applied sciences

MDPI

Article

Determining the Optimal Positions of Infill Balise Groups for ERTMS/ETCS Level 1 Applications

Christopher Wink * and Nils Nießen

Institute of Transport Science (VIA), RWTH Aachen University, Mies-van-der-Rohe-Str. 1, 52074 Aachen, Germany; niessen@via.rwth-aachen.de

* Correspondence: wink@via.rwth-aachen.de

Featured Application: This paper presents a fully working, easily usable algorithm for planning the placement of the trackside balise groups for ETCS level 1 applications to transmit infill information to the passing trains by minimizing the operational impact.

Abstract: This paper presents a method to determine the optimal positions of infill balise groups for applications of European Train Control System (ETCS) level 1. ETCS will be the only train protection system in the European Union of the future. Level 1 specifically enables a less complex system while retaining full compatibility to and interoperability of ETCS level 2. In level 1, the Movement Authority (MA), which contains information on how far and at what speed the train is authorized to run, is transmitted at fixed locations where switchable radio beacons—infill balise groups—are placed. Due to technical and economical constraints, their locations need to be optimized so that they provide the maximum operational benefit. Such a method with universal adaptability has not been available to date, so we developed an algorithm that determines the optimal locations of infill balise groups. The algorithm uses the basic laws of motion combined with domain knowledge about railways in general and specifically ETCS. All important parameters are considered while retaining simplicity in the application, making it convenient to use for planning new ETCS deployments and as a foundation for future research on ETCS level 1. The algorithm was implemented using Python and tested with real-world train characteristics.

Keywords: railway; train protection system; ETCS; ERTMS; infill; balise group; optimization

check for updates

Citation: Wink, C.; Nießen, N. Determining the Optimal Positions of Infill Balise Groups for ERTMS/ETCS Level 1 Applications. *Appl. Sci.* **2024**, *14*, 8165. https://doi.org/10.3390/app14188165

Academic Editors: Valerio De Martinis and Raimond Matthias Wüst

Received: 29 July 2024
Revised: 4 September 2024
Accepted: 5 September 2024
Published: 11 September 2024

1. Introduction

1.1. Introduction to ETCS

The European Train Control System (ETCS) is part of the European Rail Traffic Management System (ERTMS) and provides a harmonized, standardized Automatic Train Control (ATC) system for interoperable rail transport in the European Union (EU) [1]. Interoperability has many facets, including track gauge, overhead lines, and signaling systems. ETCS is intended to be the single train protection system in the EU in the future and is therefore the sole Class A System of train control systems. Someday, when the whole infrastructure, all the traction units, control cars, etc., are equipped with ETCS, only this single system will be needed onboard and trackside, massively reducing complexity and potentially enabling better economies of scale than the current proprietary national systems (Class B Systems). As such, ETCS defines the target state of a train control system for most of the European railways, both trackside and trainside, replacing the existing variety of national, proprietary train protection and train control systems.

All terms and abbreviations in this paper are—as long as they directly concern ETCS—used in accordance with their published specifications [2].

The current state of specifications has a long history of standardizing the subsystems of rail transport in Europe over the last 30 years. Council Directive 96/48/EC [3] laid the groundwork for interoperability in the railway sector in 1996 by introducing the concept of Technical Specifications for Interoperability (TSI), starting with high-speed lines. In 2001, this was extended to include conventional rail lines with Directive 2001/16/EC focusing on trans-European transport [4]. Three years later, both directives were amended and concretized by Directive 2004/50/EC [5] and then recast in 2008 by Directive 2008/57/EC [6]. More recently, Directive 2012/34/EU [7] laid the groundwork for and established the term "single European railway area", stressing the importance of overall interoperability in the railway sector. The aim is to have a unified railway network where national borders are not relevant for or hinder the trans-European railway transport of passengers and freight. In 2016, a major revision and recast of the interoperability of the railways in the European Union was adopted by the new, still-in-force Directive (EU) 2016/797 [8].

In itself, ETCS is a positive train control system as it uses a Movement Authority (MA) to authorize a train to move until a specified location within a specified speed envelope [9,10]. Additional information may be provided by ETCS as well. In contrast, a negative train control system allows for unhindered movement as long as no restriction—approaching a stop signal, for example—is imposed on the train. All necessary information of ETCS for the train driver is presented on the harmonized Driver Machine Interface (DMI) [11] in the driver's cab, making ETCS a cab signaling system. This information includes the actual speed, the currently permitted speed, the target speed, and the target distance. The need for a MA results in the need to update the currently valid MA to further authorize the movement. This process is carried out by providing "infill information" from the trackside to the train, which updates the MA. Under normal circumstances, infill extends the current MA.

The information required on the train can either be transmitted by means of balises (electronic beacons at fixed locations in the track) or by means of radio. Transmitting by means of balises is defined as ETCS (application) level 1, whereas transmitting via radio is defined as ETCS (application) level 2. Level 2, per the latest specifications [10], includes the former ETCS level 3, where information is transmitted via radio and the train integrity is checked on-board, so no trackside train detection system is needed and moving block operations are possible. For radio transmission, Global System for Mobile Communications — Railway (GSM-R) and Future Railway Mobile Communication System (FRMCS) can be used, with the first approaching its technical end of life and the latter set to be its enhanced replacement [12].

For ETCS level 1, it is specified in [13] that infill information can be transmitted to the train by means of balise groups (Eurobalises), Euroloop, or radio infill. Their respective specifications can be found in [14–17]. Eurobalises only transmit information at distinct locations (spot transmission), whereas Euroloop and radio infill provide a continuous transmission of infill over a limited distance, e.g., in the approach zone of a main signal. This potentially reduces the system latency in responding to a main signal changing from a "stop" aspect (closed main signal) to a "clear" aspect (open main signal) and therefore increases capacity. Although both Euroloop and radio infill have been implemented and are in use today on select infrastructures, recent deployments of ETCS level 1 all exclusively use balise groups to transmit infill information, indicating a better cost-to-performance or complexity-to-performance ratio than those of semi-continuous solutions. As a downside, providing infill information to a train is only possible at distinct locations where the infill balise groups are located. The current TSI for control-command and signaling systems [18] do in fact prohibit the new deployment of Euroloops and Radio Infill Units (RIUs) with specific, minimal exceptions. This highlights the need to optimize the locations of the infill balise groups to increase capacity. The literature review and work below were therefore focused solely on using standard Eurobalises to transmit infill information via the so-called air gap [19] from the trackside infrastructure to the train.

Most of the European countries implementing ETCS have focused on deploying ETCS level 2, as it benefits the infrastructure manager in further reducing the necessary lineside equipment such as full lineside signaling to just static marker boards. However, such lines need to have redundant GSM-R or FRMCS equipment for the necessary continuous radio communication in place—expenditures many non-federally owned infrastructure managers are not willing or able to afford. Unfortunately, high costs due to the complexity of the system and especially due to the need to upgrade or completely replace existing interlockings, the rollout has to date been much slower than anticipated. For example, the National Implementation Plan of Germany [20] published in 2017 does not reflect the current state of deployment anymore. The current migration strategy [21] for the network of Germany's federally owned infrastructure manager DB InfraGO AG covers the planned commissioning of ETCS until the end of 2029. Until then, lines with a combined length of 5090 km shall be equipped with ETCS, which equals around 15% of the total line length of 34,065 km and covering most of the core parts of the network with the highest number of trains. The non-federally owned infrastructure managers, of which Germany has about 190 [22], are therefore to date cautious with the implementation of ETCS, as long as no concrete planning is underway for DB's connecting lines.

Their preferred option is to deploy a low-complexity, cost-effective solution to enable interoperable operations and therefore plan to implement ETCS level 1 Full Supervision. In the neighboring country of Austria, federal railway Österreichische Bundesbahnen (ÖBB) has commissioned around 300 km of ETCS level 2 and will increase that number to about 3700 km until 2038 to cover their core network [23]. The remaining network might in part be fitted with ETCS level 1. Banedanmark, the state-owned infrastructure manager in Denmark, aims to have ETCS level 2 deployed on all lines by the year 2033 [24]. Countries with (almost) complete coverage of active ETCS on their standard gauge networks include Switzerland and Luxembourg. The first one today uses a combination of ETCS levels 2 and 1 (the latter in the operating mode "Limited Supervision"); the last one makes use of ETCS level 1 Full Supervision exclusively for different reasons, presented by Arend et al. [25].

With modern, digital interlockings directly controlling the switchable balise groups [26], such as those in service in Luxembourg since 2022 [27,28], as well as other trends identified by Stadlbauer [29], ETCS level 1 still holds a future for less demanding network segments and enables automation features. As there is an increasing number of applications of ETCS level 1 even outside of Europe—for example, in Thailand [30]—the impact might be much broader than meets the eye at first sight.

1.2. Literature Review

The research and literature covering ETCS level 1 in particular is sparse. Most papers concerning ETCS focus on level 2, which uses radio to transmit the MA and updates directly to the train. The focus to date lies on determining the capacity of ETCS level 2 compared to legacy train control systems [31] and how to increase this capacity. Papers focussing on ETCS level 1 or other, spot-transmission, balise-based train control systems are rarely published. The few instances where the capacity of ETCS level 1 was calculated and compared to other ETCS levels or legacy systems often do not disclose the exact method of where and why infill balise groups were placed. A complete algorithm for determining where to optimally place infill balise groups taking the current System Requirements Specifications (SRS) into account as well as the train characteristics is not publicly available. The engineering guidelines of infrastructure managers are generally neither available to entities outside their organization nor well documented in terms of transferability to other applications. The following literature review is therefore highly limited in the number of references, as we were not able to obtain any more sources concerning the aim of our research.

Wendler [32] defined the "buffer time equivalent"—elaborating on the work of Schwanhäußer [33]—and used it to calculate optimal distances of infill balise groups. He derived rules of thumb for the placement, as different parameters yielded similar results and ratios between the distance to the End of Authority (EOA) and the optimal placement of infill balise groups. This method unfortunately is neither fully documented nor easily accessible to and applicable by persons responsible for planning the positions of infill balise groups.

Moreover, given the place in time this study has been conducted, the redesigned and from then on harmonized braking curves introduced with version 3.0.0 of the ETCS SRS at the end of 2008 were not available then. This leads to a trainside system behavior different from the one found on most traction units today. The results, therefore, cannot be fully translated and applied anymore.

