0.00036
	B	0.00659	0.00562	0.00164	0.00040	0.00008
	C	0.00611	0.00518	0.00160	0.00045	0.00014
(100/10,000)	A	0.00264	0.00204	0.00094	0.00035	0.00013
	B	0.00162	0.00144	0.00053	0.00015	0.00003
	C	0.00195	0.00170	0.00056	0.00014	0.00003
(100/100,000)	A	0.00082	0.00063	0.00027	0.00011	0.00003
	B	0.00058	0.00053	0.00015	0.00003	0.00001
	C	0.00051	0.00046	0.00016	0.00004	0.00001

Table 2: Standard deviation of the marginal distributions for different (number of runs/number of trains)-combinations of the grade separated junction

deviation of stations B and C is approximately equal whereas at the at-grade junction the preference of B over C can be seen.

Figures 1 and 2 suggest that the selection of the number of trains and runs greatly influences the desired accuracy of the simulation. With an increasing number of trains and runs the accuracy of the simulation is increasing as well. In particular the standard deviation of the marginal distribution decreases which serves as measurement of convergence. The decision yet not only influences the quality of the approximation, but the computational time, too. The corresponding simulation time in seconds are depicted from Table 3.

# trains	10	50	100	500	1,000	5,000	10,000	100,000
# runs								
5	0.357	0.460	0.614	2.051	3.980	17.103	33.723	336.349
10	0.552	0.838	1.518	4.278	7.981	36.054	68.728	663.808
50	2.869	5.338	6.149	22.050	38.625	183.822	343.191	3436.729
100	5.866	9.667	12.772	42.858	77.982	358.799	699.200	6675.990

Table 3: Running time of the simulation in seconds for various combinations of the number of trains and runs

The computation within the analytical model takes 0.066513 seconds which is distinctly smaller than the simulation results. The computational complexity of the simulation is approximately $O(t \cdot r)$ with number of trains t and number of runs r which can be verified by the results in Table 3. Nevertheless the model may not only achieve more precise results, but fast computation time for manageable problem sizes, too.

Comparison of analytical and simulation results

For the simulation data depicted in the following figures 100 runs with 100,000 trains each have been completed. We decided to use the quite high number of runs and trains due to no time limitations and a good quality of the results as analyzed in the previous part. Arrival rates are set to 0.2 for station A and 0.1 for B and C, each. In the following figures "Queue A/B/C" refer to the results of the analytical model and are plotted as a cross whereas the simulation results are referred as "Queue A/B/C (sim)" and marked as circles.

There are several key properties to be analyzed: The first one is the distribution of the number of trains in the system (queue length) with respect to their origins. The number of trains in the system is defined as the sum of trains waiting in A, B and C, respectively, including all trains currently receiving service.

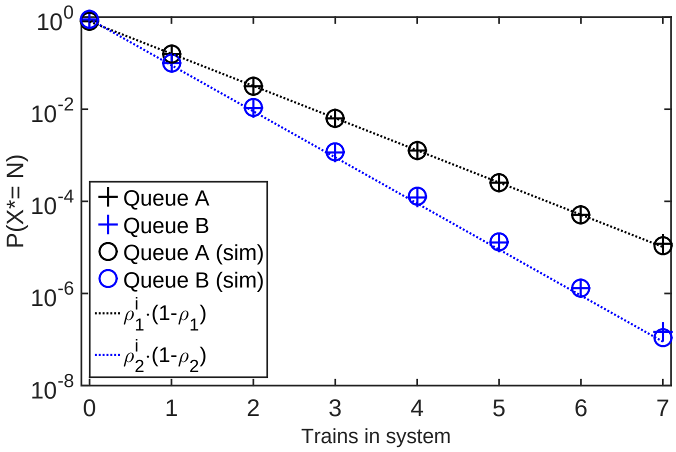

Figure 4: Marginal distributions of the number of trains in the system with origins $X^* \in \{A, B\}$ for a grade-separated junction. Service times are Markovian (variation coefficient 1). The exact results given by the geometric sequence with $\rho_1 = 0.2$ and $\rho_2 = 0.1$ in this case are depicted for comparison.

