TU WIEN Informatics

Enhancing Train Network Simulation Model with a Moving Block System

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Data Science

eingereicht von

Fabian Hartmann, Bsc
Matrikelnummer 11824496

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: DI Dr.techn. Nikolas Popper
Mitwirkung: DI Dominik Rothschedl
Daniele Giannandrea, BSc MSc
DI Matthias Rößler
DI.Dr Martin Bicher

Wien, 29. April 2025

Fabian Hartmann　　Nikolas Popper

TU WIEN Informatics

Enhancing Train Network Simulation Model with a Moving Block System

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Data Science

by

Fabian Hartmann, Bsc
Registration Number 11824496

to the Faculty of Informatics

at the TU Wien

Advisor: DI Dr.techn. Nikolas Popper
Assistance: DI Dominik Rothschedl
Daniele Giannandrea, BSc MSc
DI Matthias Rößler
DI.Dr Martin Bicher

Vienna, April 29, 2025

Fabian Hartmann | Nikolas Popper

Erklärung zur Verfassung der Arbeit

Fabian Hartmann, Bsc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel" habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, haben ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT- Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 29. April 2025

Fabian Hartmann

Acknowledgements

I want to thank my supervisor, Nikolas Popper, for his guidance, expertise, and support throughout the development of this thesis. I also appreciate Matthias Rößler for his role as co-supervisor and for participating in our discussions.

A big thanks to my other co-supervisors: Dominik Rothschedl and Daniele Giannandrea. Our regular meetings every other week became something I actually looked forward to. Thank you for patiently listening to my sometimes meandering thoughts, offering vital insights on railway network simulation, and helping me navigate the technical challenges I encountered. Your thoughtful feedback and willingness to engage with my ideas, even the less developed ones, were invaluable.

To my parents, thank you for making my studies possible, both financially and emotionally. Your support never wavered.

And I couldn't have maintained my sanity without Anja, who listened to more talk about railway simulations than anyone should have to endure. Thanks also to Andrea, Matthias, and Alexandra for being there when I needed to vent, bounce ideas around, or just take a break from staring at simulation models.

Abstract

Modern railway networks are reaching their capacity limits. Advanced train protection systems like Moving Block aim to alleviate this issue by decreasing the headway between trains. This thesis extends the functionality of the macroscopic simulator RailwaySim to better capture the continuous nature of Moving Block systems.

RailwaySim and FiBlo (a reimplementation of the former) emulate a Fixed Block system by modeling the network through a graph of operational control points as nodes and segments between them as edges. Trains occupy these infrastructure elements during their travel, and capacity limits lead to queuing and secondary delays.

The thesis compares two methods for estimating the capacity limit of a segment and demonstrates their impact on line capacity consumption as well as delay propagation. The first method uses the number of tracks as a capacity measure, while the second uses the historic maximum number of trains observed on a segment. Applied to the Meidling-Wiener Neustadt corridor, the track-based approach resulted in higher capacity consumption with 79% versus 70% with the second approach.

MoBlo refines the FiBlo model by splitting the segments into micro-segments and integrating train movement. The micro-segments allow MoBlo to reserve a movement authority based on the position and braking distance of a train (Moving Block). The train movement parameters—acceleration, deceleration, and maximum speed—are calibrated for each train category by minimizing the error between simulated and scheduled travel durations. The resulting Mean Absolute Relative Error ranges from 13.5% to 33.6% and highlights the limitations of this simple approach.

A sensitivity analysis on the micro-segment length in MoBlo showed that reducing them from 4096m to 64m decreases capacity consumption from 87% to 73%, although computational demands rose by a factor of 60. MoBlo only employs the first method to estimate the capacity of a segment (number of tracks) and results in lower capacity consumption than FiBlo with the same capacity measure.

Kurzfassung

Moderne Eisenbahnnetze stoßen an ihre Kapazitätsgrenzen. Fortschrittliche Zugbeeinflussungssysteme wie Moving Block versuchen, dieses Problem zu entschärfen, indem sie den Zugfolgeabstand verringern. Diese Arbeit erweitert die Funktionalität des makroskopischen Simulators RailwaySim, um das kontinuierliche Verhalten von Moving-Block-Systemen besser zu erfassen.

RailwaySim und FiBlo (eine Neuimplementierung des ersteren) emulieren ein Festblocksystem, indem sie das Netzwerk durch einen Graphen von Betriebsstellen als Knoten und Segmente dazwischen als Kanten modellieren. Züge belegen diese Infrastrukturelemente während ihrer Fahrt, und Kapazitätsgrenzen führen zu Engpässen und Verspätungen.

Die Arbeit vergleicht zwei Methoden zur Schätzung der Kapazitätsgrenze eines Segments und zeigt deren Auswirkungen auf den Streckenkapazitätsverbrauch sowie die Verspätungsausbreitung. Die erste Methode verwendet die Gleisanzahl als Kapazitätsmaß, während die zweite die historisch beobachtete maximale Zugbelegung eines Segments heranzieht. Angewandt auf die Strecke Meidling-Wiener Neustadt führte der gleisbasierte Ansatz zu einem höheren Kapazitätsverbrauch mit 79% gegenüber 70% beim zweiten Ansatz.

MoBlo verfeinert das FiBlo-Modell, indem es die Segmente in Mikrosegmente unterteilt und die Zugbewegung explizit simuliert. Die Mikrosegmente ermöglichen es MoBlo, eine Fahrterlaubnis auf Basis der Position und des Bremswegs eines Zuges zu reservieren (Moving Block). Die Zugbewegungsparameter – Beschleunigung, Bremsrate und Höchstgeschwindigkeit – werden für jede Zugkategorie kalibriert, indem der Fehler zwischen simulierten und geplanten Fahrzeiten minimiert wird. Der resultierende mittlere absolute relative Fehler (MARE) liegt zwischen 13,5% und 33,6%, und verdeutlicht die Limitierungen dieses einfachen Ansatzes.

Eine Sensitivitätsanalyse zur Länge der Mikrosegmente in MoBlo zeigte, dass eine Reduzierung von 4096m auf 64m die Kapazitätsauslastung von 87% auf 73% senkt, jedoch der Rechenaufwand um das 60-fache stieg. MoBlo verwendet nur die erste Methode zur Schätzung der Kapazität eines Segments (Anzahl der Gleise) und resultiert in gerninger Kapazitätsauslastung als FiBlo mit demselben Kapazitätsmaß.

Contents

CHAPTER 1

Introduction

1.1 Problem Statement

With modern railway networks reaching their capacity limits, ensuring efficient and reliable train operations becomes increasingly important [SP21].

Train protection systems are important for safety and prevent accidents, and they can also influence capacity consumption by affecting the train headway between two trains [Mas22].

Major parts of the Austrian railway network currently employ Punktförmige Zugbeeinflussung (PZB), a Fixed Block system for train protection. The system divides the railway line into fixed, discrete signal block sections, where only one train is permitted to occupy a signal block section at any given time. While this system ensures safety, it limits the number of trains that can be on the railway line simultaneously, which might cause inefficiencies compared to more advanced train protection systems.

Moving block systems, such as European Train Control System (ETCS) Level 2, a part of the broader European Rail Traffic Management System (ERTMS), promise improvements by dynamically calculating a safe "movement authority" zone for each train. This allows trains to run more closely together while still having the necessary safety margin. Using such a system could improve track capacity and reduce delays by not being constrained to fixed signal block sections.

A key task in freight and passenger train operation is scheduling traction units to trains. RailwaySim [RWJ+20], an agent based model for train circulation planning, employs a macroscopic approach to train network modeling. The network consists of Operational Control Point (OCP) as nodes and segments between them as edges. OCPs are groupings of infrastructure elements with shared traffic or operational functions, such as stations, signals, or junctions. These points serve as time measurement points in the simulation and in real-world schedule making, where arrival and departure times are specified at these locations. The model uses *Stop* and *Drive* tasks to simulate train movements between these OCPs without explicitly modeling specific train protection systems. Each segment connecting two OCPs is treated as a single infrastructure element with limited segment capacity, which represents the maximum number of trains that can occupy that segment simultaneously.

While the simulation could model the signal block sections of PZB, it is unable to capture the continuous nature of ETCS Level 2's Moving Block system. The model is therefore limited in its ability to represent the system's operational constraints.

These limitations show the need to extend the existing agent based model to incorporate finer-level details of the Moving Block system. The goal is to enable more accurate simulation and better support for infrastructure planning and operational decision-making.

1.2 Contributions

This thesis extends the simulation model by dividing the current OCP-to-OCP segments [RWJ+20] into multiple smaller micro-segments. These micro-segments are significantly smaller than the traditional signal block sections and are not intended to replicate them but to better capture the finer dynamics of train movement under a Moving Block system. Trains will be allowed to reserve multiple micro-segments, dynamically calculating a "movement authority" based on their speed and braking distance, simulating the continuous adaptability of a Moving Block system in a discretized manner. While RailwaySim does not explicitly model train speed profiles but assumes trains need a fixed amount of time between two OCPs, the Moving Block system requires more detailed modeling of train behavior, including acceleration and deceleration, to accurately calculate movement authority.

Additionally, this thesis aims to analyze the impact of Fixed Block and Moving Block system on railway network capacity consumption and delay propagation.

To evaluate the effectiveness of the extended simulation model, the following metrics will be applied:

- Travel Time Accuracy: The accuracy of simulated train travel times will be assessed using Root Mean Squared Error (RMSE), comparing simulated times to expected values.
- Line Capacity Consumption: An adapted UIC 406 method [UIC13] compresses timetables based on track occupation and enables a comparison of line capacity consumption.

1.2.1 Research Questions

The research questions to be answered are:

- RQ1: How accurately does the simulated acceleration and deceleration behavior replicate travel times between stations?
- RQ2: How does the integration of a Moving Block system in the extended agent based model affect delay propagation compared to the Fixed Block system in the original model?
- RQ3: How does the adoption of a Moving Block system, as simulated in the agent based model, impact railway capacity consumption compared to the Fixed Block system?
- RQ4: How sensitive is the extended agent based model to variations in key parameters like micro-segment length?

1.3 Methodology

The methodological approach to achieving these aims comprises several key phases:

1. **Literature Review and State of the Art**
   Conduct a literature review to understand existing methodologies and tools for simulating train protection systems, focusing on strengths, limitations, and comparative studies involving ETCS.
2. **FiBlo-Model: Re-implementation of the Existing Model**
   Develop a new simulation model that mimics the functionalities of the existing agent based model (RailwaySim) based on OCP-to-OCP segments.
3. **MoBlo-Model: Extension with Moving Block**
   Extend the FiBlo-Model model by incorporating detailed Moving Block systems (mimicking Moving Block ETCS L2, formerly L3)
   Calibrate simplified train movement models using timetable data.
4. **Model Validation**
   Validate the acceleration and braking behavior extension by comparing simulated travel times between stations with real-world timetables.
5. **Comparative Analysis**
   Examine the impacts of integrating Moving Block system on delay propagation and capacity consumption. Analyze the changes introduced, listing both improvements and potential limitations.

CHAPTER 2

Preliminaries

Railway networks are complex systems involving multiple resources like infrastructure, vehicles, and personnel. Among these, locomotives, or traction units, are especially valuable components [BBP+23]. Circulation planning is the process of assigning traction units to specific trains and routes, and efficiently planning this is important for railway operations. There is an important balance between minimizing costs and maintaining robustness against delays and disruptions [RWJ+20].

Highly efficient systems can minimize resource usage, costs, and idle times, but they often become more vulnerable to primary delays, potentially leading to network-wide disruptions and delays [BBP+23]. In a schedule without any buffer times, even small delays may cause significant delay propagation. Trains are interconnected, as they require the same resources (infrastructure or traction unit), which can cause delays to cascade through the system [RWJ+20].

Simulation models try to replicate the behavior seen in real-world scenarios and can be used to assess and improve the robustness of railway schedules. These models, such as RailwaySim [RWJ+20], simulate train movement, including delays and their propagation. This gives valuable feedback to planners as it allows them to play through different scenarios, analyze their outcomes, adjust, and repeat as necessary. By combining optimization techniques with simulation models, it is possible to optimize a schedule, run the simulation to determine the robustness, and use the result to guide the optimization into efficient and resilient solutions [RWJ+20].

The subsequent sections explore key components of railway simulation. Train Protection Systems are important to avoid accidents and determine how trains utilize the infrastructure. Agent Based Modeling and Discrete Event Simulation provide methodological frameworks for representing and simulating the complex interactions within railway systems. Finally, the section on train network modeling contains an overview of existing simulators and explores the differences between macroscopic and microscopic approaches.

2.1 Train Protection Systems

Train protection systems are important for security and for overcoming the difficulties presented by increasing rail traffic and higher train speeds. These systems are designed to prevent collisions by maintaining a safe distance between trains and managing train speeds. Various approaches to regulating the distance between trains have been developed to ensure safe and efficient travel.

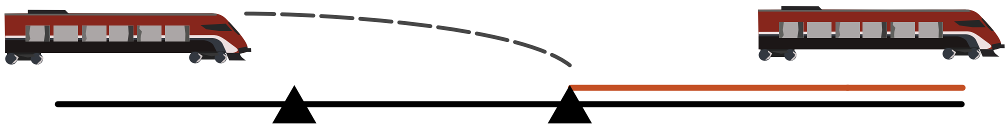
The **Fixed Block system** is a widely used method where the track is divided into discrete signal blocks. Each signal block can either be occupied or free, which is monitored by trackside equipment like axle counters. A train is allowed to enter a signal block only if it is declared free from other trains [Pac21]. In this system, the follow-up train must not enter an already occupied signal block, even if the leading train is nearing the end of the signal block, which can result in less efficient use of track capacity.

In contrast, the **Moving Block system** maintains a *movement authority* around each train that incorporates the position as well as the braking distance. These block boundaries are continuously adjusted as the train moves and prevent other trains from entering this dynamically shifting block. This allows the following train to operate based only on its speed and the headway to the leading train, potentially enabling the following train to brake later and more efficiently compared to the Fixed Block system. This method results in a dynamic and variable distance between trains depending on their speed and braking characteristics [Pac21].

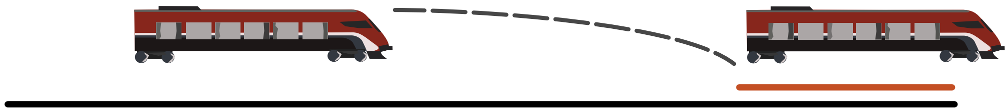
As illustrated in Figure 2.1, the Moving Block system can enhance track capacity by allowing the following train to utilize the track more efficiently, compared to the Fixed Block system where the train cannot enter an occupied block section even if the leading train is approaching the end of the corresponding block section. This capability of the Moving Block systems helps decrease track capacity consumption compared to fixed blocks.

Relative Braking Distance is a concept that involves maintaining a distance based on the difference between the braking distances of successive trains. Unlike the Moving Block system, where the following train must be able to stop at the position the leading train currently occupies, the relative braking distance method only requires the following train to be able to stop before the stopping point of the leading train. The headway is determined by the emergency braking difference between the two trains. This concept is primarily explored in research on virtually coupled train convoys traveling at the same speed. The approach is innovative but often less practical compared to the Fixed and Moving Block systems [Pac21].

Train protection systems such as PZB and ETCS are designed to prevent train collisions using these distance approaches and ensure the safe movement of trains across a network. They regulate train speeds and automate braking processes.



(a) Fixed Block system



(b) Moving Block system

Figure 2.1: Comparison of breake points for the following train in Fixed and Moving Block systems. Triangles indicate signal positions and red lines visualize active reservations for the preceding train.

2.1.1 PZB

PZB is an Inductive Train Control system widely used in Austria and Germany. The original version was limited to point-based transmission and monitoring, but the introduction of computer-controlled vehicle devices allows a partial continuous monitoring. PZB monitors train movement at critical sections, like main signals, distant signals, and speed restrictions.

Main signals (Hauptsignale) mark the start of a signal block section, and only one train is permitted to enter at any given time. The main signal indicates whether a train must stop or can proceed. Each main signal is accompanied by a distant signal (Vorsignal) that serves as an indicator to the train driver about the status of the main signal ahead. The PZB system enforces that these signals are respected and monitors the train's speed profile to verify that appropriate actions are taken. If a train driver ignores the stop indicating distant signal for too long, the PZB triggers automatic emergency braking to prevent violations and train crashes [Mas22].

The braking capabilities and braking distances can differ between trains. PZB therefore categorizes them into three types: Type O (Oberer or upper), Type M (Mittlerer or middle), and Type U (Unterer or lower) [Pac21]. Trains in Type O have the highest braking performance, while Type U trains have the longest braking distance. The speed restrictions are different for each train type, for example, trains with high decelerations require less restrictive speed limits.

To transmit information from trackside to the train itself, PZB uses track magnets at critical points along the line. A 1000 Hz magnet is positioned at the distant signal, a 500 Hz magnet approximately 260 meters before the main signal, and at the main signal itself, there is a 2000 Hz magnet. If a train drives over an active magnet, different protocols are initiated based on the frequency.

1000 Hz Magnet: This magnet is located near the distant signals and is active when the signal indicates a stop ahead or a speed restriction of up to 60 km/h (in specific cases, up to 70 km/h) [Mas22]. Upon activation, the train driver has to acknowledge this signal by pressing a vigilance button within four seconds. If the driver fails to do so, emergency brakes are activated. Even if acknowledged, the onboard system monitors the speed profile of the train to ensure it decelerates. For example, for type O trains, the train has to decelerate to 85 km/h within 23 seconds or else the emergency brakes are triggered [Mas22] [Pac21].

500 Hz Magnet: Approximately 260 meters before the main signal is the 500 Hz magnet. If the main signal indicates a stop, this magnet enforces a stricter deceleration profile. If a type O train traverses over an active 500 Hz magnet, the train has to reduce its speed from 65 km/h to 45 km/h within 153 meters, similar speed-profiles exist for the other train types. Once again, if the train fails to do so, emergency brakes are automatically deployed [Mas22].

2000 Hz Magnet: The last magnet is placed directly at the main signal and if it is active as the train passes over it, the train immediately stops by applying the emergency brakes [Mas22].

In summary, PZB is a train monitoring system with track-side magnets with different frequencies and onboard systems monitoring speed profiles. It tries to mitigate the risk of human error and ensures that trains respect distant and main signals by automatically applying brakes if necessary [Pac21].

2.1.2 ETCS

Every nation has its own system for signaling and train protection, which makes operating train services across multiple nations complicated. The ETCS, a component of the European Rail Traffic Management System (ERTMS), is designed to replace and unify the diverse national train protection systems [Pac21].

ERTMS consists of ETCS, which deals with train protection, and GSM-R, a specialized radio system for data communication between trains and trackside control centers (radio block centers) [BBGS12].

ETCS has two levels with varying degrees of complexity and operational capabilities.

ETCS Level 1 is an interoperable pointwise train protection system that can be implemented on top of national signaling infrastructure. Eurobalises can transmit data from trackside to the train and contain information like speed restrictions or signaling aspects [Mas22]. The information is processed by the onboard equipment and allows for continuous speed monitoring. Additional data can be transmitted through either infill elements or GSM-R, which allows the onboard equipment to update speed restrictions even between eurobalises. While the national trackside signals remain in place, ETCS Level 1 also includes a driver machine interface (DMI) that displays the information to the driver with an in-cab display [Mas22]. The train supervision is strongly related to PZB90 [Pac21].

