SUMO User Conference 2025
Conference paper
https://doi.org/10.52825/scp.v6i.2605

Published: 15 Jul. 2025

# Overcoming Data Scarcity in Calibrating SUMO Scenarios With Evolutionary Algorithms

Jakob Kappenberger[1,*] and Heiner Stuckenschmidt[1]

[1]University of Mannheim, Mannheim, Germany

*Correspondence: Jakob Kappenberger, jakob.kappenberger@uni-mannheim.de

**Abstract.** Traffic simulations play a crucial role in urban planning and mobility management by providing insights into transportation systems. However, their effectiveness heavily depends on accurate demand modeling, with calibration often requiring large amounts of observational data. This poses a challenge in settings with limited data availability. In this paper, we propose a methodology for calibrating SUMO scenarios under data-scarce conditions. To contextualize our approach, we first review existing SUMO scenarios and their demand modeling strategies. We then introduce the Mannheim SUMO Traffic Model (MaST) as a case study and employ the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) to optimize route probabilities as input for the existing *routeSampler* tool provided by SUMO. Results indicate that our method significantly improves calibration accuracy compared to baseline approaches both for 3-hour and 24-hour scenarios. While our findings suggest that the proposed methodology can support model calibration in data-limited environments, further research is needed to assess its generalizability and effectiveness in different contexts.

**Keywords:** Traffic Simulation, Calibration, Evolutionary Algorithms, Data Scarcity

## 1. Introduction

Urban traffic creates intricate and complex systems, making anticipating and evaluating potential changes difficult. Therefore, traffic simulations have become essential tools for urban planning and traffic management. They enable researchers and policymakers to analyze and optimize traffic flow, evaluate infrastructure changes, and forecast future transportation needs in a safe environment without impeding urban populations' mobility needs.

Like all models created to represent reality, the usefulness of traffic simulations hinges on the degree to which they can reproduce real-world traffic conditions. The process to improve said ability is called model calibration [1]. Model calibration is often separated into different steps, ranging from general error checking, capacity calibration, and route choice calibration to performance validation [2]. This work will focus on route choice calibration, determining the routes travelers choose to reach their destinations. It constitutes a crucial step for replicating traffic flows in dense urban networks where multiple connections between endpoints exist [3]. For the traffic simulator Simulation of

Urban MObility (SUMO), several approaches to tune traffic flows in this vein are available. Chief among them, the *routeSampler* tool allows for incorporating a large variety of different data sources (e.g., induction loop counts, turn counts, Origin-Destination-Matrices, etc.). However, literature on existing SUMO scenarios indicates the need for large amounts of data to properly align the simulations to real-world data, which not all municipalities may be able to produce.[1] Consequently, this paper aims to demonstrate a methodology for modeling demand in SUMO scenarios where comparatively little data is available. To this end, our contributions are the following:

- **Background**: In §2, we briefly introduce relevant literature, listing existing SUMO scenarios with a particular focus on the data they are built upon.
- **Methodology**: We present a new SUMO scenario (Mannheim SUMO Traffic Model (MaST)) in §3, which serves as a case study to demonstrate our method. Moreover, we describe our calibration pipeline, from data preprocessing and map matching of GPS traces to the optimization process based on the already existing Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm, and, finally, evaluation.
- **Analysis**: We systematically examine the performance of our calibration pipeline compared to a standard procedure proposed in the SUMO documentation (presented in §4). Finally, we summarize our findings and list the limitations of our approach in §5.

## 2. Related Work

Before presenting our methodology, it is crucial to review the avenues available for demand modeling in SUMO as well as the approaches chosen in the SUMO community. SUMO includes a variety of tools to model demand [4]. These encompass countless

***Table 1.*** *Overview input data for demand of public microscopic SUMO scenarios.*

| Scenario | Size | Input data for demand |
|---|---|---|
| Bologna (three small scenarios) [5] | - | Induction loops (636, every 5 minutes) |
| Luxembourg SUMO Traffic (LuST) [6] | 156 km$^2$ | public statistics for *activitygen* |
| Berlin Sumo Traffic (BeST) Scenario [7] | 800 km$^2$ | demand extracted from MATSim scenario, transferred using iterative traffic assignment |
| Dublin [8] | 17.5 km$^2$ | Induction loops (480, every 6 minutes) |
| Ingolstadt Traffic Scenario for SUMO (InTAS) [9] | 51.54 km$^2$ | Induction loops (24 junctions, every lane, every 15 minutes), detailed population statistics |
| Ingolstadt [10], [11] | 100 km$^2$ | ”[Induction] loops at all traffic lights in the city“, demographic data |
| Monaco SUMO Traffic (MoST) Scenario [12] | 22 km$^2$ | public statistics for *activitygen* |
| Tokyo SUMO traffic scenario (ToST) [13] | 33.22 km$^2$ | Induction loops (7 junctions, each direction, every 60 minutes), population statistics |

[1] A SUMO scenario denotes a single simulated environment, in most cases depicting an individual city or municipality.

tools that either create demand randomly (e.g., *randomTrips.py*) or based on population statistics and/or the map used (*activitygen*). Conversely, there are several tools that are able to incorporate various observational data on traffic volume and flow. These differ both with regards to which data types they can process (e.g., induction loops or Origin-Destination-Matrices) as well as the degree of data coverage they require (e.g., *dfrouter* needs full coverage of all source and sink edges).[2]

Table 1 provides an overview of publicly available microscopic SUMO scenarios, focusing on the respective input data used to model demand. The different approaches can be roughly separated into three groups:

1. The models simulating Luxembourg [6] as well as Monaco [12] constitute a group relying solely on public statistics (i.e., mainly demographic data as well as data on activity goals in the modeled area). This data is then fed into the *activitygen* tool provided by SUMO. Neither paper features a quantified evaluation of the empirical validity of said demand (most likely since the authors lack the data to do so). In a similar vein, the BeSt scenario is built upon the demand of a MATSim simulation of Berlin, which in turn was calibrated using census data [7].
2. The three scenarios situated in Bologna [5] and the urban scenario in Dublin [8] are calibrated using solely observational data on traffic volume, namely induction loops provided by the respective municipality. While the modeled demand is not specifically evaluated for the Dublin model, the Bologna models appear to be relatively close to the actual traffic volumes they are meant to replicate. It is worth noting that both scenarios rely on very good data coverage, both in terms of the number of sensors and the resolution with which they report traffic counts.
3. Lastly, both models based on the city of Ingolstadt [9], [10], [11] and the ToST scenario of Tokyo [13] rely on a combination of demographic statistics and observational data on traffic volume in the respective urban centers. For Ingolstadt, both projects evaluate the empirical validity of the traffic created, with the authors behind InTAS reporting a Normalized Root Mean Square Error (NRMSE) of 0.33 and the other Ingolstadt scenario posting Absolute Relative Error (ARE) values between 13% and 18% in the simulated timeframe.

Based on this overview, the subsequent chapter will serve to introduce our simulation as well as our methodology for route choice calibration.

