Probabilistic routing for agent-based simulations

Working Paper

Author(s):
Meister, Adrian; Liang, Zheng; Balac, Milos

Publication date:
2024-08

Permanent link:
https://doi.org/10.3929/ethz-b-000689046

Originally published in:
Arbeitsberichte Verkehrs- und Raumplanung 1889

PROBABILISTIC ROUTING FOR AGENT-BASED SIMULATIONS

Adrian Meister
IVT
ETH Zürich
CH-8093 Zurich
adrian.meister@ivt.baug.ethz.ch
ORCID: 0000-0002-3350-9044

Zheng Liang, Corresponding Author
Department of Civil and Environmental Engineering
The Hong Kong University of Science and Technology
zliangam@connect.ust.hk
ORCID: 0000-0002-1677-3257

Milos Balac
CSFM
ETH Zürich
CH-8092 Zurich
balacm@ethz.ch
ORCID: 0000-0002-6099-7442

Word Count: 6403 + 4 table(s) $\times$ 250 = 7403 words

Submission Date: August 1, 2024

ABSTRACT

Traditional route assignment approaches in agent-based models often rely on least-cost path algorithms, which may not accurately reflect the complex decision-making processes of cyclists. This research addresses these limitations by incorporating probabilistic elements into the routing model, thus accommodating the variability in route choices observed in real-world scenarios. The proposed model integrates a Recursive Logit framework to account for the influence of various factors such as gradient, surface quality, traffic conditions, and dedicated cycling infrastructure on cyclists' route selection. A case study using a detailed Zurich scenario demonstrates the model's application and effectiveness. Results show that the probabilistic routing model not only aligns more closely with observed cyclist behavior but also offers a robust tool for urban planning and policy evaluation aimed at promoting sustainable and active transportation modes.

Keywords: cycling, route-choice, agent-based, MATSim

INTRODUCTION

The transport research community is increasingly interested in agent-based simulations to study active mobility. On the one hand, this interest is driven by a shift of perspective towards more sustainable, healthy, and livable urban environments. On the other hand, micro-/mesoscopic agent-based frameworks are specifically well-suited to model active mobility as they allow for high spatio-temporal granularity and inter-dependencies. The representation of route choices, that is the traffic assignment of any such framework, typically has to capture high levels of heterogeneity when modeling active modes. The routing behaviour of motorized travel can arguably be well reproduced through known routing algorithms, with time and/or distance being minimized under different assignment schemes. However, when considering active modes, the existing literature reports many more factors that determine route choices and systematic deviations from shortest routes (*1, 2*). Cyclists, for instance, consider gradient, surface, traffic flow, continuity, intersection design, and dedicated cycling infrastructure (*3*).

The current standard practice for routing within agent-based simulations relies on generating least-cost paths. Cost functions, which are minimized, can be defined using various attributes other than just time and distance. The parameters of these cost functions are typically derived from empirical results or manually fitted to observed data. However, this approach (referred to as all-or-nothing assignment) comes with notable drawbacks, one of them being the exploration of the solution space, where the achieved equilibrium does not include different, behaviorally consistent routes for the same origin-destination (OD) pair. A realistic assignment ideally considers: (1) dynamic network effects (see dynamic user equilibrium), i.e., that the route choice depends on network attributes that dynamically change, and (2) stochastic agent behavior (see stochastic user equilibrium), which accounts for non-deterministic behavior of individuals. The former can be accounted for through cost functions that include network attributes that dynamically change throughout different simulation stages. The latter requires introducing some sort of stochasticity, as e.g. proposed by Nagel et al., Ziemke et al. (*4, 5*). These high-level approaches are used in all open-source agent-based transport simulation frameworks (e.g., MATSim (*6*), SUMO (*7*), POLARIS (*8*), METROPOLIS (*9*)). The remaining drawback relates to the underlying core principle of generating routes based on cost-minimization algorithms; the interpretation of the generated behavior is limited as the results are not based on actual behavioral models.

This paper presents the integration of explicit discrete route choice models into the agent-based framework MATSim. It represents an obvious research direction, which, to the best of the authors' knowledge, has not yet been presented for any other agent-based transport simulation framework. Discrete route choice models, estimated from stated- or revealed preference data, are backed by years of research and can be effectively used for prediction. They allow to realistically model behavioral heterogeneity using econometric theory and typically allow for faster simulation convergence towards user equilibria (*10*). We describe the technical integration of such a model into MATSim and demonstrate the results using a case study for the city of Zurich. In the first step, we implement the route choice model only for cycling but stress that our method is applicable to any mode given a respective model.