ETCS Level 2 extends the system's capabilities by incorporating constant communication between trains and trackside control units (radio block center) via GSM-R radio. The Radio Block Centre (RBC) transmits the Movement Authority (MA)s to trains based on the train route and blocked segments. Trackside signals are not required anymore as the MA is communicated to the train and displayed to the driver via the in-cab display. Although fallback signals may be present for non-ETCS equipped trains or as backup [Pac21].

The RBC generates and updates MAs based on track occupations and the train's position, which is determined through Eurobalises and onboard odometry (distance measuring equipment) [Pac21]. Trackside systems such as axle counters still monitor if trains have fully cleared a section. The system verifies that the entire train has left the section and that no cars have become detached by counting the axles as the train enters and exits.

With the release of CCS TSI 2023, ETCS Level 2 has incorporated functionalities previously associated with Level 3. In the previous modes, trackside equipment like axle counters were used to determine the status of a signal block section. With the introduction of train integrity monitoring, where trains ensure they have no detached cars, there is no need for axle counters to determine the status of a signal block section. Position and train integrity monitoring are enough to ensure a section is free of any train cars.

This allows for two additional modes within Level 2:

1. *Hybrid Train Detection (HTD)*: This mode splits the physical signal block sections into virtual subsections (VSS) [KOF24]. Because trains monitor their position and integrity, virtual subsections can be marked free even without trackside equipment and allow trailing trains to move into the corresponding part of track just after the leading train leaves it. The physical block sections can still serve as a fallback for trains without train integrity monitoring [Mas22].

2. *Moving Block*: In this mode, trains equipped with continuous position reporting and integrity monitoring can operate based on the Moving Block principle. This reduces the need for trackside equipment and potentially increases network capacity [Pac21].

Both HTD and Moving Block systems require trains to communicate their location and integrity to the RBC via GSM-R [Pac21]. While the Moving Block system promises the best performance for capacity consumption, the train integrity monitoring of freight trains is still a challenge [Pac21]. HTD can address some of the difficulties by allowing trains with integrity monitoring to utilize the benefits of virtual subsections while it provides the fallback of physical signal block sections for trains without it.

In summary, ETCS can bring interoperability to the European rail network and also enhance efficiency through the different levels of functionality.

2.2 Agent Based Modeling and Discrete Event Simulation

Discrete Event Simulation (DES) is a simulation framework where events are processed at discrete timestamps. It is good at modeling systems where state variables change (nearly) instantaneously at specific points in time [Law15]. In the context of railway simulation, the departure and arrival of trains at a station can be represented as events changing the state of the train and infrastructure at a discrete time.

DES has a clock that keeps track of the current time within the simulation [Law15]. There are two common ways to advance the clock within the simulation, either through the next-event time advance (NETA) or the Fixed-Increment Time Advance (FITA) approach. The former jumps the simulation clock from one event to the next event, skipping over inactive periods. For example , with one event scheduled for 10:00 and the next for 11:30, the clock does not consider the time in-between and advances the time to 11:30 after completing the first event to process the next one. This approach is effective at modeling systems with varying time intervals between events. The FITA approach advances the clock in fixed time intervals. Even though historically Agent Based Models are mostly FITA, the next-event time approach can improve computational efficiency [Law15].

Agent Based Modeling (ABM) is a simulation framework that focuses on modeling individual entities. Agents are self-contained, identifiable individuals within the system. They sense their environment (including other agents) and use this information to make decisions and act upon the environment [MN05]. Even if the rules of a single agent are simple, the interaction with others can result in complex system-level behavior [Mac16]. ABM is used across various fields to simulate the interaction of autonomous entities over time [MK15].

Interestingly, ABM can be viewed as a variation of DES [Law15]. Agents evaluate their environment and perform their actions at discrete points in time, which matches the DES framework.

2.3 Train Network Modeling

Simulations are an important tool for planning, designing, and evaluating railway systems. They can provide insights into infrastructure development, timetable design, and conflict resolution [Qua11] [RH04]. For example, a model can test a timetable for robustness by simulating the railway network with primary delays. With a fast simulation model, this process can be repeated within an optimization loop, where one component tries to optimize an objective function and the simulation tests for robustness to verify feasibility in a setting with delays [RWLP18].

2.3.1 Levels of Detail in Train Network Models

Different train simulation models might have different objectives and therefore require a different level of detail in the railway network. A high level of detail allows for accurately modelling railway operations, but computational demand makes it inefficient for long-term planning.

- **Macroscopic Models:** Macroscopic models represent the railway network at a high level of abstraction, with nodes typically denote a stations or junctions and links representing the tracks in between. They are efficient at large-scale simulations but limited in their accuracy [Qua11]. Additionally, they require less infrastructure information, which makes maintenance easier [ZBW+18].
- **Mesoscopic Models:** Mesoscopic models are between macro and microscopic models and try to find a balance between modelling everything necessary while still having the benefits of a macroscopic approach [Qua11].
- **Microscopic Models:** Microscopic models provide the highest level of detail and can capture infrastructure information like speed limits, gradients, curvature radii, and the exact position of signals. They can also support train characteristics like maximum speed, length, weight, as well as acceleration and deceleration behavior. These models typically compute train movements using Newton's laws of motion. This includes calculating the air and rolling resistance, modeling the acceleration behavior of traction units, and combining that to derive the position and speed of a train [SP21]. With this level of detail, the model is able to calculate running times between stations but also analyze the differences of varying signaling systems. The high level of detail leads to accurate results but also high computational costs [Qua11].

Simulation models usually represent train networks as graphs. Table 2.1 gives an overview of what the different abstraction layers use for their nodes and links.

Table 2.1: Infrastructure levels and their respective nodes and links in a graph representation [CM12].

Infrastructure Level	**Node**	**Link**
Macroscopic	Station, Junction	Line between Station/Junction
Mesoscopic	Route Node	Railway Line, Segments
Microscopic	Points, Crossing, Signal	Signal Block Section

2.3.2 Synchronous vs. Asynchronous

There are two different processing techniques to simulate train movement through a network. In a **synchronous simulation**, all components are updated at the same time. The events are processed in the same order as they are in reality. At each step, the position, speed, and status of every train, as well as the signaling aspects, are updated, making it suitable for visualizations [CM12]. This can be implemented through DES, with a simulation clock and discrete updates to trains and signaling.

The **asynchronous simulation**, on the other hand, processes train journeys one after the other. The train with the highest priority gets simulated first in its entirety, before introducing the next train into the simulation [CM12]. There is no simulation clock, and the newly introduced trains model their journey based on the already included trains but have no idea of lower-ranking trains. Therefore, trains with high priority, like high-speed intercity, are not influenced by trains with lower priority, for example, freight. Asynchronous simulations are well suited for conflict-free timetable construction [Qua11].

2.3.3 Notable Simulation Tools

- **RailSys:** RailSys [RH04] is a prominent commercial simulation tool developed for the detailed analysis and planning of railway operations. It is a microscopic model with a highly detailed infrastructure plan that includes signal and switch positions as well as track gradients. It uses a time-driven approach to derive train speed and position based on resistance and force calculations. RailSys contains a tool for capacity analysis depending on the signaling system and track configuration[Ver20]. However, it suffers from the computational drawback of microscopic models, where simulations are slower because of the high level of detail. This limits its application in scenarios where multiple iterations are necessary [Qua11] [SRK+24]. Additionally, because it is a commercial model, the functionalities are not easily extendable [Ver20] [CML18].
- **OpenTrack:** OpenTrack [Hü02] has a similar feature set to RailSys but also suffers from the same limitations. The model is designed to update the position and speed of a train in fixed intervals, with 1 to 5-second time steps showing good results [Hü02]. OpenTrack also has a headway calculator to analyze the minimal time between two consecutive trains [KOF24].
- **NEMO:** NEMO (Network Evaluation Model by ÖBB) is a macroscopic simulation model and is designed to evaluate different infrastructure scenarios for passenger and freight traffic. It represents the railway network with a high-level graph where nodes represent major elements such as stations and junctions [Qua11] [RH04].
- **RailwaySim**: RailwaySim [RWLP18] [BBP+23] [SRK+24] is an agent based macroscopic simulation tool, designed to evaluate the robustness of train schedules and optimize traction unit circulations. A train schedule is represented as a list of tasks,

where each task gets executed through a series of events by a traction unit (the agents), see section 3.1 for more details.

- **EGTRAIN:** EGTRAIN [Qua11] is a synchronous agent based microscopic simulation tool with a focus on open architecture. It attempts to combat the problems of RailSys and OpenTrack, where extensions to the existing model are hard to implement. This allows it to communicate with external software, like optimization tools [Qua14].

- **PULSim:** PULSim [CML18] is another approach to open architecture. It incorporates a scripting language to interact with the simulation. Python scripts allow for user-defined workflows and integration of new functions into the existing simulation tool.

- **PROTON (formerly PRISM):** PROTON (Punctuality and Railway Operation Simulation) [Sip23] [ZBB+19], developed by DB Analytics (Deutsche Bahn AG), is a macroscopic simulation tool. Similar to RailwaySim, PROTON does not model movement inside stations but only the arrival and departure events. It allows the schedule to include supplement times that can be used in simulations to compensate for delay if needed. PROTON's efficiency makes it ideal for large-scale network evaluations, for example, how scheduled maintenance of a track can disrupt traffic.

2.3.4 Limitations of Existing Tools

RailSys and OpenTrack are powerful and commonly used commercial railway simulators. They are good at providing detailed simulations based on their microscopic view of infrastructure and signaling systems. But both tools are inefficient at simulating large-scale networks or multi-day time-windows because of their computational requirements [Qua11] [CM12] [HB22]. In optimization scenarios, multiple simulations are required to tune and verify the results, RailSys and OpenTrack lack the speed to make this process efficient [SRK+24].

Additionally, RailSys and OpenTrack are commercial tools and not freely available to modify and extend. EGTRAIN, PROTON, and NEMO, are also not publicly released, which limits the ability to make modifications and makes it complex to improve upon previous work.

A challenge all simulation tools face is the calibration of train movement. Even though RailSys derives the travel duration based on detailed calculations, a study showed that deceleration values can deviate significantly from real-world behavior [RT20]. RailSys used a 22% higher acceleration and a 48% higher deceleration than measured. This highlights the need for validation to ensure the simulation replicates train traffic observed in reality.

2.4 Simulating Different Signaling Systems

Simulation tools like RailSys and OpenTrack are able to model different signaling systems. Analyzing simulation results with the same schedule and infrastructure but a different signaling system, illustrates the influence a train protection system has on capacity consumption and delay propagation. This allows the exploration of how advanced systems like ETCS Level 2 Moving Block might improve train operations in comparison traditional systems like PZB.

Simulations of ETCS Level 2 and HDT (formerly Hybrid Level 3) systems have been shown to significantly reduce capacity consumption compared to legacy systems [Jan19]. This study used RailSys on a Dutch railway network from Utrecht to Den Bosch. This part of the network is interesting due to the mixed traffic (Intercity, Sprinters, and freight trains). The simulations showed a decrease in capacity consumption from 90% for Level 2 to 83% using ETCS Level 2 with Moving Block (formerly Level 3). Integrating Automatic Train Operation (ATO) can further improve running times and capacity consumption [Ver20].

Analyses of the Stockholm commuter train network could replicate the improvements gained by switching from the Swedish legacy system ATC2 to ETCS Level 2 and ETCS Level 2 HTD (with the current track-side train detection). ATC2 had a capacity consumption of 51%, while ETCS Level 2 used 46% and Level 2 HTD 37%.

ETCS Level 2 HTD uses virtual subsections to further split existing signal blocks. There are also studies exploring the optimal way of setting up these subsections in Level 2 HTD. Modifying the subsection lengths, for example, shortening them near stations, can lead to capacity improvements [KOF24].

In contrast to the other studies, which showed improvements when moving from a legacy system to ETCS Level 2, transitioning from PZB to ETCS Level 2 in Vienna, Austria, may increase train headways due to less optimal braking curves under ETCS. Although implementing Level 2 HTD or optimizing signal block section lengths can improve headways and thus capacity [WS20]. A simulation study on the line between Vienna and St. Pölten showed a reduction in headway for Level 2 with Moving Block (formerly Level 3) compared to Level 2 with Fixed Blocks (151 to 67 seconds for passenger and 294 to 78 seconds for cargo), with the potential of doubling the number of high-speed trains operating on these lines [SP21].

Knutsen et al. [KOF23] provided an overview of the academic interest in ETCS Level 2 with and without HTD, showing the current research priorities and challenges in developing these systems.

CHAPTER 3

FiBlo: Re-Implementation of RailwaySim

RailwaySim [RWLP18, RWJ+20, BBP+23, SRK+24] is a macroscopic simulation tool for railway networks. The model represents the network as a graph, with nodes corresponding to OCPs such as stations and junctions and edges representing the segments connecting them [RWLP18]. This high level view is best fitted for large-scale planning. The re-implementation of RailwaySim (termed FiBlo) allows for extensions such as integrating Moving Block systems with the more dynamic movement authority [1].

3.1 Conceptual Model

RailwaySim models the rail network as a graph within a macroscopic simulation framework where OCPs function as nodes and OCP-to-OCP segments as edges.

The core of RailwaySim is a multi-agent model where the entities interact through tasks and events. The primary entities in this model are [RWLP18]:

- OCPs (i.e. stations, junctions) (Nodes)
- OCP-to-OCP segments (Edges)
- Traction Units or Trains (Active agents that execute tasks)

While moving through the network, a train occupies infrastructure elements such as nodes and edges. Each infrastructure element has a capacity limit, which is the maximum number of trains that can simultaneously occupy it. Any task that would exceed this limit must wait until capacity becomes available due to the completion of another task

[1] The source code for FiBlo and the MoBlo model can be found in the GitHub repository: https://github.com/e11824496/PyTrainSim

by a different train. This approach allows the model to simulate congestion and delay propagation within the network.

In this model, OCP-to-OCP segments (edges) do not necessarily correspond with signaling block sections, and a single edge may encompass multiple signaling block sections. Therefore, the effectiveness of the macroscopic graph representation depends on how closely these segments map to the signal blocks. Since the model reserves the underlying infrastructure element of an edge, this approach works best when the edges are not excessively long and closely relate to the signaling block section. For instance, in Austrian railways, the average length of these segments is 2.33 km [RWLP18], ensuring that this approach accurately reflects the operational constraints.

Determining the capacity of a segment is an important aspect of this simulation model. This thesis analyzes two approaches: the first method, **plan infrastructure**, uses the number of parallel tracks as the capacity limit. Because the segments do not directly correspond to the signal block sections, using this approach can underestimate the capacity. On long single-track railway lines between two OCPs, there might be multiple signal block sections, allowing several trains to be on this segment simultaneously, although in different signal block sections. The second method, **estimated infrastructure**, heuristically estimates the capacity as the maximum number of trains observed on this segment simultaneously. By using this approach, the model avoids the previous mistake of underestimating the capacity but might in turn overestimate it. Revisiting the long single-track line, the macroscopic model's representation of this as a single segment with capacity greater than one could permit multiple trains to depart simultaneously, whereas real-world operations would require maintaining proper headway between consecutive trains.

Within RailwaySim, each agent, such as a train or traction unit, has a list of tasks that represent the schedule. RailwaySim evolved from focusing only on *Drive* tasks [RWLP18], which represent the train movements between OCPs, to also include *Stop* tasks [SRK+24]. This additional task allows the model to capture effects that occur in stations, like extended dwell time due to increased passenger volume. Each task requires that all associated agents and infrastructure elements (segments and stations) are available before execution. The specific infrastructure is occupied during the execution of this task. The task list of an agent is able to capture every element of a schedule and can replicate the movement through *Stop* and *Drive* tasks within the simulation. The model can also be extended to include operational duties like train handover and breaks for train staff.

Tables 3.1a and 3.1b demonstrate how a train schedule is translated into a task list within the RailwaySim framework. Table 3.1a displays the schedule for a train traveling from Vienna (Meidling) to Wiener Neustadt, with intermediate stops at Liesing and passing through Mödling. Table 3.1b shows how this schedule is converted into a task list for the simulation.

Each task has a scheduled end time and a scheduled duration. The scheduled end time is the time the task is expected to finish according to the schedule, while the scheduled duration corresponds to the minimum time required for the corresponding task. Within the simulation, the simulated end time of a task can deviate from the scheduled one due to delays or blocked infrastructure. If the train is already delayed, the completion of the task is the maximum of the scheduled end time and the simulated start time plus the scheduled task duration. The train does not leave before the scheduled time but requires at least the scheduled duration for completion. Additionally, for any given task, primary delays can be added to simulate the impact of external factors, such as unexpectedly high passenger volumes or deviations in travel durations due to suboptimal acceleration and deceleration behavior.

Table 3.1: Translation of Train Schedule to Task List in RailwaySim and FiBlo.

(a) Train Schedule

Station	**Arrival**	**Departure**
Meidling	-	10:00
Liesing	10:10	10:12
Mödling (Pass Through)	10:20	10:20
Wiener Neustadt	10:45	10:50

(b) Corresponding Task List

No.	**Task**	**From**	**To**	**Sched. End**	**Duration**
1	*Stop*	Meidling	-	10:00	-
2	*Drive*	Meidling	Liesing	10:10	10 min
3	*Stop*	Liesing	-	10:12	2 min
4	*Drive*	Liesing	Mödling	10:20	8 min
5	*Drive*	Mödling	Wiener Neustadt	10:45	25 min
6	*Stop*	Wiener Neustadt	-	10:50	5 min

Primary delays refer to the unexpected extension of a process due to external factors not modeled within the simulation. They are an important part of simulating realistic train travel and enable planners to check for schedule robustness. Some primary delays may be caused by equipment failure, temporarily unavailable infrastructure, or simply high passenger flow. These delays can propagate through the network and affect other trains, creating secondary delays. Secondary delays originate when there are conflicts with other trains. A train might need an already fully occupied infrastructure element and therefore has to wait for the completion of a different task before continuing [RWJ+20]. To evaluate the impact of primary delays on a schedule, RailwaySim can extend the required duration for a task and simulate the effects on other trains and the secondary delay created by that interaction [BBP+23].

3.2 Event-Driven Approach

Each task (such as *Drive* and *Stop*) assigned to an agent is completed through the execution of several events. The original event-handling process is as follows [SRK+24]:

- **`Start` Event**: This event starts when it is time for an agent to begin a new task. It verifies that the required infrastructure element is available. If the infrastructure is free, the `AttemptEnd` event is scheduled and the infrastructure reserved. If the infrastructure is occupied, the `Start` event is rescheduled based on the next time the infrastructure is expected to be free.
- **`AttemptEnd` Event**: This event evaluates the availability of the next task's infrastructure when the current task is nearing completion. If the next task's infrastructure elements are free, the `End` event is scheduled. If any of the required infrastructure elements are blocked, the `AttemptEnd` event is rescheduled. This mechanism can lead to queuing and secondary delays within the simulation as a train has to wait for other trains to release infrastructure elements.
- **`End` Event**: This event advances the task list of the agents and schedules the `Start` events of the new current tasks. The infrastructure utilized by the current task is released.

