38th International Electric Vehicle Symposium and Exhibition (EVS38) Göteborg, Sweden, June 15-18, 2025

A free to use webtool for electric bus fleet analysis

Moritz Schiel[1], Paul Simon Scheer
[1]*Reiner Lemoine Institute, Rudower Chaussee 12, 12489 Berlin, Germany,*
moritz.schiel@rl-institut.de

Executive Summary

This paper introduces an open source and free to use webtool for bus fleet analysis in the context of full electrification. It calculates results such as system cost, number of necessary buses and placement of opportunity charging points primarily based on block data. The webtool allows for extensive parameter variation and aims to support bus operators to quickly analyze and compare various electrification scenarios. Besides introducing the underlying simulation frameworks, this paper focuses on showcasing the features of the webtool and discusses possible use cases with the aid of example scenarios and results.

Keywords: Modelling & Simulation, Optimal charging locations, Heavy Duty electric Vehicles & Buses, Environmental Impact, Sustainable Energy

1 Introduction

1.1 Motivation

The aim of achieving climate goals has intensified the push towards electrifying urban transportation, with buses playing a key role in reducing greenhouse gas emissions and improving urban air quality. The transition to electric buses supports the European Union target of a 55% reduction in emissions by 2030 [1]. The EU's Clean Vehicles Directive [2] provides a clear framework for electrification, requiring a portion of new bus procurements to be low- or zero-emission vehicles, thus accelerating adoption.

Among available technologies, battery electric buses stand out as the best solution for urban transport. They produce zero tailpipe emissions, operate more quietly, and have lower long-term operating costs compared to hydrogen alternatives [3], making them particularly suitable for high-density urban areas.

In Germany, despite a growing number of electric buses in cities like Berlin and Hamburg, overall bus fleet electrification remains limited [3]. Small and medium-sized bus operators, who make up a significant part of Germany's public transport system, face unique challenges in transitioning to electric fleets due to cost constraints and limited personnel. However, supporting these operators is essential to meet national climate goals.

1.2 Project overview

To support the transition of bus fleets to electric vehicles, a webtool is being developed in the research project *E-Bus 2030+*, which is funded by the German Federal Ministry for Digital and Transport. The project partners Reiner Lemoine Institute (RLI) and TU Berlin on the research side together with BVG, the Berlin bus operator, aim to develop a robust strategy for the full electrification of the Berlin bus fleet by 2030. Prior to starting *E-Bus 2030+*, RLI and TU Berlin have both developed simulation frameworks for electric bus fleet analysis, each with a different focus. Combined, these tools can cover the full spectrum of bus operation that is relevant for decisions on electrification.

To make the results and methodology available to all bus fleet operators in Germany, the project combines the different simulation frameworks into an easy-to-use webtool. This empowers bus operators to analyze their own fleet and compare different electrification scenarios. Usually, bus electrification concepts are created at a specific point in time, often by external contractors. The bus operators must deal with shifting demands through infrastructure projects, new bus lines, market developments or a shortage of skilled workers. This new approach enables companies to flexibly develop strategies that can be adjusted to those changing demands, even with limited resources.

2 Methodology

This chapter gives an overview of the backend structure behind the webtool. It goes in depth on the data model and specific tool functionality.

2.1 Webtool overview

The main function of the webtool is to simulate and evaluate a specific electrification scenario for buses. This requires the following input:

- Vehicle schedule / blocks
- Vehicle parameters
- Depot charging parameters
- Optional: opportunity charging parameters

Out of those, only the schedule must be provided as a file. Every other input offers default values and is adjustable during scenario creation.

Common variations of electrification scenarios that the tool can compare include:

- Depot vs opportunity charging
- Battery sizes
- Placement of opportunity chargers
- Block scheduling

To compare these scenarios, the webtool offers extensive result pages including a download option. The results cover KPIs like cost, environmental impact and amount of necessary vehicles, which allow for an easy comparison of scenarios at first glance. More in-depth results include plots of the state of charge (SoC) covering every vehicle (Figure 1), load timeseries for each electrified station and depot and a map-based scenario visualization.

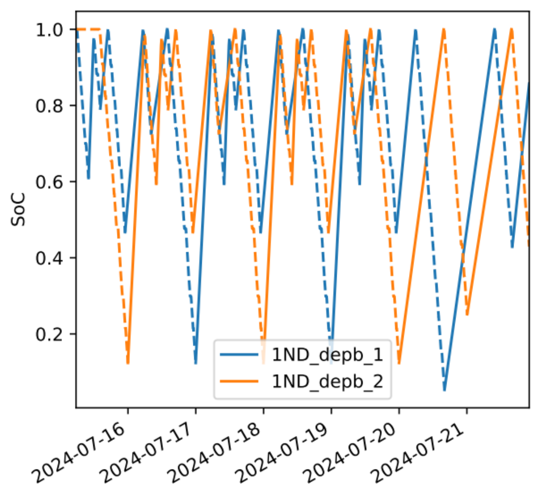

Figure 1: SoC plot of two buses over a week

2.2 Toolchain

The webtool combines the RLI-developed open-source tool SimBA [4] and TU Berlin's tool eFLIPS-depot [5].

SimBA (Simulation toolbox for Bus Applications) is an open-source Python tool developed by RLI through the research project Buffered-HLL [6]. Its core features are the event-based time-step resolved simulation of a bus operation including consumption, charging, dispatching and application of several charging strategies. It is an expansion of the open-source tool SpiceEV (Simulation Program for Individual Charging Events of Electric Vehicles) [7], which focuses on charging strategies.

eFLIPS (Electric Fleet and Infrastructure Planning/Simulation) is a simulation tool developed by TU Berlin to facilitate several tasks of a holistic bus simulation. It includes

- eFLIPS-model: Interface for a common database structure between the different tools
- eFLIPS-ingest: Allows import of bus schedules in the above database format. Supports: VDV 452 .x10 format
- eFLIPS-depot: Allows spatial and temporal resolved simulation of depot processes with consideration of parking patterns, shunting and other processes
- eFLIPS-eval: Visualization of results

The webtool combines both tools to achieve a simulation framework which contains the complete technical and operational side of a bus operation. The concept of the tool coupling is shown in Figure 2.