# 3. Methodology

## 3.1 MaST - Mannheim SUMO Traffic Model

Figure 1 shows the simulated area of our SUMO scenario for Mannheim.[3] It currently covers the city center, namely the districts Innenstadt/Jungbusch, Schwetzingerstadt/Oststadt, and Lindenhof. The initial topography of the model is based on OpenStreetMap data collected via the *OSMWebWizard* provided by SUMO, which was subsequently manually checked and improved to account for any potential errors in the importing process. The created network includes sidewalks as well as bike paths (although the scenario does not simulate pedestrians and bike riders as of yet). Moreover, the model features the public transport infrastructure present in Mannheim (stops

[2]See [4] for a more thorough discussion of the different tools.

[3]In principle, the proposed method could have also been evaluated on an existing scenario. However, many published scenarios do not share all data they were built with. More importantly, none of the discussed models ships with an empirical route distribution that our pipeline relies on.

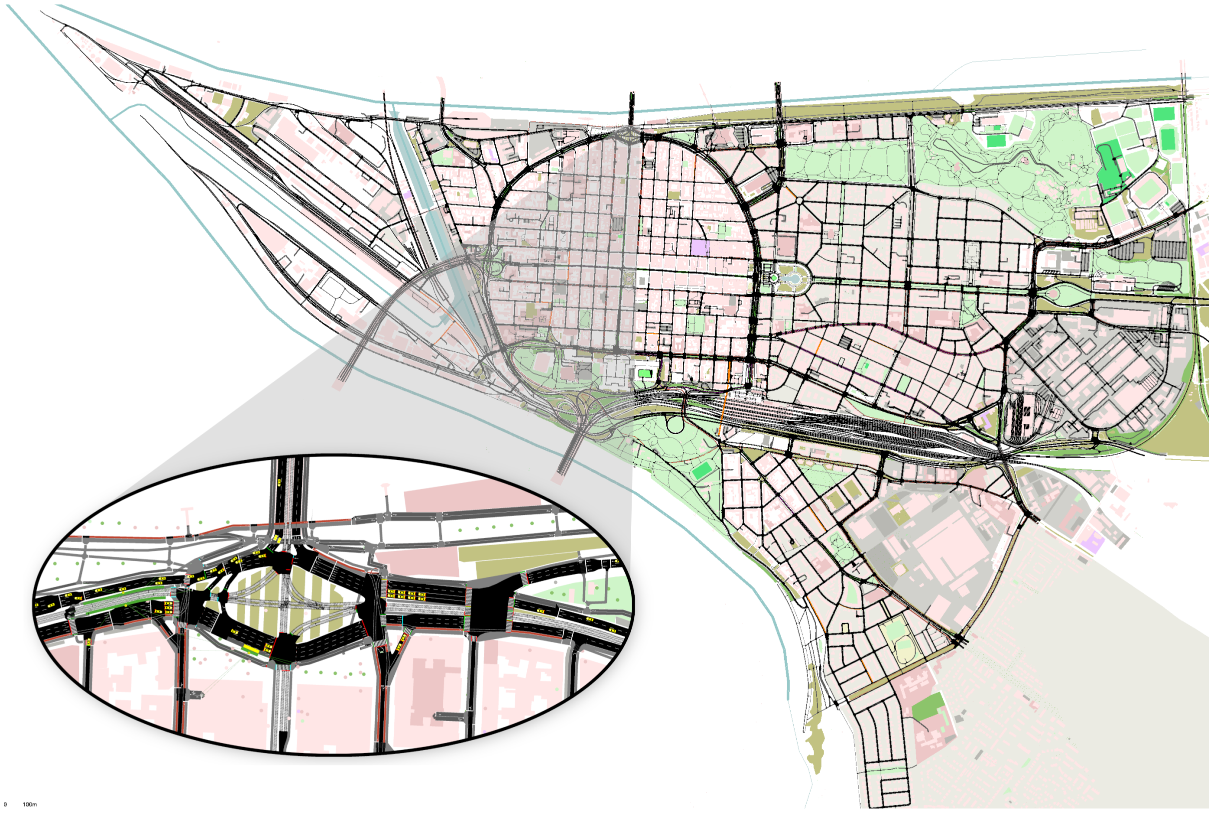

***Figure 1.*** *Visual interface of Mannheim SUMO Traffic Model (MaST).*

and rails) as well as the accurate timetables imported using data from Rhein-Neckar-Verkehr GmbH (rnv). Finally, the traffic light programs for 20 crucial junctions around the central ring of Mannheim were imported using the plans provided by the municipality. We run the scenario in two versions: One, which simulates the time frame from 6 am to 9 am, including the morning peak, and another, covering 24 hours from midnight to midnight. Both are based on data from May 16, 2023, as this date allowed for maximum data coverage in terms of the traffic cameras (some of the sensors went offline afterward). Table 2 offers an overview of descriptive statistics on MaST.

***Table 2.*** *Descriptive statistics on MaST.*

| **Metric** | **Value** |
|---|---|
| Size | 11.33 $km^2$ |
| Edge Length | 635.7 km |
| Inserted Vehicles | 41,793 (3-hour), 225,947 (24-hour) |
| Teleports | 39 (3-hour), 609 (24-hour) |
| Avg. Travel Time (seconds) | 326.15 (3-hour), 361.52 (24-hour) |
| Avg. Speed (m/s) | 4.38 (3-hour), 7.18 (24-hour) |

The following data sources were available for modeling the demand of the simulation (visualized in figure 2):

- **Induction loops**: Mainly located at the entries and exits of the network, there are eight induction loops operational in the simulated area (7 of which deliver bi-directional data). They deliver one data package per direction per 60 minutes and do not discriminate between different kinds of vehicles. The induction loops proved significantly more reliable when evaluating the different data sources at the junctions for which both induction loop and camera data were available. Therefore, they will form the sole basis for assessing the demand created for MaST.

- **Traffic cameras**: Moreover, 13 traffic cameras offer turn counts at various junctions in the network in 10-minute intervals. While they can theoretically classify different types of vehicles, internal evaluation has shown this functionality to be unreliable. Thus, we only rely on the overall counts for the specified turns. Since even these sporadically contained undercounts, we plausibilized those cameras for which induction loop data was also available, effectively redistributing the volume indicated by the loops according to the turn probabilities computed by the cameras.
- **GPS traces**: In addition to the two previous data sources deployed solely for representing the empirical demand, to enable our optimization pipeline and feed the *RouteSampler* tool provided by SUMO, we obtained GPS traces from a commercial provider. The data set spans two weeks, from June 24 to July 8, 2024, and contains 19,598 trips after filtering for trips of passenger cars traversing the simulated area during our initial target time frame from 6 am to 9 am.[4] We use this data both to create the set of routes used for calibration and to derive an initial estimate of the overall frequency of an individual route.