The outcome of the simulation is a list of execution timestamps and their corresponding delays in relation to the planned time [SRK+24].

3.3 Adaptations in FiBlo

In FiBlo, the re-implementation of RailwaySim, some adaptations were made to simplify the process and allow extensions like Moving Block system.

These adaptations include:

- **Unified Event Mechanism**: Instead of having three separate events (`Start`, `AttemptEnd`, and `End`) per task, the new model uses a single comprehensive event to manage all transitions. This event first attempts to complete the current task by checking if the necessary infrastructure for the follow-up task is available (similar to `AttemptEnd`). If not, it schedules itself to retry. If successful, it frees the current infrastructure (similar to `End`), occupies the next needed infrastructure, and schedules the next event (similar to `Start`).

  This approach simplifies implementation and improves clarity, as all transitions are handled in one place. Additionally, it addresses a limitation of the previous design: the release of resources before the follow-up task could be reserved. For example, in the multi-event approach, the `AttemptEnd` event might release infrastructure after confirming the follow-up task is possible. But before the corresponding `Start` event is executed, another task could occupy the infrastructure, making the follow-up task impossible to execute, thus leaving the train without any occupied infrastructure. RailwaySim assigns priorities to different event types to address this issue, but this essentially replicates the behavior of the unified mechanism.

- **Active Agents**: Instead of relying on trains and traction units as the active agents, the adaptation only uses trains. This simplifies event execution as a given task is only associated with one agent (instead of potentially multiple traction units that are part of the train).
- **Simplified Traction Unit Handling**: In the RailwaySim model, multiple traction units could be associated with a task, requiring readiness checks for each traction unit before executing a task. The FiBlo model simplifies this by ensuring each train only starts when all preceding trains are complete. If two trains rely on the same traction unit, one must wait for the other to finish before starting. This method introduces an initial *Start-Up* Task to verify that everything is in place before proceeding with *Stop* and *Drive* tasks.
- **Multiple Infrastructure Elements per Event**: While RailwaySim was designed to utilize estimated infrastructure as a simulation basis, FiBlo can also work with plan infrastructure data. Because a scheduled traversal might not always have a corresponding segment in the plan network, FiBlo needs to find an alternative path. This issue commonly arises from directional OCPs (see Section 5.1.3 for details). In typical schedule data, only one directional OCP is specified while the OCP for the other direction is omitted, but both are present within the infrastructure data. To address this, FiBlo implements a pathfinding solution that identifies the shortest possible connection between corresponding OCPs and requires all segments along this path to be available before executing the movement.

3.4 Stochasticity

The only random component in FiBlo and RailwaySim is the primary delay injected into tasks. All other aspects of the simulation, including the execution of events, the queuing at infrastructure elements and the resulting secondary delays are the result of a deterministic process.

Primary delays can be introduced into the simulation through two methods:

- Statistical distributions: Synthetic delays can be drawn from probability distributions. These distributions can be parameterized separately for different task types or train categories.
- Historical data segregation: Primary delays can be extracted from historical operational data, separating them from secondary delays to create realistic delay patterns [SRK$^+$24].

Once primary delays are introduced, the subsequent event processing, including the calculation of secondary delays due to infrastructure conflicts, is purely deterministic. This methodology guarantees that for any given set of primary delays, the simulation will produce identical results across multiple runs.

3.5 Validation

To ensure that FiBlo can replicate the behavior observed in RailwaySim, the validation uses the primary delay and results produced by RailwaySim, as presented by Schwab et al. [SRK+24]. The experiment was conducted using train operations across Austria on a reference day. By using the same primary delay and network, this approach guarantees a controlled environment for evaluating whether FiBlo behaves consistently with the original.

3.5.1 Use of Primary Delay Data

In the original RailwaySim model [SRK+24], primary delays were extracted through a network-based disaggregation process. This approach used historical delay data and separated them into primary and secondary delays. The primary delays were then injected into the simulation to reproduce realistic conditions, allowing RailwaySim to derive the secondary delays. By using the same primary delay and network, FiBlo should be able to reproduce the same congestion, delay propagation, and therefore secondary delays.

3.5.2 Comparison of Results

Results indicate that the FiBlo model performs similarly to the RailwaySim model, with only minor deviations observed in 0.43% of trains.

These deviations were attributed to two factors:

1. **Race Conditions:** A common source of deviation was race conditions, specifically in cases where two trains were scheduled to depart from the same OCP in the same direction at the same time. Small differences in implementations occasionally led to variations in which train was able to access the shared segment first, resulting in minor differences in subsequent delays and travel times.

2. **Propagation of Deviations:** Minor initial differences, such as those arising from race conditions, occasionally led to further deviations downstream. For instance, if one train's travel time was impacted by race conditions at the start of its journey, other trains interacting with that train later in the network were similarly affected and deviated.

The re-implemented model (FiBlo) successfully replicated the delay propagation and congestion mechanisms of the original RailwaySim model. All observed differences between the models can be explained by either race conditions or by delay propagation effects arising from previous discrepancies. These findings show that FiBlo reproduces the original model's behavior under identical conditions, validating that congestion effects, delay propagation, and capacity constraints, are consistent with the RailwaySim model.

CHAPTER 4

MoBlo: Extension with Moving Block

RailwaySim and FiBlo are macroscopic models that represent the network as OCPs and segments between them. The existing models replicate a Fixed Block system, where railway lines are divided into discrete signal block sections that can be occupied by at most one train at any given time. This approach fails to capture the potential benefits and dynamic movement authority of Moving Block systems, such as those proposed in previous ETCS Level 3 specifications (CCS TSI, 2023) [KOFR25]. Moving Block systems calculate a safe block around trains, incorporating their position and braking distance. This block is continuously updated to reflect the position and speed of the train. This approach promises improved infrastructure utilization by reducing headway between trains.

Additionally, FiBlo requires the estimation of segment capacity that limits how many trains can be on a segment. FiBlo supports two methods, either based on plan infrastructure data or on historical traversals. Both approaches have their drawbacks and might under- or overestimate the capacity limit of a given segment and therefore misrepresent real-world constraints.

To address this limitation and more accurately represent train movement dynamics under a Moving Block paradigm, this chapter introduces the MoBlo model. MoBlo serves as an extension to the previous models, it still maintains the same event and task structure but introduces new ones to emulate a Moving Block system.

4.1 Conceptual Approach for Moving Block Implementation

The core concept of the MoBlo model is to divide the existing OCP-to-OCP segments into multiple smaller micro-segments, each spanning a few meters. Instead of updating the position and infrastructure reservations only after completing entire segments, MoBlo does so after every micro-segment. Reducing the length of the micro-segments increases the frequency of the updates and also the fidelity of track reservations. This approach creates a approximation of a true Moving Block system, where ideally the track would be divided into infinitesimally small sections [EW24].

By updating the train every few meters instead of every few seconds, MoBlo discretizes space rather than time and maintains the familiar segment and task structure established in RailwaySim. Unlike the original model where trains occupy entire OCP-to-OCP segments, MoBlo allows trains to reserve the smaller micro-segments. The event and task driven approach of RailwaySim stays, but the *Drive* task is adapted to capture the Moving Block dynamic and train movement. Each OCP-to-OCP segment is now represented by multiple *Moving Block Drive* tasks, one for each micro-segment. During task execution, the model calculates the safe speed for the train based on the available infrastructure ahead and the train's braking capabilities.

MoBlo also modifies which infrastructure elements are reserved. FiBlo only occupies the segment the train is currently on, whereas MoBlo reserves the micro-segment of the current position but also all micro-segments needed to come to a safe stop. This extended reservation emulates the movement authority found in Moving Block systems and is updated at every micro-segment. This approach allows for a simple representation of the Moving Block operations while maintaining computational efficiency. Shorter segments more accurately capture the continuous nature of an ideal Moving Block system but require greater computational resources, while longer segments reduce update frequency at the cost of model fidelity.

This approach requires several modifications to the existing FiBlo model:

- **Micro-Segments:** MoBlo introduces a new parameter, the micro-segment length, which dictates the splitting of OCP-to-OCP segments into smaller micro-segments. Each segment is divided into the minimum number of equidistant micro-segments, with at most the specified micro-segment length.
- **Infrastructure Capacity:** FiBlo supports two methods for estimating the capacity of a segment, either through plan infrastructure data (the number of tracks) or through historic estimates by using the maximum number of trains that occupied a segment simultaneously. Both approaches are insufficient for capturing real-world operations. MoBlo is designed to use the number of tracks as the capacity limit for a segment. This approach is not entirely accurate in the RailwaySim and FiBlo due to the coarse granularity of OCP-to-OCP segments. Long segments between

OCPs could accommodate multiple trains simultaneously with the appropriate headway, even on single-track lines. The finer granularity of the new model and the movement authority enable trains to only occupy the required parts of the segment and therefore allow multiple trains to be on the same segment unlike in FiBlo.

- **Train Dynamics:** The MoBlo model incorporates acceleration and deceleration profiles of trains. As a train accelerates to full speed after a stop, it requires a different amount of time to traverse the first micro-segment compared to the subsequent micro-segment. Additionally, the model needs to calculate braking distances and therefore uses the current speed to determine the movement authority for each train.

- **Task and Infrastructure Management:** Instead of occupying a single infrastructure element with each task, the new model dynamically calculates and reserves multiple micro-segments based on the train's current speed and braking distance. This change is required because a high-speed train traveling at 160 km/h may need to reserve 1 km ahead to ensure sufficient braking distance, while a slower regional train at 80 km/h might only need 250 m.

4.1.1 Comparison to RailwaySim/FiBlo

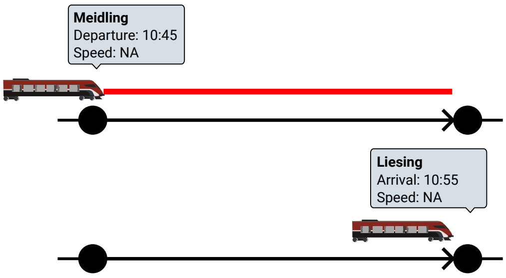
Figure 4.1 illustrates the fundamental differences between RailwaySim/FiBlo and MoBlo. The former models have one task for the traversal of the entire segment and do not model train speed or intermediate steps. The arrival of a train at the next OCP depends on a predefined minimum task duration and potential delays. In MoBlo, on the other hand, the train position, speed, and the reservations are updated after every micro-segment (each with its own associated task and arrival event). The speed profile of a train is modeled explicitly and is used to calculate the travel time for a micro-segment.

Within MoBlo, primary delays are still the only stochastic element of the simulation. The train movements and reservations are calculated deterministically. Applying the same primary delay to multiple MoBlo simulation runs results in identical outcomes. MoBlo is also able to apply the same primary delay as FiBlo, which enables a fair comparison between the two. One important implementation detail is that when a primary delay is applied to a *Drive* task in FiBlo, it is split equally among all micro-segments in MoBlo that compose the original OCP-to-OCP segment. This allows MoBlo to incorporate the same primary delay while spreading the effect to all corresponding *MoBlo Drive* tasks.

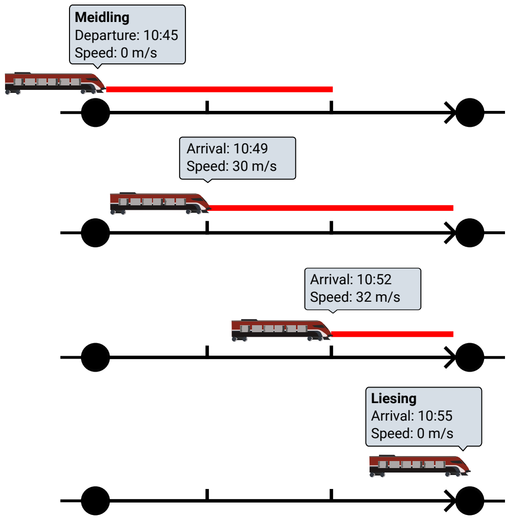
4.1.2 Speed Calculations and Infrastructure Reservations

When the train is about to enter a new micro-segment, MoBlo calculates and reserves the required infrastructure. This process can be described in a few steps:

1. **Infrastructure Availability Check**: The system first verifies if the micro-segment ahead has capacity left. If not, the train has to wait.



(a) RailwaySim/FiBlo Concept



(b) MoBlo Concept

Figure 4.1: Comparison of reservations and tasks between RailwaySim/FiBlo concept and the MoBlo Implementation. Circles indicate the OCPs and the smaller sections correspond to the newly introduced micro-segments. MoBlo captures train speed in addition to arrival times.

2. **Proposed Speed Calculation**: The movement calculations have an underlying assumption: the train wants to arrive at its destination as quickly as possible and therefore attempts to accelerate for the upcoming micro-segment. This results in the *proposed exit speed* and is based on the micro-segment length, the entry speed, and the train's acceleration capabilities.

3. **Forward-Looking Speed Adjustment**: To ensure that the train respects segment capacity limits and scheduled stops, the model looks ahead to upcoming micro-segments and evaluates if the *proposed exit speed* is possible. During these calculations, MoBlo propagates the *proposed exit speed* forward and the *safe exit speeds* backward. This recursive process involves:

    a) **Deceleration and Calculate Proposed Exit Speed:** Starting with the *proposed exit speed* from the previous micro-segment, the model wants to check if the train could come to a full stop before encountering a forced stop. A forced stop could either be a blocked infrastructure or a scheduled stop. Therefore, the *proposed exit speed* of this section is calculated by applying the train's brakes (deceleration).

    b) **Recursive Speed Adjustment:** The model checks if it can safely enter the next micro-segment with the *proposed exit speed* by applying the forward-looking speed adjustment to the next micro-segment. The result is a safe entry speed for the next micro-segment. The *safe entry speed* of a blocked micro-segment or a train stop is 0.

c) **Calculating Safe Entry Speed:** Given the *safe entry speed* of the next micro-segment, the *safe entry speed* for the current micro-segment is the maximum speed at which the train would still be able to decelerate to the safe speed at the end of the micro-segment.

4. **Reserving micro-segments**: Based on the calculated safe entry speeds, the system reserves the current micro-segment and a sufficient number of upcoming micro-segments for the braking distance. This reservation pattern simulates the "movement authority" concept in Moving Block.

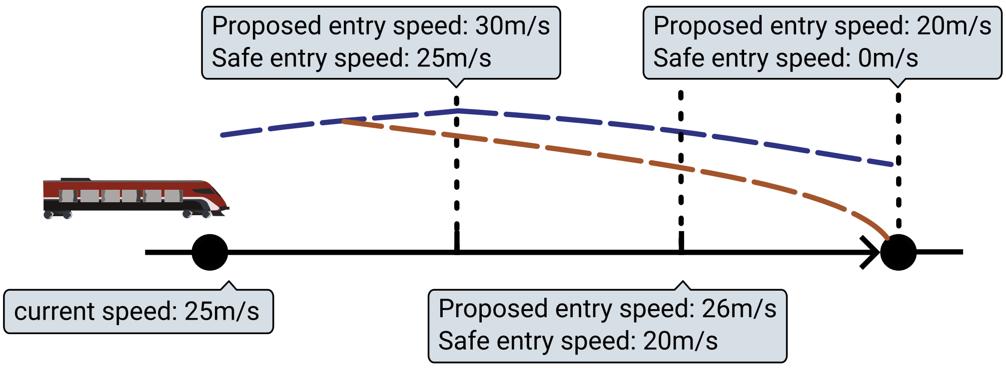


Figure 4.2: Forward-Looking Speed Adjustment for safe train movement. The blue dashed lines indicate the initially calculated proposed speeds. Through recursive adjustments, these proposed speeds are checked against infrastructure constraints to yield the safe speeds indicated by the red dashed lines.

Figure 4.2 illustrates the speed adjustment process. The train initially proposes to accelerate in the upcoming micro-segment with the overall goal of arriving at the destination as fast as possible. This proposed speed (blue dashed line) gets propagated forward until either the train has sufficient headway to come to a stop or needs to stop due to a constraint, such as a blocked infrastructure element or a scheduled train stop. In the illustrated example, the train cannot maintain its proposed speed because doing so would make it impossible to stop in time at the upcoming station. The recursive adjustment process calculates safe speeds (red dashed line) that are propagated backward. This speed guarantees that the train can come to a stop at the required point.

The model recalculates the time needed to traverse the current micro-segment based on the train's entry speed and the calculated safe exit speed.

4.2 Train Behavior Modeling

MoBlo requires a simple but realistic model for train movement to calculate travel times and braking distances. Complex models might require too much computational power and more information to accurately calculate the speed and distances for a train. This thesis proposes a model based on constant acceleration and deceleration.

The core equations for train movement are derived from kinematics:

$$\begin{aligned} v_{exit} &= v_{entry} + at \\ s &= \frac{v_{entry} + v_{exit}}{2} t \\ &= \frac{v_{entry} + v_{exit}}{2} \frac{v_{exit} - v_{entry}}{a} \end{aligned}$$

Where v_{exit} is the velocity at the end, v_{entry} is the initial velocity, a is acceleration (or deceleration), t is time, and s is the distance traveled.

From these basic equations, we derive the following formulas used in the MoBlo model:

1. **Acceleration and Braking Distance:**

$$s = \frac{v_{exit}^2 - v_{entry}^2}{2a} \tag{4.1}$$

   This formula calculates the distance required to change speed from an initial speed v_{entry} to a final speed v_{exit} with a constant acceleration a. For acceleration, $a = a_a > 0$ and $v_{exit} > v_{entry}$; for braking, $a = a_d < 0$ and $v_{exit} < v_{entry}$.

2. **Maximum Entry Speed:**

$$v_{entry} = \sqrt{v_{exit}^2 - 2a_d s} \tag{4.2}$$

   Given a distance s and a required exit speed v_{exit}, this formula calculates the maximum speed at which a train can enter a micro-segment and still be able to decelerate to the required exit speed with a constant deceleration $a_d < 0$.

3. **Exit Speed:**

$$v_{exit} = \sqrt{v_{entry}^2 + 2as} \tag{4.3}$$

   This formula calculates the exit speed of a train at the end of a micro-segment of length s, given an entry speed v_{entry} and constant acceleration a. For maximum exit speed, use $a = a_a > 0$; for minimum exit speed (deceleration), use $a = a_d < 0$.

The infrastructure model provides maximum speed information for each segment that trains within MoBlo must not exceed. To fully capture the limitations of different train categories, MoBlo incorporates a relative maximum speed variable that restricts the maximum speed of a train relative to the segment's maximum speed. A freight train might have limited braking capabilities and is therefore restricted to a different maximum speed than a passenger train. For example, PZB categorizes trains according to their braking rate into three groups and restricts their speeds differently (see Section 2.1.1). Additionally, even the fastest trains might not operate at the maximum allowed speed because of energy efficiency considerations or other limitations.

With those three train movement parameters: acceleration, deceleration, and relative maximum speed, MoBlo can generate speed profiles for trains and calculate the travel time as well as braking distance. The model does not differentiate between normal braking and emergency braking and might therefore overestimate the braking distance for the movement authority.