BACKGROUND

Discrete route choice models

Discrete route choice models come in two conceptually different types. Path-based approaches estimate model parameters by comparing complete routes with each other. Link-based approaches model the choice process sequentially at each link.

Path-based models rely on two fundamental steps. The first is to derive a set of candidate routes - the choice set. The second is to choose one route based on a decision model, typically a Logit formulation. The derivation of choice sets is anything but trivial and substantially affects estimated model parameters. Most used methods are deterministic and based on some sort of repeated shortest path calculations paired with labeling, link elimination, or induced stochasticity (see Ton et al. (*11*) for a detailed review). The decision model is estimated based on observed routes for which individual choice sets are created and compared to. The most widely used are so-called correction-terms models that adjust the choice probabilities to account for correlation across routes, i.e., the Path Size- and C-Logit (*12, 13*).

Link-based models were specifically designed to circumvent the choice set generation. They decompose the route choice into a problem of sequential link choices. Specifically, decision-makers choose at each link, choosing which link to take in the next step until they reach the destination. The choice set for each decision is explicitly given through the network. Doing so, the approach allows for an unbiased estimation of model parameters. In the context of integrating any such models in large-scale simulation software, the link-based models has a substantial advantage w.r.t. the required computational efforts as, once estimated, one only needs to compute a closed-form probability function. The most widely used models are the Recursive Logit (RL) (*14*), as well as its nested version (*15*) that allows for correlated error terms.

Route choices in agent-based transport simulations

Route choices are implemented in different ways across agent-based transport simulation frameworks. Nguyen et al. (*16*) provides an overview of the most relevant agent-based simulation frameworks currently used by the research community and how those model route choice decisions. We restrict our review to those that best compare to MATSim, i.e., micro-/mesoscopic traffic simulators which model mid-term decisions (mode, route and destination choice), applied to demand modeling on large-scale networks with several hundred thousand agents. Throughout all frameworks, a given OD pair is routed using the well-known A* or Dijkstra algorithms based on a given cost function.

Frameworks like POLARIS and METROPOLIS model the routing behaviour solely using the above-mentioned routing algorithms. They account for dynamic network effects (e.g., live traffic information) by allowing agents to re-route during the actual simulation on the network. In METROPOLIS, the agents' route choices are revised during the simulation on the network at every intersection based on currently measured and historical travel times. Similarly, POLARIS models route choice in two ways. One is a general re-routing between iterations if the gap between experienced and predicted travel times surpasses a specified threshold or is based on live information about the network using a bounded rationality en-route switching model. Both frameworks do not generate an actual choice set, and the traffic assignment depends on repeated least-cost path calculations under changing network conditions. Neither use an explicit discrete choice model or routing with stochasticity, i.e., for a set of given network conditions, they produce an all-or-nothing assignment.

In SUMO, agents are routed based on generalized travel costs for which weights can be specified. Each agent's choice set consists of the last *n* iterations' least-cost routes. The choice of routes is performed using probabilistic models, including different Logit formulations and the C-Logit. MATSim has a similar approach, where agents build a choice set throughout the iterative simulation process. Each agent has a predefined daily activity chain for which it can decide on departure time, mode, and route choice. The resulting plan is routed using least-cost paths and scored using a global scoring function, which considers all choice dimensions through dedicated parameters. At each iteration, agents can change their plan, including re-routing, and select from the *k* plans kept in memory using choice mechanisms. Among others, these include an MNL as well as a Path Size Logit. MATSim also supports re-routing during an iteration to incorporate dynamic decision-making.

While SUMO and MATSim are often referred to as using discrete choice models for (route) choices, the formulations do not include proper additive utility functions with estimated parameters. SUMO only uses the generalized costs originating from the routing, and MATSim uses the score from the scoring function. The parameters of these cost/score functions are derived from the literature and/or manually fitted. Both frameworks also do not build an actual choice set for each iteration but rely on enumerated past choices throughout the simulation and are solely based on changing network conditions relevant to routing (mostly time). No behaviorally consistent stochastic agent behavior can be incorporated. A notable extension of MATSim includes the BicycleModule from Ziemke et al. (*5*). The authors included several cycling relevant attributes in the routing cost functions and additional random parameters, however not using explicit discrete choice models.

METHODS

Conceptual integration

Discrete choice models have already been paired with MATSim to model mode choice decisions (*10*). In that case, the choice relevant variables like travel times and costs, are calculated for each possible mode of a trip (or complete tours), based on the aforementioned routing methods. An agent then chooses the mode(s) based on an MNL model formulation. For the integration of the route choice models, we follow the same hierarchical structure of choice decision; that is, the route is computed individually for each mode before the actual mode is selected, see Figure 1.