A study commissioned by UIC to the Institute of Transport Science at RWTH Aachen University [34] calculated the capacity consumption of trains under the supervision of different levels of ETCS on different generic lines and led to a revision of the Union Internationale des Chemins de fer/International Union of Railways (UIC) Code 406 [35]. For the base calculations of ETCS level 1, one infill balise group was placed at the indication point of the train with the longest braking distance. In another scenario, a second infill balise group was added 400 m ahead of each main signal. Compared to level 2, the scenario of level 1 with one additional infill balise group proved to be between 3% and 12% worse in terms of capacity. The authors note that the practical benefit of level 2 might be higher with increasing variance of train characteristics and therefore braking distances. As further scenarios with varying numbers and positions of infill balise groups were not investigated, and major change requests were implemented in the ETCS specifications—especially concerning the calculation of the braking curves—additional research might be able to clear up the picture as to when each ETCS level performs better than the other.

In a following study by VIA Consulting & Development GmbH from Aachen on behalf of UIC, the capacity effects of ETCS on railway nodes were investigated [36]. Once again, the maximum number of infill balise groups in level 1 was set to two. Their positions were set at the maximum Indication Point (IP) and—if two infill balise groups were assumed—additionally at the mean IP of the trains covered. All scenarios covering level 2 implemented speed changes at the switch, whereas level 1 scenarios did not, although ETCS level 1 Full Supervision is fully capable of that. Taking all of this into account, the differences in capacity of no more than 5% and less than 3%, respectively, in favor of level 2 over level 1 indicate potential for level 1, even in larger nodes. Additional research into the effect of the placement of infill balise groups might help to bridge the existing gap of information.

The latter two of the aforementioned studies did not aim at optimizing the positions of infill balise groups on a microscopic level and therefore used approximations to efficiently position them for the network segments investigated. It is possible that in some cases, a microscopic optimization of the infill balise groups using an algorithm like the one presented in this paper could have yielded slightly better performance results.

Shanker [37] presented the concept of using basic probability and the laws of motion to approximate the position of a single infill balise group. He introduced the idea of segmenting the approach zone based on the runtime being equal in each of these segments. The method described by Shanker does not take the acceleration phase after receiving infill information into account, but focuses entirely on the braking phase up to the EOA. The achievable acceleration, however, can have significant impact on the operations and capacity after a train had to apply its brakes. No testing with real-world data was performed to verify the model. In summary, this method needs to take additional parameters into account to be applicable for planning purposes considering different train types.

A calculation based on a mathematical model using approximated braking and acceleration curves was presented by Nikolov et al. [38]. To simplify the approach, they used different models for constant as well as variable deceleration and acceleration. The software developed calculates the optimal position of a single infill balise group in addition to one at the distant signal. No real-world train characteristic was used for testing purposes. This proposed method is limited to a constant or parabolic deceleration and acceleration and therefore is not adaptable to closely model the ETCS braking curves. Only a single infill balise group can be positioned and apparently no Key Performance Indicator (KPI) or optimization target to evaluate different settings has been defined.

Yin et al. [39] determined the positioning of balises in the context of urban metro systems using a genetic algorithm. However, their aim was to optimize the stopping positions of passenger trains at their respective platforms and not to optimize the operations when approaching a closed main signal about to open.

1.3. Contribution of This Paper

There is—to the best of our knowledge—no completely published method available for the optimal placement of infill balise groups for ETCS level 1 applications that satisfies the requirements for determining the number and exact positions of infill balise groups in real-world applications. Placing infill balise groups at non-optimal positions potentially has significant negative effects during operations, as it could increase the headway times unnecessarily when a train needs to slow down because the track section ahead is not yet clear. The results from the literature review yield some interesting starting points; however, most of them do not take all the important aspects, like system latency and sometimes the acceleration phase, into account.

We therefore propose a method to optimally place infill balise groups and present it in the following sections. Our method combines different theoretical approaches from existing research such as segmenting the approach zone based on the balise groups and using the basic laws of motion as Shanker [37] did, and approximating the acceleration and deceleration curves akin to Nikolov et al. [38] but on a much more granular scale. Our work contributes to existing research by providing a comprehensive algorithm and program enabling researchers and engineers to model a wide variety of scenarios with ample customizability and minimal limitations. It provides the capabilities of realistically modelling the train characteristics as well as the infrastructure in an easy-to-use format not available to date. With our work, it is now not only possible to determine the optimal locations of infill balise groups but also to precisely quantify the effect of non-optimal locations on the operations in applications of ETCS level 1. Moreover, it enables benchmarking other strategies and existing rules of thumb for placing infill balise groups against the optimum locations. Finally, our work can be used in the context of capacity assessments of ETCS based on the latest specifications by providing researchers a tool to automate placing infill balise groups on a microscopic level, so far often performed with approximate or global settings.

1.4. Structure of the Paper

The paper itself is structured into five sections. After this introductory section, we present the basic idea of our research and define the scope of our work in Section 2. After that, the algorithm is presented together with its input parameters before we point out our implementation. The algorithm developed and the initial results from testing and first applications will be presented in Section 3. We discuss and evaluate different weighting methods, build a test case, and present its results. From the results, we can draw conclusions on the effects of different input parameters on the optimal positions of infill balise groups. After that, we discuss our work in Section 4. The results produced are compared to other research findings, and we highlight similarities as well as differences and their probable causes. In addition, the current limitations of the algorithm are pointed out and possible future work is derived from that. Finally, we will conclude this paper with Section 5, where we give a condensed overview of our work and highlight possible applications for our work.

2. Materials and Methods

2.1. Algorithm

This section covers the fundamentals of our algorithm. Information on the implementation in Python is given in Section 2.2.

The aim is to develop a fast, easy-to-use algorithm to calculate the optimal positions of infill balise groups in ETCS level 1 on a microscopic level. Track and train characteristics are to be factored in and shall be easily customizable by the user. The calculation is based on the additional runtime a train endures when the following block section is not free and available, and it therefore needs to apply its brakes. We do not consider scenarios where the train comes to a complete standstill. As updating the MA is only possible at the discrete locations of infill balise groups, their positions highly influence the knock-on effects of such a scenario.

The driving dynamics of the train(s) involved play an important role; therefore, braking and acceleration performance must be appropriately modelled to obtain robust results. Each calculation focuses on a single train and optimizes the infill balise groups for only this specific train. Additional calculations necessary to find a global optimum of the infill balise group locations when dealing with different kinds of trains with differing driving dynamics and therefore differing optima must—for now—be performed outside the algorithm presented here. A high-level flowchart of the fundamental algorithm is presented in Figure 1.

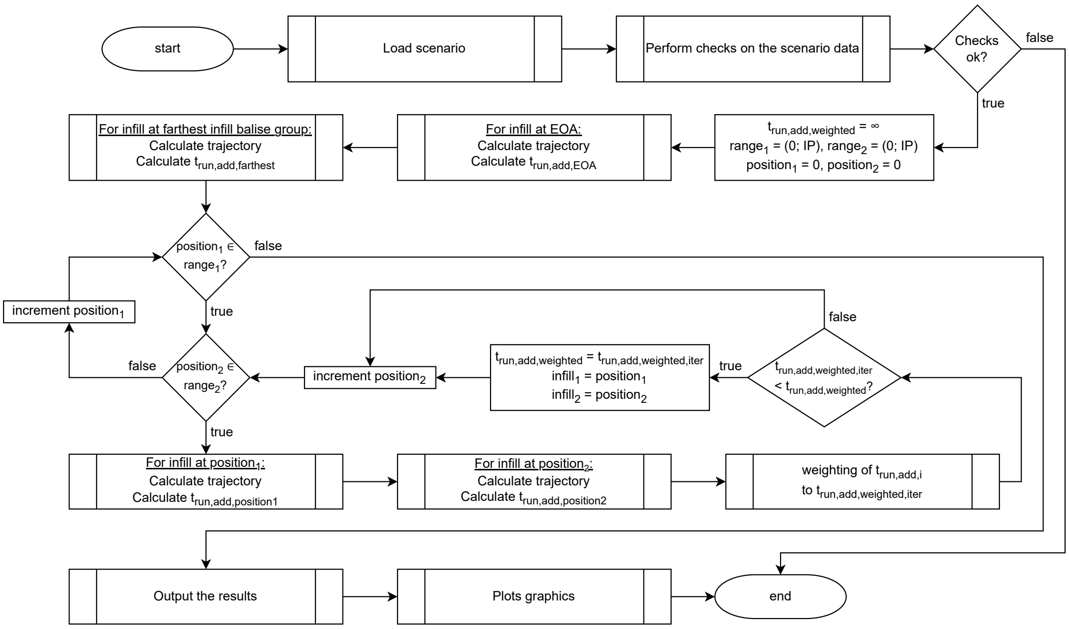

start
Load scenario
Perform checks on the scenario data
Checks ok?
false
true
$t_{run,add,weighted} = \infty$
$range_1 = (0; IP)$, $range_2 = (0; IP)$
$position_1 = 0$, $position_2 = 0$
For infill at EOA:
Calculate trajectory
Calculate $t_{run,add,EOA}$
For infill at farthest infill balise group:
Calculate trajectory
Calculate $t_{run,add,farthest}$
$position_1 \in range_1$?
false
true
increment $position_1$
false
$position_2 \in range_2$?
true
increment $position_2$
$t_{run,add,weighted} = t_{run,add,weighted,iter}$
$infill_1 = position_1$
$infill_2 = position_2$
true
$t_{run,add,weighted,iter} < t_{run,add,weighted}$?
false
For infill at $position_1$:
Calculate trajectory
Calculate $t_{run,add,position1}$
For infill at $position_2$:
Calculate trajectory
Calculate $t_{run,add,position2}$
weighting of $t_{run,add,i}$ to $t_{run,add,weighted,iter}$
Output the results
Plots graphics
end

Figure 1. High-level flowchart of the algorithm to determine the optimal positions of the infill balise groups.

The KPI the algorithm optimizes for is the weighted additional runtime $t_{run,add,weighted}$ of a single train. The method of weighting the runtimes is described in Section 2.1.4. The additional runtime $t_{run,add}$ is defined as the difference in runtime between a train passing unhindered at the maximum allowed speed without any braking, leading to the shortest runtime possible on the segment in question $t_{run,unhindered}$, and the runtime when applying the brakes and—after receiving an updated MA at one of the discrete locations of an infill balise group—accelerating to the maximum allowed speed $t_{run,hindered}$:

$$t_{run,add} = t_{run,hindered} - t_{run,unhindered}. \tag{1}$$

By minimizing the additional runtime, we roughly approximate the effect this has on the headway time between two trains and therefore the practical capacity of a line.