4.3 Advantages

MoBlo offers some advantages over FiBlo:

- **Use of Infrastructure Plan:** RailwaySim and FiBlo utilize two different segment capacity methods, but neither can handle long segments where two trains can traverse the segment at the same time but with headway between them. The MoBlo model is able to utilize infrastructure plans directly and mitigates this issue by allowing trains to reserve only part of the long OCP-to-OCP segment. As a result, if a train is far enough ahead, another train can enter the segment. This extension mitigates the need to estimate the capacity limit of a segment.

- **Speed Profiles:** With the incorporation of acceleration and deceleration, the model can generate speed profiles for trains as they travel through the network.

- **Better Support for Moving Block systems:** MoBlo is able to approximate advanced signaling systems like ETCS Level 2 with Moving Blocks. RailwaySim and FiBlo emulate the Fixed Block system and are unable to simulate movement authorities that update as a train traverses a segment.

- **Travel duration if blocked:** Train protection systems make sure trains slow down when there is an obstruction ahead. For instance, under PZB, trains must begin decelerating after passing a distant signal (Vorsignal) that indicates an occupied block ahead. The train needs to be able to stop before the upcoming signal block section. If the section clears before the train reaches the main signal, the train can accelerate again but because of the deceleration, the travel duration for this and the upcoming signal block section is longer than expected.

RailwaySim and FiBlo are unable to model this. A train only reserves the segment it is currently on, and if the train gets blocked, the models cannot capture the deceleration required. The next segment has the same minimum duration regardless of whether the train was blocked beforehand or not.

MoBlo is able to simulate this behavior by implementing the movement authority and explicitly modeling train speed. When a micro-segment ahead is occupied, the train reduces speed accordingly, which extends the travel time for the micro-segments. If the train has to come to a full stop, the next micro-segment will have an extended traversal time. This more granular representation of train speeds and travel durations creates a more realistic model for blocked trains and the effects of delays.

4.4 Limitations

The MoBlo model is better at simulating the Moving Block concept compared to RailwaySim/FiBlo, but still lacks some details. It is important to understand the limitations of MoBlo before interpreting the simulation results.

1. **Simplified Train Movement:** The train movement within MoBlo assumes constant acceleration and deceleration. Real-World trains experience a much more complicated acceleration and braking curve than MoBlo models, as well as external influences, including:

    - Track gradients
    - Curve resistance
    - Changes in air resistance due to tunnels
    - Train weight
    - Traction characteristics

    These simplifications in train movement may lead to inaccurate travel times, especially in areas with tunnels and steep climbs. While real-world freight trains can be impacted by their load, the simulation of freight trains is independent of the train weight.

2. **Simplification of Single-Track Segments:** MoBlo is unable to accurately model segments that consist of a single physical track used in both directions. In reality, a single-track line requires that trains traveling in different directions coordinate, as only one train is able to traverse the track. In RailwaySim, FiBlo and MoBlo, each direction is represented by a different infrastructure element, making movement in different directions independent of each other. Therefore, the models allow simultaneous bidirectional traffic on such a segment, which is physically not possible. This can lead to underestimation of conflicts and delays in scenarios with single-track configurations.

3. **Discrete Representation of Continuous Systems:** The Moving Block concept is a continuous system, and MoBlo approximates it with high frequency updates to position, speed and braking distances. While this approach is a good approximation, the model might not capture every detail, especially at high micro-segment lengths.

4. **Limited Representation of Train Protection System Protocols:** The MoBlo model, as well as RailwaySim and FiBlo, do not fully simulate all details of a train protection system, like:

   - Communication delays in updating movement authorities
   - Release and Buffer durations for segment reservations
   - Specific protocols required for different levels of ETCS.

   Even though the MoBlo model gets closer to microscopic simulation, it is still limited in resolution and detail.

5. **Increased Computational Runtime:** The extension from FiBlo to MoBlo introduced micro-segments that are significantly smaller. Instead of a single task per segment, the new model requires multiple, one for each micro-segment. This increased granularity results in higher computational requirements, potentially limiting the scale of the network that can be efficiently simulated.

6. **Limited Train-Specific Behavior Modeling:** The MoBlo model uses broad categories (e.g., Schnellbahn, Regionalexpress) to define train behavior, rather than modeling specific train or traction unit configurations. The same category can employ different traction units, each with a specific acceleration profile, but the simplification in MoBlo assigns one constant acceleration to the entire category, ignoring these variations.

7. **Train Length Missing:** The movement authority in MoBlo accounts for the current position and the braking distance but ignores the length of the train. The physical train length affects block occupation and therefore the headway between trains, but the model is unable to capture this.

8. **Conventional vs Emergency Braking:** The MoBlo model uses a single braking rate for both track reservations and train movements. The emergency braking rate is usually stronger than the braking rate at conventional stops. MoBlo calibrates the braking rate based on schedule data and does not include any emergency braking rate. The movement authority in a Moving Block system incorporates a braking distance that could be calculated based on the emergency rate, but MoBlo uses the same conventional braking. This makes the movement authority in MoBlo longer than necessary.

CHAPTER 5

Methodology

The extension from the Fixed Block model (FiBlo) to include a Moving Block system (MoBlo) allows for comparing traditional OCP-to-OCP segment approaches with the more detailed Moving Block approach. This chapter explains the simulation methods used to analyze differences in train behavior, infrastructure usage, and overall system performance between both models.

5.1 Data Overview

The train network modeling in this thesis relies on two main datasets provided by ÖBB: an extensive delay dataset and a detailed infrastructure model dataset.

5.1.1 Delay Dataset

The delay dataset contains records of train movements within the Austrian Train Network for December 14th and 15th, 2022. This dataset documents both freight and passenger trains, recording each stop or pass at an OCP.

This dataset tracks approximately 14,000 trains moving through 2,370 OCPs, generating 574,000 recorded events. Each record contains both scheduled and actual arrival and departure times. For non-stop OCPs, arrival and departure times are recorded as identical. A complete schedule for passenger and freight trains can be extracted from this dataset. While the overall data quality is robust, some trains have unusually large distances between two entries, indicating data anomalies and missing entries at intermediate OCPs.

For this thesis, analysis was focused on data from December 14th.

5.1.2 Infrastructure Model Dataset

The infrastructure model dataset has detailed information on OCPs and connecting segments. Each OCP contains metadata including operational type (station, junction, undefined) and references to connecting segments. Track metadata encompasses maximum permitted speeds, track classifications (track-running, siding, or undefined), and length.

This dataset contains 2,842 OCPs connected by 5,253 segments. Throughout this thesis, the network derived from this dataset is referred to as the plan infrastructure.

For simulation purposes, when scheduled movement between two OCPs lacks an explicit segment in the plan infrastructure, the algorithm in FiBlo and MoBlo calculates the shortest path between those OCPs. Because some entries span unusually long distances and miss intermediate OCPs, this analysis removed trains where a single movement would exceed 10 nodes in length.

5.1.3 Estimated Network and Differences

FiBlo and RailwaySim are designed to work on a network that is derived from schedule and delay data. For each OCP in the schedule, a corresponding one exists in the network. The connections between the OCPs are established based on observed train movement. For instance, a train moving from OCP A to OCP B creates the segment A-B.

The resulting estimated network consists of 2,370 OCPs connected by 4,303 segments. In comparison, the plan infrastructure model includes 2,842 OCPs with 5,253 segments. The difference between those two can be attributed to segments and OCPs that were not traversed during observation. Additionally, some tracks are merged due to the directional nature of certain OCPs.

Directional OCPs log train movement only in one direction and typically exist in pairs located in close physical proximity. For example, a route $A - B_1 - B_2 - C$, where B_1 and B_2 are directional OCPs, appears in the infrastructure model as three segments per direction. In the estimated network, the same route appears as $A - B_1 - C$ for one direction and $C - B_2 - A$ for the opposite direction, therefore only two segments per direction. The estimated network captures this directionality through observed train movements, even when not explicitly represented in the infrastructure data. FiBlo and MoBlo can deal with the plan infrastructure by routing the train through the shortest path between two scheduled OCPs.

The estimated network can also contain different OCP-to-OCP segment configurations than those in the plan infrastructure. It is possible that for a sequence of OCPs such as $A - B - C$, different segment configurations may emerge in the estimated network. For instance, some trains might have recorded arrivals at all three OCPs (A, B, and C), establishing segments $A - B$ and $B - C$ in the estimated network. However, other trains might have missing entries at the intermediate point B. This would result in an additional direct $A - C$ segment in the estimated network, even though the train physically traversed the same infrastructure through B. In the FiBlo model, these derived segments operate independently, meaning that occupation of the derived $A - C$ connection does not affect capacity or availability of the $A - B$ and $B - C$ segments, despite sharing the same physical infrastructure. It should be noted that such missing intermediate entries are rare in the dataset. Since only about 25 out of 4303 segments can be attributed to this inconsistency, it is ignored in both MoBlo and FiBlo.

5.2 Meidling - Wiener Neustadt

For specific analyses, including blocking time visualization and capacity consumption, this research focuses on a defined subset of the Austrian railway network: the line between Wien Meidling and Wiener Neustadt, as shown in Figure 5.1. This section shows some properties of interest:

- High traffic volume (multiple trains per hour)
- Diverse train composition including local commuter, regional express, and high-speed intercity services
- Straightforward track configuration with minimal crossings and junctions, consisting primarily of two tracks (one in each direction), facilitating capacity analysis
- Varied stopping patterns, with commuter trains making multiple stops while intercity trains typically only stop at Meidling and Wiener Neustadt, creating interesting dynamics

The analysis in this thesis focuses on the hours from 8:00 am to 10:00 am. During this time, the network experiences peak demand and has multiple train categories operating between these two cities, including both high-speed and commuter trains. EuroCity, InterCity, and Railjet are high-speed services offered on this railway line, designed for long-distance travel with few stops. During the specified time period, a total of 4 high-speed trains travel between Meidling and Wiener Neustadt. Additionally, there are 4 local passenger trains with more frequent stops and slower speeds using this infrastructure. These include 1 Schnellbahn (suburban commuter), 1 Regionalzug (local, regional), and 2 Regionalexpress (faster, regional) trains.

Because these services operate at different speeds and stop at different stations, there is an opportunity to analyze the interactions on a constrained infrastructure. During the time window, there are also two overtaking maneuvers involving faster EuroCity and InterCity trains passing slower services. These overtaking events, illustrated in Figure 6.5, occur at stations with dedicated passing lanes.

In terms of infrastructure characteristics, the subset in the infrastructure model dataset includes 45 OCP-to-OCP segments with a total length of 44,681 meters. Of these, 3,083 meters have two parallel tracks for each direction. The estimated infrastructure used

Figure 5.1: Selected railway subsection from Vienna (Meidling) to Wiener Neustadt marked in blue, other railway lines are shown in gray.

in FiBlo and RailwaySim deviates from the plan infrastructure by containing only 35 segments covering the same total length, but with 15,638 meters configured as quad-track segments (two tracks per direction). The difference in OCPs is due to the representation of directional OCPs, as explained in the estimated network section. Segment capacity estimation is also the reason that an additional 12 km are estimated as four parallel tracks, even though there are only two physical tracks there.

While the Meidling to Wiener Neustadt corridor is mainly operated as a double-track line with one track in each direction, the initial few kilometers from Meidling station have a quad-track configuration. Meidling station is a major junction with 8 platforms and multiple tracks heading in the direction of Wiener Neustadt before narrowing to the standard double-track configuration. The infrastructure plan includes an additional segment with four parallel tracks, but this represents a merging line rather than a true quad-track segment. When faster trains need to overtake slower services, this typically occurs at stations equipped with additional passing tracks, requiring careful timetable planning.

5.3 Train Movement Calibration

The MoBlo model, discussed in Chapter 4, utilizes acceleration, deceleration, and relative maximum speed parameters to model train movement throughout the network. The calibration aims to determine the parameter settings that best approximate the scheduled journey times between OCPs.

Root Mean Square Error (RMSE) is the main metric for evaluating the accuracy of simulated travel times against scheduled travel times:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5.1}$$

where y_i is the scheduled time and $\hat{y}_i$ is the simulated time for each segment. The calibration compares simulated times against scheduled runtimes rather than observed actual runtimes, as the latter may contain primary or secondary delays that should emerge from the simulation rather than be built into the train movement model.

Mean Absolute Relative Error (MARE) serves as a secondary evaluation metric:

$$MARE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{5.2}$$

MARE is a valuable complementary metric to RMSE because it looks at errors in relative terms, making it easier to interpret with different journey durations. It tends to disproportionately penalize errors in short journey segments, where even slight deviations can lead to large percentage differences. RMSE places higher importance on large absolute time differences, which better matches the focus of having consistent timing across the railway network.

To find the best parameter setting for each train category, the calibration process uses an extensive grid search that tests every combination of parameters. The process consists of finding all schedules for a given category, calculating the simulated travel time based on the train movement parameters, and then comparing them to the scheduled runtime. The lowest RMSE indicates the parameter setting with the least deviation.

The grid search explores the following parameter ranges:

- Acceleration values: 0.1 to 1.1 m/s² (11 linearly spaced values)
- Deceleration values: -0.1 to -1.1 m/s² (11 linearly spaced values)
- Relative maximum speed values: 0.4 to 1.0 (7 linearly spaced values)

These ranges are based on previous research: Rosberg et al. [RT20] observed braking rates of 0.6 m/s^2 and acceleration rates between 0.3 and 0.6 m/s^2 for passenger trains in Sweden, while Vergeosen [Ver20] utilized a braking rate of 0.5 m/s^2 in ERTMS/ETCS simulations.

The grid search was performed twice: on the subset (Wien Meidling to Wiener Neustadt, 8:00 to 10:00 time interval) and on the entire Austrian railway network. This has two advantages:

1. The capacity estimation process can use the parameters specifically tuned for this subset to get the runtimes that best match the scheduled runtimes.
2. It allows an assessment of how well these parameters generalize by comparing subset-tuned parameters to those derived from the entire network.

5.4 Sensitivity to micro-segment length

The MoBlo model introduces an additional parameter: the micro-segment length. To capture the continuously updating movement authority of Moving Block systems, the micro-segment length would ideally be very small. However, computational limitations require a tradeoff between accurately modeling Moving Block and computational resources. As the micro-segment length decreases, there are more micro-segments for the same OCP-to-OCP segment and therefore more *Drive* tasks to process. A sensitivity analysis varies the micro-segment length and can thus show the impact it has on simulated delay propagation, capacity consumption and computational runtime.

5.5 Blocking Time Diagram Analysis

A blocking time diagram visualizes the time periods during which signal block sections are reserved or occupied by trains. These diagrams are essential tools for understanding train operation dynamics, including overtaking maneuvers, stopping patterns, and traffic flow.

Occupation time is fundamental to these diagrams and to railway capacity evaluation. According to UIC 406 leaflet [UIC13], occupation time encompasses several components:

1. **Setup time**: Time required for route formation and signal system preparation before a train enters the block.
2. **Sight and reaction time**: Period between when a train operator first sees the distant signal and when the train physically reaches it.
3. **Approach time**: Time a train requires to traverse the distance from the distant signal to the main signal.

4. **Running time**: Journey time of the train through the occupied block section.

5. **Clearing time**: Time required for the tail of the train to clear the block section after the head has passed the main signal.

6. **Release time**: Time needed for the signaling system to release the block for the next train.

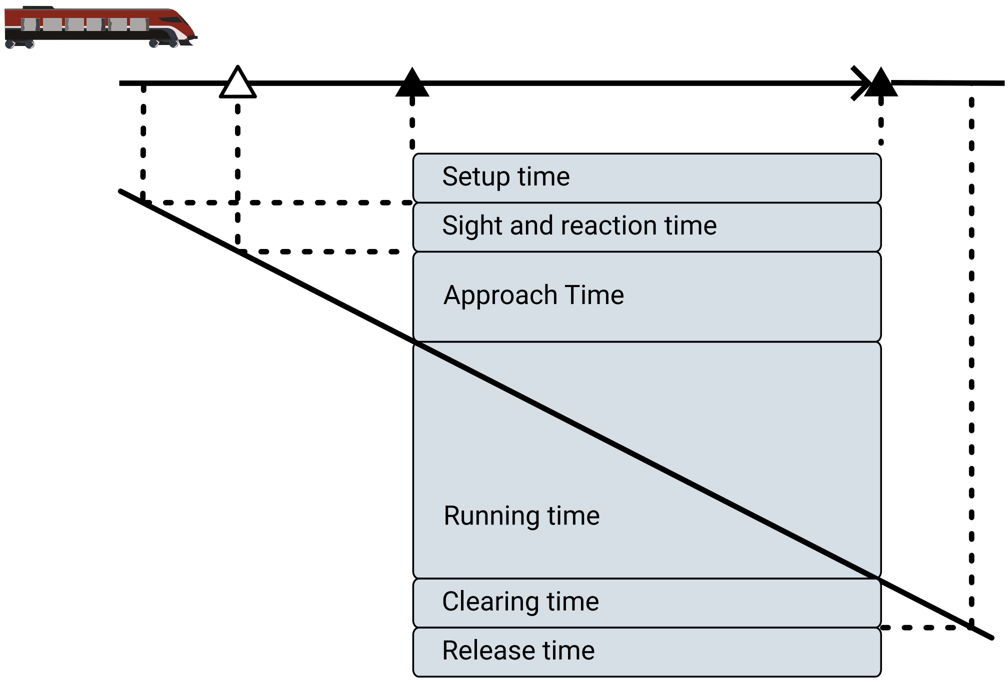


Figure 5.2: Blocking time for three-aspect signaling like PZB, showing the main signal (solid triangle) and distant signal (hollow triangle). Adapted from [GCD13].

Figure 5.2 displays the different components of block occupation time. In Fixed Block systems like PZB, the distant signal provides information about the status of the main signal ahead. If the distant signal indicates a stop ahead, trains need to reduce speed to a predefined limit within 23 seconds after passing it (see Section 2.1.1 for more details). This ensures that the train is able to stop before it reaches the main signal if necessary.

In the context of blocking time calculations, this is important as it effectively extends the occupation period of a signal block section. For a train to move unimpeded by other trains, the next section needs to be available even before the train reaches the distant signal; otherwise, it is required to slow down. A block section needs to be considered occupied not only for the duration that the train actually traverses it but also for the setup, sight, and approach time before that. Consequently, these additional times contribute to the minimum headway time between two consecutive trains.

5.5.1 Block Reservations in FiBlo

The FiBlo model does not account for all components of the occupation time; specifically, it only models the running time and ignores other parts like approach or clearing time. Instead of using signal block sections, FiBlo operates on OCP-to-OCP segments with blocking times derived from the departure and arrival of the neighboring OCPs. When a train is simulated to depart at time x from OCP A and arrive at time y at OCP B, the corresponding segment is blocked from x to y. As a result, FiBlo creates discrete, non-overlapping block reservations where a train occupies only one segment at any given time. Within the simulation, the train moves through each segment and only checks the occupation status of the following segment at the end. This simplified approach might underestimate the capacity needed for a train to move unimpeded through the network.