In Figure 4 the probability of a specific number of trains in the system is visualized in a semilogarithmic scale. The number of trains for origin C is not depicted for better readability and because the results for B and C are identical in the grade-separated case due to the symmetry of the infrastructure and service rules (cf. Appendix B).

The results correspond well with the geometric sequences $(1 - \rho_i) \cdot \rho^i$ ($\rho_1 = 0.2$, $\rho_2 = 0.1$), which is the exact result for the number of customers in queueing systems of type $M/M/1$.

For the at-grade variant, which is presented in Figure 5, the number of trains waiting in B and C cannot be expected to be equal anymore due to the additional conflicts between directions $A \to C$ and $B \to A$. These conflicts

- break the symmetry between the queues in B and C and
- introduce correlations between trains of opposite directions. The queueing process for requests originating in A is no longer decoupled from service of requests originating in B or C and vice versa.

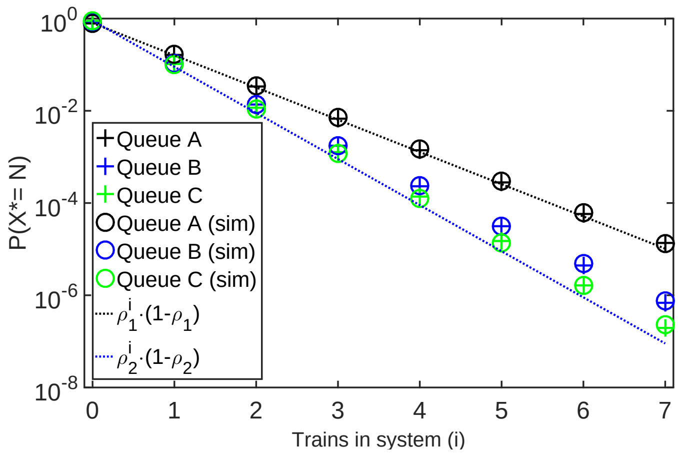

Figure 5: The probability of the number of trains in the system for origins $X^* \in \{A, B, C\}$ for an at-grade junction versus the geometric sequence with $\rho_1 = 0.2$ and $\rho_2 = 0.1$

Indeed, as can be seen in Figure 5, Queue C is now significantly shorter than Queue B and Queue lengths in A, B and C deviate from the geometric distribution obtained in the grade-separated case.

This is to be expected by the infrastructure layout and the service strategy discussed in detail in Appendix B. As Queue C does not suffer any conflicts with trains originating in A and Queue B is not given preferential treatment w.r.t. Queue C, it can be expected to be shorter than Queue B. Correlations between the two directions also ensure the system no longer corresponds to independently operating queueing systems of type $M/M/1/\infty$, such that deviations from the geometric sequence are to be expected. However, it can be noted the results obtained with the Markovian approach discussed in Section 3 correspond well with simulation results.

In Figures 6 and 7 the expectation value of queue lengths is depicted for different variation of train service times. The latter are taken to be normally distributed with variation coefficients varying between 0.75 and 1.25.

It can be noted that the expected queue lengths grow as the variance does. Note that the expectation value of service times is identical for all data points (set to 1), but the variance of service times is altered such that different variations are obtained.

When comparing analytical results and simulation for the grade-separated case in Figure 6 a good match between the corresponding data is found. As before, the data for the queues in stations B and C is equal for symmetry reasons.

For the at-grade case depicted in Figure 7 several interesting observations can be made. First, as already observed in Figure 5, a bias between B and C is observed. Moreover, our Markovian approach seems to underestimate the expected queue lengths for all three origins.