Infrastructure wise, we consider an ETCS system where the placement of infill balise groups is highly flexible. We therefore do not need to adhere to the positioning of fixed lineside signals. Also, we do not consider restrictions resulting from big metal masses, etc., which limit the positions where balises can be positioned. However, as the algorithm is designed in a way to respect certain restrictions, an iterative approach for most of the realistic scenarios can be used without alterations of its core needed. As the minimal working scenario, the infrastructure consists of a main signal balise group at the EOA and one balise group serving infill information. The latter must be placed farther from the

EOA than the IP of the train modelled. This complies with the principle of enabling the train to have an unhindered journey if the next block section is free and available. Up to two additional infill balise groups can be placed between the EOA and the farthest infill balise group. This leads to up to three infill balise groups ahead of the EOA in addition to the main signal balise group at the EOA. Both the line speed and the track gradient are constant over the area covered by the algorithm. The EOA is marked by a harmonized ETCS marker board [40]. An example of this setup is presented in Figure 2.

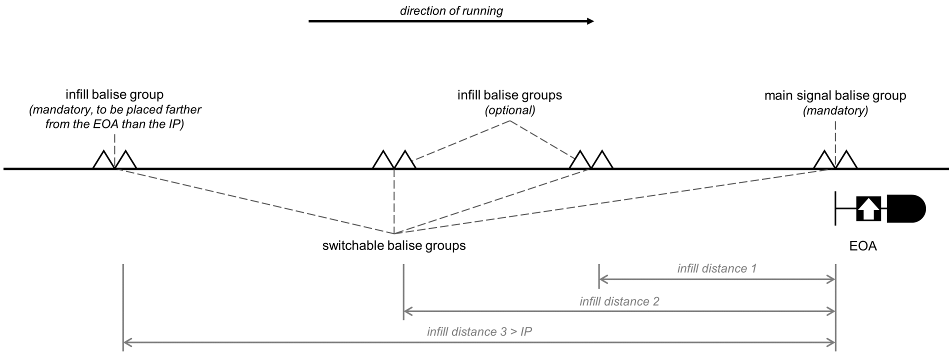


Figure 2. Schematic of the infrastructure modelled.

As part of the initial plausibility checks, the algorithm examines whether the farthest infill balise group satisfies the requirement of being placed farther from the EOA than the IP and then optimizes the locations of the other, additional infill balise group(s). This is carried out by finding the position or positions of infill balise groups where the weighted additional runtime of a specified scenario is minimal, and therefore, the lowest operational impact of the track section ahead not being free on time is to be expected. The additional runtime is calculated for each possible trajectory corresponding to a position or a combination of positions of infill balise groups. The trajectories are constructed from the ETCS braking curves—in general, the Indication Curve—as the deceleration curve and the acceleration a specific train can archive. How the ETCS braking curves can be optimized for best usage has been worked on by Busse [41], for example. This is outside this paper's scope.

The individual train trajectories modelled can be segmented into the running regimes cruising, deceleration, and acceleration. Trains are not able to transition from deceleration to acceleration and vice versa in an instant. This is due to the nature of the braking systems and the time needed to build up or release the braking force. Train length, brake position, and the type of the braking system are the most important factors influencing the build-up and release times. The same is true for the traction effort; however, modern electric locomotives have significantly better longitudinal agility and dynamics than their predecessors or types with an internal combustion engine. Both effects combined accumulate to roughly less than 10 s for a typical passenger train. This minimum time during the transition between these two regimes is modelled as cruising. To be physically correct, it should be classified as coasting, as neither braking nor traction effort is applied and the forces in play all reduce the train's speed. For practical reasons, this simplification has been made. The ratio of mass to resistance is large enough that for the short time spans in question here, cruising instead of coasting can be used. The typical trajectory of a train needing to apply its brakes and receiving infill consists of the following phases:

- approaching the IP at the maximum allowed speed (cruising);
- slowing down according to the braking curve starting at the IP until passing an infill balise group providing a new MA (deceleration);
- processing the information (deceleration or cruising);
- transitioning from deceleration to acceleration (cruising);
- accelerating to the maximum allowed speed (acceleration);
- clearing the investigation area at the maximum allowed speed (cruising).

Whether processing the information takes place while decelerating or cruising is determined by the train's speed. When the train has already reached the Release Speed (RS) before passing the infill balise group or reaches the RS while processing, it does not slow down further but transitions to cruising. Receiving infill information at any speed higher than that, the train will have processed the information before reaching the RS and a transition to cruising takes place immediately. With these prerequisites, we need to consider running with constant speed and running with constant acceleration or deceleration.

2.1.1. Input Parameters

The basic input parameters shown in Table 1 are necessary to perform the optimization. These are categorized into track parameters and train parameters.

Table 1. Necessary input parameters for the algorithm to perform the optimization.

Category	Parameter	Unit
Track	Line Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$
	Release Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$
	Gradient	[‰]
	Balises	[-]
	Balise Group Distance	[m]
	Balise Positions	[m]
Train	Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$
	Acceleration	$[\mathrm{m}\cdot\mathrm{s}^{-2}]$
	Deceleration	$[\mathrm{m}\cdot\mathrm{s}^{-2}]$
	Rotating Masses	[%]
	Indication Point	[m]
	Minimum Cruising Time	[s]
	Processing Time	[s]

The parameters were chosen to balance between the minimum amount necessary to perform the optimization and the desire to be flexible in respect to the scenarios that can be covered by the algorithm presented. Therefore, not all requirements from [42] are explicitly modelled, but can be respected by means of the available input parameters. These parameters impact the exact optimal positions of the infill balise groups as they each influence the possible trajectories of the train, be it through changes in the braking curves, the deceleration, the acceleration, or the exact position where data are transmitted to the train. The applicable National Values (NV) for a particular question need to be considered in preprocessing, as they can have significant impact on the shape of the braking curves and therefore can shift the optimal positions of infill balise groups.

2.1.2. Track Parameters

Trackside information needed to perform the optimization includes the line speed and the gradient as fixed parameters of the track alignment. Both parameters are considered to be constant over the whole section of track examined.

As we consider ETCS level 1 without continuous infill by means of Euroloop or RIU, a RS higher than 0 km/h must be permitted. This enables trains to obtain a new MA or update their existing MA at the EOA after receiving clearance via optical (lineside signal) or digital (using a private or public wireless network directly to the train or to a personal mobile device) means. Typical values for the RS range from 5 km/h to 40 km/h.

The algorithm needs to have knowledge of some boundary conditions regarding the balise groups used to transmit infill information. First, the number of infill balise groups must be set. Currently, the algorithm is designed to handle two or three infill balise groups besides the necessary main signal balise group at the target (EOA). Fewer balise groups (meaning zero or one) most often do not have practical relevance and no optimization is possible. The installation of four or more infill balise groups yields ever diminishing marginal returns. The algorithm presented in this paper could, however, be adapted and applied to those scenarios as well. Balise groups need to be a certain distance apart to allow the train passing over them to read and process all information transmitted before receiving further information. The exact minimum distances vary depending on the speed and the number of single balises inside each of the balise groups and can be obtained from [42] in conjunction with [14]. Typical values for speeds up to 160 km/h range between 3 m and 15 m. A global, constant value can be specified and will limit the freedom on where the flexible balise groups can be placed. The algorithm ensures that this lower boundary is respected for all balise groups. Finally, positions of infill balise groups to be placed at fixed locations are the last input parameter of the category "track". At least one balise group—the one farthest away from the target and in rear of the Indication Point of the reference train—must be set. Depending on the total number of infill balise groups (two or three), the remaining balise groups can either be fixed as well with a position in rear of the target or left flexible for the algorithm to optimize their locations.

2.1.3. Train Parameters

The necessary parameters of the train to perform the calculation can be grouped into the three categories of driving dynamics, ETCS braking performance, and latency.

The delays for processing the balise group information between receiving the balise group message and updating the status on-board according to [43] are considered by adding the so-called "processing time" of the European Vital Computer (EVC) and the DMI, which adds latency of the specified amount to the modelled system.

The braking curves are obtained using the official European Union Agency for Railways (ERA) braking curves simulation tool [44]. With that approach, the complexity of the algorithm developed can be limited to the core functions needed. We therefore can calculate trains both modelled as lambda trains as well as gamma trains [45]. Gamma trains are fixed train compositions for which the exact braking capabilities and properties are known. The braking performance is modelled by brake deceleration values as a step function over the train's speed. Lambda trains are the norm to model the braking behavior of changing compositions. The calculation is based on the known braking percentage of the train. Using the braking percentage of the train as well as its braking position and train length, the UIC Conversion Model [46] is used to calculate the estimated braking deceleration as a function of the speed.

With this approach, possible future changes in the calculation of the braking curves can easily be integrated. Different scenarios using different supervision limits—for example, the Indication Curve for conventional trains driven by a human driver and the Permitted Curve for automated trains—can be calculated without the need to develop and rewrite essential parts of the algorithm. Even completely different spot-transmission train protection systems could be modelled inside the developed framework if they can be abstracted using approximated braking and acceleration curves.

2.1.4. Weighting the Runtimes

The weighting is used to calculate the weighted additional runtime $t_{run,add,weighted}$ as the KPI of the optimization. The algorithm calculates the runtime on each trajectory and individually compares them to the trajectory of the unhindered run, resulting in the additional runtime $t_{run,add,i}$ for each trajectory associated with receiving infill information at a specific balise group.

Figure 3 will be used to illustrate the approach behind the weighting. The green trajectory depicts the unhindered run at maximum speed after having received infill information at balise group "IF_3". Because infill information is received before passing the IP, no braking must be initiated. The three other trajectories in this case are associated with receiving infill information—after having initiated braking—at "IF_2" (yellow), at "IF_1" (red), and at the EOA (purple). Calculating the additional runtime observes the train running from the farthest infill balise group (IF_3 in this example) until the point where all trajectories have reached the initial speed again, depicted by the dashed vertical line on the far right in Figure 3.