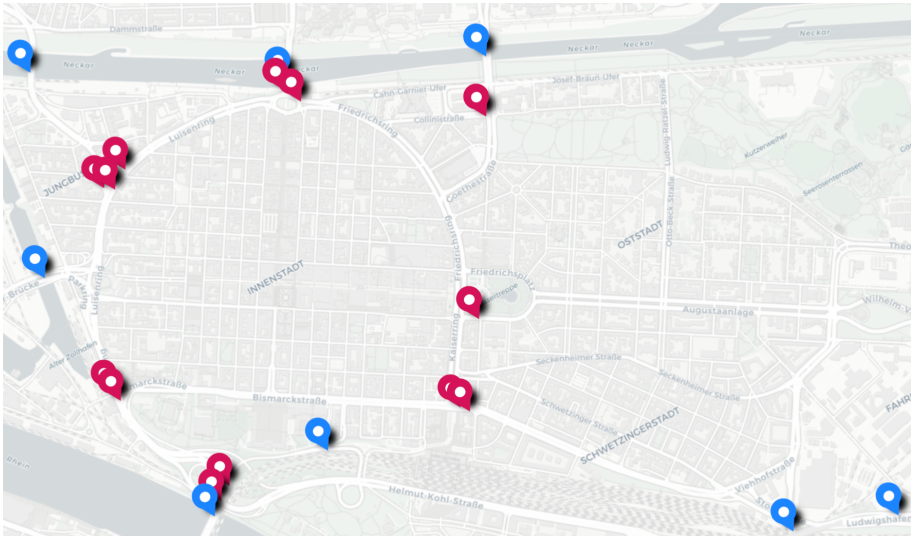


***Figure 2.*** *Location of data sensors in Mannheim. Traffic cameras are marked red. Blue markers show induction loops.*

When comparing the data coverage described above to the SUMO scenarios introduced in §2, it is noticeable that there is significantly less observational data available in Mannheim than in those scenarios relying solely on traffic counts for calibration (Dublin and Bologna). Moreover, among those models that pair demographic statistics and observational traffic data, InTas uses more induction loops with higher temporal and spatial resolutions (i.e., individual lanes covered, reported every 15 minutes). The other scenario situated in Ingolstadt has comparable data coverage to the Dublin and Bologna models, with induction loop data available from every traffic light-controlled junction in the simulation. In the following, we will describe our proposal to navigate this relative scarcity of traffic data in Mannheim without relying on population statistics.

[4]We acknowledge the temporal mismatch between the different data sources (since the empirical data used is from May 2023). However, this should not prove overly problematic as we are only using the information on the empirical traffic distribution inherent to the routes as a prior. Moreover, the deployed map matching procedure (see §3.2) only allows for routes valid in the created network.

### 3.2 Map Matching Using Hidden Markov Models

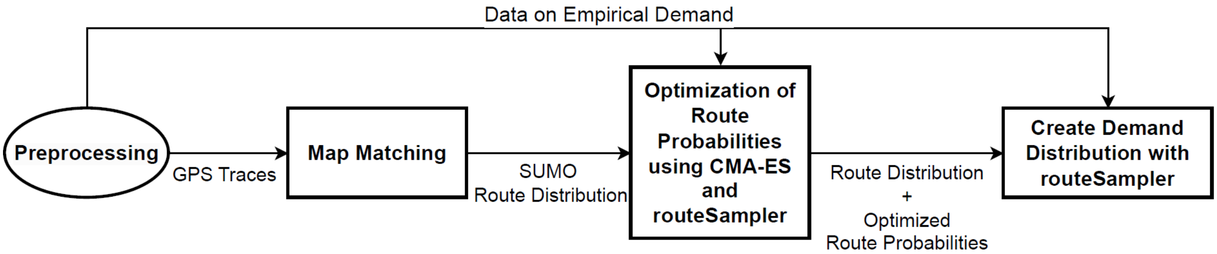


***Figure 3.*** *Overall workflow of the presented method, from preprocessing and map matching to finding the optimal route distribution for creating the final demand.*

In contrast to other approaches (e.g., Stang and Bogenberger [14]), we do not aim to develop a calibration method for route choices in SUMO from the ground up. Instead, we utilize existing tools provided by SUMO and interject the results of our optimization pipeline from outside. Figure 3 visualizes the proposed pipeline in its entirety. Among the toolkit included in SUMO, the *routeSampler* module appears particularly well-suited for our task [4]. For one, it offers great versatility concerning the counting data it can handle as it can be used, among other alternatives, with edge counts provided by our induction loops as well as the turn counts derived from the traffic cameras in Mannheim. Additionally, it allows for providing a probability distribution together with the set of routes from which it iteratively samples until the traffic counts are met. This is especially important since the sampled route distribution, which in theory satisfies the traffic data, might lead to traffic jams when running the actual SUMO simulation if the network does not support the sampled traffic flows, preventing the counts from being met. Intuitively, then, our method relies on having *routeSampler* enforce the fulfillment of the observed traffic counts while our optimization pipeline produces the route probability distribution, which best realizes these counts during simulation. The main advantage of following this approach lies in the reduced dimensionality and complexity of the optimization problem compared to sampling a demand distribution without such constraints.

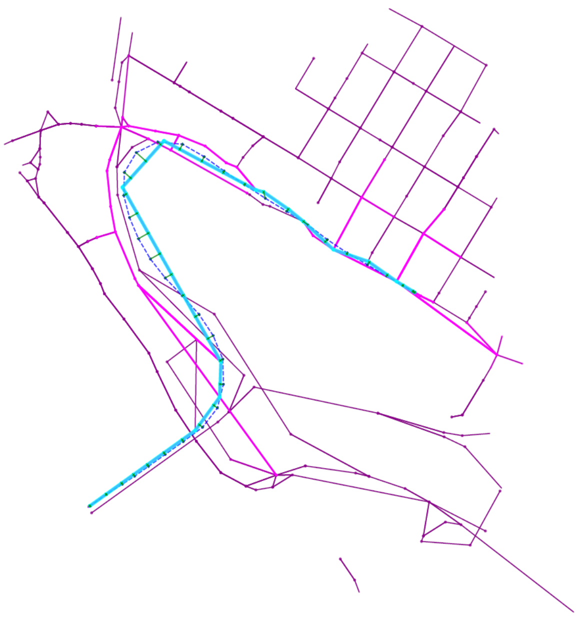

***Figure 4.*** *Exemplary map matching from GPS trace (dashed line) to SUMO edges (path in light blue).*

As *routeSampler* relies on an initial route distribution (denoted $\mathcal{R}$ in the following) from which to sample, it is crucial to provide it with a set of high-quality routes. Thus, the first step is to convert the GPS traces available to SUMO routes. This task of matching vehicle trajectories to a road network is commonly referred to as map matching [15]. While SUMO offers map matching tools of its own to handle GPS traces (e.g., *duarouter* or *tracemapper*), we found them to be inadequate in cases of missing GPS data points or complex, overlaying network architectures.[5]

Therefore, we opted for deploying a Hidden Markov Model (HMM)-based approach for map matching. A HMM is a robust statistical method for modeling generative sequences, where an underlying process produces an observable sequence.[6] While they have a large variety of applications, HMMs are particularly well-suited to map matching as they can account for the connectivity of the road network via transition probabilities, allowing them to better cope with complex and multi-layered networks. The goal, then, is to find the most likely road segment (hidden state) for a given GPS point (observation), incorporating the transition probabilities from the last emitted road segment [17]. To this end, we deployed the implementation provided by the *LeuvenMapMatching* library [18]. Figure 4 shows an example of this process. After feeding all of our GPS traces through the map matching process, we used the *routecheck* tool and *duarouter* to detect and potentially fix unconnected routes. This procedure yielded an overall number of 17,649 routes (6,617 unique). If one were to replace the HMM-based map matching with *tracemapper*, one would instead receive 8,049 routes (3,014 unique) while only marginally increasing the mean implausibility score from 1.11 to 1.21 as computed by the *implausibleRoutes* tool.[7][8]