Whereas the first observation can be explained by infrastructure layout and service strategy rules, the situation is less clear for the second observation. The difference between the Markovian approach and simulation could possibly be due to the fact that Markovian processes only allow for a single transition at a given time. In the simulation, by contrasts, two events can happen at the same time: For instance, two trains may start in different stations

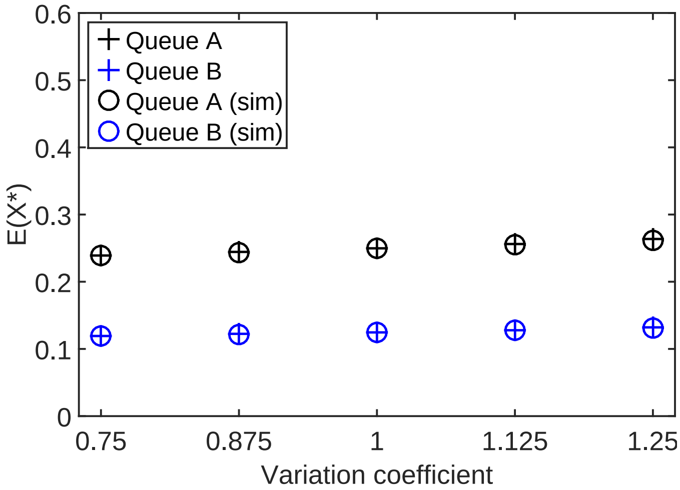

Figure 6: The expected length of the queues for origins $X^* \in \{A, B, C\}$ in a grade-separated junction for variation coefficients for the service times $0.75, 0.875, 1, 1.125$ and 1.25

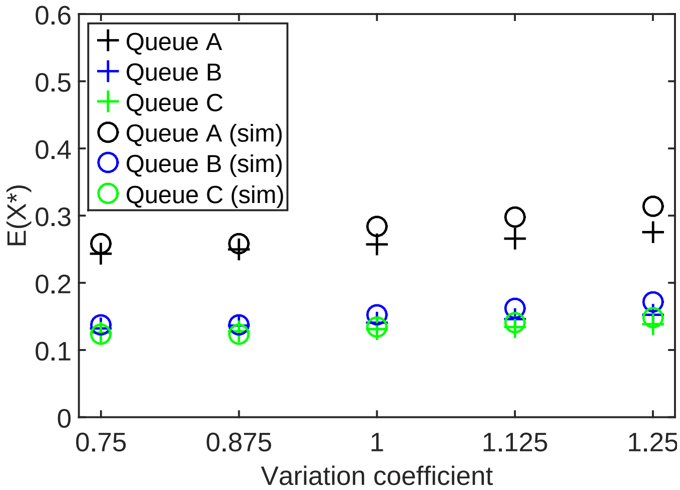

Figure 7: The expected length of the queues for origins $X^* \in \{A, B, C\}$ in an at-grade junction for variation coefficients for the service times $0.75, 0.875, 1, 1.125$ and 1.25

at the same time or arrivals and starting or ending of service can appear simultaneously. This introduces a higher degree of irregularity in the system as compared to Markovian processes. The deviation could possibly be healed by considering Batch-Markov-Processes in the analytical model instead.

6 Conclusions

In this paper railway junctions as an example of critical infrastructure elements often becoming bottlenecks in operations have been analyzed. In order to analyze their impact on line capacity the number of trains needs to be modeled in a detailed, yet computationally accessible way.

To meet these requirements we have introduced a Markovian model that allows computing queue lengths on the line segments adjacent to railway junctions, thus indicating how congested the corresponding track segments are. The model allows to treat different infrastructure layouts and can serve to assess the cost for revenue ratio in terms of line capacity for different variants in construction projects.

By comparing model results to simulation it has been shown the model meets expectations for a standard case. It is now to be extended to a map more complex situations including different train types and velocities as well as more complex network segments.

7 Acknowledgements

This project has been supported by the Excellence Initiative of the German federal and state governments.

References

Emma V Andersson, Anders Peterson, and Johanna Törnquist Krasemann. Quantifying railway timetable robustness in critical points. *Journal of Rail Transport Planning & Management*, 3(3):95–110, 2013.