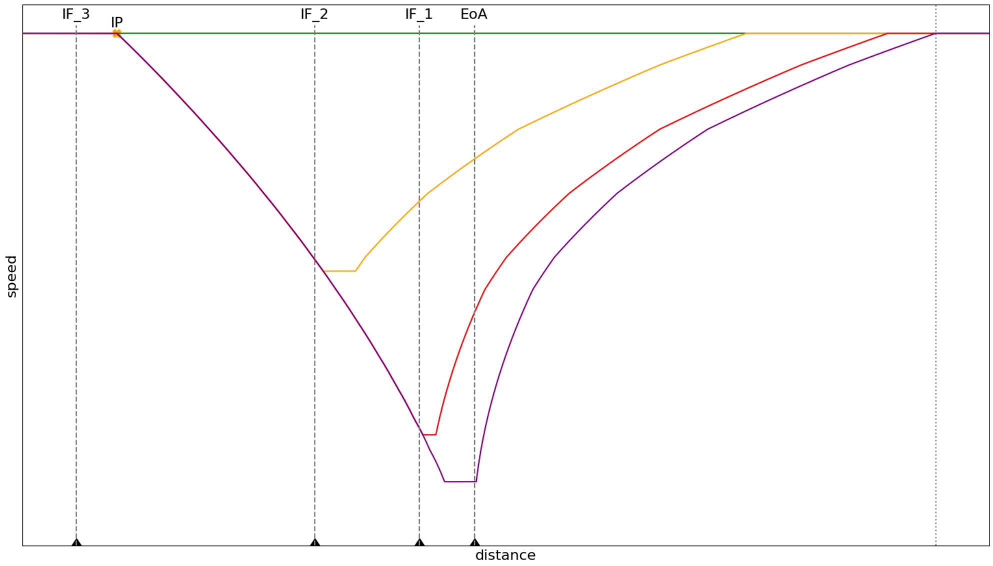


Figure 3. Qualitative example of train trajectories depending on the infill balise group where infill information is received. The direction of travel is from left to right. The colors correspond to receiving infill information at IF_3 (green), IF_2 (yellow), IF_1 (red), or at the EOA (purple).

The approach zone between the farthest infill balise group IF_3 and the EOA is divided into three segments used to determine the individual weights g_i depending on the weighting method. The first segment spans from IF_3 to IF_2, the second one from IF_2 to IF_1, and the last one from IF_1 to the EOA. If only two infill balise groups are present, just two segments would be distinguished.

The weighted additional runtime is calculated by using the individual weights g_i for a weighted arithmetic mean, as shown in the following equation:

$$t_{run,add,weighted} = \frac{\sum_{i=1}^{n} g_i \cdot t_{run,add,i}}{\sum_{i=1}^{n} g_i}. \tag{2}$$

Three different methods of defining the individual weights g_i were developed and implemented: “Time”, “Distance”, and “Equal”.

The “Time” method uses the runtime of the slowest trajectory (purple in Figure 3) in each of the segments as the corresponding weight. The weights change with different positions of the infill balise groups. By this, a system behavior is modelled in which the time the main signal changes from closed to open and infill information becomes available is unknown and completely random timewise.

Taking not the runtimes but the distances between the balise groups (length of the segments) into account is possible by selecting the “Distance” method. The weights then change with different positions of the infill balise groups as well. This leads to a slight, but important change in view compared to the first weighting method, as it describes a system behavior in which the moment the main signal changes from closed to open is unknown and randomly distributed over the distance of the train run between the farthest infill balise group and the EOA.

At last, “Equal” enables the user to apply an equal weighting to the segments, so the weighted additional runtime is the ordinary arithmetic mean of the additional runtimes corresponding to each of the segments. There is no analogy to the real world to be drawn.

It must be noted that the calculated weighted additional runtime in all cases only takes trains into account that do not receive an updated MA before passing the farthest infill balise group, and therefore, ETCS Target Speed Monitoring (TSM) becomes active and supervises the braking curves. This only becomes relevant in disturbed conditions, as normally, timetabling ensures that all trains can run on a "green wave" without hitting a closed main signal outside planned stops.

2.2. Implementation

The algorithm was implemented using Python in version 3.11.9. Additional Packages used—in alphabetical order—are enum, json, logging*, math, matplotlib, numpy, os*, pandas, progress*, and time*. The packages indicated by an asterisk are not necessary for the actual computation but rather provide comfort features and additional user-facing information. The source code of around 1100 source lines of code is split into five modules:

- optimize_infill.py: main function with input, iterative optimization, and output of the results;
- checks.py: performs checks on the input data provided to raise errors on implausible parameters;
- calc_movements.py: contains the functions which each calculate one specific type of movement;
- trajectory.py: calculates granular time-distance data to be used to plot the trajectories;
- plots.py: prepares the plots described in Section 2.3.

Each of the modules may contain several functions specifically designed to handle small parts of the calculation to ensure readability and maintainability. The source code with comments in German in the version used to prepare this article is publicly available under the GNU General Public License 2.0 [47].

As the result of preparatory research, we decided not to use an optimizer, but focused on an iterative approach. Considerations on processing times as well as complexity led to this decision. We landed on a two-iteration process in which first, all possible and allowed positions of the infill balise groups to be optimized are tested with a larger step size (defaults to 10 m). In the second iteration, a precise localization with an accuracy of 1 m in the vicinity of the former, less accurate optimum is performed. Both times, the iteration starts at the EOA and progressively shifts the infill balise groups towards the IP opposite to the direction of running. A criterion to terminate the iterations early is not implemented to date, as it would deny valuable insights on how the system behaves in a specific scenario. It can be added when the algorithm shall be used outside of science in a production environment.

The program is based on basic physics and the laws of motion to calculate the runtimes. We explicitly did not want to design a complex runtime calculator. To model non-constant accelerations and decelerations, we chose an approximation of sectional constant values. Hence, the driving dynamics are to be provided as step functions of the acceleration and deceleration values dependent on the train's speed. The effect of rotating masses f_p is taken into account with the standard gravity g_0 when a non-zero gradient i is applied and changes the input nominal acceleration and deceleration $a_{nominal}$ to the corresponding effective acceleration and deceleration $a_{effective}$:

$$a_{effective} = a_{nominal} - \frac{g_0 \cdot i}{1 + f_p}. \tag{3}$$

Table 2 expands Table 1 and presents the input parameters of the current implementation of the algorithm. Additionally, technical parameters to customize the program are included. Due to Python's nature of dynamic typing, some variables are either processed as an integer or a float depending on the input provided. All input parameters are held structured in a JavaScript Object Notation (JSON) file and therefore are easily accessible and customizable, even to a future overlaid automation to batch-process different scenarios. This might be used to automate the planning process of all infill balise groups of one station or larger portions of an infrastructure. When the number of fixed infill balise groups matches the total number of infill balise groups, no optimization is possible. Rather, an evaluation of the specified positions regarding the reference train is performed. This enables users to compare different scenarios from other sources in a common application.

Table 2. Input parameters of the algorithm's implementation in Python.

Category	Parameter	Unit	Variable	Data Type
Track	Line Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$	line_speed	integer
	Release Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$	release_speed	integer
	Gradient	[‰]	gradient	integer/float
	Balises	[-]	balises	integer
	Balise Group Distance	[m]	balise_group_distance	integer
	Balise Positions	[m]	balise_positions	list of integers
Train	Speed	$[\mathrm{km}\cdot\mathrm{h}^{-1}]$	speed	integer
	Acceleration	$[\mathrm{m}\cdot\mathrm{s}^{-2}]$	acceleration: steps/values	object of two integer-/float-arrays
	Deceleration	$[\mathrm{m}\cdot\mathrm{s}^{-2}]$	deceleration: steps/values	object of two integer-/float-arrays
	Rotating Masses	[%]	rotating_masses	integer/float
	Indication Point	[m]	indication_point	integer
	Minimum Cruising Time	[s]	min_cruise_time	integer/float
	Processing Time	[s]	processing_time	integer/float
Tech	Steps	[m]	steps	integer
	Weighting	[−]	weighting	string ∈ {"TIME"; "DISTANCE"; "EQUAL"}
	Plot Trajectories	[−]	plot_trajectories	Boolean
	Plot 3d contour	[−]	plot_3d	Boolean
	Rotate 3d contour	[−]	rotate_plot	Boolean
	Locale	[−]	locale	string ∈ {"en"; "de"}

The parameters of the category "Tech" are used to control parts of the program as well as the desired outputs. The "Steps" specify the initial step size for which the weighted additional runtime shall be calculated. Larger values mean a less granular but faster calculation. Inside the interval where the first run has detected the minimum value, a second run with a fixed step size of 1 m is performed to obtain the results. For most applications, an initial step size of 10 m has been near optimal in terms of compute time. How the single additional runtimes from the different trajectories are weighted to obtain the final, weighted additional runtime as the KPI is specified by "weighting", cf. Section 2.1.4. The two following Boolean values "plot trajectories" and "plot 3d contour" switch between plotting and not plotting the graphs, as presented in Figures 4 and 5, respectively. The latter of these two can be plotted as 360 single frames to combine them to a rotating animation of the three-dimensional visualization by setting "rotate 3d contour" to "True". Finally, the user can decide whether the labels and further information on the plots shall be printed in English or German.

2.3. Output

The most important outputs of the program are the positions of the optimized infill balise group locations and the corresponding weighted additional run time. To better understand the results and enable further use of them, different output files can be generated:

- input parameters and key results as a JSON file;
- two-dimensional plot of the trajectories (optional);
- three-dimensional plot of the weighted additional runtime over all possible distances (optional);
- 360 single frames of the three-dimensional plot to generate an animation (optional).

The JSON file contains all the input parameters and appended the positions of the infill balise groups and the additional runtime, cf. Table 3. This enables an easy comparison of different scenarios, even at a later point in time.

Table 3. Output parameters of the algorithm as the result of the optimization.

Parameter	Unit	Variable	Data Type
Infill Positions	[m]	infill_positions	list of integers
Weighted Additional Runtime	[s]	additional_-runtime	float

As one of the graphical outputs, the calculated trajectories overlaid with the optimal positions of the infill balise groups accompanied by key parameters and results can be plotted as shown in Figure 4. In the example provided, three infill balise groups were placed, leading to four different trajectories. The green trajectory serves as the baseline where the train travels at the maximum allowed speed without the need to brake because the block section ahead is free and the main signal at position 0 m is open. This trajectory is only possible if the updated MA containing the information of the open main signal is available at the farthest infill balise group (IF_1) at position −1500 m (meaning 1.5 km ahead of the main signal) when the train passes over it. If this information is not available at that time, the ETCS On-board Unit (OBU) will invite the driver to apply the brakes starting at the IP, situated at position −1400 m. The next possible infill (IF_2) is found at position −665 m and would allow the yellow trajectory. One can distinguish the processing time after passing the infill balise group, during which the OBU processes the information received but has not yet displayed the new MA to the train driver, so the braking continues. After that, the cruising time of constant speed while transitioning from braking to acceleration is visible as the horizontal part of the trajectory. If infill can be provided by IF_1, the train will follow the red trajectory, and if a new MA is available at the EOA, the train must slow down all the way to RS and will follow the violet trajectory, leading to the largest additional runtime.