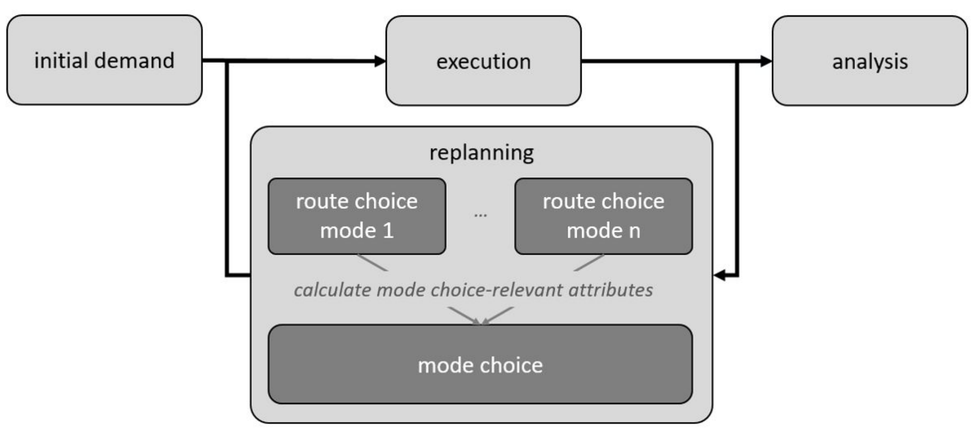

FIGURE 1 Conceptual integration of a discrete route choice model into the MATSim loop.

Different route choice models were evaluated in a preliminary comparative study (*17*) that covers the same scenario used later in this paper. The main evaluation criteria were the computational burden and accuracy when predicting the model fit, as well as the behavioural validity of the obtained model parameters. The results showed that, while link- and path-based models allow for a similar behavioural interpretation (i.e., parameter signs and relative effect sizes), the path-based models are highly sensitive to the choice set generation. These findings are aligned with growing consensus within the research community that, if possible, choice set generation methods should generally be avoided to increase reproducibility and comparability and decrease related biases across different studies. The computational burden for predicting is another practical argument for choosing the RL over path-based methods. The fact that trips need to be routed independently of which mode is ultimately assigned to them results in a potentially high number of required OD pair predictions. Furthermore, these predictions potentially need to be repeated throughout different iterations to allow for dynamic network effects.

Technical integration

The following presents the relevant technical aspects required to integrate an RL model within the MATSim environment. We use the regular RL for demonstration purposes; however, the same methodology applies to the nested version. A decision maker at link k will choose the next link a from the set of outgoing links $A(k)$ that maximizes the sum of the instantaneous utility $u(a|k)$ and the expected downstream utility $V^d(a)$, where d is the destination link. The next-link probability from link k to link a is given in equation (1):

$$P^d(a|k) = \frac{e^{\frac{1}{\mu}(v(a|k)+V^d(a))}}{\sum_{a'\in A(k)} e^{\frac{1}{\mu}(v(a'|k)+V^d(a'))}} \tag{1}$$

The expected downstream utility is given by the continuation of this link choice process based on the Bellman equation (*18*), shown in equation (2):

$$V^d(k) = E[\max_{a\in A(k)}(v(a|k)+\mu\varepsilon(a)+V^d(a))] \quad \forall k\in A \tag{2}$$

where d is the destination link and $u(a|k) = v(a|k)+\varepsilon(a)$ the instantaneous utility. We note that the downstream utility of the destination link $V^d(d) = 0$, and the second equality holds when the random error $\varepsilon(a)$ is assumed to be IID with extreme value type I. The random error term has a scale parameter μ, which is fixed for estimation. We do not include the link-size attribute that can be used to account for correlated routes (*14*), due to the commonly reported large additional computational burden (*19, 20*).

Using the model for prediction involves calculating the downstream utility V^d for all links towards a given destination d, and then successively applying equation (1) from the origin until d is reached. Given estimated parameters for the instantaneous utility u, V^d can be calculated in a straightforward way by propagating backward through the network starting from d (similar to the "backward-pass" in the assignment model of Dial (*21*)) until all links are processed. While this appears to be a suited problem for applying well-known shortest-path (SP) algorithms like Dijkstra (i.e., calculating the least-cost path from the destination node to every potential origin node), the models applied in this study are estimated on bi-directional graphs and include a u-turn parameter. Hence, one has to consider each link's cost depending on its previous downstream link toward the destination. We consequently implement a more general breadth-first graph search procedure that ultimately processes all link/incoming-link combinations in the graph. While this leads to an additional computational burden, it must be noted that simplifying the model, i.e., neglecting the directional dependency of the downstream utility, showed to have a notable impact on generated routes, especially given that the networks typically used in MATSim applications (and in this study) are almost fully bi-directional and have a considerable amount of dead-ends. Especially w.r.t. the latter, omitting the u-turn penalties leads to unrealistic exploration of dead-ends along the path. The pseudo-code of the developed procedure is defined in **Algorithm 1**.