Dimitri P Bertsekas, Robert G Gallager, and Pierre Humblet. *Data networks*, volume 2. Prentice-Hall International New Jersey, 1992.

Gunter Bolch, Stefan Greiner, Hermann de Meer, and Kishor S Trivedi. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons, 2006a.

Gunter Bolch, Stefan Greiner, Hermann de Meer, and Kishor S Trivedi. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons, 2006b.

Thorsten Büker and Bernhard Seybold. Stochastic modelling of delay propagation in large networks. *Journal of Rail Transport Planning & Management*, 2(1):34–50, 2012.

Valentina Cacchiani, Dennis Huisman, Martin Kidd, Leo Kroon, Paolo Toth, Lucas Veelenturf, and Joris Wagenaar. An overview of recovery models and algorithms for real-time railway rescheduling. *Transportation Research Part B: Methodological*, 63:15–37, 2014.

Martin P Clark. *Networks and telecommunications*. Wiley, 1991.

Europese Commissie. *White paper, Roadmap to a single European Transport Area, Towards a competitive and resource efficient transport system*. COM (2011), 2011.

Francesco Corman. *Real-time Railway Traffic Management: Dispatching in complex, large and busy railway networks*. TU Delft, Delft University of Technology, 2010.

David R Cox. A use of complex probabilities in the theory of stochastic processes. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 51, pages 313–319. Cambridge Univ Press, 1955.

Andrea Dariano, Dario Pacciarelli, and Marco Pranzo. A branch and bound algorithm for scheduling trains in a railway network. *European Journal of Operational Research*, 183 (2):643–657, 2007.

Klaus Fischer and Günter Hertel. *Bedienungsprozesse im Transportwesen: Grundlagen und Anwendungen der Bedienungstheorie*. Transpress-Verlag, 1990.

Erol Gelenbe and Isi Mitrani. *Analysis and synthesis of computer systems*, volume 4. World Scientific, 2010.

Erol Gelenbe, Guy Pujolle, and JCC Nelson. *Introduction to queueing networks*. Wiley Chichester, 1998.

Rob MP Goverde. Railway timetable stability analysis using max-plus system theory. *Transportation Research Part B: Methodological*, 41(2):179–201, 2007.

Günter Hertel. Exakte Lösung zur Berechnung der Wartegleiszahl vor im Einrichtungsbetrieb befahrenen Streckengleisen bei Nicht-Poisson-Ankünften (G/M/1-Wartesystem). *Wissenschaftliche Zeitschrift der Hochschule für Verkehrswesen*, 31:195–205, 1984.

Tijs Huisman, Richard J Boucherie, and Nico M Van Dijk. A solvable queueing network model for railway networks and its validation and applications for the netherlands. *European Journal of Operational Research*, 142(1):30–51, 2002.

David Janecek and Frédéric Weymann. LUKS-analysis of lines and junctions. In *Proceedings of the 12th World Conference on Transport Research (WCTR)*, 2010.

Leonard Kleinrock. *Theory, Volume 1, Queueing Systems*. Wiley-Interscience, 1975.

Edward D Lazowska, John Zahorjan, G Scott Graham, and Kenneth C Sevcik. *Quantitative system performance: computer system analysis using queueing network models*. Prentice-Hall, Inc., 1984.

Christian Liebchen, Marco Lübbecke, Rolf Möhring, and Sebastian Stiller. The concept of recoverable robustness, linear programming recovery, and railway applications. In *Robust and online large-scale optimization*, pages 1–27. Springer, 2009.

Christian Liebchen, Michael Schachtebeck, Anita Schöbel, Sebastian Stiller, and André Prigge. Computing delay resistant railway timetables. *Computers & Operations Research*, 37(5):857–868, 2010.

Matlab2016b. The mathworks. *Inc., Natick, MA*, 2016.

Lorenzo Mussone and Roberto Wolfler Calvo. An analytical approach to calculate the capacity of a railway system. *European Journal of Operational Research*, 228(1):11–23, 2013.