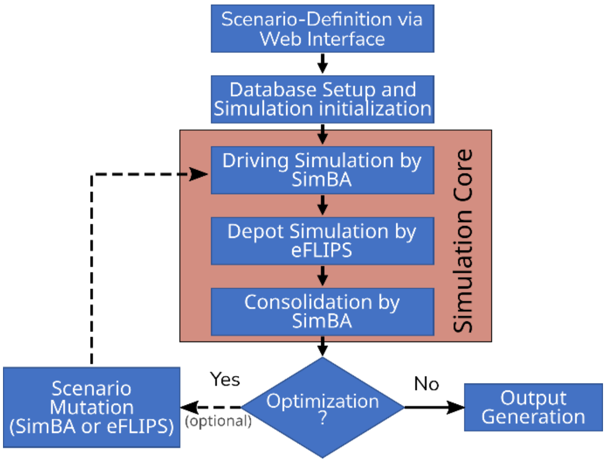

Figure 2: Concept of simulation modules in the webtool

The web interface allows the user to define a custom scenario based on one of several bus schedule input formats as well as the customization of used vehicle types, technical parameters of electrification, depot specifications and more. Inputs are validated, a scenario for simulation is generated and the simulation is started. At the core of the simulation are the alternating executions of SimBA and eFLIPS which solve parts of the scenario respectively. Optional optimization calls mutate the scenario in some way, for example add electrification, change blocking or depot assignments.

2.2.1 Consumption Calculation

The calculation of consumption is done on a trip basis. Inputs for the calculation are the average speed during the trip, temperature, level of loading, incline and vehicle type. The lookup of a trip specific consumption is based on a multidimensional linear interpolation. The table which is looked up can be generated through measurements, simulations or a mix of both, e.g. expanding Standardized On-Road Test cycle (SORT) measurements with estimates for increased consumption under incline or climatization needs over temperature. The webtool contains consumption tables, which were generated using a physical longitudinal dynamics model described in [8] for a series of generic vehicle types such as an 8m, 12m or 18m bus with the possibility of an additional diesel heater.

2.2.2 Database

The tools were adjusted to work with a common database structure, defined in *eFLIPS-model*. Both tools have a modular structure and enable various combinations of individual functions. The tools are coupled using a Django backend and a shared Postgres database. The database contains the relevant information regarding the scenario which is provided by the user. Each Scenario contains a multitude of instances of several models. In this case a *model* references a Python class that represents a database table, with its attributes corresponding to a record in the table. This allows the description of components or concepts of a bus operation. At the core of each simulation is the Scenario model, which bundles the different simulation models in a flat hierarchy.
Some of the core models the webtool uses are listed in Table 1.

Table 1: Core models of the database

Scenario	Instance to bundle models related by a simulation. All following models reference their respective scenario by a foreign key.
Block	Instance to bundle **Trips** to be serviced by a single **Vehicle**. References **VehicleType** and **Vehicle**.
Trip	Instance, that defines start and end time. References the **Block** it is contained in and references the **Route.**
Route	Instance, that defines the spatial definition of the **Trip**.
VehicleType	Instance, that defines selected technical properties such as the charging curve, battery capacity and length.
Vehicle	Instance that represents a concrete vehicle and references its **VehicleType**.
Station	Instance, that defines technical properties of the bus stations. This includes the flags if the installation of charging stations is possible or in the case of electrified stations technical parameters such as maximum power per charger, number of chargers and the maximum power which can be provided by the grid connection
Consumption	Instance, that defines the consumption of a **VehicleType** as a function of speed, temperature, level of loading and incline.
Depot	Instance, that references a **Station** with extra capabilities like handling bus storage, cleaning, maintenance and charging of buses in between **Blocks**.
Area	Instance, that represents a certain location, which is used to define the location of a P**rocess**.
Process	Instance, that represents a certain action, which is executed on **Vehicles** in a **Depot**.

2.2.3 Simulation Core

The webtools' core functionality is the Simulation Core that simulates a given bus operation scenario through coupling of SimBA and eFLIPS. It consists of the three steps driving simulation, depot simulation and consolidation, which are described below.

SimBA calculates vehicle resolved consumption based on the input schedule of the bus operation as well as the opportunity charging which is computed respecting nonlinear charging curves. This is translated in expected SoCs through the defined battery capacity of the vehicle type. The starting, minimum and end SoC are passed to eFLIPS. Each Block is simulated with a fully charged vehicle. Figure 3 shows an explanatory result of this first step.