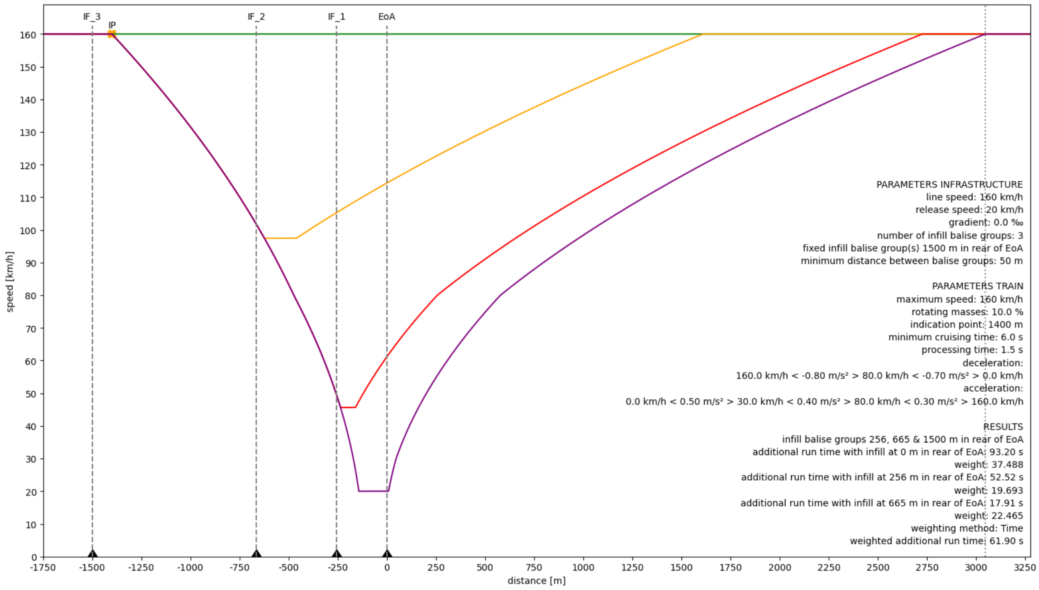


Figure 4. Example of the two-dimensional plot of the trajectories. The direction of travel is from left to right. The colors correspond to receiving infill information at IF_3 (green), IF_2 (yellow), IF_1 (red), or at the EOA (purple).

When transferring the theoretical optimal results to the real world or when dealing with more than one type of train the infrastructure shall be optimized for, the question arises of how significant the impact of (slightly) shifting the positions of the infill balise groups on the additional runtime might be. This question can be answered by calculating the weighted additional runtime for each possible and allowed position of the infill balise groups and determining the increase in weighted additional runtime when moving away from the globally optimal positioning.

Graphically, Figure 5 allows one to accomplish exactly that. This three-dimensional shape was produced by the same input parameters as before and depicts the weighted additional runtime on the vertical axis. Please note that this axis does not start at 0 s for readability reasons. The two horizontal axes represent the positions of the two infill balise groups to be positioned. The third infill balise group was fixed at position 1500 m as before to allow a timely update of the MA before reaching the IP. From the shape of this contour, it is obvious that we deal with a clear global minimum of the additional runtime. By further investigating, the slope around this optimum is quite low, indicating a broad range of positions where practically no significant increase in the additional runtime is to be expected. We conclude that we deal with a good-natured system behavior, which helps when planning the positions of the infill balise groups as restrictions apply. These restrictions in real-world applications might stem from switches, large metal objects, and other constraints.

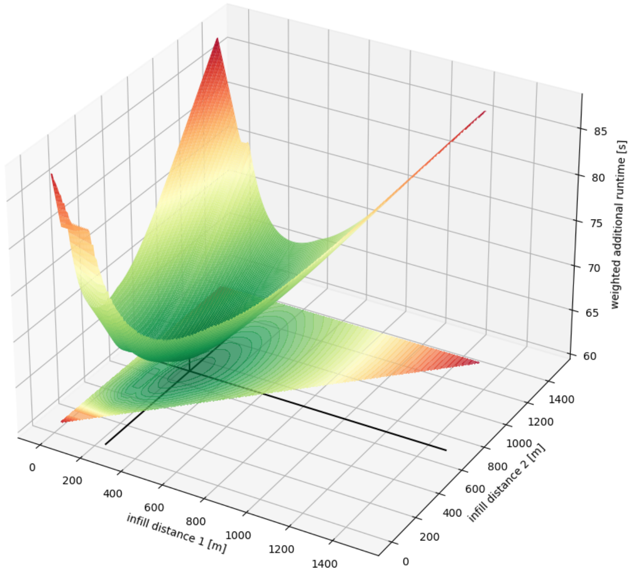


Figure 5. Example of the three-dimensional plot showing the additional runtime over all possible positions of the infill balise groups. The color gradient indicates low additional runtimes in green and high additional runtimes in red.

2.4. Optimization for Multiple Trains

When dealing with multiple trains with differing speeds or driving dynamics, the positions of the infill balise groups need to be determined in a way that overall best results are achieved. The farthest infill balise group always needs to be placed based on the largest IP of all trains expected so that it is technically possible to allow an unhindered journey if the following section is free. The next infill balise group's positioning depends on the number of train types to be considered and the total number of infill balise groups possible or desired. If two train types are present, the second infill balise group can be placed at the IP or Permitted Point of the second train type. If more than two train types need to be considered, the second infill balise group can either be placed as presented above or at the mean IP of all trains. A weighting factor based on the traffic mix can be applied. Any additional infill balise groups should then be placed in such a manner that the operational impact measured by the KPI "weighted additional runtime" is minimized.

These considerations are not yet part of the algorithm presented in this paper and need to be developed in detail, analyzed, and later implemented. The version of the algorithm presented here then can be used to evaluate the weighted additional runtime for each train type individually.

3. Results

3.1. Evaluating the Weighting Methods

We have developed three weighting methods presented in Section 2.1.4. These three methods normally yield different results in terms of the positioning of the infill balise groups. We therefore evaluated them to narrow down their application scenarios. One thing to note is that the calculated weighted additional runtimes cannot be compared across the different weighting methods. To positively compare the methods, the resulting positions of the infill balise groups must be used as input for another weighting method to calculate the corresponding weighted additional runtime.

During testing, it became obvious that using the "Equal" method yields neither realistic nor applicable results. Due to the nature of the weighting method, all infill balise groups are placed closest to the IP with their spacing only limited by the required minimum distance. With that, the algorithm achieves its optimum, as the trajectories receiving infill at those locations endure additional runtimes in the order of only a few seconds. The slowest trajectory with "infill" at the EOA cannot be optimized but always remains the same with its associated additional runtime. We therefore strongly advise against using the method with equal weighting of the individual additional runtimes.

The method "Distance" yields promising results, close to that of the "Time" method. Upon further inspection, it returns slightly worse results in nearly all scenarios tested. Furthermore, the working of this method cannot be linked to a system behavior in the real world, as it would mean randomly distributing the probability of infill being available over the distance between EOA and the farthest infill balise group.

In conclusion, only the weighting method "Time" can satisfy the requirements while also having a representation in the real world. It models a random behavior regarding the time the next section becomes available and the MA is updated via infill information at the next infill balise group passed, allowing the train to accelerate to its initial speed again.

With this framework now set, testing and sensitivity analyses of different parameters influencing the positions of infill balise groups can be performed.

3.2. Test Case

3.2.1. Setup

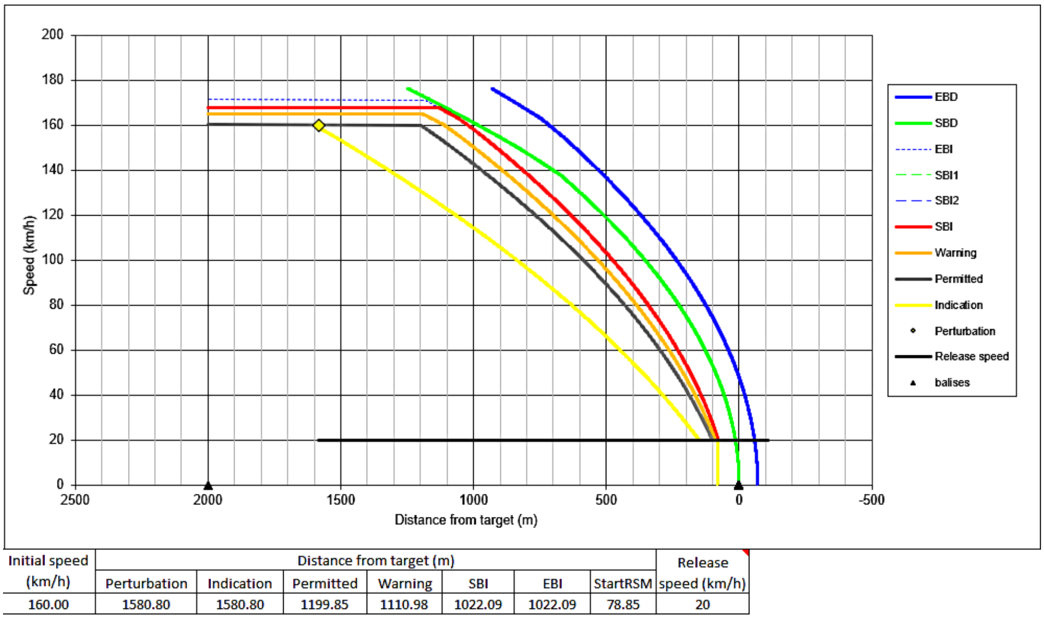
To test the algorithm and to quantify the effect of different parameters, a test case using one train was designed. The calculation itself takes less than 1.5 s to complete on a standard notebook with an i7-10510U processor and 16 GB of system memory running on Windows 11. Additional time is needed to plot the desired graphical outputs.

The line is considered to be entirely flat and to allow speeds of up to 160 km/h with a fixed RS of 20 km/h. The train is modelled after a recent regional Electric Multiple Unit (EMU) deployed in various German regions. It has an overall length of about 70 m, a total weight of around 120 t, and can apply up to 2.6 MW of traction power, leading to a fast acceleration. The acceleration values were derived from the recognized and proven runtime calculator of the software LUKS® [48] in version 3.8.1. The deceleration values used for the test case are derived from the indication curve of this particular vehicle modelled in the official ERA braking curves simulation tool [44]. We use the current NV of Germany's largest infrastructure manager DB InfraGO AG [49] and model the train as a passenger train in brake position "P" with 200 brake percentages. The resulting values of the step functions for acceleration and deceleration from this preprocessing are presented in Table 4.

Table 4. Step functions approximating the deceleration and acceleration of the train used for testing.