Sundaravalli Narayanaswami and Narayan Rangaraj. Scheduling and rescheduling of railway operations: a review and expository analysis. *Technology Operation Management*, 2 (2):102–122, 2011.

Nils Nießen. Waiting and loss probabilities for route nodes. In *Proc. of the 5th International Seminar on Railway Operations Modelling and Analysis, Rail Copenhagen*, 2013.

Nils Nießen. Queueing. In IA Hansen and J Pachl, editors, *Railway timetabling and operations*. Hamburg: DVV Media Group Eurailpress, 2014.

Gerhart Potthoff. *Die Bedienungstheorie im Verkehrswesen*. Transpress VEB, 1965.

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

Wulf Schwanhäußer. *Die Bemessung der Pufferzeiten im Fahrplangefüge der Eisenbahn*. PhD thesis, Verkehrswissenschaftliches Institut der Rheinisch-Westfälischen Technischen Hochschule, 1974.

Wulf Schwanhäußer. The status of german railway operations management in research and practice. *Transportation Research Part A: Policy and Practice*, 28(6):495–500, 1994.

Daniel Sparing and Rob MP Goverde. An optimization model for periodic timetable generation with dynamic frequencies. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 785–790. IEEE, 2013.

Code 406 – Capacity. UIC, 2nd edition, 2013.

Norman Weik, Nora Niebel, and Nils Nießen. Capacity analysis of railway lines in germany–a rigorous discussion of the queueing based approach. *Journal of Rail Transport Planning & Management*, 6(2):99–115, 2016.

Ekkehard Wendler. The scheduled waiting time on railway lines. *Transportation Research Part B: Methodological*, 41(2):148–158, 2007.

Frdric Herbert Georges Weymann and Nils Nieen. Optimisation processes to assist with the fine compilation of timetables. *Railway technology review : ETR / International Edition*, 2015(1):24–27, 2015. ISSN 0013-2845.

Jianxin Yuan. *Stochastic modelling of train delays and delay propagation in stations*, volume 2006. Eburon Uitgeverij BV, 2006.

Appendix A

This is a list of all transitions for a system representing a double-tracked level junction as shown in figure 1. Trains coming from A (queue k_1) proceed either to B (job type $t_1 = 1$) or to C ($t_1 = 2$). Trains in the other direction are serviced by the second resource, where $t_2 = 1$ represents a train from B (queue k_2) and $t_2 = 2$ one from C (k_3). A train from A to C cannot be served in parallel with one from B to A. When the first resource is idle, then its job type represents the type of the next train that will be served. The job type of the second resource is not relevant when it is idle. The service time for all job types is given by $\widetilde{Y}_{m,\alpha}$ as defined before, and $\mu = \frac{m}{\mathrm{E}(\widetilde{Y}_{m,\alpha})}$ represents the service intensity of the first phase. Unless more restrictive conditions are stated, we have $k_1, k_2, k_3 \in \mathbb{N}_0$, $p_1, p_2 \in \{0, \ldots, m-1\}$, $t_1, t_2 \in \{1, 2\}$. The following transitions exist:

$$\begin{aligned}
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\lambda_1}{\longrightarrow} (k_1 + 1, k_2, k_3, p_1, t_1, p_2, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\lambda_2}{\longrightarrow} (k_1, k_2 + 1, k_3, p_1, t_1, p_2, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\lambda_3}{\longrightarrow} (k_1, k_2, k_3 + 1, p_1, t_1, p_2, t_2)
\end{aligned}$$

For $p_1 \leq m - 2$:

$$\begin{aligned}
(k_1, k_2, k_3, m-1, t_1, p_2, t_2) &\underset{\beta\alpha^{m-1}\mu}{\longrightarrow} (k_1, k_2, k_3, 0, 1, p_2, t_2)\\
(k_1, k_2, k_3, m-1, t_1, p_2, t_2) &\underset{(1-\beta)\alpha^{m-1}\mu}{\longrightarrow} (k_1, k_2, k_3, 0, 2, p_2, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\beta(1-\alpha)\alpha^{p_1}\mu}{\longrightarrow} (k_1, k_2, k_3, 0, 1, p_2, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{(1-\beta)(1-\alpha)\alpha^{p_1}\mu}{\longrightarrow} (k_1, k_2, k_3, 0, 2, p_2, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\alpha^{p_1+1}\mu}{\longrightarrow} (k_1, k_2, k_3, p_1 + 1, t_1, p_2, t_2)
\end{aligned}$$