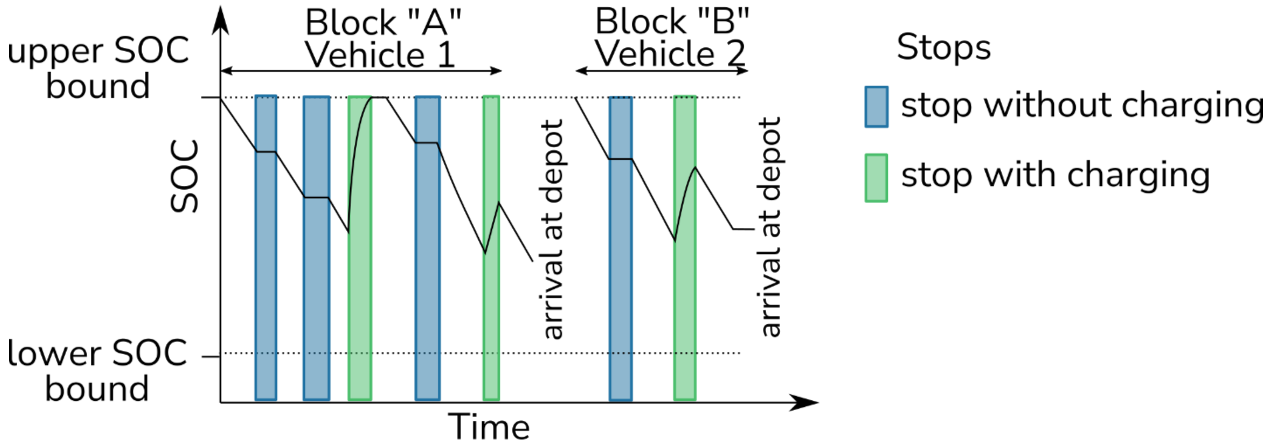

Figure 3: SoC delta calculated by SimBA

Vehicle dispatchment is done by eFLIPS in the following step, which incorporates a more detailed approach towards depot simulation. Depot processes like cleaning, shunting and charging are simulated. This gives eFLIPS the ability to calculate the SoCs of the fleet at any point in time. Existing vehicles at the depot can service a block if their current SoC is greater or equal to the delta of the start and minimal SoC of that block. This ensures the SoC does not drop below the lower SoC bound. If no such vehicle exists a new vehicle is generated. If blocks require more energy than a fully charged battery a vehicle will be dispatched anyways. Using **Station Optimization** might resolve this issue. If the issue is not resolved the Block will be tagged as *critical* during the result visualization. The input schedule is repeated three times in total, to analyze if SoCs stay stable through multiple simulations of the input schedule. If drift is detected, eFLIPS-depot uses given constraints to compensate for that drift, e.g. increasing charging power or the introduction of an extra vehicle. The dispatchment tries to minimize the number of vehicles by using the same vehicle to service multiple blocks.

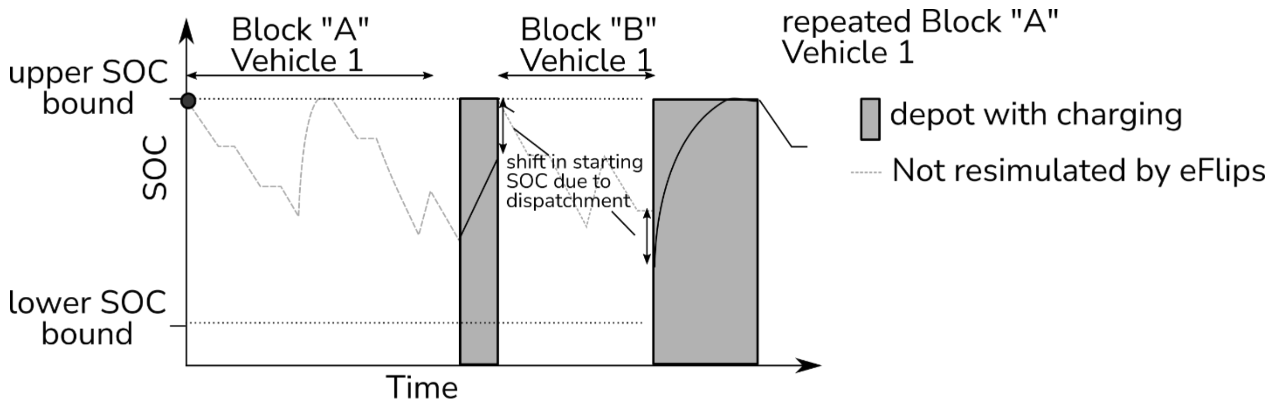

Figure 4: Vehicle SoC is inconsistent after dispatchment

The output of this simulation (Figure 4) is passed to SimBA. Through non-linear effects of opportunity charging, the resulting SoC of opportunity charging vehicles would be unknown. Therefore, SimBA takes the dispatchment and expected SoCs at the start of each block and re-calculates the scenario (Figure 5). Through the assumption of monotonically decreasing charging curves, e.g. higher charge at lower SoCs, the resulting scenario can be evaluated. The assumption guarantees that a block which was deemed drivable stays drivable, even when the starting SoC is dropped from 100% by not more than the minimal SoC of this block. The energy gained through opportunity charging will be greater or equal to the amount of energy which was charged at a higher starting SoC.

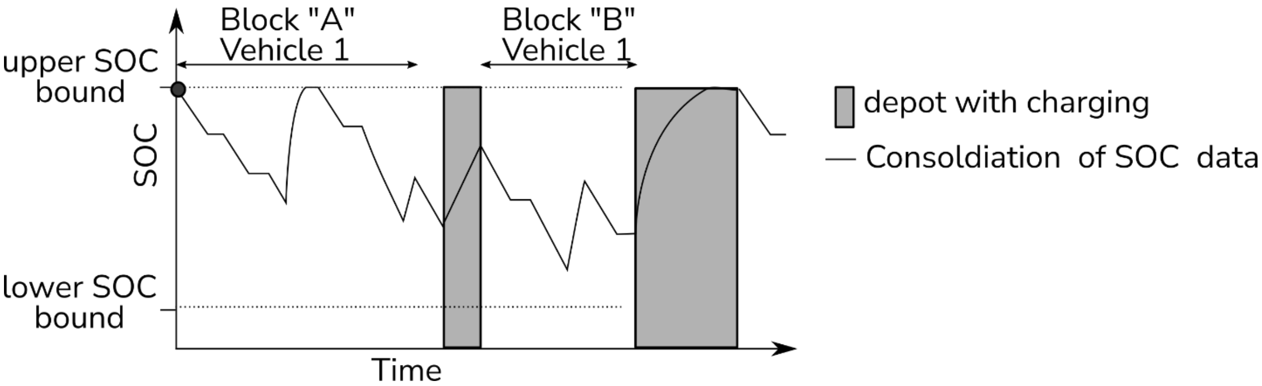

Figure 5: Vehicle SoC graph after consolidation

The described coupling can be repeated to increase the resolution of the predicted SoCs. Further optional steps can be taken to change the scenario in differing aspects like increasing the number of drivable blocks by adding electrification at stations or reducing the requirements for depots by changing the depots that blocks are serviced by.