Speed [km/h]	Acceleration [m/s^2]	Deceleration [m/s^2]
[0; 10]	0.926	−0.072
(10; 20]	0.926	−0.194
(20; 30]	0.926	−0.295
(30; 40]	0.926	−0.379
(40; 50]	0.926	−0.450
(50; 60]	0.926	−0.512
(60; 70]	0.926	−0.565
(70; 80]	0.926	−0.612
(80; 90]	0.694	−0.653
(90; 100]	0.556	−0.690
(100; 110]	0.556	−0.723
(110; 120]	0.463	−0.753
(120; 130]	0.463	−0.780
(130; 140]	0.347	−0.804
(140; 150]	0.347	−0.826
(150; 160]	0.309	−0.833

The IP is taken from the braking curves simulation tool as well, as shown in Figure 6. The rotating masses are not used, as the input data already take them into account and no gradient is present. As a minimum cruising time, 6 s is set. The processing time of the onboard unit is set to 1.5 s. Balise groups are allowed to be as close as 50 m to each other.



Initial speed (km/h)	Distance from target (m)							Release speed (km/h)
	Perturbation	Indication	Permitted	Warning	SBI	EBI	StartRSM	
160.00	1580.80	1580.80	1199.85	1110.98	1022.09	1022.09	78.85	20

Figure 6. Example of the ETCS braking curves for the train modelled with an initial speed of 160 km/h. Screenshot from [44]. The direction of travel is from left to right.

3.2.2. Results

As a first test, we evaluated the effects of different initial speeds and the number of infill balise groups on the optimal positions as well as the weighted additional runtime. We chose initial speeds from 40 km/h to 160 km/h in increments of 10 km/h—as typically, permitted speeds are engineered with that increment, although ETCS now allows for increments of as low as 5 km/h—and two or three infill balise groups. The farthest infill balise group was positioned by taking the calculated IP and adding the distance travelled in 4 s at maximum line speed, leading to its position shown in Table 5. In the same table, the compressed results from the 26 calculated scenarios are also presented.

Table 5. Results from the optimization of the infill balise groups for the test train.

		Two Infill Balise Groups		Three Infill Balise Groups	
Speed [km/h]	Indication Point [m]	Positions [m]	Weighted Additional Runtime [s]	Positions [m]	Weighted Additional Runtime [s]
40	283	146 – 327	15.1	146 – 216 – 327	13.8
50	359	183 – 415	20.2	145 – 244 – 415	17.9
60	441	198 – 508	24.4	144 – 267 – 508	21.5
70	529	215 – 607	28.3	143 – 291 – 607	24.8
80	622	239 – 711	31.7	141 – 319 – 711	27.8
90	721	256 – 821	34.9	140 – 342 – 821	30.7
100	826	274 – 937	38.2	139 – 366 – 937	33.7
110	937	303 – 1059	41.4	185 – 439 – 1059	36.6
120	1054	322 – 1187	44.7	194 – 483 – 1187	39.5
130	1176	342 – 1320	48.0	201 – 520 – 1320	42.4
140	1305	364 – 1461	51.5	219 – 577 – 1461	45.6
150	1439	397 – 1606	55.0	227 – 618 – 1606	48.8
160	1581	420 – 1759	56.7	243 – 705 – 1759	52.0

The first result that meets the eye are the differences in the additional runtime between the scenarios with two infill balise groups versus three of them. In this setting, the absolute delta is calculated to be between 1.3 s at 40 km/h and 6.3 s at 150 km/h. The relative delta lies between 8.2% and 12.1% in favor of three infill balise groups. It does not surprise us that more infill balise groups lead to lower additional runtimes, as an additional location is introduced where infill information can be transmitted to the train, but the benefit most likely does not outweigh the costs when dealing with only one train type. If the train characteristics become more diverse, the additional infill balise group will be able to significantly improve the overall system performance. For example, two infill balise groups might be placed to optimally serve two different train types with largely different braking capabilities and a third one providing infill near the EOA.

Still comparing two infill balise groups to three of them, the optimized location of the infill balise group closest to the EOA in the two-infill-balise-group scenario always lies between the two optimized locations of the two infill balise groups closest to the EOA in the three-infill-balise-group scenario.

One can see that the IP increases proportionally more than the optimal positions of the infill balise groups when increasing the initial speed. It seems that positions of infill balise groups closer to the EOA are more favorable than ones farther out. Calculating the ratio between the infill balise groups and the IP at their respective speeds leads to a ratio between 0.52 and 0.27 for the scenario of two infill balise groups, the former for slow speeds, the latter for high speeds. When considering three infill balise groups, the two ratios come out at 0.76 to 0.45 and 0.52 to 0.15, respectively. This can be explained by the relatively small additional runtime at the beginning of the braking, i.e., close after passing the IP. The more the train must slow down, the longer it takes to accelerate to the initial speed and the greater the additional runtime becomes. Therefore, the minimal additional runtime can be found when placing the infill balise groups in the space laid out by the ratios calculated above. Notably, the ratios seem to stabilize at the upper end of the speed spectrum.

Figure 7 depicts the optimal positions when assuming an initial speed of 160 km/h and using three infill balise groups at 243 m, 705 m, and 1759 m ahead of the EOA. This leads to a weighted additional runtime of 52.0 s compared to an unhindered journey, as depicted in the green trajectory.

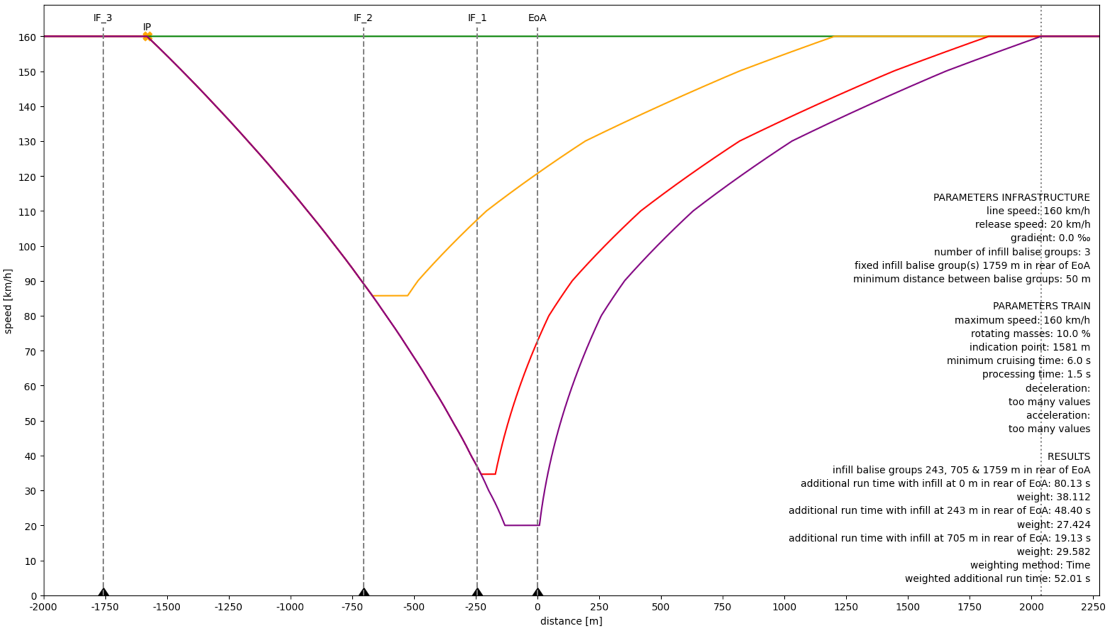


Figure 7. Plot of the trajectories with three optimally placed infill balise groups for the test train. The direction of travel is from left to right. The colors correspond to receiving infill information at IF_3 (green), IF_2 (yellow), IF_1 (red), or at the EOA (purple).

Comparing the results to the scenario with only two infill balise groups at 420 m and 1759 m ahead of the EOA, the weighted additional runtime increases by 4.7 s or 9.0% to 56.7 s, as shown in Figure 8.

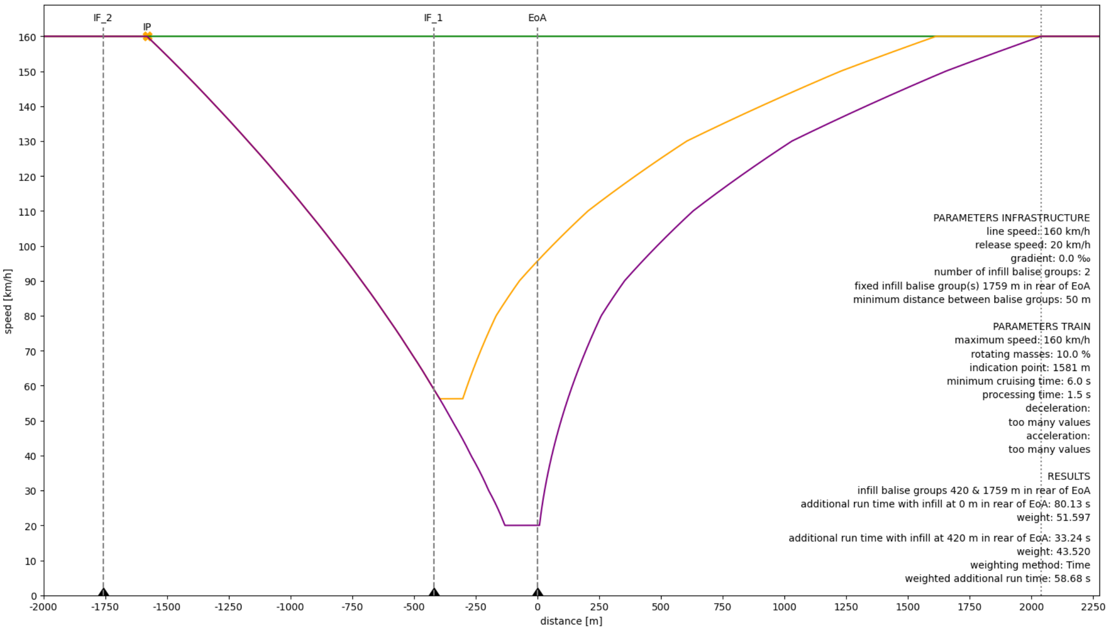


Figure 8. Plot of the trajectories with two optimally placed infill balise groups for the test train. The direction of travel is from left to right. The colors correspond to receiving infill information at IF_2 (green), IF_1 (yellow), or at the EOA (purple).