For $p_2 \leq m - 2$:

$$\begin{aligned}
(k_1, k_2, k_3, p_1, t_1, m-1, t_2) &\underset{\alpha^{m-1}\mu}{\longrightarrow} (k_1, k_2, k_3, p_1, t_1, 0, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{(1-\alpha)\alpha^{p_2}\mu}{\longrightarrow} (k_1, k_2, k_3, p_1, t_1, 0, t_2)\\
(k_1, k_2, k_3, p_1, t_1, p_2, t_2) &\underset{\alpha^{p_2+1}\mu}{\longrightarrow} (k_1, k_2, k_3, p_1, t_1, p_2 + 1, t_2)
\end{aligned}$$

For $k_1 \geq 1$:

$$\begin{aligned}
(k_1, k_2, k_3, 0, 1, p_2, t_2) &\underset{\beta(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 1, p_2, t_2) \\
(k_1, k_2, k_3, 0, 1, p_2, t_2) &\underset{(1-\beta)(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 2, p_2, t_2) \\
(k_1, k_2, k_3, 0, 1, p_2, t_2) &\underset{\alpha\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 1, 1, p_2, t_2) \\
(k_1, k_2, k_3, 0, 2, 0, 1) &\underset{\beta(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 1, 0, 1) \\
(k_1, k_2, k_3, 0, 2, 0, 1) &\underset{(1-\beta)(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 2, 0, 1) \\
(k_1, k_2, k_3, 0, 2, 0, 1) &\underset{\alpha\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 1, 2, 0, 1) \\
(k_1, k_2, k_3, 0, 2, p_2, 2) &\underset{\beta(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 1, p_2, 2) \\
(k_1, k_2, k_3, 0, 2, p_2, 2) &\underset{(1-\beta)(1-\alpha)\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 0, 2, p_2, 2) \\
(k_1, k_2, k_3, 0, 2, p_2, 2) &\underset{\alpha\mu}{\longrightarrow} (k_1 - 1, k_2, k_3, 1, 2, p_2, 2)
\end{aligned}$$

For $k_2 \geq 1$:

$$\begin{aligned}
(k_1, k_2, k_3, p_1, 1, 0, t_2) &\underset{(1-\alpha)\mu}{\longrightarrow} (k_1, k_2 - 1, k_3, p_1, 1, 0, 1) \\
(k_1, k_2, k_3, p_1, 1, 0, t_2) &\underset{\alpha\mu}{\longrightarrow} (k_1, k_2 - 1, k_3, p_1, 1, 1, 1)
\end{aligned}$$

For $k_2 \geq 1$, $k_2 \geq k_1$:

$$\begin{aligned}
(0, k_2, k_3, 0, t_1, 0, t_2) &\underset{(1-\alpha)\mu}{\longrightarrow} (0, k_2 - 1, k_3, 0, t_1, 0, 1) \\
(0, k_2, k_3, 0, t_1, 0, t_2) &\underset{\alpha\mu}{\longrightarrow} (0, k_2 - 1, k_3, 0, t_1, 1, 1)
\end{aligned}$$