5.5.2 Block Reservations in MoBlo

MoBlo does not occupy entire segments but uses micro-segments (e.g., 128 m each) and reserves those to simulate the movement authority around the train. The model incorporates an additional aspect of the occupation time concept neglected in FiBlo: the approach time. By reserving all micro-segments required to come to a full stop if needed, the model already accounts for the approach time. For an unimpeded run at full speed, the train must continuously reserve its braking distance, if this is not possible, the train decelerates accordingly. As a train traverses the network in MoBlo, it simultaneously occupies multiple micro-segments, creating overlapping block reservations that better reflect actual railway operations. This model still simplifies block occupation by neglecting the setup, sight, clearing, and release time.

Additionally, because train movements follow physical acceleration and deceleration profiles based on the calibration, the simulation produces simulated travel times that deviate from scheduled times.

5.5.3 Methodological Differences and Considerations

The concept of block occupation time differs between MoBlo and FiBlo in four major areas:

1. **Block granularity**: FiBlo operates on coarse OCP-to-OCP segments, while MoBlo uses fine-grained micro-segments, enabling more precise modeling of train movements.

2. **Occupation time components**: FiBlo only accounts for running time, while MoBlo additionally implements approach time through its braking distance reservation mechanism. Neither model fully implements all components defined by UIC 406.

3. **Block reservation pattern**: FiBlo maintains non-overlapping, sequential block reservations. MoBlo creates overlapping reservations that incorporate the breaking distance.

4. **Schedule adherence**: FiBlo strictly follows the predefined schedule, whereas MoBlo's physical movement model can produce deviations from scheduled times.

These differences should be considered when interpreting capacity assessments derived from either simulation approach.

5.6 Capacity Consumption

The capacity consumption of a railway line can be measured using the UIC 406 method [LSKST08] [UIC13]. The method tries to estimate how much of the infrastructure is utilized by a given timetable. It compresses the timetable graph to remove the buffer-times between trains and determines the minimum time that is required for all trains to travel through the network subset [LKSST06].

The UIC 406 method includes these steps:

1. **Line and Time Interval:** Select a railway line and time interval for analysis. UIC 406 recommends at least two hours to analyze peak traffic [UIC13].

2. **Splitting into Line Sections:** Split the railway line into smaller sections based on junctions, crossings, overtakings, signaling system changes, or transitions between single and double tracks [UIC13]. Analyze each of the resulting line sections separately by performing the following steps on each [UIC13].

3. **Train Path Collection** Identify all trains that enter the selected section within the selected time interval. The first train path is then copied and added to the last position within the time interval as part of the calculation process [UIC13].

4. **Timetable Compression** The next step involves compressing the timetable graphs. This compression is performed by pushing all train paths together to the minimum headway time, eliminating any buffer times. The compression must respect original running times, dwell times, and block occupation times [LSKST08]. This can be visualized with the Blocking Time Diagram by compressing train paths until some occupation blocks touch but never overlap. Occupation time includes pre-entry, running, and clearing time, see section 5.5.

5. **Line Occupation Time Calculation** Calculate line occupation time as the period between the entry of the first train and the last (copied) train into the section.

6. **Capacity Calculation** The occupation time rate is the ratio between the line occupation time and the selected time interval. To account for necessary buffers and ensure a high quality service, the capacity consumption adds additional margins based on the Additional Time Rate [UIC13]:

$$\text{Occupancy Time Rate [\%]} = \frac{\text{Occupancy Time}}{\text{Defined Time Period}} \tag{5.3}$$

$$\text{Capacity Consumption [\%]} = \frac{\text{Occupancy Time} \times (1 + \text{Additional Time Rate})}{\text{Defined Time Period}} \tag{5.4}$$

For mixed traffic, the Additional Time Rate is 0.33 for peak hour and 0.67 for daily period [UIC13].

5.6.1 Adaptations for this thesis

This thesis slightly modifies the UIC capacity consumption method for the FiBlo and MoBlo models:

Line Section Adjustments
The railway line between Wien Meidling and Wiener Neustadt is the basis for this evaluation. The subset of the networks has some branching and overtakings that would require a split into multiple line sections, each independently analyzed by the UIC compression method. Because the line mostly has double-track configuration, it is maintained as a single line section for simplicity. To handle overtakings at stations, this thesis adopts Denmark's approach of maintaining train order as specified in the schedule before and after overtaking [LSKST08].

Track Occupation Times
The UIC compression method is designed to use every part of the block occupation concept introduced in section 5.5. Both models simplify this approach by omitting the setup, sight, clearing and release times. While MoBlo incorporates the approach time by reserving micro-segments for the breaking distance, FiBlo omits this part and only models the running time without additional margins.

Handling Estimated Segment Capacity in FiBlo
As discussed in Section 5.2, both the estimated network and the plan infrastructure network indicate segments with a segment capacity greater than one per direction. In the infrastructure plan, the first few kilometers after Meidling are fitted with four parallel tracks. Additionally, a different railway line merges into this one along the way, shortly increasing the number of parallel tracks. The FiBlo model is designed to work with an estimated network derived from observed train movements, which can lead to segment capacity estimations that do not perfectly match the physical infrastructure. In some cases, this estimation process suggests capacities exceeding one per direction, even when there is only one physical track between those OCPs.

To examine the effect of different interpretations on segment capacity, the following compression techniques are implemented:

1. **One Track per Direction:** Assumes a capacity of one on each segment throughout the corridor and ignores the initial quad-track setup and merging lines, but allows for overtaking at stations.
2. **Plan infrastructure:** Uses the capacity values specified in the infrastructure model dataset. For example, if there are two physical tracks at a specific segment is marked with a capacity of two. The compression therefore allows for overlap in occupation times with up to two trains occupying the same segment simultaneously.
3. **Estimated segment capacity (FiBlo only):** Utilizes the segment capacity as calculated by the heuristic method based on historical train movements. This allows trains to overlap at specific segments according to the maximum simultaneous trains observed on this segment in historical data.

By comparing these approaches, it is possible to analyze how different interpretations of segment capacity affect line capacity calculations and timetable compression results.

This analysis additionally tries to evaluate the impact of the micro-segment length in the MoBlo model on the calculated line capacity. The hypothesis is that shorter micro-segment lengths should allow for smaller headways between trains, potentially increasing the theoretical line capacity as trains can follow each other more closely.

It is important to note that a direct comparison between FiBlo and MoBlo models is difficult due to fundamental differences in how each model derives running times and defines block occupation. Section 5.5.3 lists the differences in modeling block occupations.

5.7 Comparative Analysis of Delay Propagation

Line capacity analysis through timetable compression focuses on the theoretical limits of a railway network. However, it does not completely account for the complexity of real-world operations. Train delays are a common occurrence in railway systems and can propagate through the network, affecting overall performance. Understanding how delays spread and affect other trains is a different topic worth investigating.

To quantify the propagation of delays in railway systems and assess the amount of secondary delay generated within the simulation, this thesis introduces the concept of markup.

5.7.1 Markup Definition

Markup is a measure of the ratio between the actual runtime a train has in a simulation with injected primary delays and the expected runtime based on the base simulation and primary delays. The markup for each individual train is defined as:

$$M_i = \frac{R_{d,i}}{R_{b,i} + D_{p,i}} \tag{5.5}$$

Where:

- M_i is the markup for train i
- $R_{d,i}$ is the runtime for train i in the simulation with injected primary delay
- $R_{b,i}$ is the base runtime for train i (the runtime in a simulation with the same model but without primary delay)
- $D_{p,i}$ is the sum of primary delays for train i injected through its journey for $R_{d,i}$

The overall markup for the entire simulation is then calculated as the average of all individual train markups:

$$M_{total} = \frac{1}{n}\sum_{i=1}^{n} M_i \tag{5.6}$$

Where n is the total number of trains in the simulation.

It is important to note that the base runtime $R_{b,i}$ is different depending on the type of model analyzed. This approach is useful when comparing different simulation models, such as MoBlo and FiBlo. MoBlo does not strictly adhere to the schedule as it models train movement through acceleration, deceleration and maximum speed. Therefore the results of MoBlo and FiBlo can differ even without any primary delay. By using scenario-specific base runtimes, we can reduce the effect of runtime deviations introduced by the train movement modeling in MoBlo. This normalization is important when comparing the fundamentally different concepts, and allows for a fair assessment of how Moving Block systems might improve or alter delay propagation compared to Fixed Block systems.

A markup value greater than 1 indicates the presence of secondary delays, as the actual runtime exceeds the expected runtime (base runtime plus primary delay). Conversely, a markup value less than 1 suggests that the system is able to recover from the primary delay, effectively "catching up" during the journey. The magnitude of the markup indicates the extent of secondary delay or recovery within the system.

Three scenarios are analyzed to understand delay propagation under various settings:

1. FiBlo with estimated network
2. FiBlo with the plan network
3. MoBlo with plan network and various micro-segment lengths

These three scenarios enable a evaluation of how different modeling approaches and infrastructure representations affect delay propagation. By comparing FiBlo simulations with both the estimated network and the infrastructure plan, the analysis can quantify the impact of infrastructure modeling on delay propagation dynamics.

Furthermore, the analysis of MoBlo with various micro-segment lengths explores the relationship between granularity and delay propagation. As micro-segment lengths decrease, the model approaches a more continuous representation of train movement, potentially capturing delay interactions with greater precision. However, this increased fidelity comes with computational costs that must be balanced against the practical benefits.

5.7.2 Primary Delay Distribution

Primary delays are important to accurately simulate real world train networks. For example, a extended dwell times at a station can lead to secondary delay in a different train. This study uses an existing method for generating primary delays, based on previous work by Viehauser [Vie24]. The primary delays are modeled to best capture the external influence found in real world scenarios.

Viehauser's model distinguishes between four different distributions:

1. Freight trains at their first stop
2. Freight trains at subsequent stops
3. Passenger trains at their first stop
4. Passenger trains at subsequent stops

This differentiation accounts for the varying nature of delays encountered at the beginning of a journey versus those occurring during the trip, as well as the distinct characteristics of freight and passenger services.

Using these distributions, ten different primary delay scenarios were sampled. Each scenario represents a complete set of primary delays for every task in the simulation. These ten samples were then applied consistently across all three simulation scenarios (FiBlo with estimated network, FiBlo with planned network, and MoBlo with planned network). This methodology guarantees that any differences observed in delay propagation and overall system performance are attributable to the characteristics of the simulation model rather than variations in the input delays.

5.8 Computational Runtime Analysis

MoBlo best approximates an ideal Moving Block systems when the micro-segment lengths are really short and the update frequencies high. But as micro-segment lengths decrease in the MoBlo approach, the number of events that must be processed increases substantially. This can lead to unacceptable runtimes.

To quantify this trade-off, runtime measurements were conducted alongside the delay propagation experiments described in Section 5.7. The time measurements only include the simulation execution and ignore the preprocessing steps such as reading primary delay data or constructing the infrastructure model.

For each simulation model (FiBlo with estimated network, FiBlo with planned network, and MoBlo with planned network), ten different primary delay scenarios were applied. The average runtime across these ten scenarios per model was calculated as a measure of computational performance.

All experiments were executed on a virtual machine with the following specifications:

- 8 virtual CPU cores
- 32 GB RAM
- 4 experiments running in parallel
- Single-threaded execution per experiment

This controlled environment tries to reduce the impact of background activities that could potentially influence the runtime of the simulations. While experiments were run in parallel to reduce overall execution time, each individual simulation was confined to a single thread. This mitigates potential inferences between experiments.

CHAPTER 6

Results and Discussion

The following chapter presents the results for the experiments discussed in Chapter 5. The main focus of the analysis is to examine the influence of different simulation models and infrastructure settings on capacity consumption and delay propagation. FiBlo and MoBlo approximate different train protection systems, comparing them gives insights into how the train protection system can alter the dynamics in a railway network. The FiBlo model is closely related to the Fixed Block approach where trains occupy entire OCP-to-OCP segments with limited capacity. The MoBlo model refines this approach by splitting each segment into smaller micro-segments and allowing trains to occupy micro-segments based on their movement authorities. The movement authority also incorporates their braking distance and is updated at every micro-segment as the train moves through the network. This closely resembles the Moving Block concept.

The next sections analyze how the different models, and their respective train protection systems, alter capacity consumption and delay propagation.

6.1 Results of Train Movement Calibration

MoBlo requires accurate calibration of the movement parameters (acceleration, deceleration and maximum relative speed).

This section outlines the results of the train movement calibration process described in Section 5.3. The grid search method was applied to both a subset (Meidling to Wiener Neustadt) and the entire Austrian railway network to determine the best train movement parameters. The section evaluates how different parameter combinations affect simulation accuracy (RMSE), identifies optimal values for each train category, and assesses parameter generalization from the subset to the broader network.

6.1.1 Grid Search Visualization

Figure 6.1 visualizes the grid search results for the Regionalzug category across both calibration areas. The heatmap shows that parameter combinations with either a low acceleration or deceleration perform poorly in terms of RMSE across both datasets. Using either extremely low or extremely high relative maximum speed also results in increased error rates, which demonstrates the importance of this parameter for getting good results. The two heatmaps from the different calibration datasets (Meidling to Wiener Neustadt and entire network) display similar patterns, suggesting good generalization potential for parameters calibrated on the subset.

For the Railjet category, the heatmap for the subset Meidling to Wiener Neustadt is shown in Figure 6.2. The relative maximum speed parameter exhibits an even more pronounced impact on model performance. This can be attributed to reduced intermediate stops, which makes the maximum cruising speed an important factor in calculating the runtime.

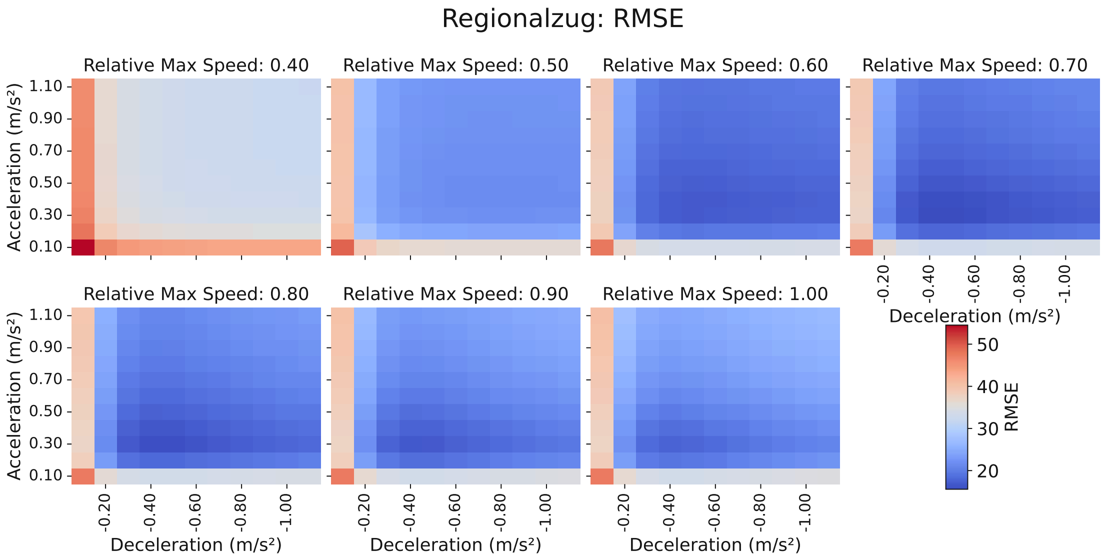
6.1.2 Optimal Parameters and Error Metrics

Table 6.1 presents the optimal parameter combinations and corresponding error metrics for each train category on both the subset and entire network. The results show notable differences in optimal parameter values between train categories. This reflects that each train category has different operational characteristics and might accelerate and decelerate at a different rates.

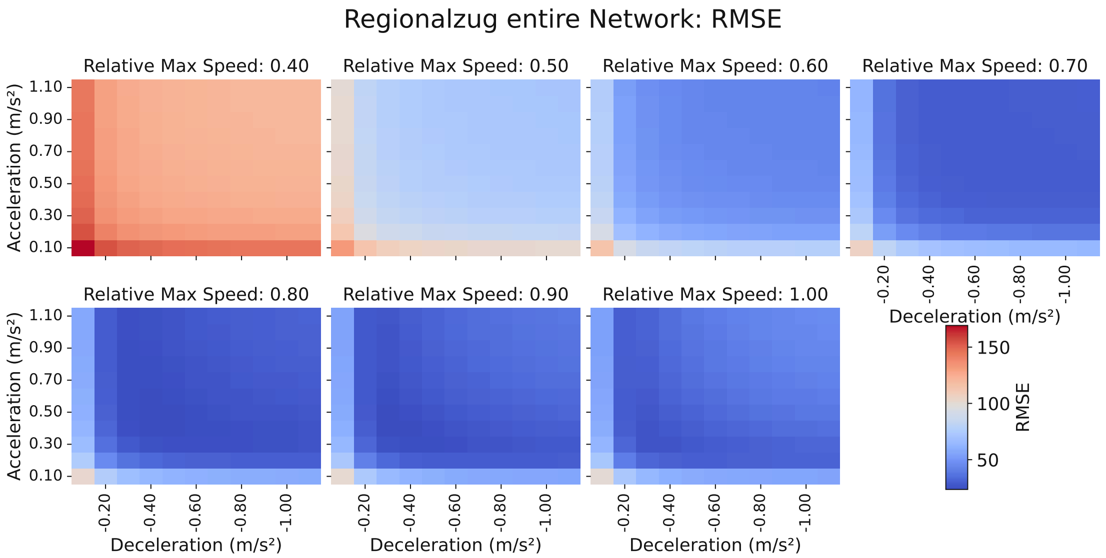
Several noteworthy patterns emerge from the calibration results:

- Long-distance train categories (Railjet, Eurocity) consistently demonstrate lower optimal acceleration values compared to regional services on the subset from Meidling to Wiener Neustadt.
- The InterCity category shows an unusually high optimal acceleration value (0.8 m/s^2) on the subset but aligns with other long-distance services in the network-wide calibration.
- The Mean Absolute Relative Error (MARE) ranges from 13.5% to 33.6%, indicating significant deviations between simulated and scheduled runtimes.

Figure 6.3 shows the Mean Absolute Relative Error for each OCP-to-OCP segment, emphasizing areas where predictions deviate most from the schedule. Central Austria stands out with several bright segments, indicating high error rates. Interestingly, these tracks are primarily used by freight rather than passenger trains, which might explain the discrepancy since freight operations often follow different patterns. The Vienna area also has elevated error rates, likely due to the complex nature of its dense railway network. While some mountain regions show moderate errors, there's no clear pattern suggesting that topography alone drives prediction accuracy.



(a) Parameter optimization results for the Subset Meidling - Wiener Neustadt



(b) Parameter optimization results for the entire network

Figure 6.1: Grid search results for optimizing train movement parameters for the **Regionalzug** train category in the MoBlo simulator. Each heatmap represents a different relative maximum speed value, with acceleration rates on the x-axis and deceleration rates on the y-axis. Color gradients indicate the root mean square error (RMSE) between simulated and scheduled travel durations between OCPs, with blue regions representing lower error (better fit) and red regions representing higher error.