2.2.4 Station Optimization

Some blocks can only be operated via opportunity charging if their energy requirements are higher than a fully charged battery can deliver. Splitting these blocks into multiple blocks might be possible but would increase the number of needed vehicles and increase the need for non-service trips. Another solution could be to electrify selected terminal stations. Complex bus systems can create complex situations for the bus operator to decide which stations should have charging capabilities. The Webtool allows simulating the bus system with electrified terminal stations, terminal stations which cannot be electrified, e.g. for reasons of limited space or grid connections, and an automatic setting. In this automatic setting if a scenario results in SoCs below a given threshold an optimization process will automatically electrify terminal stations to achieve full electrification. This is done by evaluating the system and identifying stations to electrify which improve the drivability of the schedule. Figure 6 shows different stops at different times of a block with opportunity charging. Only the two stations marked in blue have an impact on the low SoC event and are considered for electrification. The standing time and SoC at the time are considered as well and allow for an approximation of charge.

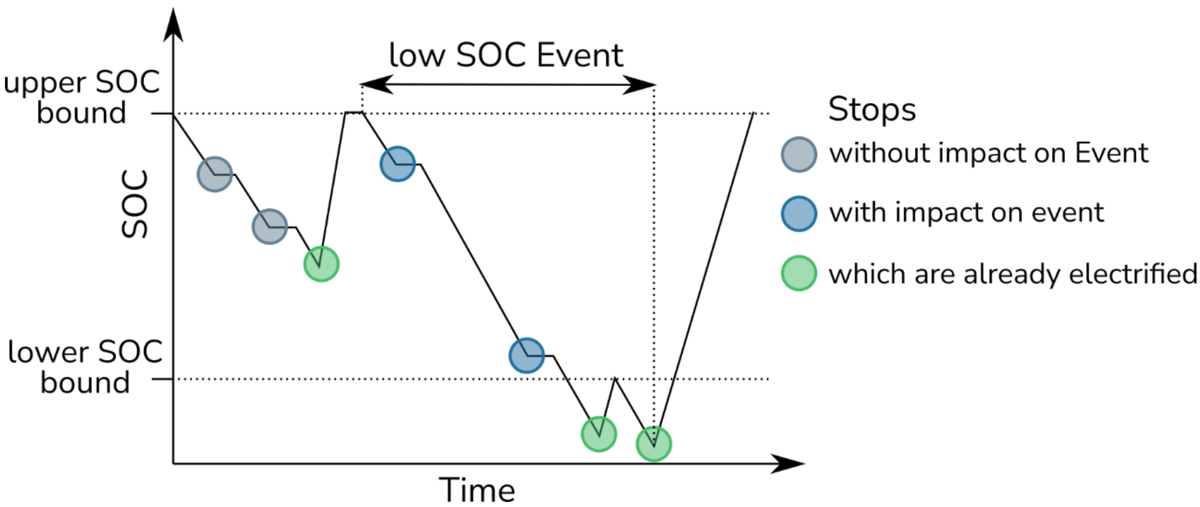

Figure 6: Vehicle SoC time series during a Station Optimization step, with relevant stations for electrification highlighted in blue

After processing the whole system with the current state of electrification the station with the highest potential is electrified. This loops until full electrification is achieved. The bus system is represented by a tree with each node representing a state of electrification. A vertex describes the addition or subtraction of an electrified station. The first node in this tree is the empty node with no electrified opportunity station, while the last node represents full electrification of each station (Figure 7).

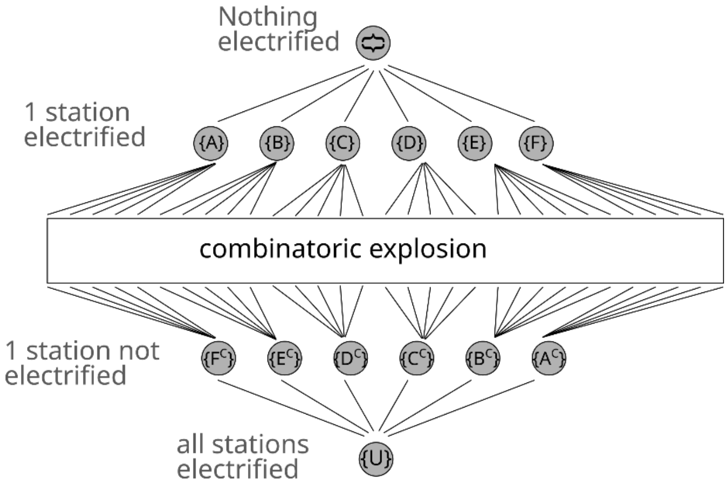

Figure 7: Possible complexity of Station Optimization

While the given algorithm guarantees to find a solution in N steps with N being the number of stations that could be electrified, this solution is not necessarily the optimal solution. If wished, the tree can be further searched for more optimal solutions using a depth first search. If nodes can be identified as non-ideal, they are removed from the search. The algorithm currently only optimizes for the number of electrified stations.
By dividing the bus system into independent subsystems and finding stations to electrify which are mandatory for a full system electrification, the scope of search can be drastically reduced.