For $k_3 \geq 1$:

$$\begin{aligned}
(k_1, k_2, k_3, p_1, 2, 0, t_2) &\underset{(1-\alpha)\mu}{\longrightarrow} (k_1, k_2, k_3 - 1, p_1, 2, 0, 2) \\
(k_1, k_2, k_3, p_1, 2, 0, t_2) &\underset{\alpha\mu}{\longrightarrow} (k_1, k_2, k_3 - 1, p_1, 2, 1, 2) \\
(k_1, 0, k_3, p_1, 1, 0, t_2) &\underset{(1-\alpha)\mu}{\longrightarrow} (k_1, 0, k_3 - 1, p_1, 1, 0, 2) \\
(k_1, 0, k_3, p_1, 1, 0, t_2) &\underset{\alpha\mu}{\longrightarrow} (k_1, 0, k_3 - 1, p_1, 1, 1, 2)
\end{aligned}$$

For $k_3 > k_2$ (implies $k_3 \geq 1$):

$$\begin{aligned}
(0, k_2, k_3, 0, t_1, 0, t_2) &\underset{(1-\alpha)\mu}{\longrightarrow} (0, k_2, k_3 - 1, 0, t_1, 0, 2) \\
(0, k_2, k_3, 0, t_1, 0, t_2) &\underset{\alpha\mu}{\longrightarrow} (0, k_2, k_3 - 1, 0, t_1, 1, 2)
\end{aligned}$$

Appendix B

We subsequently give a detailed overview of the service discipline applied in the simulation.

Notation

- s_i: Possible start of service for the next request in queue i ($i \in \{A, B, C\}$).
- $route_A$: Route of the next request entering the system in station A. $t_A \in \{1, 2\}$ where 1 corresponds to the route with destination B, 2 to the one with destination C.
- l_i: length of queue i at the instant of consideration ($i \in \{A, B, C\}$).

At-grade junction (°)

In the at-grade infrastructure variant trains are served according to the following rules.

if $s_A < min(s_B, s_C)$ **then**
    Request from A enters service
else if $s_B < min(s_A, s_C)$ **then**
    Request from B enters service
else if $s_c < min(s_A, s_B)$ **then**
    Request from C enters service
else if $s_A = s_C$ & $s_A < s_B$ **then**
    Requests from A and C enter service simultaneously
else if $s_A = s_B$ & $s_A < s_C$ & $route_A = 1$ **then**
    Requests from A and B enter service simultaneously
else if $s_A = s_B$ & $s_A < s_C$ & $route_A = 2$ **then**
    Request from A enters service. Request from B waits.
else if $s_B = s_C$ & $s_B < s_A$ & $l_B > l_C$ **then**
    Request from B enters service
else if $s_B = s_C$ & $s_B < s_A$ & $l_C > l_B$ **then**
    Request from B enters service
else if $s_B = s_C$ & $s_B < s_A$ & $l_B = l_C$ **then**
    Origin of next request (B or C) is decided at random
else if $s_A = s_B = s_C$ & $route_A = 1$ **then**
    Requests from A and B enter service simultaneously
else if $s_A = s_B = s_C$ & $route_A = 2$ **then**
    Requests from A and C enter service simultaneously
end if

Grade-separated junctions (°°)

In the grade separated infrastructure variant trains are served according to the following rules.

if $s_A < min(s_B, s_C)$ **then**
  Request from A enters service
else if $s_B < min(s_A, s_C)$ **then**
  Request from B enters service
else if $s_c < min(s_A, s_B)$ **then**
  Request from C enters service
else if $s_A = s_C$ & $s_A < s_B$ **then**
  Requests from A and C enter service simultaneously
else if $s_A = s_B$ & $s_A < s_C$ **then**
  Requests from A and B enter service simultaneously
else if $s_B = s_C$ & $l_B > l_C$ **then**
  Request from B enters service
else if $s_B = s_C$ & $l_C > l_B$ **then**
  Request from B enters service
else if $s_B = s_C$ & $l_B = l_C$ **then**
  Origin of next request (B or C) is decided at random
end if