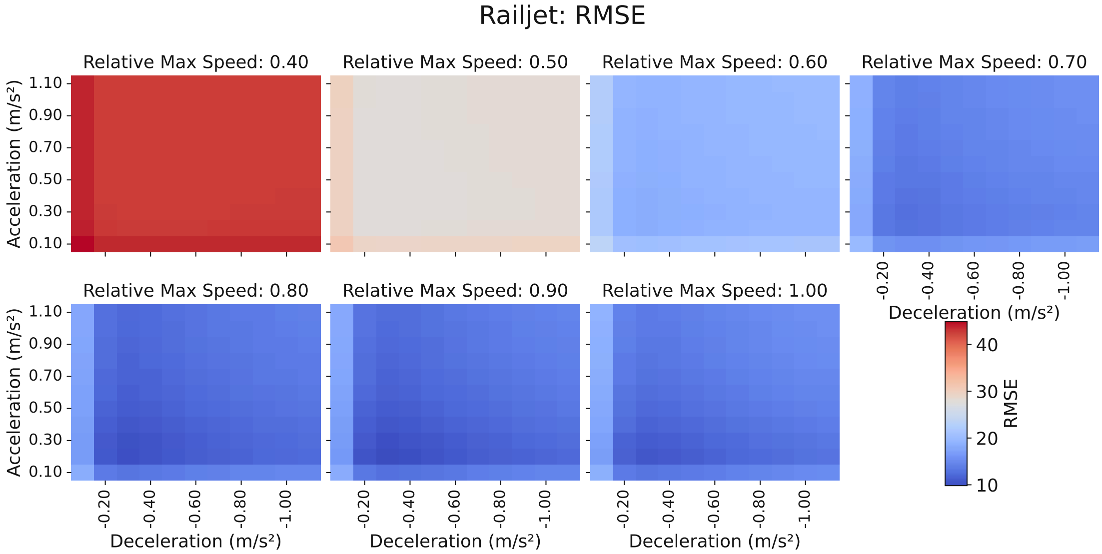


Figure 6.2: Grid search results for optimizing train movement parameters for the **Railjet** train category on the subset Meidling - Wiener Neustadt. Color gradients indicate the RMSE between simulated and scheduled travel durations between OCPs.

Table 6.1: Optimized train movement parameters and resulting error metrics for different train categories. Error metrics show RMSE and MARE between scheduled and simulated travel times, calibrated and evaluated separately for the Meidling-Wiener Neustadt subset and the entire Austrian network.

Category	**Acc.** (m/s^2)	**Dec.** (m/s^2)	**RMS**	**RMSE** (s)	**MARE** (%)
		subset			
Railjet	0.2	-0.3	0.9	9.8	16.1
Eurocity	0.2	-0.3	0.9	9.8	16.1
InterCity	0.8	-0.3	0.7	15.9	33.6
Regionalzug	0.3	-0.4	0.7	15.6	20.4
Regionalexpress	0.3	-0.4	0.7	16.7	18.5
Schnellbahn	0.4	-0.5	0.6	13.9	13.5
		Entire Network			
Railjet	0.3	-0.3	0.8	18.6	18.2
Eurocity	0.3	-0.4	0.8	22.9	19.6
InterCity	0.3	-0.4	0.8	22.6	18.4
Regionalexpress	0.4	-0.4	0.9	22.9	19.2
Regionalzug	0.5	-0.4	0.8	23.6	15.2
Schnellbahn	0.4	-0.4	0.9	20.5	16.6

Note: Acc. = Acceleration, Dec. = Deceleration, RMS = Relative Max Speed

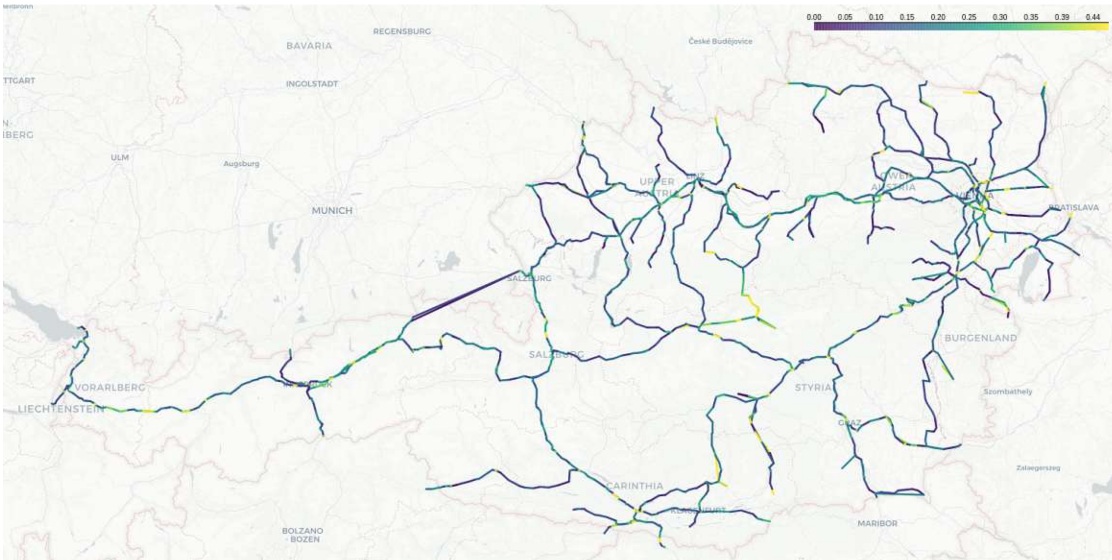


Figure 6.3: Spatial distribution of Mean Absolute Relative Error (MARE) between scheduled and MoBlo-simulated travel durations across the Austrian railway network. Each segment connecting Operational Control Points (OCPs) is color-coded according to its error magnitude, with brighter segments (yellow) indicating higher MARE values and darker segments (blue) representing lower error where the simulation more closely matches the scheduled times.

Figure 6.4 shows the Mean Relative Error (MRE), which is a good measure to see the bias in the calculation of travel times. This visualization highlights segments where the model systematically overestimates runtime (red) or underestimates it (blue). Segments appearing in neutral colors in the middle of the spectrum either demonstrate well-calibrated predictions or contain offsetting errors in both directions.

6.1.3 Factors Contributing to Model Discrepancies

Several factors contribute to the discrepancies between the simplified train movement model and actual scheduled runtimes:

1. **Limited infrastructure data granularity:** The model lacks detailed information about track characteristics such as gradient profiles, curve radii, and temporary speed restrictions that significantly influence train performance in real-world operations.

2. **Overgeneralized train categorization:** The broad categorization scheme may inadequately capture variations within each category. For instance, different traction units with varying performance characteristics may operate within the same train category, leading to inconsistent behavior patterns.

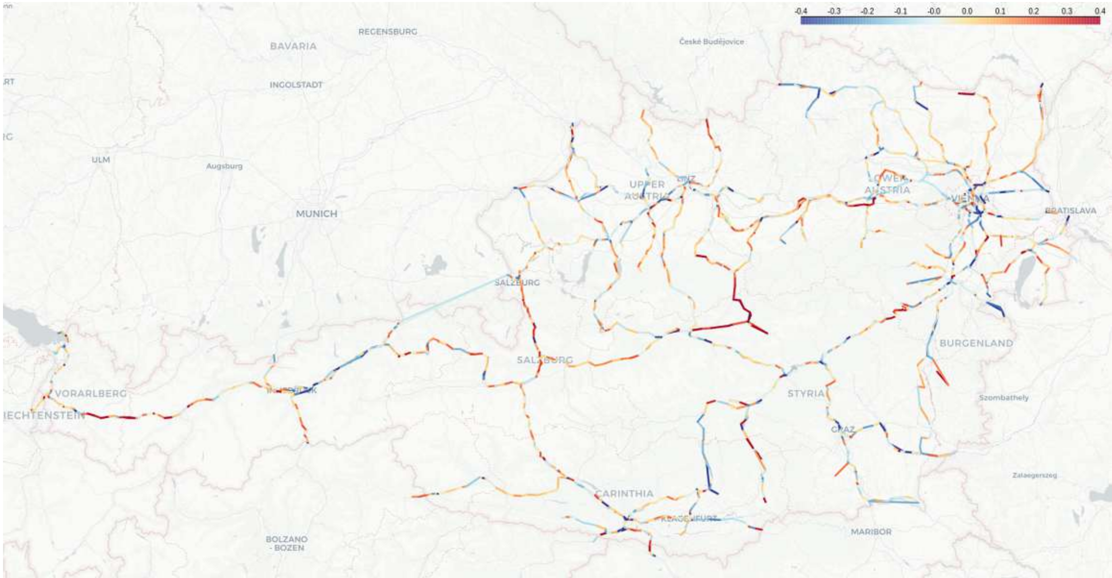


Figure 6.4: Geographical visualization of Mean Relative Error (MRE) between scheduled and MoBlo-simulated travel durations across the Austrian railway network. Blue segments indicate underestimated travel times (simulation faster than schedule), while red segments show overestimated travel times (simulation slower than schedule), with color intensity representing error magnitude.

3. **Simplified physics representation:** The constant acceleration and deceleration model represents a significant simplification of actual train dynamics, which involve complex interactions between power, resistance, and track conditions.

4. **Timetable buffer allocation:** Timetables often include buffer times that are not accounted for in the simplified model.

6.1.4 Parameter Generalization Assessment

To evaluate how well parameters calibrated on the subset generalize to the entire network, a comparative analysis was conducted using the Root Mean Square Error (RMSE) metric. Table 6.2 presents this comparison.

The analysis of parameter generalization reveals several important findings:

1. **Consistent performance degradation:** All train categories exhibit increased RMSE values when subset-calibrated parameters are applied to the entire network, confirming that localized calibration does not fully capture the complexity of the broader system.

2. **Variable generalization success:** The extent of performance degradation varies substantially across categories, with Eurocity showing minimal RMSE increase (3.1%) while Schnellbahn exhibits considerable deterioration (67.3%).

3. **Long-distance vs. regional services:** Parameters for long-distance services (particularly Eurocity and Railjet) generally demonstrate better generalization capabilities compared to regional services (Schnellbahn, Regionalexpress, Regionalzug).

Table 6.2: Comparison of Root Mean Square Error (RMSE) in seconds between scheduled and simulated travel times on the entire network when train movement parameters are calibrated using different network scopes (entire Network and Meidling - Wiener Neustadt subset). Lower RMSE values indicate better fit between simulation and schedule.

	Calibration Scope		
Train Category	**Entire Network (s)**	**Subset (s)**	**Error Increase (%)**
Railjet	18.6	22.3	19.9
Eurocity	22.9	23.6	3.1
InterCity	22.6	28.8	27.4
Regionalzug	23.6	34.1	44.5
Regionalexpress	22.9	30.2	31.9
Schnellbahn	20.5	34.3	67.3

Multiple factors may contribute to these observed differences in generalization capacity:

1. **Network heterogeneity:** The subset represents only a limited portion of the network's operational conditions. Train services operate differently across regions due to varying infrastructure quality, operational practices, and geographical constraints. For instance, operations in mountainous regions fundamentally differ from those in flat terrain.

2. **Fleet heterogeneity:** Categories with more standardized rolling stock (such as Eurocity) demonstrate better generalization than categories that may utilize diverse vehicle types across different regions (such as Schnellbahn).

These results emphasize both the potential and limitations of applying locally-tuned parameters to the entire railway network.

6.2 Blocking Time Diagram Analysis Results

Building on the methodological framework established in Section 5.5, this section presents and analyzes the blocking time diagrams generated by both the FiBlo and MoBlo implementations. These visualizations showcase how different modeling approaches represent infrastructure occupation patterns. As previously established in Section 5.5.3, the two implementations exhibit fundamentally different approaches to infrastructure occupation.

Figure 6.5 shows FiBlo operating with plan infrastructure data and adheres strictly to the scheduled timetable, releasing each segment precisely when entering the next one, without any safety margins.

A notable feature in this visualization is the presence of blocks spanning multiple OCP-to-OCP segments. This occurs because the schedule data skips directional OCPs that exist in the other direction, see section 5.1.3 for more details. The schedule might mention the OCPs B_1 and C but skip B_2 in between, which is for the opposite direction. When FiBlo's pathfinding algorithm processes such a schedule entry, it assigns multiple infrastructure elements to a single task to ensure connectivity. A train scheduled to go from B_1 to the next regular OCP C needs to traverse both the B_1-B_2 segment and the B_2-C segment. Since both segments are assigned to the same task, they appear as one continuous block in the blocking time diagram.

It's worth noting that if FiBlo were used with the estimated infrastructure, each task would block only a single infrastructure element (directly connecting the OCPs B_1 and C), but the visualization would still show the skipped directional OCP to maintain consistency with the physical layout and the other modeling approaches, therefore creating a similar spanning appearance.

In contrast, the MoBlo implementation (Figure 6.6) incorporates train movement physics and works directly on smaller micro-segments. The train movement calibration process leads to deviations between simulated and scheduled travel times, as they can not be simulated without error. The smaller micro-segments result in finer "stairs" in comparison to FiBlo. Crucially, MoBlo reserves not only the micro-segment a train currently occupies but also all micro-segments within its braking distance, thereby implicitly incorporating a basic safety margin.

6.3 Capacity Consumption Results

This section presents the results from the capacity consumption analysis for the FiBlo and MoBlo simulation approaches, described in Section 5.6. The analysis focuses on evaluating the level of infrastructure utilization between Meidling and Wiener Neustadt.

Capacity consumption is the percentage of time that railway infrastructure is occupied during a defined time window. The UIC 406 methodology [UIC13] forms the foundation of

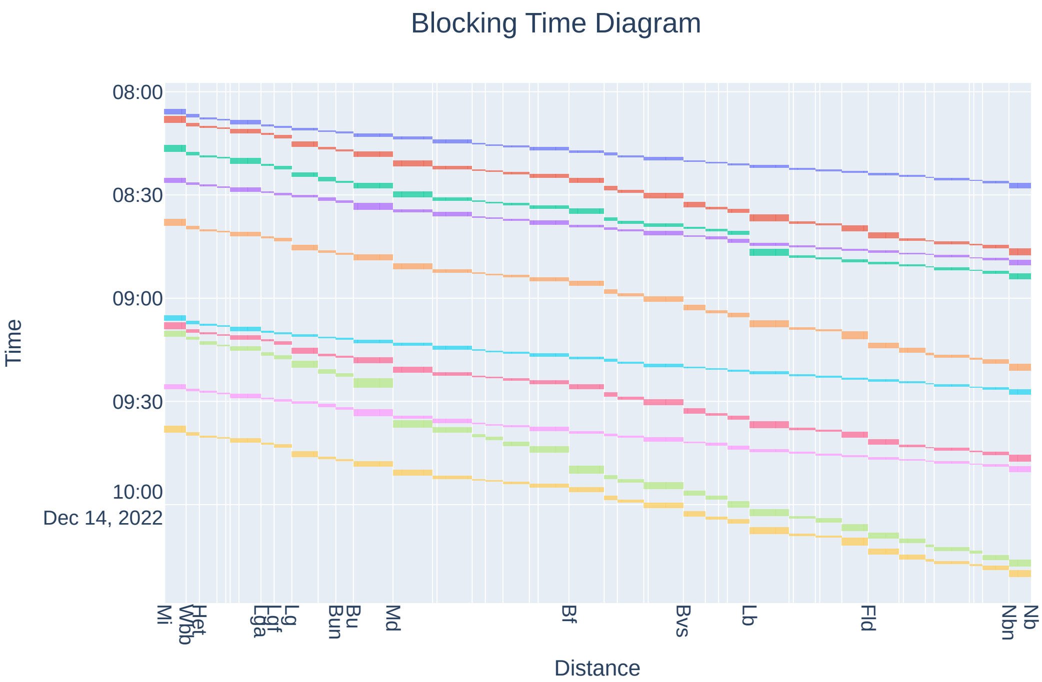


Figure 6.5: Visualization of infrastructure occupation times in the **FiBlo** model from Meidling to Wiener Neustadt. Each vertical band represents a segment betweenOCPs according to the infrastructure plan. Trains occupy exactly one segment at a time based on their position without considering braking distances.

the capacity analysis conducted in this thesis, although with adaptations to accommodate the specific characteristics of both simulation approaches.

FiBlo and MoBlo emulate two different train protection systems (Fixed Block vs Moving Block). Comparing the capacity consumption between the two systems requires careful attention to the fundamental differences. The models have a different interpretation regarding occupation time and also use different travel times between OCPs. However, the comparison still offers useful indications that Moving Block systems may provide improvements over Fixed Block in terms of capacity consumption.

6.3.1 FiBlo Capacity Utilization Results

The FiBlo simulation model was evaluated using three distinct infrastructure configurations to assess their impact on network capacity consumption. Table 6.3 summarizes the occupation times and corresponding capacity consumption percentages and reveals substantial differences between the different infrastructure settings. FiBlo operating on the estimated infrastructure shows lower occupation time (1:03:30) and capacity consumption (70.2%) compared to the other configurations. Since the estimated infrastructure assigns a capacity limit for each segment based on the historic maximum number of

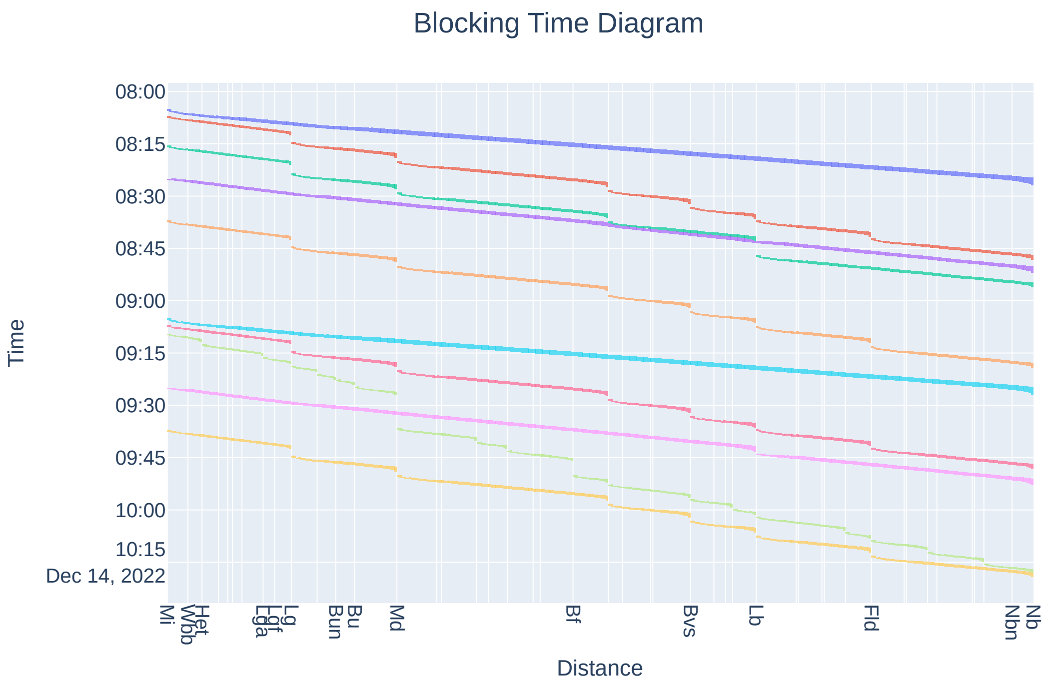


Figure 6.6: Visualization of infrastructure occupation times in the MoBlo model from Meidling to Wiener Neustadt using 256m micro-segments. Each vertical band represents a segment betweenOCPs according to the infrastructure plan. MoBlo reserves not only the micro-segment a train is traversing but also additional micro-segments required for its braking distance.

trains observed on the segment, it might allow parallel occupation even on segments with only one physical track. This permits the simulation to overlap the block occupation at critical points near Wiener Neustadt and at overtaking stations.