### 3.3 Optimizing Route Probabilities

Having created an appropriate set of routes $\mathcal{R}$, it remains to design the optimization process to produce a probability distribution for said routes. Since our target function, namely, the deviation of simulated versus empirical induction loop counts, depends entirely on the simulation and is thus non-differentiable, this task represents a derivative-free or black-box optimization problem, which can be formalized thusly [19]:

Let $\boldsymbol{x} \in \mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}$. The goal is to find:

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}),$$

where the objective function $f(\boldsymbol{x})$ can only be evaluated for any $\boldsymbol{x} \in \mathbb{R}^n$. No explicit form, derivatives, or structure of $f(\boldsymbol{x})$ are assumed to be available.

In our case $\boldsymbol{x} = [x_1, x_2, \ldots, x_m] \in \mathbb{R}^m$ represents a probability vector, where $x_i$ is the probability assigned to route $i$ in a given route set $\mathcal{R} = \{r_1, r_2, \ldots, r_m\}$. The objective function$f(\boldsymbol{x})$ then corresponds to :

$$f(\boldsymbol{x}) = \text{EvaluateDemand}(\boldsymbol{x}, \mathcal{R}, \text{Observational Data}),$$

where EvaluateDemand encompasses:

[5]See https://sumo.dlr.de/docs/FAQ.html#how_do_i_generate_sumo_routes_from_gps_traces for a discussion of the different tools.

[6]See [16] for a detailed introduction to HMMs.

[7]See https://sumo.dlr.de/docs/Tools/Routes.html#implausibleroutespy for an explanation of the module.

[8]It is important to note that this comparison is not conclusive but rather meant to illustrate the fact that, for our specific data, the HMM-based approach produced significantly more routes, while preserving the overall quality of the route distribution.

1. Generating a demand distribution by feeding $\boldsymbol{x}$, the route set $\mathcal{R}$, and our observational traffic data (both induction loops and traffic cameras) to *routeSampler*.
2. Running the MaST scenario with the generated demand.
3. Computing and returning the absolute differences between observed and simulated induction loop counts.

There is a large variety of methods available to tackle problems of this nature [20]. We opted to deploy the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm due to its relative simplicity and favorable performance characteristics.[9] The core principle behind CMA-ES is aligned with general evolutionary concepts as it continuously samples candidate solutions according to an adapting distribution.

At every time step, a population of candidate solutions or individuals (we denote an individual candidate vector with $\boldsymbol{z} \in \mathbb{R}^m$) is sampled from a multivariate normal distribution, which is parametrized by a mean vector $\boldsymbol{m}$ (our current best guess for every element of $\boldsymbol{z}$) and a covariance matrix $\boldsymbol{C}$ controlling the shape and orientation of the search distribution as well as accounting for dependencies between the different elements in the distribution. After sampling, all candidate solutions are evaluated (using $f(\boldsymbol{x})$), and $\boldsymbol{m}$ is updated based on a weighted average of successful individuals (in our case, the best 25%) to increase the likelihood of them being sampled in subsequent generations. Another crucial parameter in this process is the step size $\sigma$, which determines how quickly or slowly the search space is traversed. $\boldsymbol{C}$ and $\sigma$ are adapted to balance both exploration and exploitation [22].

As a starting point for our optimization process, we want to exploit the information contained in our route set $\mathcal{R}$ created using the GPS traces provided to us. The natural avenue of doing so lies in reframing the frequencies of the routes in $\mathcal{R}$ as an initial guess for the mean vector $\boldsymbol{m}$ since they, even though most likely not representative of the overall traffic in Mannheim due to the limited vehicle fleet of the commercial provider, encode some information about the flow of traffic throughout our simulated area. Given our route set $\mathcal{R} = \{r_1, r_2, \ldots, r_m\}$, let the relative frequencies of the routes derived from GPS traces be represented by $\boldsymbol{p} = [p_1, p_2, \ldots, p_m]$, where:

$$p_i = \frac{\text{count}(r_i)}{\sum_{j=1}^{m} \text{count}(r_j)}, \quad i = 1, 2, \ldots, m,$$

and $\text{count}(r_i)$ denotes the number of occurrences of route $r_i$ in the GPS trace data. However, as indicated by figure 8 in §4, the scale of the individual values in $\boldsymbol{p}$ differs greatly, ranging from 0.0057 % (occurring once) to 3.52 % (620 occurrences). Since CMA-ES uses a single step size parameter $\sigma$ for all dimensions (i.e., routes), we thus cannot directly optimize $\boldsymbol{x}$. Instead, we use a candidate solution $\boldsymbol{z}$ sampled from $\boldsymbol{m}$ as a weight vector for $\boldsymbol{p}$, effectively fine-tuning the probability distribution yielded by the GPS traces, which gives the input $\boldsymbol{x}$ for our target function as:

$$\boldsymbol{x} = \boldsymbol{z} \cdot \boldsymbol{p},$$

where the starting mean vector $\boldsymbol{m}_0$ is initialized as uniform weights ($m_i = 1 \, \forall i$). Intuitively, we thus start off with the relative frequencies contained in our data as a best guess, which we then nudge up or downwards proportional to their size and depending on their effect on the objective function EvaluateDemand.

[9] See [21] for a thorough introduction of CMA-ES.

We first run this optimization process for the 3-hour scenario for 100 generations, each with a population of 120 candidate solutions.[10] The number of iterations was chosen to approach the number of function evaluations of $30n$ (where $n$ is the problem dimension, in our case, 6,617) cited in the literature on CMA-ES as necessary for achieving consistent improvement [21]. While our experiments fall well short of this number (12,000 vs. 198,510), the "head start" induced by the prior information of the GPS traces should balance this out. Moreover, this should also help avoid moving too much probability mass in the route distribution so as to create unrealistic demand where, e.g., one route dominates the distribution entirely. Subsequently, we utilize the outcome of this optimization round as input $\boldsymbol{p}$ for the 24-hour model, which we then optimize for 50 generations (again with populations of 120 individuals each) as the execution times greatly increase with the added simulation length.[11]