2.2.5 Tool Access

The webtool will be made available through a public web service. The code is accessible via GitHub under an open-source license. Considering a broad group of interested users, the webtool supports a setup via Docker image. This lowers the technical skill requirements in setting up a local version of the tool. The ease of local hosting can also solve potential issues regarding data security if bus operational data has higher confidentiality requirements, as well as increase performance through better hardware.

3 Frontend features

The tool collects input data in a specified order via a wizard (Figure 8). A scenario always starts with **block data**, either through a file upload or by copying data from an existing scenario. While the block data mainly contains information about Trips and Routes, it also contains vehicle types as well as terminal stops.
For each identified **vehicle type** the user must set a consumption and a battery capacity. At time of writing, the user must choose a preset vehicle type such as “articulated bus” and is only able to change the battery capacity, since consumption is handled by the Consumption model. There will be an option to set static consumption for each vehicle type in the future.
Terminal stop options are expanded upon in chapter 3.1.
The two remaining input steps are handled by the project partners at TU Berlin and are not focused on in this paper. The **cost page** contains various default values for cost calculation as well as a Life Cycle Assessment.
Depots can either be automatically created by the tool to fulfill any demand, be set via simple input fields for charging point number and capacity or be created in a visual interface.
Lastly, the **summary** contains all relevant information from the data collection process. The user can check and edit any value except for the block data. Once satisfied, the user can start a simulation, which will redirect them to the results page when finished.

Figure 8: Webtool wizard

3.1 Charging station placement

Bus operators are often interested in the feasibility of opportunity charging for their fleet. The webtool provides features to compare opportunity charging scenarios by allowing for extensive parametrization. All relevant stations from the block data are listed and can each be set to one of three electrification modes:

- **Electrified**: This station will always have charging infrastructure. Each station has inputs for maximum chargers and charging power. If no values are set, the number of chargers is unlimited and will be placed as necessary. Charging power is set as default at the top of the page and only needs to be set for specific stations if it differs.
- **Automatic**: This station can be electrified by the Station Optimizer. It uses the default charging power and unlimited charging points.
- **Not electrified**: This station is not electrified and is excluded from the list of possible stations that are considered by the Station Optimizer.

These options cover most real-world applications. If a station is chosen by the optimizer but has more charging points than possible, the station can be set as electrified in a copied scenario and limit the charging points via the station specific setting.
To make the input interface viable for scenarios with many stations, there is a bulk editing option as well as search and filter functionality (Figure 9).

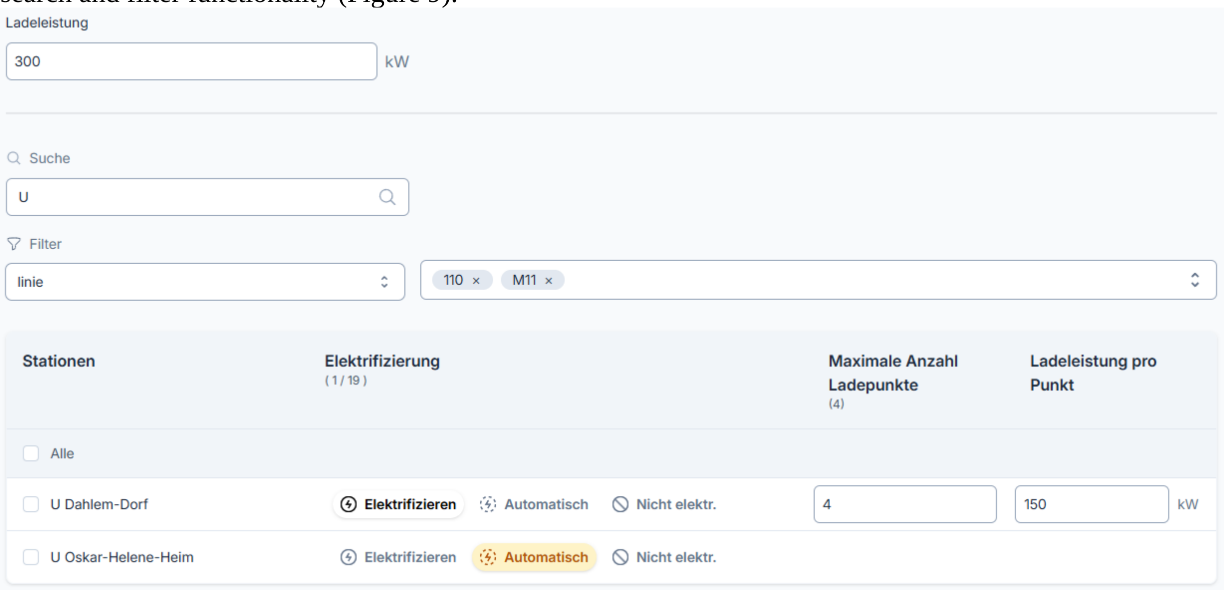

Figure 9: Terminal station electrification inputs. Filters are set to include Stations that are used by the lines "110" and "M11" where the Station name contains the letter U

3.2 Result overview

A main discussion topic during development and in user testing is the simulation output. The goal of the result page is to show the viability of a scenario and to make it comparable to other scenarios. If a scenario fails, the user also needs to be able to identify points of failure such as battery capacity, insufficient charging points and power or a few demanding blocks.
To achieve these goals, the team has decided on three types of visualization: Key performance indicators (KPIs), plots and a map.
KPIs are any relevant results that can be represented by a number, such as the amount of vehicles needed or the share of electrified terminal stations. Plots show details of the specific scenario. They include events for each vehicle, depot utilization and state of charge diagrams (Figure 10).
The map (Figure 11) shows stations and depots, each clickable to reveal detailed information about charging capacity and utilization.