Figure 6.7 illustrates the compressed blocking time diagram for FiBlo with estimated network infrastructure. It demonstrates how the compression works and shows overlap at the line end and at stations after overtakings. Due to these overlaps, it is possible to shorten the headway between trains significantly compared to the other infrastructure scenarios.

All scenarios show capacity consumption below 100%, remaining within UIC recommendations [UIC13]. The additional time rate of 0.33 already accounts for buffer time needed for stable operations.

The differences in capacity consumption demonstrate one significant finding: the estimation method for the segment capacity can significantly alter the capacity consumption. Using the number of tracks as the capacity limit of a segment increases the capacity estimation by 9.2 percentage points over the heuristic method.

Table 6.3: Occupation time and capacity consumption under FiBlo with different infrastructure networks between Meidling and Wiener Neustadt in a 2h time window.

Infrastructure Setting	Occupation Time	Capacity Consumption
Estimated Infrastructure	1:03:30	70.2%
Infrastructure Plan	1:11:48	79.4%
One Track per Direction	1:14:54	82.8%

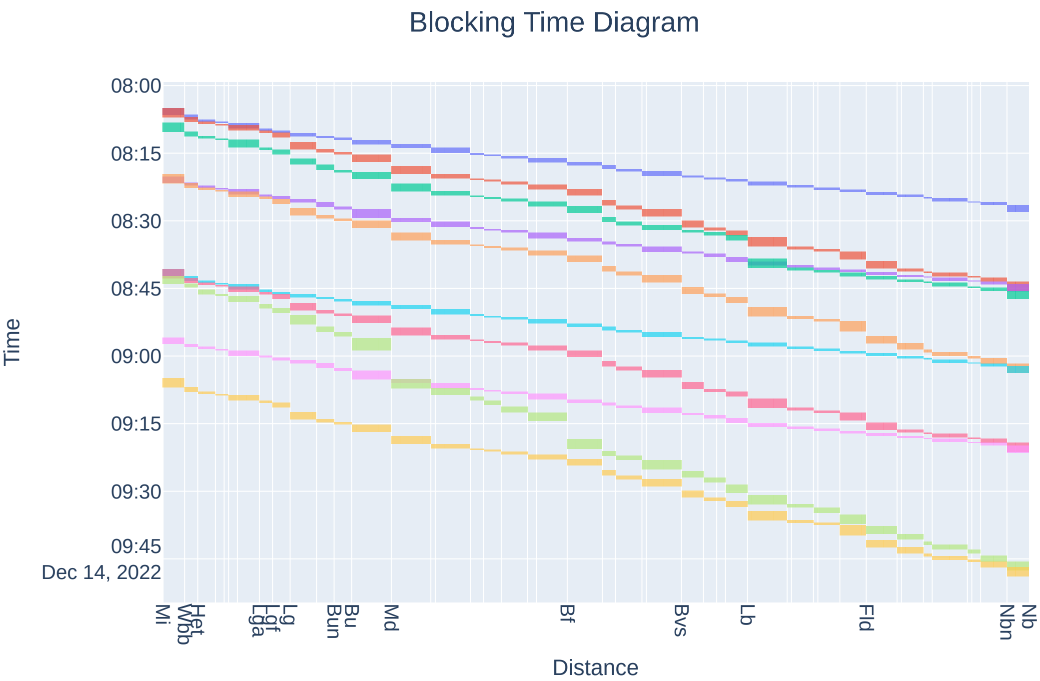


Figure 6.7: Compressed blocking time diagram for FiBlo with estimated network infrastructure.

On one hand, the estimated infrastructure approach might overestimate the available capacity in some cases by allowing parallel traversal where only one physical track exists without any headway between the trains. On the other hand, the plan infrastructure might underestimate it at long OCP-to-OCP segments, where two trains might be able to occupy this block with adequate headway, but they are modeled as a single resource with capacity limit of one.

This variance in capacity consumption underscores that they are highly sensitive to infrastructure modeling assumptions. This finding suggests that capacity studies should carefully consider the level of infrastructure detail incorporated in the model, as different approaches may yield significantly different capacity consumption estimates.

6.3.2 MoBlo Capacity Utilization Results

MoBlo is designed to work with the plan infrastructure, as it can already account for long segments with potentially multiple trains on a single track. Because trains only occupy the micro-segment they are currently on plus the ones required for their braking distance, multiple trains can be on a OCP-to-OCP segment with appropriate headway between them.

But MoBlo introduces micro-segments and the granularity of those has an impact on the capacity consumption. Figure 6.8 presents the capacity consumption for different micro-segment lengths, ranging from 16 meters to 4096 meters. Smaller micro-segments lead to lower capacity consumption because the position and braking distance are updated more often than with larger micro-segments. Additionally, the smaller segments allow the model to reserve the braking distance in finer chunks. This leads to higher fidelity, which better approximates the ideal continuous Moving Block section at the cost of greater computation demands. Larger segments require fewer updates and are computationally more efficient but sacrifice some model accuracy.

Figure 6.8 also illustrates the differences between the one-track-per-direction and plan infrastructure scenarios. Because the plan infrastructure has a quad track setup near Meidling, it allows for some overlap at those segments and therefore shorter headways between trains and less capacity consumption. The deviation between the two infrastructure scenarios is mostly less than 1 percentage point and increases to approximately 2 percentage points at the coarsest granularities. The difference is smaller than the one observed with FiBlo, especially between the estimated and the plan infrastructure.

The MoBlo simulations in this chapter utilized the train behavior parameters tuned specifically for these trains and this network subset to produce runtimes close to scheduled times. Because MoBlo does not differentiate between conventional braking and emergency braking, the movement authority gets calculated based on the calibrated braking rate. This might be too conservative as emergency braking might enable shorter braking distances and therefore shorter movement authority and less headway between trains.

Figure 6.9 shows the blocking time diagram for MoBlo with one track per direction using 256m micro-segments. Because it uses only one track throughout the corridor from Meidling to Wiener Neustadt, no block occupations overlap. The compression is primarily limited by the line section's end and start points.

6.3.3 Comparison and Discussion

MoBlo calculates the travel duration between OCPs based on the acceleration, deceleration and maximum speed. Because these derived travel times do not entirely match with the scheduled duration, there is some deviation between MoBlo and FiBlo. Additionally, both models have different concepts regarding block occupation. While FiBlo only incorporates the running time, MoBlo also includes the approach time by reserving all micro-segments

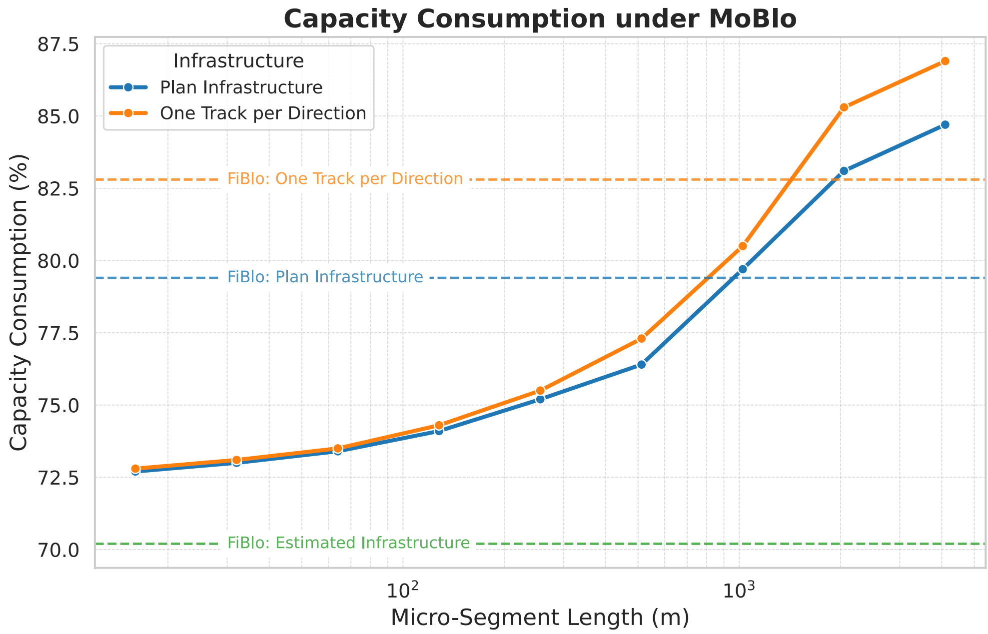


Figure 6.8: Capacity consumption under MoBlo with different micro-segment lengths and infrastructure networks

required for the train to come to a safe stop. These two differences require additional attention when comparing the capacity consumption across modeling approaches.

Because the approach time is not considered within FiBlo but is in MoBlo, it should favor FiBlo in terms of capacity consumption. The included braking distance requires trains to allocate micro-segments even before they traverse them. In FiBlo the segments are only considered occupied if the train is running through the segment and does not include additional margins for approach time that would require the segment to be blocked beforehand.

However, MoBlo's Moving Block system allows trains to only occupy the required distance and does not block entire segments. In systems like PZB, signal blocks must be longer than the braking distance, because a train needs to be able to stop between distant and main signals. This means that the braking distance in MoBlo might be shorter than the remainder of the OCP-to-OCP segment in FiBlo, allowing trains to reserve a smaller part of the track. This is an inherent advantage of the Moving Block system over the Fixed Block system and therefore allows for a better utilization of existing infrastructure.

When examining results from identical infrastructure configurations (specifically, the plan infrastructure and one-track-per-direction scenarios), MoBlo with small micro-segments yields lower capacity consumption than FiBlo. For instance, with the plan infrastructure, FiBlo reports a capacity consumption of 79.4%, while MoBlo results range from 72.7% to 84.7% depending on micro-segment length, with only the coarsest granularities exceeding

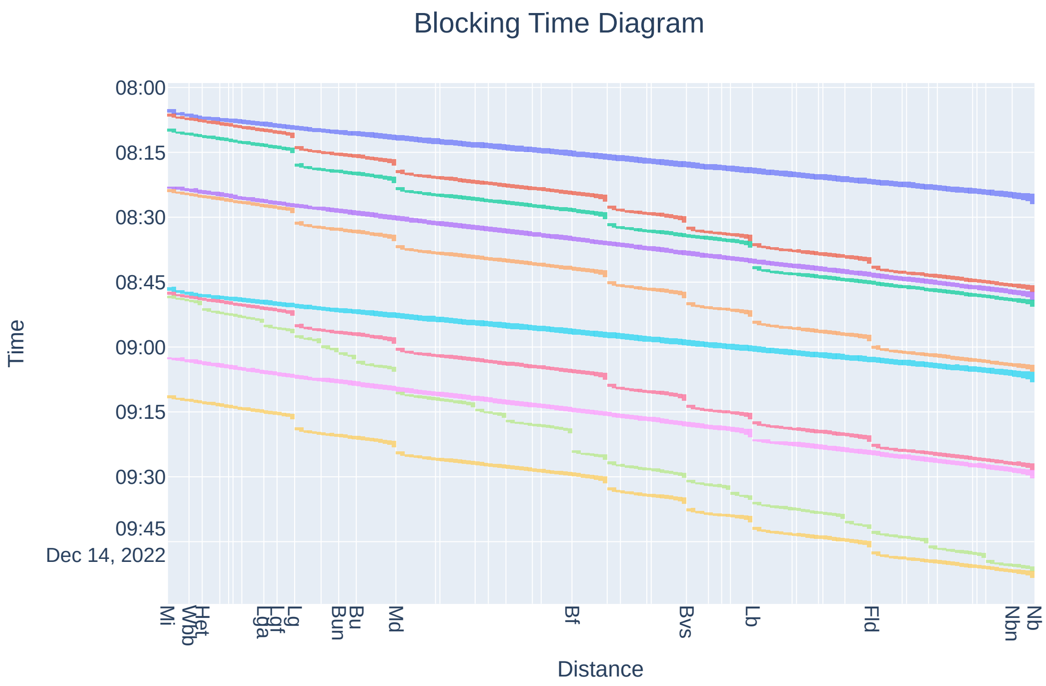


Figure 6.9: Compressed blocking time diagram for MoBlo with one track per direction, using a micro-segment length of 256m.

FiBlo's.

Assuming that travel time deviations between the two models do not systematically favor MoBlo, a reasonable assumption given the calibration process, these results suggest that MoBlo's finer-grained representation of train movements enables a more efficient use of infrastructure than FiBlo.

6.4 Delay Propagation and Runtime Results

Timetable compression looks at theoretical capacity limits of infrastructure but does not consider delay. The delay propagation experiments described in 5.7 aim to capture the differences between FiBlo and MoBlo in terms of secondary delay.

Table 6.4 presents the average runtimes and average markup values for different simulation configurations. FiBlo shows a substantial difference in average markup depending on the infrastructure. The estimated infrastructure results in a significantly better markup value than using the number of tracks as the capacity limit (plan infrastructure). Comparing MoBlo and FiBlo on the same infrastructure (plan) shows that MoBlo is less prone to secondary delay than FiBlo. Within the MoBlo Model, decreasing the micro-segment length also decreases the markup value, indicating that the higher fidelity not only improves capacity consumption (see Section 5.6) but also delay propagation.

Table 6.4: Comparison of MoBlo and FiBlo Average Runtimes and Markup based on 10 simulation runs with different pre-sampled primary delay.

System	Average Runtime (s)	Average Markup
MoBlo (by Micro-Segment Lengths)		
64 m	227.7	1.08
128 m	68.5	1.08
256 m	27.1	1.09
512 m	13.0	1.10
1024 m	7.3	1.11
2048 m	4.8	1.11
4096 m	3.9	1.13
FiBlo (by Infrastructure source)		
Estimated	2.5	1.06
Plan	2.4	1.18

6.4.1 Markup Distribution and Catch-up Mechanisms

To illustrate the distribution of markups across individual trains, Figure 6.10 presents histograms of train markups for each scenario, given a specific input primary delay. All three scenarios display a common pattern: many trains with low or no markup, and a few with really high markups. This skewed distribution highlights the non-uniform impact of delays across the network. Notably, an outlier is present in all scenarios, corresponding to a train severely impacted by a freight train with a high primary delay. Due to this train's short trip, the resulting secondary delay leads to an exceptionally high markup.

Despite all three scenarios having a similar distribution, FiBlo with the estimated infrastructure does not have any train with a markup below one. Trains are unable to recover from delays in this simulation configuration, while they can in the others.

FiBlo with the estimated network adheres strictly to the schedule, and the run and stop durations are as specified in the schedule and do not account for any buffer times. If a train is scheduled to require 5 minutes between Liesing and Mödling, the minimum duration for the corresponding task is 5 minutes. Any primary delay that is injected into the simulation is propagated until the end, and the train is unable to catch up on any delay.

For the FiBlo with plan network scenario, not only do the simulations with primary delay deviate from the schedule but also the base run. A fictional example can illustrate this: the section from Liesing to Mödling has one track for each direction, but the segment is long enough that two trains are able to occupy it with appropriate headway. In the base simulation, one train starts the journey and reserves the segment. The other train is scheduled to depart with appropriate headway but has to wait until the other train leaves the segment and therefore deviates from the schedule. In a simulation run with primary delay, if the first train is delayed long enough, it enables the second one to immediately depart from Liesing where it had to wait in the base simulation. This appears as recovering from delay within the markup calculations.

MoBlo scenarios have the ability to catch up due to discrepancies between scheduled and simulated travel times. MoBlo simulates train movement through acceleration and deceleration, which has some deviations from the scheduled duration. Assuming the scheduled duration for Liesing to Mödling is 5 minutes but the simulated duration is 4 minutes, the train arrives earlier in Mödling and waits for an additional minute until the scheduled departure time, creating a buffer time at the station. In a simulation with primary delays, these buffers can be utilized and allow trains to recover from delay.

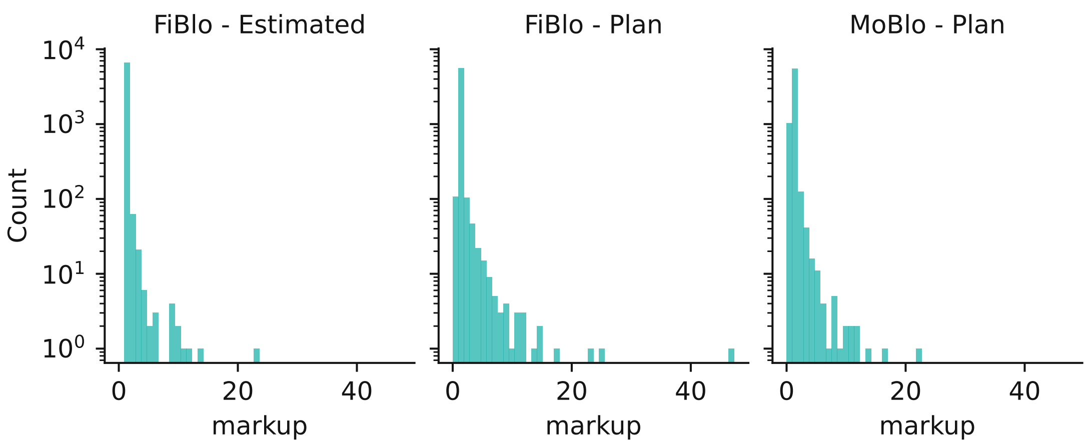


Figure 6.10: Histograms of train markups for different simulation models using one primary delay sample. MoBlo with 128m micro-segment length.

6.4.2 Computational Runtime

The FiBlo model demonstrates better runtime efficiency, with average runtimes of 2.5 and 2.4 seconds for the estimated and plan scenarios, respectively (Table 6.4). This performance can be attributed to its simplified approach: reserving exactly one segment at a time per train, using a single *Drive* task for each edge in the network, and avoiding detailed calculations of braking distances or train movements.

In contrast, the MoBlo implementation offers more detailed simulations but at a higher computational cost. MoBlo divides segments into smaller micro-segments, incorporates train movement dynamics, reserves multiple micro-segments for the full stopping distance, and performs forward-looking calculations for speed allowances. These factors contribute to longer runtimes, which vary significantly based on the chosen micro-segments length.

As shown in Table 6.4, MoBlo runtimes range from 227.7 seconds for 64-meter micro-segments to 3.9 seconds for 4096-meter micro-segments, exhibiting a clear inverse relationship between micro-segments length and runtime. Shorter micro-segments lead to longer runtimes because more micro-segments means more *Drive* tasks. This results in more frequent calculations for speed allowances and train movement. The runtime for MoBlo with 4096-meter micro-segments, while shorter than for smaller micro-segments, is higher than FiBlo. This is despite FiBlo's average OCP-to-OCP segment length being around 2 km, where most segments consist of a single micro-segments. As a result, both FiBlo and MoBlo handle a similar number of *Drive* tasks. However, MoBlo computes a speed profile to derive the travel duration, which FiBlo does not, making MoBlo more computationally intensive and resulting in longer runtimes.