Appendix C

(# runs/# trains)	m	Queue length 0	1	2	3	4	5	6	7
(5/10)	A	0.066161043	0.093730679	0.010564648	-	-	-	-	-
	B	0.112567188	0.092352950	-	-	-	-	-	-
	C	0.090452121	0.057928640	0.058241754	-	-	-	-	-
(10/10)	A	0.121439450	0.081423161	0.020157948	0.039677679	0.019051999	-	-	-
	B	0.072663649	0.066280829	-	-	-	-	-	-
	C	0.082241038	0.083979052	-	-	-	-	-	-
(100/100)	A	0.026146454	0.019679689	0.010232689	0.004267402	0.001762892	0.000404786	0.000218387	-
	B	0.022866976	0.018383766	0.007577367	0.001999697	0.000146478	-	-	-
	C	0.020993243	0.017824339	0.005719016	0.000978752	0.000399978	-	-	-
(100/1,000)	A	0.008002613	0.005810845	0.003140910	0.001251374	0.000470614	0.000199253	0.000009297	-
	B	0.006601015	0.005752820	0.002071574	0.000593651	0.000169229	-	-	-
	C	0.005776997	0.004994537	0.001624271	0.000463340	0.000119813	0.000034688	-	-
(100/10,000)	A	0.002767401	0.002104010	0.001114075	0.000408485	0.000133992	0.000050178	0.000017436	0.000004507
	B	0.002361174	0.001986837	0.000729372	0.000210824	0.000057625	0.000014757	0.000007988	0.000001332
	C	0.001938445	0.001727339	0.000545314	0.000147825	0.000030978	0.000006611	-	-
(100/100,000)	A	0.000689209	0.000617563	0.000298223	0.000123782	0.000048390	0.000017220	0.000006055	0.000002026
	B	0.000648830	0.000529093	0.000185779	0.000060968	0.000016418	0.000005137	0.000001240	0.000000062
	C	0.000655394	0.000559908	0.000178669	0.000041433	0.000010679	0.000002051	0.000000464	-

Figure 8: Standard deviation of marginal distribution for different (number of runs/number of trains)-combinations of the at-grade junction

(# runs/# trains)	m	Queue length							
		0	1	2	3	4	5	6	7
(5/10)	A	0.032913894	0.052318268	0.024807431	0.002282239	-	-	-	-
	B	0.064726757	0.063852943	-	-	-	-	-	-
	C	0.104279705	0.055992178	-	-	-	-	-	-
(10/10)	A	0.081176448	0.074285875	0.037663662	0.001076343	-	-	-	-
	B	0.079977604	0.065426065	0.019985054	-	-	-	-	-
	C	0.096517277	0.053822014	0.018289871	-	-	-	-	-
(100/100)	A	0.027977381	0.020315260	0.010213914	0.004623327	0.001868092	0.001765828	0.000388498	-
	B	0.019766520	0.016825589	0.004857131	0.001240463	0.000312719	-	-	-
	C	0.021165964	0.017743796	0.005120951	0.001093533	0.000719960	0.000148676	-	-
(100/1,000)	A	0.007737485	0.006396292	0.002930319	0.001109086	0.000362368	0.000110412	0.000011784	0.000008879
	B	0.006595361	0.005627620	0.001640630	0.000409024	0.000087157	0.000076515	0.000034175	-
	C	0.006113524	0.005181027	0.001602268	0.000454732	0.000148379	0.000046532	-	-
(100/10,000)	A	0.002643559	0.002049136	0.000949406	0.000359729	0.000135155	0.000042220	0.000015224	0.000002345
	B	0.001625148	0.001449281	0.000536924	0.000159006	0.000034377	0.000006772	0.000000478	-
	C	0.001955141	0.001702051	0.000566031	0.000140617	0.000033864	0.000011174	0.000001082	-
(100/100,000)	A	0.000824025	0.000632156	0.000276804	0.000110013	0.000037446	0.000131336	0.000004595	0.000002147
	B	0.000586230	0.000538553	0.000154919	0.000034951	0.000010188	0.000002395	0.000000511	0.000000002
	C	0.000511838	0.000460812	0.000162198	0.000044987	0.000010006	0.000002302	0.000000725	0.000000280

Figure 9: Standard deviation of marginal distribution for different (number of runs/number of trains)-combinations of the grade separated junction