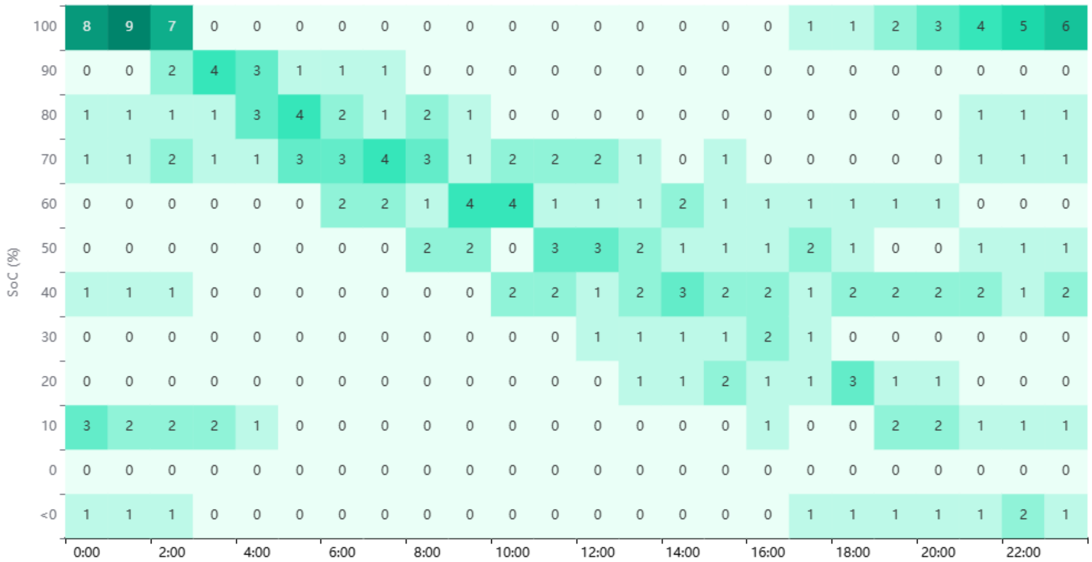

Figure 10: State of charge diagram showing the number of vehicles in each state of charge bracket per hour of the day

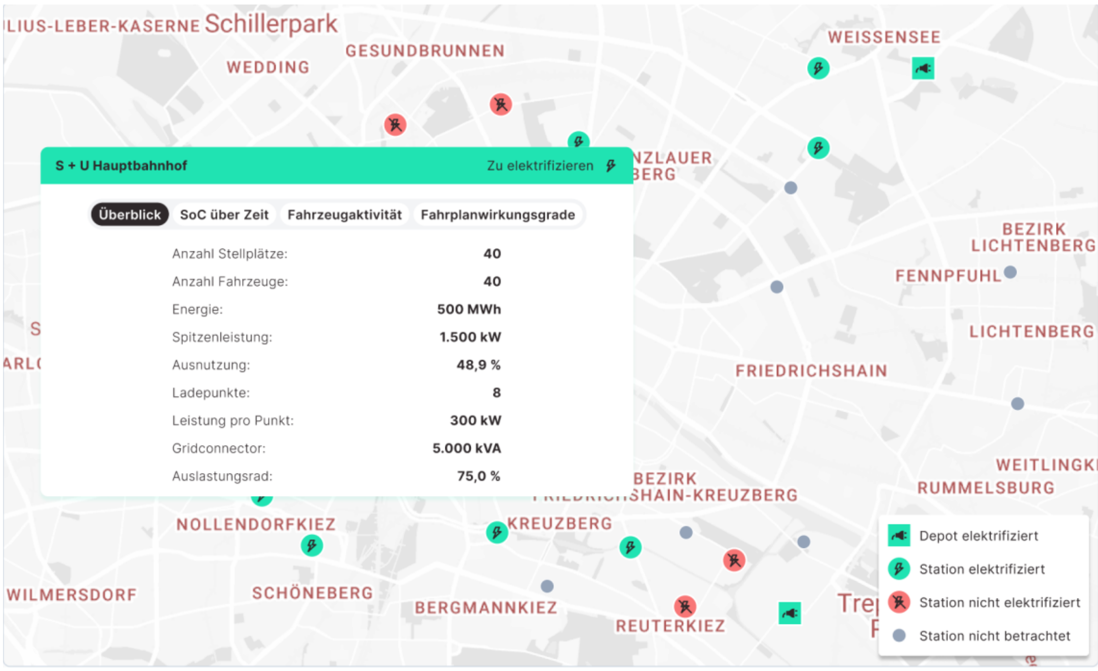

Figure 11: Example map from a mockup. Green icons show an electrified depot or station. Pop-ups include multiple tabs with data and graphics.

4 Use Cases

Bus operators might encounter various questions while tackling fleet electrification. This chapter highlights example use cases and shows how the tool can provide answers.

4.1 Are the current blocks feasible with electric buses?

The first steps in creating a new scenario are to upload the block data and to name the scenario. For this example, a file generated via General Transit Feed Specification (GTFS) data for a few lines in Berlin is used. All blocks are served by the same vehicle type. For vehicle options (Figure 12), this scenario will use an articulated bus with 650 kWh of usable capacity. The consumption will be calculated assuming an ambient temperature of -10°C, since the energy demand of buses is highest at low temperatures.