## 4. Results

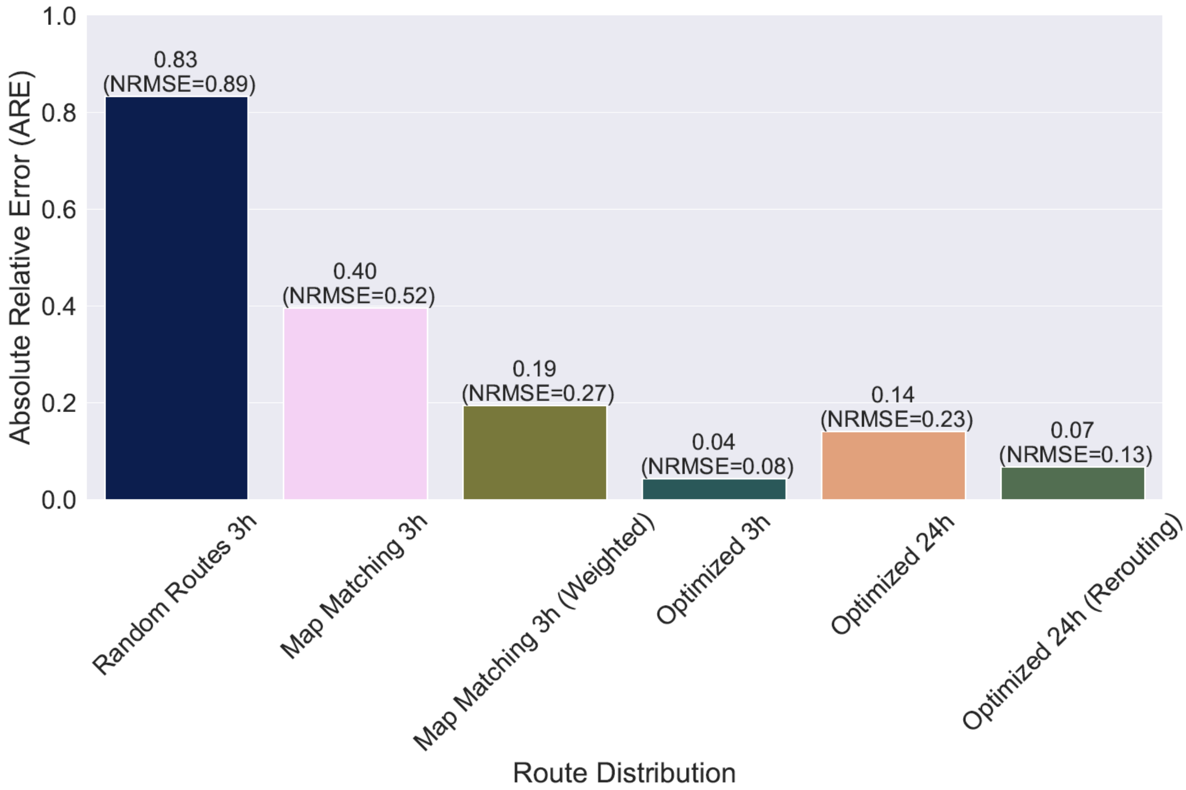


***Figure 5.*** *Absolute Relative Error (ARE) and Normalized Root Mean Square Error (NRMSE) between the simulated counts in the different experimental settings and the empirical induction loop data.*

Figure 5 visualizes the overall results of our experiments. We compare the following route distributions as inputs to our objective function EvaluateDemand in terms of how closely they are able to match the induction loop counts in Mannheim:

- **Random Routes 3h**: To establish a simple baseline, we first created a random route set using the *randomTrips* module shipping with SUMO. This avenue is the simplest starting point for creating demand for a scenario. Out of our different approaches, the random demand leads to the worst performance, posting an Ab-

[10] Running 60 simulations in parallel on an AMD Milan EPYC 7513 processor, the optimization took approximately 15 hours.

[11] Using the same setup as for the 3-hour scenario, optimizing took 148 hours.

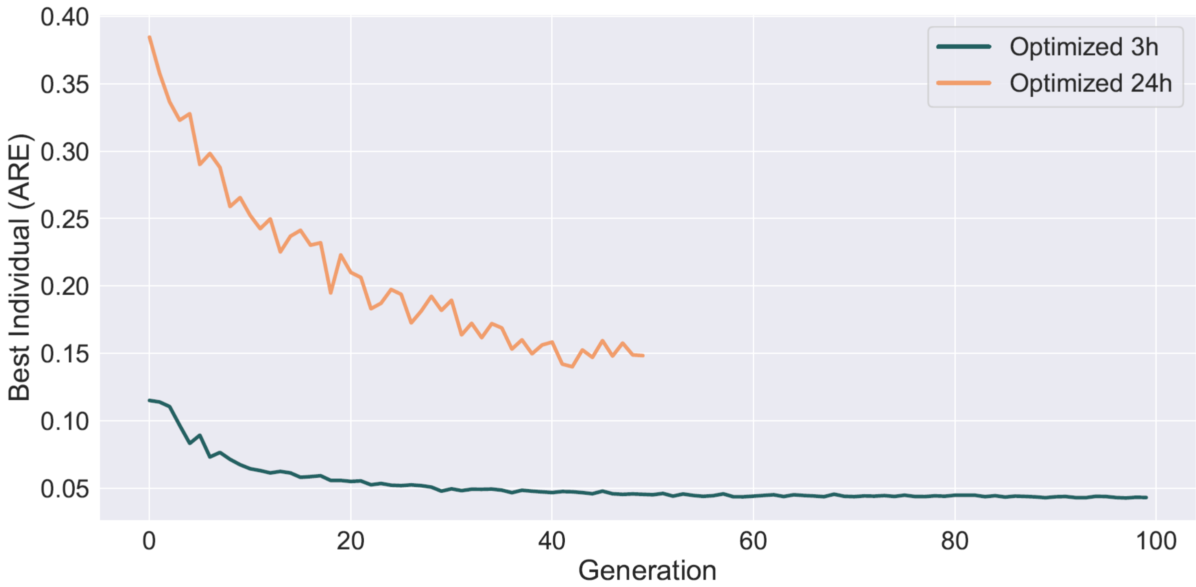


***Figure 6.*** *Best candidate solution in terms of ARE per generation for the 3-hour and 24-hour scenario, respectively.*

solute Relative Error (ARE) of 0.83 as well as a Normalized Root Mean Square Error (NRMSE) of 0.89. Thus, less than 20% of the demand in our 3-hour scenario is actually simulated. It is worth noting, though, that more tuning of the route construction process would most likely have led to some improvements.[12]

- **Map Matching 3h**: Simply using a route distribution based on empirical data, namely the output for our map matching pipeline described in §3, already brings a large improvement to the simulation as the ARE decreases to 0.40 (NRMSE of 0.52).
- **Map Matching 3h (Weighted)**: More improvements become apparent when we additionally use the information on how often a specific route occurs in the data set (using the *–weighted* argument of *routeSampler*). With an ARE of 0.19 and a NRMSE of 0.27, the route distribution already offers an acceptable performance when compared to the approaches introduced in §2 (albeit in a smaller time frame than some of them and, obviously, a very different context).
- **Optimized 3h**: Running the optimization process detailed above further improves upon this. For the 3-hour scenario, the resulting demand distribution leads to an ARE of 0.04 and NRMSE of 0.08, almost completely matching the counts recorded by the induction loops. This is particularly noteworthy as the counts between the different sensor types are not entirely compatible with one another due to measurement errors (even after plausibilization). That is to say, the sampling provided by *routeSampler* contains an initial error caused by these incompatibilities approaching the ARE shown here. Figure 6 visualizes the optimization process, where the ARE steadily decreases until it plateaus, thus indicating that the number of generations is sufficient.
- **Optimized 24h**: Optimizing the route probability distribution $x$ for the 24-hour scenario yields an ARE of 0.14 and a NRMSE of 0.23. While this is considerably worse when compared to the shorter setup, the difficulty of meeting the empirical counts is also much higher. Comparing this result to the existing and published SUMO scenarios, it appears in line with or slightly better than the two scenarios

[12]We used no additional arguments to *routeTrips*, simply creating 20,000 random routes to be able to match the empirical counts, at least in theory.