CHAPTER 7

Conclusion and Outlook

This thesis presented and compared two approaches to railway simulation: the Fixed Block model (FiBlo) and the Moving Block model (MoBlo). While FiBlo operates by reserving entire segments between OCP, MoBlo divides these segments into micro-segments and incorporates train movement dynamics, including acceleration and deceleration.

The thesis explored two different methods for determining the segment capacity. The *estimated* infrastructure is based on the historic maximum number of trains observed on a segment at the same time. The *plan* infrastructure uses the number of parallel tracks. While FiBlo can use both methods, MoBlo is designed to work with the plan infrastructure, as it can already model multiple trains on the same segment, if they have the required headway and braking distance.

The research focused on four key aspects: the accuracy of train movement calibration, the effects of different train protection systems on delay propagation and capacity consumption, and the sensitivity of the MoBlo model to variations in micro-segment length.

7.1 Train Movement Calibration

The introduction of micro-segments in the MoBlo model requires a more explicit simulation of train movement. In the Fixed Block approach employed by FiBlo and RailwaySim, trains traversed OCP-to-OCP segments based on scheduled travel durations. However, with these segments divided into smaller micro-segments, the travel duration for each micro-segment is unclear. Moreover, after a stop, trains generally require more time to cover the initial hundred meters than they do for the next due to their acceleration. To accurately represent this, the model requires parameters like acceleration and deceleration. Furthermore, the movement authority within Moving Block depends on the braking distance.

In order to address these requirements, MoBlo employs calibrated parameters for acceleration, deceleration, and relative maximum speed to accurately simulate train movement. The calibration process aims to minimize deviations between simulated and scheduled travel durations across OCPs. Recognizing that different train categories may exhibit distinct acceleration and braking characteristics, these parameters are individually adjusted for each category to achieve the best results.

This led to the formulation of the first research question:

Research Question 1: *How accurately does the simulated acceleration and deceleration behavior replicate travel times between stations?*

The simplified train movement model , in combination with the calibration process , is able to approximate the travel times with varying degrees of accuracy across different train categories. The best parameter combinations showed Mean Absolute Relative Errors (MARE) between simulated and scheduled travel durations ranging from 13.5% to 33.6%. The high error rates highlight the limitations of this simple approach.

7.2 Delay Propagation

Moving Block systems, such as the one implemented in MoBlo, have the potential to improve capacity consumption by reducing headway between trains. Because FiBlo emulates a Fixed Block system and MoBlo emulates a Moving Block system, comparing the results of these two models presents an opportunity to evaluate the impact of train protection systems on secondary delays. Using an identical schedule, infrastructure, and primary delay inputs allows the thesis to reduce external influences.

Research Question 2: *How does the integration of a Moving Block system in the extended agent-based model affect delay propagation compared to the Fixed Block system in the original model?*

The delay propagation experiments showed that FiBlo and MoBlo have significantly different secondary delays, indicated by the markup values. If both models used the plan infrastructure, MoBlo results in less markup than FiBlo regardless of the micro-segment length. MoBlo reached values ranging from 1.08 to 1.13, whereas FiBlo resulted in 1.18. If FiBlo is used with the estimated infrastructure, it achieves a markup value of 1.06, even better than MoBlo with the smallest micro-segment length. This can be attributed to the more lenient segment capacity limit. With the estimated infrastructure, FiBlo allows trains to depart simultaneously if there were ever two trains recorded on the segment at the same time, whereas MoBlo enforces proper headway on single-track infrastructure.

Within the MoBlo system, different micro-segment lengths varied in their markup value ranging from 1.08 to 1.13. Shorter micro-segments enable more frequent and precise updates to infrastructure reservations based on train position and braking distances. This leads to more efficient handling of delays and lower markup values.

7.3 Capacity Consumption

Beyond examining the delay propagation effects of train protection systems, this thesis also analyzes their implications on capacity consumption. Using the subset from Meidling to Wiener Neustadt as a case study, capacity consumption was assessed with reference to the UCI leaflet 406, to discern the impact of Moving Block versus Fixed Block systems using the MoBlo and FiBlo models.

Research Question 3: *How does the adoption of a Moving Block system, as simulated in the agent-based model, impact railway capacity consumption compared to the Fixed Block system?*

The capacity consumption analysis reveals a similar pattern to the delay propagation. When comparing identical infrastructure configurations (plan infrastructure or one-track-per-direction scenarios), MoBlo yielded lower capacity consumption estimates than FiBlo for all but the coarsest micro-segment lengths.

With the plan infrastructure configuration, FiBlo reports 79.4% capacity consumption, while MoBlo estimates range from 72.7% to 84.7% depending on micro-segment length. Only the simulations with the longest micro-segments exceeded FiBlo's results. Similarly, for the one-track-per-direction scenario, FiBlo shows 82.8% capacity consumption, while MoBlo ranges from 72.8% to 86.9% across different micro-segment lengths. The reasons why shorter micro-segments resulted in better results are the same as discussed for delay propagation. Longer micro-segments result in less frequent updates, which in turn requires wider track reservations.

It is important to emphasize that these comparisons must be interpreted with caution due to fundamental differences between the two models. First, the block occupation concepts differ: MoBlo reserves not only the micro-segment a train currently occupies but also all micro-segments required for a safe braking distance, which the latter is ignored in FiBlo. Second, MoBlo's train movement model produces runtime deviations from the schedule, which can affect capacity consumption calculations.

The results suggest that Moving Block systems, when implemented with appropriate granularity, can offer capacity benefits compared to Fixed Block systems.

7.4 Sensitivity to micro-segment length

MoBlo introduced the micro-segments to better capture the dynamics of a Moving Block system and more explicitly model train movements. In theory, as the micro-segment length approaches zero, the system would emulate a truly continuous, ideal Moving Block system. However, due to computational limitations, there is a trade-off between the granularity of the micro-segment and the computation efficiency of the model.

Research Question 4: *How sensitive is the extended agent-based model to variations in key parameters like micro-segment length?*

The previous analysis showed that MoBlo produced different results depending on the micro-segment lengths. The parameter affected capacity consumption and delay propagation but also computation efficiency. For capacity consumption, shorter micro-segments showed better results. Using 4096m micro-segments resulted in approximately 12 percentage points higher capacity consumption compared to 16m micro-segments. For delay propagation, shorter micro-segment lengths resulted in lower markup values, indicating reduced secondary delay. In both cases, the shorter micro-segments led to more frequent updates to the position and speed of the train. This allows the system to reserve less track for each train, improving the headway between trains. But shorter micro-segments resulted in higher computational requirements. The average runtime for a simulation ranged from 3.9 seconds with 4096m micro-segments to 227.7 seconds with 64m micro-segments, which is a nearly 60-fold increase in computational cost.

7.5 FiBlo Segment Capacity

An additional aspect of this research emerged in the preceding analysis: the impact of segment capacity in FiBlo. The study employed two distinct methodologies to estimate the capacity of a segment: a heuristic approach based on the maximum number of concurrently observed trains in historical data, and a limit based on the number of tracks in plan infrastructure data.

The results showed a high sensitivity, with an impact on both the capacity consumption and delay propagation. The estimated approach resulted in a capacity consumption of 70%, while the plan infrastructure indicated 79%, differing by 9.2 percentage points. FiBlo with an estimated network also outperformed FiBlo with the planned network in regard to delay propagation (1.06 vs 1.18 in markup).

These differences emphasize the sensitivity of segment capacity to delay propagation and capacity consumption analysis within FiBlo.

7.6 Model Choice

The choice of model depends on the specific simulation requirements.

The FiBlo model allows for fast macroscopic simulation and is based on a Fixed Block system. It is limited in its ability to simulate Moving Block systems and is sensitive regarding segment capacity estimation.

MoBlo emulates a Moving Block model and is better suited for simulations involving a train protection system implementing Moving Block or virtual subsections. The micro-segment length can have a huge impact. Small micro-segments better capture the benefits of Moving Block but require more computational power. Overall, simulations using large micro-segments can deviate strongly from real Moving Block implementations and should be avoided.

7.7 Future Work

While this thesis has made contributions to understanding railway network simulation through the comparison of Fixed Block and Moving Block approaches, several avenues for future research remain:

1. **Advanced Train Dynamics:** The current train movement calibration process is too simplistic to replicate travel times between OCPs. Future work should focus on adding more detailed infrastructure and traction unit data, including gradient profiles and an acceleration/deceleration model. This could improve simulation accuracy, by allowing more detailed calculations.

2. **Better Train Categorization:** Train movement in MoBlo is calibrated for a limited number of train categories. Developing a more detailed system that accounts for variations within each category (e.g., different traction units with varying performance characteristics) could improve model accuracy.

3. **Validation with Real-World Delay Data:** Comparing simulated delay propagation patterns with actual historical delay data would provide valuable validation of both models.

Overview of Generative AI Tools Used

In the development of this thesis, Generative AI (GenAI) tools were employed as assistive technologies to enhance both the writing process and code development. These tools were utilized only as complementary resources to refine my work rather than as primary content generators.

For the textual components of this thesis, GenAI (GPT-4o and Claude 3.7) was employed as an advanced writing assistant. The process typically involved:

- Drafting original content and ideas myself based entirely on my research and analysis
- Providing the AI with an outline and draft of my thoughts
- Utilizing the AI to refine sentence structure, improve readability, and enhance clarity
- Employing it as a sophisticated grammar checker to identify potential linguistic issues and suggest alternative phrasings
- Occasionally having the AI summarize my writing to verify that my intended message was clearly conveyed

It is important to emphasize that all intellectual contributions, research findings, analyses, and conclusions presented in this thesis are entirely my own. The AI tools were used exclusively to polish the expression of these ideas rather than to generate the substantive content itself.

In the technical implementation aspects of this thesis, GitHub Copilot was utilized as a programming assistant. Its application was limited to small utility functions and visualizations.

All algorithms, problem-solving approaches, and technical implementations reflect my understanding and expertise. GitHub Copilot served merely as a productivity enhancement tool that reduced time spent on repetitive coding tasks.

List of Figures

List of Tables

Acronyms

ABM Agent Based Modeling. 10

DES Discrete Event Simulation. 10, 12

ETCS European Train Control System. 1–3, 6, 8–10, 14, 23

MA Movement Authority. 9

OCP Operational Control Point. 1–3, 15, 16, 19, 20, 23–26, 29, 33–38, 40, 42, 47, 48, 54–59, 63, 65, 73, 74

PZB Punktförmige Zugbeeinflussung. 1, 2, 6–9, 14, 29, 39, 59, 73

RBC Radio Block Centre. 9

Bibliography

[BBGS12] Richard Bloomfield, Robin Bloomfield, Ilir Gashi, and Robert Stroud. How Secure Is ERTMS? In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, Frank Ortmeier, and Peter Daniel, editors, *Computer Safety, Reliability, and Security*, volume 7613, pages 247–258. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. Series Title: Lecture Notes in Computer Science.

[BBP+23] Dominik Brunmeir, Martin Bicher, Niki Popper, Matthias Rößler, Christoph Urach, Claire Rippinger, and Matthias Wastian. Four Years of Not-Using a Simulator: The Agent-Based Template. In *2023 Winter Simulation Conference (WSC)*, pages 255–266, December 2023. ISSN: 1558-4305.

[CM12] Yong Cui and Ullrich Martin. Multi-scale Simulation in Railway Planning and Operation. *PROMET - Traffic&Transportation*, 23(6):511–517, February 2012.

[CML18] Yong Cui, Ullrich Martin, and Jiajian Liang. PULSim: User-Based Adaptable Simulation Tool for Railway Planning and Operations. *Journal of Advanced Transportation*, 2018(1):7284815, 2018. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2018/7284815.

[EW24] Stefan Engels and Robert Wille. Towards an Optimization Pipeline for the Design of Train Control Systems with Hybrid Train Detection (Short Paper). *OASIcs, Volume 123, ATMOS 2024*, 123:12:1–12:6, 2024. Artwork Size: 6 pages, 1232048 bytes ISBN: 9783959773508 Medium: application/pdf Publisher: Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[GCD13] Rob M. P. Goverde, Francesco Corman, and Andrea D'Ariano. Railway line capacity consumption of different railway signalling systems under scheduled and disturbed conditions. *Journal of Rail Transport Planning & Management*, 3(3):78–94, August 2013.

[HB22] Johan Högdahl and Markus Bohlin. A Combined Simulation-Optimization Approach for Robust Timetabling on Main Railway Lines. *Transportation Science*, 57, September 2022.

[Hü02] Daniel Hürlimann. *Objektorientierte Modellierung von Infrastrukturelementen und Betriebsvorgängen im Eisenbahnwesen.* PhD thesis, ETH Zurich, 2002. Artwork Size: 139 p. Medium: application/pdf.

[Jan19] Joost Jansen. *ERTMS/ETCS Hybrid Level 3: A simulation-based impact assessment for the Dutch railway network.* PhD thesis, TU Delft, 2019.

[KOF23] Daniel Knutsen, Nils O.E. Olsson, and Jiali Fu. ERTMS/ETCS Level 3: Development, assumptions, and what it means for the future. *Journal of Intelligent and Connected Vehicles*, 6(1):34–45, March 2023. Conference Name: Journal of Intelligent and Connected Vehicles.

[KOF24] Daniel Knutsen, Nils O. E. Olsson, and Jiali Fu. Capacity evaluation of ERTMS/ETCS hybrid level 3 using simulation methods. *Journal of Rail Transport Planning & Management*, 30:100444, June 2024.

[KOFR25] Daniel Knutsen, Nils O. E. Olsson, Jiali Fu, and Tomas Rosberg. Capacity evaluation of ERTMS/ETCS HTD and moving block. *Journal of Rail Transport Planning & Management*, 33:100506, March 2025.

[Law15] Averill M. Law. *Simulation modeling and analysis.* McGraw-Hill series in industrial engineering and management science. McGraw-Hill Education, New York, NY, fifth edition edition, 2015.

[LKSST06] A. Landex, A. H. Kaas, B. Schittenhelm, and J. Schneider-Tilli. Practical use of the UIC 406 capacity leaflet by including timetable tools in the investigations. In *Computers in Railways X*, volume 1, pages 643–652, Prague, Czech Republic, June 2006. WIT Press. ISSN: 1743-3509, 1746-4498.

[LSKST08] A. Landex, B. Schittenhelm, A. H. Kaas, and J. Schneider-Tilli. Capacity measurement with the UIC 406 capacity method. In *Proceedings of the 11th International Conference on Computers in Railways*, pages 55–64, Toledo, Spain, August 2008.

[Mac16] C M Macal. Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, 10(2):144–156, May 2016. Publisher: Taylor & Francis _eprint: https://doi.org/10.1057/jos.2016.7.

[Mas22] Ulrich Maschek. *Sicherung des Schienenverkehrs: Grundlagen und Planung der Leit- und Sicherungstechnik.* Springer Fachmedien, Wiesbaden, 2022.

[MK15] David Masad and Jacqueline Kazil. Mesa: An Agent-Based Modeling Framework. In *SciPy*, pages 51–58, Austin, Texas, 2015.

[MN05] C.M. Macal and M.J. North. Tutorial on agent-based modeling and simulation. In *Proceedings of the Winter Simulation Conference, 2005.*, pages 14 pp.–, December 2005. ISSN: 1558-4305.

[Pac21] Jörn Pachl. *Systemtechnik des Schienenverkehrs: Bahnbetrieb planen, steuern und sichern*. Springer Fachmedien, Wiesbaden, 2021.

[Qua11] Egidio Quaglietta. *A microscopic simulation model for supporting the design of railway systems: development and applications*. PhD thesis, Università degli Studi di Napoli Federico II, 2011. Medium: application/pdf.

[Qua14] Egidio Quaglietta. A simulation-based approach for the optimal design of signalling block layout in railway networks. *Simulation Modelling Practice and Theory*, 46:4–24, August 2014.

[RH04] A Radtke and D Hauptmann. Automated planning of timetables in large railway networks using a microscopic data basis and railway simulation techniques. *Computers in Railways IX*, 2004.

[RT20] Tomas Rosberg and Birgitta Thorslund. Simulated and real train driving in a lineside automatic train protection (ATP) system environment. *Journal of Rail Transport Planning & Management*, 16:100205, December 2020.

[RWJ+20] Matthias Rößler, Matthias Wastian, Anna Jellen, Sarah Frisch, Dominic Weinberger, Philipp Hungerländer, Martin Bicher, and Niki Popper. Simulation And Optimization Of Traction Unit Circulations. In *2020 Winter Simulation Conference (WSC)*, pages 90–101, December 2020. ISSN: 1558-4305.

[RWLP18] Matthias Rößler, Matthias Wastian, Michael Landsiedl, and Niki Popper. AN AGENT-BASED MODEL FOR ROBUSTNESS TESTING OF FREIGHT TRAIN SCHEDULES. In *Proceedings of the 17th International Conference on Modeling & Applied Simulation (MAS 2018)*, 2018.

[Sip23] Hans Sipilä. Simulations with PROTON and RailSys : Use of a macroscopic and microscopicrailway simulation tool in Swedish applications. *TRITA-ABE-RPT.*, 2023.

[SP21] Andreas Schöbel and Olga Permiakova. Enhancing capacity on ETCS Level 2 lines in Austria. In *6th International Conference on Road and Rail Infrastructure*, pages 1111–1117, May 2021.

[SRK+24] Nadine Schwab, Matthias Röβler, Hannah Kastinger, Günter Schneckenreither, Matthias Wastian, and Niki Popper. Validation and Quantification of Possible Model Extensions for a Railway Operations Model Using Delay Data Disaggregation. In *2024 Winter Simulation Conference (WSC)*, pages 1563–1574, December 2024. ISSN: 1558-4305.

[UIC13] UIC. uic406_2013.pdf, 2013.

[Ver20] Ruben Vergroesen. *ERTMS/ETCS Hybrid Level 3 and ATO: A simulation based capacity impact study for the Dutch railway network*. PhD thesis, TU Delft, 2020.

[Vie24] Maximilian Viehauser. *Simulation-Based Disaggregation of Train Delay Data*. Thesis, Technische Universität Wien, 2024. Accepted: 2024-10-17T13:47:46Z Journal Abbreviation: Simulationsgestützte Disaggregation von Zugverspätungsdaten.

[WS20] Maximilian Wirth and Andreas Schöbel. Mindestzugfolgezeiten bei ETCS Level 2 und Level 3 auf der Wiener S-Bahn-Stammstrecke. *Signalling+ Datacommunication (112)*, 2020.

[ZBB+19] Markus Zinser, Torsten Betz, Maurice Becker, Michael Geilke, Carla Terschlüsen, Adam Kaluza, Ingrid Johansson, and Jennifer Warg. PRISM: A Macroscopic Monte Carlo Railway Simulation. In *The 12th World Congress on Railway Research (WCRR)*, 2019.

[ZBW+18] Markus Zinser, Torsten Betz, Jennifer Warg, Emma Solinen, and Markus Bohlin. Comparison of microscopic and macroscopic approaches to simulating the effects of infrastructure disruptions on railway networks. *Proceedings of 7th Transport Research Arena TRA*, 2018.