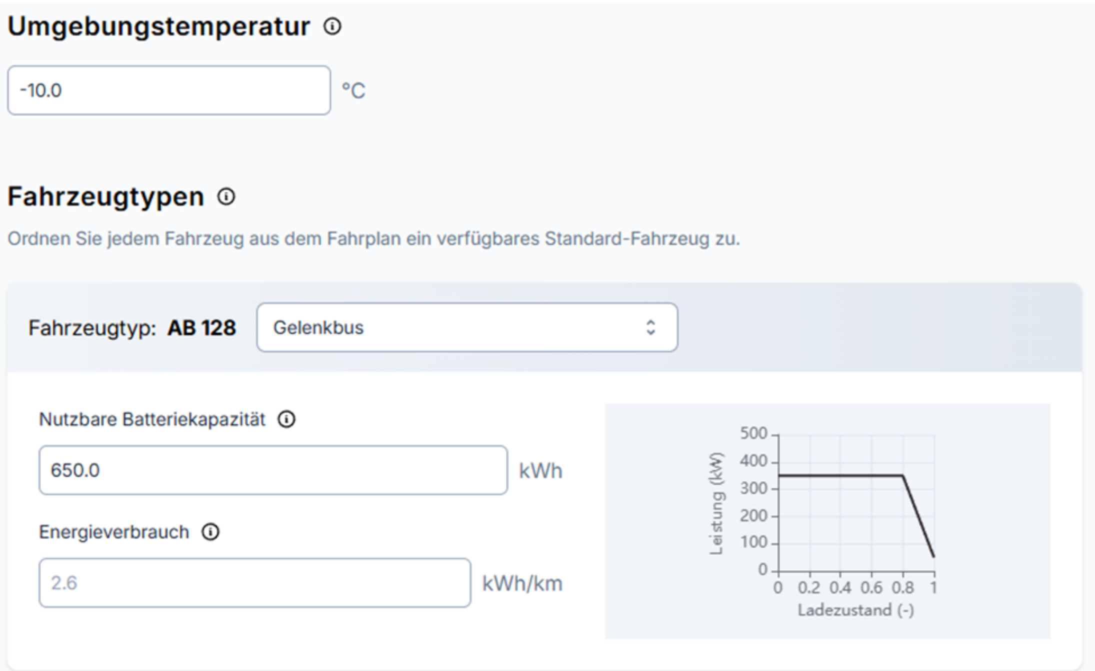

Figure 12: Vehicle input in the webtool. First input field contains the ambient temperature, the dropdown next to the vehicle type "AB 128" from the block data shows, that an articulated bus has been chosen as a preset.

Opportunity charging will not be considered for the first scenario, so all stations will be set to "not electrified". Depot and cost settings are kept as default. This results in a demand for 8 buses, 3 of which dip below 0% state of charge and are thus marked as "critical" by the tool (Figure 13).

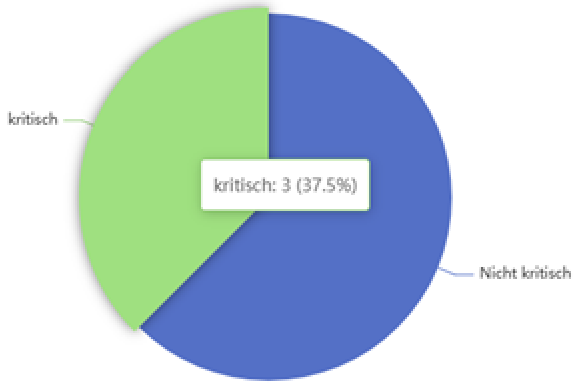

Figure 13: Amount of critical versus non-critical buses in the scenario. 3 Buses out of 8 dropped below 0% SoC in the simulated scenario.

Options to make the blocks work include:

- Change the vehicle. Either by using bigger batteries or by choosing a model with auxiliary heating.
- Increase depot charging power
- Include opportunity charging

When creating a variant of the scenario, all data and previous inputs are already set in the wizard. To make the scenario feasible, the terminal stations are set to "automatic" with a charging power of 300 kW. Running this scenario shows that with only 2 electrified stops with 1 charging point each, the blocks are now fit for electric buses.

4.2 What grid connection capacity is necessary for this depot?

For this example, the scenario from chapter 4.1 already contains all necessary information. The KPIs at the top of the result page show a depot peak load of 600 kW. For more detail, reference the associated plot for depot charging power and occupation (Figure 14). At time of simulation, charging strategies are not implemented in the webtool, which results in buses charging at maximum power. For that reason and because of the small input scenario of 8 blocks over a single day, there is no difference in peak depot power between the depot scenario and the variation with 2 opportunity charging points. In future versions of the tool, options for different smart charging strategies like peak shaving or balanced charging will be included.