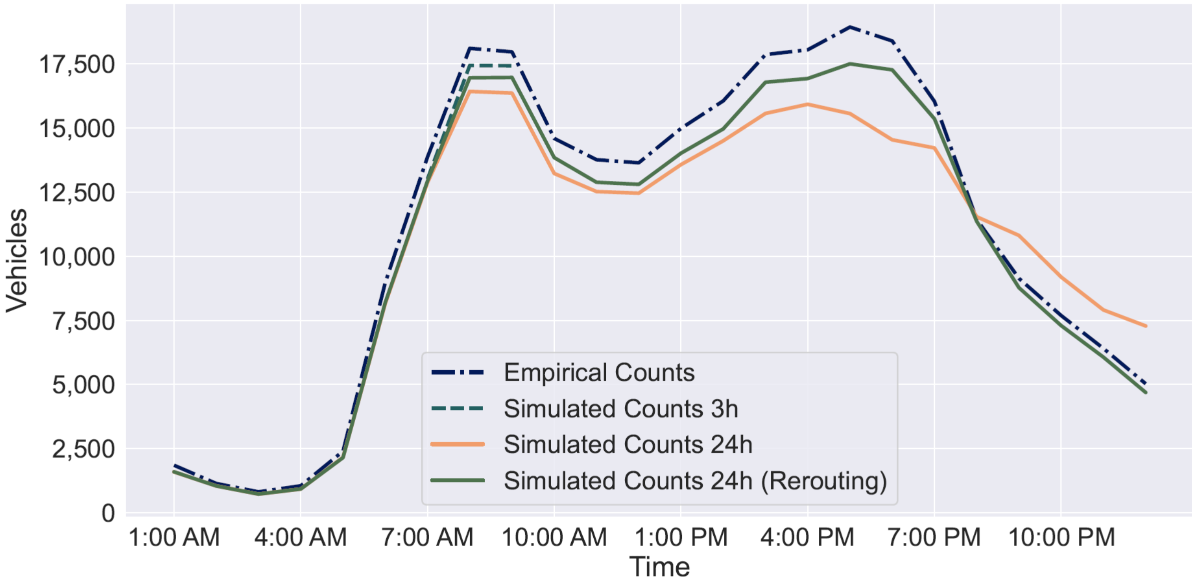


***Figure 7.*** *Amount of vehicles during the simulated time frame for the empirical induction loop data as well as the 3-hour and 24-hour simulations settings (with optimized demand, respectively).*

in Ingolstadt (NRMSE of 0.33 for InTas, ARE values between 13% and 18% for the other project simulating Ingolstadt), although we do not deploy rerouting or a warm-up phase here. When examining the optimization progress depicted in figure 6, one can observe constant improvement, which does not plateau in the same manner as the 3-hour scenario. Considering the resources already required to run the optimization process for 50 generations, we opted against optimizing for a longer period of time.

- **Optimized 24h with Rerouting**: To be able to better cope with the volume of vehicles over the whole day, we evaluate a final setting where we still use the route distribution that we optimized for the 24-hour scenario but now allow a percentage of vehicles in the scenario to perform rerouting. In particular, we follow Harth et al. [11] and set the rerouting probability of all vehicles to 0.2. Additionally, we also select a step length of 0.25 for the scenario, resulting in 4 simulation iterations per simulated second, and set the parameter *max-depart-delay* to 100, thus discarding any vehicles that cannot be inserted into the simulation after 100 seconds to avoid long traffic jams incurred by the insertion backlog. By sacrificing efficiency for maximum performance in his way, we achieve considerable improvements, halving the ARE to 0.07 and reducing the NRMSE to 0.13.

Figure 7 serves to illustrate the number of vehicles recorded by the available detectors versus the simulated counts in the respective scenarios. The curve corresponding to the optimized 3-hour scenario nearly completely shadows the empirical demand between 6 am and 9 am as indicated by its low ARE. In contrast, the optimized 24-hour simulation does not quite match the empirical traffic volume as it struggles to reach the afternoon peak of over 18,000 vehicles. Nonetheless, apart from traffic jams stemming from said peak that do not dissolve fully until late evening, simulated volume mirrors the real-world data fairly well. Finally, the optimized 24-hour setting with rerouting avoids these problems and remains very close below the empirical curve throughout the simulated day.

Finally, figure 8 is meant to serve as a reasonableness check, describing the route distributions resulting from the optimization processes vis-à-vis the initial route distribu-

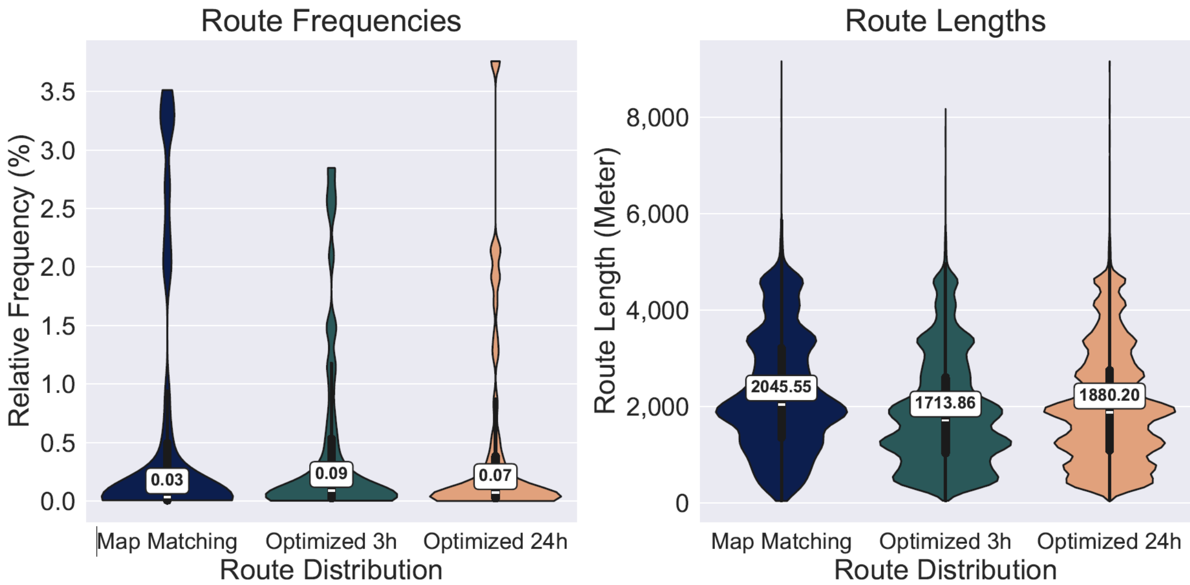


***Figure 8.*** *Distributions of route frequencies (left) and route lengths (right) for the route distributions after map matching as well as after the optimization process (3-hour and 24-hour, respectively). The white boxes designate the median values of the respective distributions.*

tion originating from our GPS trace data set in terms of how many routes approximately lie in the defined data ranges both for route frequency and length. The distributions in terms of the frequencies of individual routes remain reasonably close to one another, with the 3-hour optimization on average reducing the weight of the most frequent routes and increasing the share of more infrequent routes. The 24-hour optimization process then nudges the distribution back closer to the initial route distribution. For the route lengths, one can observe a similar pattern. Once more, the route distribution optimized for the shorter time frame leads to a reduction in the average route length, and the result of the 24-hour optimization resembles the initial distribution more closely. Overall, these findings suggest no obvious exploit induced by the optimization process whereby the initial route distribution would be altered in an unreasonable or overly drastic manner.