To gain further insights, we evaluated the weighted additional runtime from manually placing the infill balise groups according to five different strategies and comparing the results to the optimal positions. The first two use the fixed locations akin to the positions of the distant signal and the trackside magnets of the German legacy intermittent train protection system Punktförmige Zugbeeinflussung (PZB) 90 for braking distances of 1000 m and 700 m, respectively. The next two strategies divide the distance between the IP and the EOA into equally long segments delimited by the infill balise groups at half or one-third and two-thirds, depending on whether one or two additional infill balise groups are to be positioned. The last one is based on [34] and uses one additional infill balise group at a fixed distance ahead of the EOA. In summary, the five strategies are:

1. two additional infill balise groups at 1000 m and 250 m ahead of the EOA;
2. two additional infill balise groups at 700 m and 250 m ahead of the EOA;
3. two additional infill balise groups at one-third and two-thirds of the IP;
4. one additional infill balise group at one-half of the IP;
5. one additional infill balise group at 400 m ahead of the EOA like the setting of [34].

We calculated the weighted additional runtime of these five strategies for both 120 km/h and 160 km/h as the train's initial speed. The other parameters and constraints match the first part of the test case. The results are shown in Table 6 for three infill balise groups and Table 7 for two infill balise groups alongside the delta of the weighted additional runtime compared to the optimal positions with the minimum weighted additional runtime from our algorithm, as presented in Table 5.

Table 6. Results from evaluating different strategies for the manual placement of three infill balise groups for the test train.

Strategy	Initial Speed [km/h]	Positions [m]	Weighted Additional Runtime [s]	Delta to Optimum [s]
optimum	120	194 – 483 – 1187	39.5	—
1	120	250 – 1000 – 1187	43.2	3.7
2	120	250 – 700 – 1187	40.5	1.0
3	120	396 – 791 – 1187	42.8	3.3
optimum	160	243 – 705 – 1581	52.0	—
1	160	250 – 1000 – 1581	53.1	1.1
2	160	250 – 700 – 1581	52.0	0.0
3	160	527 – 1054 – 1581	55.6	3.6

Table 7. Results from evaluating different strategies for the manual placement of two infill balise groups for the test train.

Strategy	Initial Speed [km/h]	Positions [m]	Weighted Additional Runtime [s]	Delta to Optimum [s]
optimum	120	322 – 1187	44.7	—
4	120	594 – 1187	48.4	3.7
5	120	400 – 1187	45.2	0.5
optimum	160	420 – 1581	56.7	—
4	160	791 – 1581	62.4	5.7
5	160	400 – 1581	58.7	2.0

One can clearly see that all five strategies for both initial speeds are close to the optimum in terms of the absolute weighted additional runtime. One case (strategy 4 at 160 km/h) yields a delta of more than 5 s to the weighted additional runtime of the optimal locations. These results support the definition of a good-natured system behavior, as the additional time losses are relatively small compared to other factors influencing the train's driving behavior. However, in heavily utilized sections of a railway network and especially in disturbed conditions, these few seconds or up to 10% reduction in time lost can be important to increase capacity and to restore operational quality more quickly.

3.3. Further Findings

When considering multiple trains (cf. Section 2.4), the resulting positions of the infill balise groups are more likely than not a compromise of the individual optimal positions.

However, as the weighted additional runtime only increases marginally in an envelope of about ±50 m around the optimal positions, this compromise has minor negative effects on the overall performance. This was also confirmed by testing different, non-optimal placements in the previous paragraphs. Obviously, the downsides increase the more the driving characteristics of the trains in question differ because the infill balise groups are not able to provide infill information at the optimal location for each individual train. One could then increase the number of infill balise groups, but technical and economic reasons as well as diminishing returns limit this number.

During testing, it also became clear that the algorithm can handle occurring discontinuities in the additional runtime very well. Those discontinuities—which could also be observed in various test cases and are perceivable in Figure 5 on the left-hand side of the plot—stem from taking system latencies into account. In our case, these are the processing time of the OBU as well as the cruising phase between deceleration and acceleration. Such discontinuities can produce local minima of the weighted additional runtime but do not present themselves as the global minimum where the absolute lowest weighted additional runtime can be found. The algorithm always identifies the global minimum and does not erroneously confuse a non-optimal local minimum as the global minimum. The global minimum has been correctly detected by all sensible input parameters tested to date.

4. Discussion

With the work presented in this paper, a conclusive, open-source algorithm for the optimization of infill balise groups in ETCS level 1 applications is available now. We achieved this without the need for a complete runtime calculator while still retaining enough leeway to adapt to different problems and train characteristics. However, at this stage, some manual preprocessing is needed to have the data available in the necessary format and granularity.

Interestingly, when comparing the ratios of infill balise group positions to the IP of the scenario with two infill balise groups from our results, it seems to stabilize around a value of 0.30 for higher speeds. Nikolov et al. [38], with their mathematically driven model, came to a ratio of around one-third for a comparable scenario. In contrast to Wendler [32], we set the objective on minimizing the weighted additional runtime and not on directly quantifying the accompanying capacity effects. This was chosen to reduce the complexity of the algorithm developed, supporting a potential application in planning scenarios. The chosen point of view, however, does not neglect the topic of capacity entirely. The scenarios used to calculate the optimal positions of the infill balise groups mimic a situation where a train must initiate braking due to the track ahead not being free and available, which should only occur in case of delays or disturbances. As the running and clearing times are essential components of the blocking times [50] of a train on a given infrastructure, increasing them leads to longer headway times, higher capacity consumption, and ultimately to a decrease in capacity overall. Therefore, aiming for the lowest weighted additional runtime does in fact have positive effects on the capacity consumption and knock-on delays during operations.

In a future version, changes in line speed and gradient might be added to further expand the algorithm's usability and adaptability. Parts of the current preprocessing—especially obtaining the deceleration and the IP—could be integrated into the software by implementing parts of the UIC conversion model and the ETCS braking curves calculation. Additionally, a method to perform a comprehensive optimization considering different train types with individual characteristics should be developed. Some methodological approaches were presented in Section 2.4 and could form the foundation for this enhancement. With that, the best equipment for mixed-traffic infrastructures could be derived in one single framework and workflow without the need to manually decide where to place the infill balise groups in case a compromise is needed.

5. Conclusions

As is evident from the literature review, research and publications explicitly concerning ETCS level 1 and other current balise-based, spot transmission train protection systems are rare. Although there are several deployments of ETCS level 1 in operation, apparently no one has ever published a comprehensible, widely applicable method on how to effectively and optimally place the necessary infill balise groups. Some works were available concerning different aspects, but a complete workflow was not in their scope. Therefore, our work focused on designing and implementing the algorithm in Python and making it available to the community and the public.

With our work, calculating the optimal positions of infill balise groups for ETCS level 1 applications is possible. The algorithm is designed to be accessible and simple to use while still allowing it to adapt to a broad range of scenarios thanks to the input parameters available to the user. With that, we have managed to create an algorithm that is not confined to a specific geographical, regulatory, or contextual area of use but applicable universally in the context of ETCS level 1.

The chosen implementation using Python and JSON files for input and output features fast computation times and supports the application in different scenarios, as further automation could be added on top. The scientific value of our work lies in it being the possible foundation of future research on ETCS level 1 and its effects on operations and capacity. One could use the algorithm to optimally place infill balise groups when investigating the effect different ETCS configurations have on a larger scale, such as a line, a station, or a whole network, without the need for either default positions or manual placement. For that, we have made the source code available as open source [47].

During thorough testing and the first application of the software in the planning phase for a new ETCS deployment, valuable insights on the sensitivity to each of the parameters were gained and—if necessary—changes and improvements were implemented.

Owing to the chosen boundaries and interfaces of the algorithm, all ETCS trains can be modelled in this framework, covering the theoretical, technically possible speed range of up to 500 km/h, as set by the Eurobalises used [14] to transmit the data to the train. The first applications most likely will be in much lower speed regimes of regional and secondary lines where the benefits of a system of lower complexity and high economic efficiency are much needed to accelerate the wide deployment and adoption of ETCS. Additional research might add the ability to calculate the braking curves and the derived deceleration values on its own without relying on the official Braking Curves Simulation Tool [44]. This would decrease the preprocessing phase and accelerate the process by a significant amount.

The algorithm is focused on application in scenarios of ETCS; however, it could also be adapted for other technically related systems. The specifics of ETCS and its special braking curves and other influencing system parameters were considered.

While working on this task, it became clear that additional future work is needed to fully map the application scenarios of this tool, especially considering the need to automatically optimize the locations based on more than one train type. To date, this task must be performed iteratively with the support of the software now available. Future research therefore might cover the development of suitable methods to optimize the positions of infill balise groups for more than one train type. For this to work, it is necessary to develop an appropriate KPI, e.g., an additional weighting for the additional runtime of each individual train, which takes multiple train types into account and serves as the optimization goal.

For now, the optimization algorithm and software have reached a robust state, and the railway community shall be able to benefit from our work.