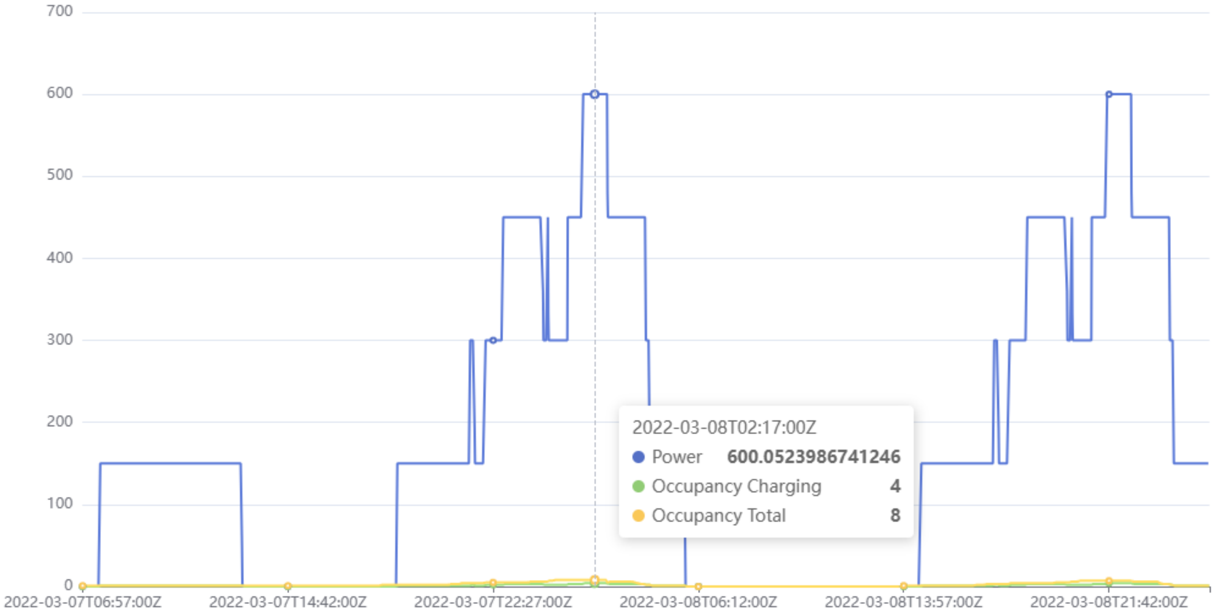

Figure 14: Depot charging power and occupancy in the example scenario without smart charging

5 Discussion

Having presented the webtool's features and demonstrated its application across various use cases, this section critically examines key limitations encountered during its development and deployment.

5.1 Tool use outside of Germany

While the webtool as well as the underlying software has been developed in German research projects and designed with local bus operators in mind, the technical nature of the model should make it usable worldwide. While the website will have language options for English and German, there are no plans for unit conversions. The most powerful data format that the tool supports is the VDV 452 [9]. It is a German format for block data and might not be available in other countries, which also comes with its own issues as discussed in chapter 5.2. An alternative import option is a CSV file with block data, an example of which can be downloaded in the webtool. This requires work and expertise from the bus operator to translate their available data into the correct format. A third option is to use General Transit Feed Specification (GTFS) data and user settings to create block data based on timetables. Backend functionality for scheduling already exists, GTFS support is planned but not implemented at time of writing. This will produce a scenario that can be simulated with open input data but will not be accurate to actual operation.

Some features of the tool currently operate on data sets limited to Germany. Of those, only the elevation data, which is used to calculate consumption, is relevant for simulation purposes. This can be resolved by either using static consumption per vehicle type or by expanding the underlying data set.

5.2 Current issues

The current design faces two major hurdles towards easy usability. Block data in Germany might have an interface specification in the VDV 452, but it is not specific enough to consistently parse. Some companies

don’t input all the data into their planning tools, which leaves the importer with less information than expected. Others use city specific coordinate systems, which makes map usage challenging. Writing a parser that can deal with any variation in VDV 452 format might not be feasible in the current research project.
The second issue is consumption data. To our knowledge, there is no publicly accessible consumption data for different bus models under various circumstances. Options to solve this are static consumption inputs, which are less responsive to individual scenarios and circumstances, or a big database of precalculated consumption tables. Consumption tables are theoretical and not validated but allow for analysis such as winter versus summer scenarios or challenging topography.

6 Conclusion

This paper introduced a webtool designed for bus fleet analysis in the context of full electrification, detailing its features and use cases. While the tool demonstrates strong capabilities, several limitations have been identified, particularly regarding block data availability and formatting. Future work will focus on addressing these challenges, improving cross-border applicability, and enhancing the tool’s flexibility to support a wider range of scenarios.

Acknowledgments

The authors gratefully thank the Reiner Lemoine Foundation for supporting this research work.

We would like to extend our sincere gratitude to the project partners from the TU Berlin at the Chair of Methods for Product Development and Mechatronics and the BVG for their invaluable support, collaboration, and commitment throughout this project.

References

[1] European Union, *EU climate target plan 2030*, ISBN 978-92-76-21764-0, 2020.

[2] Directive (EU) 2019/1161 of the European Parliament and of the Council of 20 June 2019 amending Directive 2009/33/EC on the promotion of clean and energy-efficient road transport vehicles.

[3] PwC, *E-Bus Radar 2025*, download at https://www.pwc.de/de/branchen-und-markte/oeffentlicher-sektor/pwc-e-bus-radar-2025.pdf, accessed on 2025-04-28.

[4] *SimBA*, https://github.com/rl-institut/simba, accessed on 2025-04-28.

[5] *eFLIPS-depot*, https://github.com/mpm-tu-berlin/eflips-depot, accessed on 2025-04-28.

[6] *Reiner Lemoine Institut*, https://www.reiner-lemoine-institut.de/en/buffered-hll/, accessed on 2025-04-28.

[7] *SpiceEV*, https://github.com/rl-institut/spice_ev, accessed on 2025-04-28.

[8] Simulation of e-bus Operation and Automated Planning of Charging and Tank Infrastructure, https://link.springer.com/chapter/10.1007/978-3-658-46551-3_18, accessed on 2025-04-28.

[9] VDV-Schrift 452 Version 1.6.2, *VDV-Standardschnittstelle Liniennetz/Fahrplan*, 2023.

Presenter Biography

Moritz Schiel has been a researcher in the Research Unit Mobility with Renewable Energy at RLI since April 2022. Moritz has a master's degree in Renewable Energies from HTW Berlin. His main field of research is the electrification of public transport bus fleets with an emphasis on the energy supply, the use of renewable energies and grid integration. The methodologies used focus on the development and application of open-source modelling and simulation software.