## 5. Conclusion and Future Work

Accurately modeling demand is crucial in constructing an empirically valid SUMO scenario. In the present paper, we proposed an approach for route choice calibration in SUMO based on an evolutionary algorithm feeding existing SUMO tools. Using this pipeline, we calibrated our MaST scenario to meet the empirical traffic counts available in Mannheim as closely as possible. We then performed a structured evaluation of the optimization process results and compared them against simple baselines. While this evaluation indicates that our approach succeeds in closely matching the empirical counts both for the 3-hour and 24-hour scenarios, there are a number of limitations of our work to consider.

Crucially, our approach only delivers a *possible* demand configuration closely matching the input counts as it compensates for the lack of information on traffic flows between observations. The created demand thus may or may not resemble the actual traffic flows. Moreover, we fine-tuned and evaluated the demand for one particular date, therefore not examining how well our approach generalizes from the input data. In a similar vein, even though the CMA-ES algorithm is relatively efficient compared to other optimization methods, the number of iterations necessary — particularly for the 24-hour scenario — renders our optimization pipeline currently only suitable for historical simulation. Consequently, in scenarios with abundant data, it may be preferable to directly model demand using tools like *routeSampler*, which do not necessitate continuous simulation iterations and thus offer higher efficiency. Like the vanilla *routeSampler* module, the suitability of our method directly depends on the availability of high-quality routes to sample from. In cases where such routes are not available, alternatives such as *flowrouter*, which do not require input routes, may be more promising.[13]

It remains for future research to perform extensive experiments to determine whether the route distribution fitted to a particular set of observational data points can be used to simulate other days or settings. Similarly, it might also be interesting to compare the presented method to existing approaches already usable in SUMO, such as Cadyts [23], in a structured manner as well as assuming a more microscopic perspective to examine how well the created traffic flows fit individual counting stations. While we also attempted to optimize the turn probability distribution that can also serve as input for *routeSampler* and should offer more generalizability, we were thus far unsuccessful in improving performance as the sampling method of *routeSampler* is biased towards under counts when confronted with an imbalance between turn probability data and traffic counts. Furthermore, we only utilized a small subset of the GPS traces available (trips on business days between 6 am and 9 am). It might bring further improvements or a faster convergence for the 24-hour scenario to select a route distribution more representative of the time frame modeled.

## Data availability statement

All data used to calibrate our model was provided by the municipality of Mannheim and is not freely accessible. The counts provided by the traffic cameras can be found at https://opendata.smartmannheim.de/dataset/ in 60-minute resolution. The GPS traces used as a basis for the route distribution were commercially obtained and cannot be publicly shared by us.

## Underlying and related material

All code used to reproduce our results (including the Mannheim SUMO Traffic Model (MaST) scenario) can be found at https://github.com/JakobKappenberger/mast. Unfortunately, as we are not allowed to even publish the processed GPS routes, the repository only contains random routes.

## Author contributions

**Jakob Kappenberger**: Conceptualization, Methodology, Software, Formal Analysis, Writing - Original Draft, Writing - Review & Editing, Visualization, Validation.
**Heiner Stuckenschmidt**: Supervision, Funding Acquisition.

## Competing interests

The authors declare that they have no competing interests.

[13] See https://sumo.dlr.de/docs/Tools/Detector.html#flowrouterpy for a description of *flowrouter*.

## Funding

This work was conducted as part of the grant "sMArt roots – Smart City Modellstadt Mannheim", funded by the German Federal Ministry for Housing, Urban Development and Building under Grant No. 13622847. The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG.

## Acknowledgements

We want to extend our thanks to our contacts in the municipal administration in Mannheim for providing us with the indication loop data and traffic light plans. Furthermore, we would like to thank our colleague Lea Cohausz and the reviewers for their helpful feedback on an earlier version of this article.

## References

[1] R. Dowling, A. Skabardonis, J. Halkias, G. McHale, and G. Zammit, "Guidelines for Calibration of Microsimulation Models: Framework and Applications," en, *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1876, no. 1, pp. 1–9, Jan. 2004, Publisher: SAGE Publications, ISSN: 0361-1981, 2169-4052. DOI: 10.3141/1876-01. Accessed: Jan. 14, 2025. [Online]. Available: https://journals.sagepub.com/doi/10.3141/1876-01.

[2] "Traffic analysis toolbox, volume III: Guidelines for applying traffic microsimulation software," United States Department of Transportation - Federal Highway Administration, Tech. Rep. FHWA-HRT-04-040, 2004. Accessed: Jan. 14, 2025. [Online]. Available: https://ops.fhwa.dot.gov/trafficanalysistools/tat_vol3/.

[3] J. Barceló, "Models, Traffic Models, Simulation, and Traffic Simulation," in *Fundamentals of Traffic Simulation*, J. Barceló, Ed., vol. 145, Series Title: International Series in Operations Research & Management Science, New York, NY: Springer New York, 2010, pp. 1–62, ISBN: 978-1-4419-6141-9 978-1-4419-6142-6. DOI: 10.1007/978-1-4419-6142-6_1. Accessed: Jan. 13, 2025. [Online]. Available: http://link.springer.com/10.1007/978-1-4419-6142-6_1.

[4] M. Behrisch and P. Hartwig, "A comparison of SUMO's count based and countless demand generation tools," en, *SUMO Conference Proceedings*, vol. 2, pp. 125–131, Jun. 2022, ISSN: 2750-4425. DOI: 10.52825/scp.v2i.107. Accessed: Jan. 15, 2025. [Online]. Available: https://www.tib-op.org/ojs/index.php/scp/article/view/107.

[5] L. Bieker, D. Krajzewicz, A. Morra, C. Michelacci, and F. Cartolano, "Traffic Simulation for All: A Real World Traffic Scenario from the City of Bologna," en, in *Lecture Notes in Mobility*, ISSN: 2196-5544, 2196-5552, Cham: Springer International Publishing, 2015, pp. 47–60, ISBN: 978-3-319-15023-9 978-3-319-15024-6. DOI: 10.1007/978-3-319-15024-6_4. Accessed: Jan. 14, 2025. [Online]. Available: https://link.springer.com/10.1007/978-3-319-15024-6_4.