Author Contributions: Conceptualization, C.W.; methodology, C.W.; software, C.W.; validation, C.W.; formal analysis, C.W.; data curation, C.W.; writing—original draft preparation, C.W.; writing—review and editing, N.N.; visualization, C.W.; supervision, N.N.; project administration, C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available on GitHub at https://doi.org/10.5281/zenodo.12934251, accessed on 26 July 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. European Commission—Directorate-General for Mobility and Transport. ERTMS. Available online: https://transport.ec.europa.eu/transport-modes/rail/ertms_en (accessed on 22 July 2024).
2. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-023: Glossary of Terms and Abbreviations*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index003_-_SUBSET-023_v400.pdf (accessed on 3 April 2024).
3. Council Directive 96/48/EC of 23 July 1996 on the Interoperability of the Trans-European High-Speed Rail System; 1996. No Longer in Force. Available online: http://data.europa.eu/eli/dir/1996/48/oj (accessed on 4 April 2024).
4. Directive 2001/16/EC of the European Parliament and of the Council of 19 March 2001 on the Interoperability of the Trans-European Conventional Rail System; 2001. No Longer in Force. Available online: http://data.europa.eu/eli/dir/2001/16/oj (accessed on 4 April 2024).
5. Directive 2004/50/EC of the European Parliament and of the Council of 29 April 2004 Amending Council Directive 96/48/EC on the Interoperability of the Trans-European High-Speed Rail System and Directive 2001/16/EC of the European Parliament and of the Council on the Interoperability of the Trans-European Conventional Rail System; 2004. No Longer in Force. Available online: http://data.europa.eu/eli/dir/2004/50/oj (accessed on 4 April 2024).
6. Directive 2008/57/EC of the European Parliament and of the Council of 17 June 2008 on the Interoperability of the Rail System within the Community (Recast); 2008. No Longer in Force. Available online: http://data.europa.eu/eli/dir/2008/57/oj (accessed on 4 April 2024).
7. Directive 2012/34/EU of the European Parliament and of the Council of 21 November 2012 Establishing a Single European Railway Area (Recast). 2012. Available online: http://data.europa.eu/eli/dir/2012/34/oj (accessed on 4 April 2024).
8. Directive (EU) 2016/797 of the European Parliament and of the Council of 11 May 2016 on the Interoperability of the Rail System within the European Union (Recast). 2016. Available online: http://data.europa.eu/eli/dir/2016/797/oj (accessed on 4 April 2024).
9. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-026-1: ERTMS/ETCS System Requirements Specification. Chapter 1: Introduction*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index004_-_SUBSET-026_v400.zip (accessed on 10 June 2024).
10. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-026-2: ERTMS/ETCS System Requirements Specification. Chapter 2: Basic System Description*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index004_-_SUBSET-026_v400.zip (accessed on 10 June 2024).
11. European Railway Agency. *ERA_ERTMS_015560: ETCS Driver Machine Interface*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-12/index006_-_ERA_ERTMS_015560_v400.zip (accessed on 3 April 2024).
12. International Union of Railways – UIC Rail System Department. *FRMCS and 5G for Rail: Challenges, Achievements and Opportunities*. 2020. Available online: https://uic.org/IMG/pdf/brochure_frmcs_v2_web.pdf (accessed on 22 July 2024).
13. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-026-3: ERTMS/ETCS System Requirements Specification. Chapter 3: Principles*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index004_-_SUBSET-026_v400.zip (accessed on 10 June 2024).
14. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-036: FFFIS for Eurobalise*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index009_-_SUBSET-036_v400.pdf (accessed on 3 April 2024).
15. ERA * UNISIG * EEIG ERTMS Users Group. *SUBSET-044: FFFIS for Euroloop*; Version 4.0.0; 2012. Available online: https://www.era.europa.eu/system/files/2023-09/index016_-_subset-044_v240.pdf (accessed on 3 April 2024).
16. UNISIG. *SUBSET-047: Trackside-Trainborne FIS for RADIO IN-FILL*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index019_-_SUBSET-047_v400.pdf (accessed on 3 April 2024).
17. UNISIG. *SUBSET-048: Trainborne FFFIS for RADIO IN-FILL*; Version 3.0.0; 2012. Available online: https://www.era.europa.eu/system/files/2023-09/index020_-_subset-048_v300.pdf (accessed on 16 May 2024).
18. Commission Implementing Regulation (EU) 2023/1695 of 10 August 2023 on the Technical Specification for Interoperability Relating to the Control-Command and Signalling Subsystems of the Rail System in the European Union and Repealing Regulation (EU) 2016/919 (TSI CCS). 2023. Available online: http://data.europa.eu/eli/reg_impl/2023/1695/oj (accessed on 4 April 2024).
19. UNISIG. SUBSET-100: Interface 'G' Specification. Version: 2.0.0. 2012. Available online: https://www.era.europa.eu/system/files/2023-09/index046_-_subset-100_v200.pdf (accessed on 3 April 2024).
20. Eisenbahn-Bundesamt. *Nationaler Umsetzungsplan ETCS*; Version 1.11; Eisenbahn-Bundesamt: Bonn, Germany, 2017. Available online: https://www.eba.bund.de/SharedDocs/Downloads/DE/Europa/ERTMS/Nationaler_Umsetzungsplan_ETCS.pdf (accessed on 3 April 2024).

21. DB InfraGO AG. ETCS Migrationsstrategie. 2024. Available online: https://www.dbinfrago.com/web/schienennetz/etcs/etcs-migrationsstrategie-11089586# (accessed on 8 May 2024).
22. Eisenbahnbundesamt. Liste der in Deutschland Genehmigten Betreiber von Eisenbahnstrecken. 2024. Available online: https://www.eba.bund.de/DE/Themen/Eisenbahnunternehmen/eisenbahnunternehmen_node.html (accessed on 8 May 2024).
23. ÖBB-Infrastruktur Aktiengesellschaft. ETCS Expansion. 2024. Available online: https://infrastruktur.oebb.at/en/partners/rail-network/documents-and-data/etcs-train-control/etcs-expansion (accessed on 10 June 2024).
24. Banedanmark. Digitale Signalsystemer på Fjernbanen og S-Banen. 2024. Available online: https://www.bane.dk/Borger/Baneprojekter/Signalprogrammet (accessed on 10 June 2024).
25. Arend, L.; Pott, L.; Hoffmann, N.; Schack, R. ETCS Level 2 without GSM-R. *Signal. Datacommun.* **2018**, *110*, 18–28.
26. Laumen, P. ETCS Level 1 in the Context of Digital Interlocking Technology. Ph.D. Thesis, RWTH Aachen University, Aachen, Germany, 2023. [CrossRef]
27. Finken, K.; Hamblock, T.; Klöters, G. The ZSB 2000 on the way to ETCS. *Signal. Datacommun.* **2019**, *111*, 32–37.
28. Krüger, B.; Ide, M.; Finken, K. ETCS L2 without GSM-R—Experience report on the new ETCS architecture for CFL with ZSB 2000. *Signal. Datacommun.* **2024**, *116*, 16–23.
29. Stadlbauer, R. New Trends in ETCS Level 1. *Signal. Datacommun.* **2019**, *111*, 6–12.
30. Hutter, F.; Zekl, M. ETCS—The successful commissioning of the Red Line in Thailand. *Signal. Datacommun.* **2023**, *115*, 69–76.
31. Landex, A.; Jensen, L.W. Infrastructure Capacity in the ERTMS Signaling System. In Proceedings of the RailNorrköping 2019, 8th International Conference on Railway Operations Modelling and Analysis (ICROMA), Norrköping, Sweden, 17–20 June 2019; Peterson, A., Joborn, M., Bohlin, M., Eds.; 2019; pp. 607–622. Available online: https://ep.liu.se/ecp/069/040/ecp19069040.pdf (accessed on 23 April 2024).
32. Wendler, E. *Influence of Automatic Train Control Components with an Infill Functionality on the Performance Behaviour of Railway Facilities*; International Union of Railways (UIC): Paris, France, 2006.
33. Schwanhäußer, W. *Die Bemessung der Pufferzeiten im Fahrplangefüge der Eisenbahn*; Heft 20; Veröffentlichung des Verkehrswissenschaftlichen Instituts der RWTH Aachen: Aachen, Germany, 1974.
34. Verkehrswissenschaftliches Institut—Lehrstuhl für Schienenbahnwesen und Verkehrswirstchaft der Rheinisch-Westfälischen Technischen Hochschule Aachen. *Influence of ETCS on Line Capacity. Generic Study*; International Union of Railways (UIC). ERTMS Platform. Infrastructure Department: Paris, France, 2008. Available online: https://uic.org/cdrom/2011/05_ERTMS_training2011/docs/Capacity-generic-study.pdf (accessed on 24 July 2024).
35. International Union of Railways. *UIC Leaflet 406. Capacity*, 2nd ed.; International Union of Railways: Paris, France, 2013.
36. VIA Consulting & Development GmbH. *Influence of the European Train Control System (ETCS) on the Capacity of Nodes*; International Union of Railways (UIC): Paris, France, 2010. Available online: https://uic.org/cdrom/2011/05_ERTMS_training2011/docs/Capacity-nodes-study.pdf (accessed on 24 July 2024).
37. Shanker, V. The Ideal Positioning of Infill Balises. *IRSE News* **2011**, *172*, 9.
38. Nikolov, T.; Ganchev, G. Software Model for Optimization of Infill Balise Location in ETCS, Level 1 in Railway Transport. In Proceedings of the ICEST 2015, Proceedings of the L International Scientific Conference on Information, Communication and Energy Systems and Technologies, Sofia, Bulgaria, 24–26 June 2015; Dimitrov, K., Ed.; 2015; pp. 237–240. Available online: http://rcvt.tu-sofia.bg/ICEST2015_57.pdf (accessed on 17 May 2024).
39. Yin, J.; Chen, D.; Tang, T.; Zhu, L.; Zhu, W. Balise arrangement optimization for train station parking via expert knowledge and genetic algorithm. *Appl. Math. Model.* **2016**, *40*, 8513–8529. [CrossRef]
40. EEIG ERTMS Users Group. *21E089: Engineering Rules for Harmonised Marker Boards*; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index101_-_21E089_v1-.pdf (accessed on 4 April 2024).
41. Busse, M. Der Optimierte Einsatz von ETCS-Bremskurven. Ph.D. Thesis, Technische Universität Dresden, Dresden, Germany, 2020. Available online: https://nbn-resolving.org/urn:nbn:de:bsz:14-qucosa2-744060 (accessed on 3 April 2024).
42. UNISIG. *SUBSET-040: Dimensioning and Engineering Rules*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index013_-_SUBSET-040_v400.pdf (accessed on 10 June 2024).
43. UNISIG. *SUBSET-041: Performance Requirements for Interoperability*; Version 4.0.0; 2023. Available online: https://www.era.europa.eu/system/files/2023-09/index014_-_SUBSET-041_v400.pdf (accessed on 10 June 2024).
44. European Railway Agency. Braking Curves Simulation Tool v4.2. Available online: https://www.era.europa.eu/system/files/2022-11/era_braking_curves_simulation_tool_en.xls (accessed on 3 April 2024).
45. European Railway Agency. *ERA_ERTMS_040026: Introduction to ETCS Braking Curves*; Version: 1.5; 2020. Available online: https://www.era.europa.eu/system/files/2022-11/Introduction%20to%20ETCS%20braking%20curves.pdf (accessed on 10 August 2024).
46. International Union of Railways. *UIC Leaflet 544-1. Brakes—Braking Performance*, 6th ed.; International Union of Railways: Paris, France, 2014.
47. Wink, C. Python Source Code: Infill Optimization. 2024. [CrossRef]
48. Quattron Management Consulting Gmbh. LUKS—Analysis of Lines and Junctions. Available online: https://www.quattron.com/en/services/software-it-digital-data-management/luks/ (accessed on 8 July 2024).

49. DB Netz AG. Übersicht der Nationalen Werte für ETCS auf den Strecken der DB Netz AG. 2022. Available online: https://www.dbinfrago.com/resource/blob/11089680/192c84ec7797bdaa1001c91ca0b8cb44/ETCS-Nationale-Werte-data.pdf (accessed on 9 July 2024).
50. Happel, O. Sperrzeiten als Grundlage für die Fahrplankonstruktion. *Eisenbahntechnische Rundschau* **1959**, *8*, 79–90.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.