[6] L. Codeca, R. Frank, and T. Engel, "Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research," en, in *2015 IEEE Vehicular Networking Conference (VNC)*, Kyoto, Japan: IEEE, Dec. 2015, pp. 1–8. DOI: 10.1109/vnc.2015.7385539. Accessed: Jan. 14, 2025. [Online]. Available: http://ieeexplore.ieee.org/document/7385539/.

[7] K. Schrab, R. Protzmann, and I. Radusch, "A Large-Scale Traffic Scenario of Berlin for Evaluating Smart Mobility Applications," en, in *Smart Energy for Smart Transport*, E. G.

Nathanail, N. Gavanas, and G. Adamos, Eds., Series Title: Lecture Notes in Intelligent Transportation and Infrastructure, Cham: Springer Nature Switzerland, 2023, pp. 276–287, ISBN: 978-3-031-23720-1 978-3-031-23721-8. DOI: 10.1007/978-3-031-23721-8_24. Accessed: Jan. 14, 2025. [Online]. Available: https://link.springer.com/10.1007/978-3-031-23721-8_24.

[8] M. Gueriau and I. Dusparic, "Quantifying the impact of connected and autonomous vehicles on traffic efficiency and safety in mixed traffic," en, in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, Rhodes, Greece: IEEE, Sep. 2020, pp. 1–8, ISBN: 978-1-7281-4149-7. DOI: 10.1109/ITSC45102.2020.9294174. Accessed: Jan. 14, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/9294174/.

[9] S. C. Lobo, S. Neumeier, E. M. G. Fernandez, and C. Facchi, *InTAS – The Ingolstadt Traffic Scenario for SUMO*, en, arXiv:2011.11995 [cs], Nov. 2020. DOI: 10.48550/arXiv.2011.11995. Accessed: Jan. 14, 2025. [Online]. Available: http://arxiv.org/abs/2011.11995.

[10] M. Langer, M. Harth, L. Preitschaft, R. Kates, and K. Bogenberger, "Calibration and Assessment of Urban Microscopic Traffic Simulation as an Environment for Testing of Automated Driving," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, Indianapolis, IN, USA: IEEE, Sep. 2021, pp. 3210–3216, ISBN: 978-1-7281-9142-3. DOI: 10.1109/ITSC48978.2021.9564743. Accessed: Jan. 15, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/9564743/.

[11] M. Harth, M. Langer, and K. Bogenberger, "Automated Calibration of Traffic Demand and Traffic Lights in SUMO Using Real-World Observations," en, *SUMO Conference Proceedings*, vol. 2, pp. 133–148, Jun. 2022, Publisher: TIB Open Publishing, ISSN: 2750-4425. DOI: 10.52825/scp.v2i.120. Accessed: Jan. 14, 2025. [Online]. Available: https://www.tib-op.org/ojs/index.php/scp/article/view/120.

[12] L. Codeca and J. Harri, "Towards multimodal mobility simulation of C-ITS: The Monaco SUMO traffic scenario," en, in *2017 IEEE Vehicular Networking Conference (VNC)*, Torino: IEEE, Nov. 2017, pp. 97–100, ISBN: 978-1-5386-0986-6. DOI: 10.1109/VNC.2017.8275627. Accessed: Jan. 14, 2025. [Online]. Available: http://ieeexplore.ieee.org/document/8275627/.

[13] Y. Yamazaki, Y. Tamura, X. Défago, E. Javanmardi, and M. Tsukada, "ToST: Tokyo SUMO Traffic Scenario," en, in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, Spain: IEEE, Sep. 2023, pp. 3597–3604, ISBN: 979-8-3503-9946-2. DOI: 10.1109/ITSC57777.2023.10422517. Accessed: Jan. 15, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/10422517/.

[14] C. Stang and K. Bogenberger, "Calibration of Microscopic Traffic Simulation in an Urban Environment Using GPS-Data," en, *SUMO Conference Proceedings*, vol. 5, pp. 71–78, Jul. 2024, ISSN: 2750-4425. DOI: 10.52825/scp.v5i.1099. Accessed: Jan. 15, 2025. [Online]. Available: https://www.tib-op.org/ojs/index.php/scp/article/view/1099.

[15] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data," in *Proceedings of the 31st International Conference on Very Large Data Bases*, ser. VLDB '05, Trondheim, Norway: VLDB Endowment, 2005, pp. 853–864, ISBN: 1595931546.

[16] L. Rabiner and B. Juang, "An introduction to hidden Markov models," en, *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986, ISSN: 0740-7467. DOI: 10.1109/MASSP.1986.1165342. Accessed: Jan. 21, 2025. [Online]. Available: http://ieeexplore.ieee.org/document/1165342/.

[17] P. Newson and J. Krumm, "Hidden Markov map matching through noise and sparseness," en, in *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Seattle Washington: ACM, Nov. 2009, pp. 336–343, ISBN: 978-1-60558-649-6. DOI: 10.1145/1653771.1653818. Accessed: Jan. 21, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/1653771.1653818.

[18] W. Meert and M. Verbeke, "HMM with non-emitting states for Map Matching," en, in *European Conference on Data Analysis 2018*, Paderborn, Germany, 2018.

[19] S. Alarie, C. Audet, A. E. Gheribi, M. Kokkolaras, and S. Le Digabel, "Two decades of blackbox optimization applications," en, *EURO Journal on Computational Optimization*, vol. 9, p. 100 011, 2021, ISSN: 21924406. DOI: 10.1016/j.ejco.2021.100011. Accessed: Jan. 22, 2025. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2192440621001386.

[20] J. Larson, M. Menickelly, and S. M. Wild, "Derivative-free optimization methods," en, *Acta Numerica*, vol. 28, pp. 287–404, May 2019, ISSN: 0962-4929, 1474-0508. DOI: 10.1017/S0962492919000060. Accessed: Jan. 20, 2025. [Online]. Available: https://www.cambridge.org/core/product/identifier/S0962492919000060/type/journal_article.

[21] N. Hansen and A. Ostermeier, "Completely Derandomized Self-Adaptation in Evolution Strategies," en, *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, Jun. 2001, ISSN: 1063-6560, 1530-9304. DOI: 10.1162/106365601750190398. Accessed: Jan. 17, 2025. [Online]. Available: https://direct.mit.edu/evco/article/9/2/159-195/892.

[22] G. Shala, A. Biedenkapp, N. Awad, S. Adriaensen, M. Lindauer, and F. Hutter, "Learning Step-Size Adaptation in CMA-ES," en, in *Parallel Problem Solving from Nature – PPSN XVI*, T. Bäck et al., Eds., vol. 12269, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2020, pp. 691–706, ISBN: 978-3-030-58111-4 978-3-030-58112-1. DOI: 10.1007/978-3-030-58112-1_48. Accessed: Jan. 22, 2025. [Online]. Available: http://link.springer.com/10.1007/978-3-030-58112-1_48.

[23] K. Nagel, M. Zilske, and G. Flötteröd, *CaDyTS: Calibration of Dynamic Traffic Simulations*, 2016. DOI: 10.14279/DEPOSITONCE-7753. Accessed: Mar. 26, 2025. [Online]. Available: https://depositonce.tu-berlin.de//handle/11303/8619.