Algorithm 1: Adapted BFS

Data: Graph *G(n,l)*, destination node n_d, origin node n_o, links l
Result: Map *DU(l)* with maximum downstream utility $\forall l \in G$
Initialize queue Q with initial node-link pair $p(n_d, null)$
Initialize empty list V of visited pairs
Initialize map *DU(l)* = -Inf for $\forall l \in G$
while *Q not empty* **do**
  Current pair $p_{cur}(n_{cur}, l_{prev})$ = top entry in Q, remove top entry from Q
  if $p_{cur} \in V$ **then**
    continue
  end
  for *All combinations of incoming link* l_{cur} *from* n_{cur} *and* l_{prev} **do**
    calculate $utility = DU(l_{prev})$ + instantaneous utility $IU(l_{prev})$
    if l_{cur} = *reversed* l_{prev} **then**
      $utility$ += u-turn penalty
    end
    if $utility > DU(l_{cur})$ **then**
      $DU(l_{cur}) = cost$
    end
    add $p(n_{next}, l_{cur})$ to Q with n_{next} origin node of l_{cur}
  end
  add p_{cur} to V
end

The algorithm starts from the destination node n_d, evaluates and stores its incoming links' downstream utility (which are zero, $DU(null) = IU(null) = 0$) and adds the incoming links' origin nodes (i.e., the neighbors of n_d) to a queue. The queue uses a *pair* object to keep track of each node's preceding link, e.g., the links connecting the neighbors at depth 1 to n_d in the second iteration. This allows identifying potential u-turns in the evaluated trajectories. The downstream utility of any link is defined by the sum of the downstream- and instantaneous utility of the preceding link, plus an additional u-turn penalty if detected. The map *DU* keeps track of the largest known utility for any given link and ensures optimality. Every node/link pair is only evaluated once, and it is not dependent on which order these are processed. In practice, the computational burden of the developed BFS can be reduced by, e.g., restricting the search space to only relevant parts of the network, which is the reason for designing the procedure as breadth-first as opposed to depth-first search (see Section 4.1).

Zurich scenario

The following presents the Zurich MATSim scenario used to evaluate the RL integration while only focusing on the cycling-relevant aspects. The supply side, i.e., the network, is based on OpenStreetMap (OSM), generated using the SNMAN package (*22*). The network is filtered such that it only includes cycling-relevant edges, i.e., highways are excluded based on the OSM high-way tags `motorway`, `motorway_link`, `trunk`, `trunk_link` . The cycling infrastructure attributes are derived from the relevant OSM tags. Bike paths are physically separated from the motorized infrastructure. Bike lanes consist of painted markings / lanes on motorized streets. They are not physically separated from motorized traffic. The speed limit information is sourced from OSM, with most streets having a 50 kmlimit. Information about gradients is derived from the Google Elevation API. Traffic levels are derived from the national aggregated transport model (*23*) and represented through annual average daily traffic (AADT) counts. The demand side, i.e., the synthetic population and its travel patterns, is sourced from the existing MATSim scenario developed based on the eqasim framework (*24*) and adapted to Switzerland (*25*). We only consider the static cycling demand from the relaxed scenario within the network boundaries. For the 100% scenario, this includes around 200,000 cycling trips from 80,000 agents. The agents are attributed with an e-bike ownership indicator based on the latest national household travel survey (*26*).

The RL model was estimated for four different subgroups, differentiating between regular and e-bikes, as well as between males and females. The data for the estimation come from the EBIS study (*27*) that collected GPS tracking information of cyclists in Switzerland. For the Zurich region, the data includes around 36,000 successfully map-matched cycling trips from 2022 and 2023. Around 20,000 are used to estimate the four models (with around 5,000 observations per model), and the rest is used for validation. Table 1 shows four different sets of parameters and their respective distance-equivalents (DE, sometimes called Value-of-Distance, or VoD, indicators). A DE-value of e.g. -0.1 means that a cyclist perceives the distance 10% shorter if the respective attribute is present. The estimation results align with previous studies conducted in this area (*17, 28*). In particular, bike lanes are perceived to be more favorable than separated bike paths, which is due to the high share of lanes over paths in the study area (forced choices). Furthermore, streets with higher average traffic loads are perceived as more favorable than those with less traffic. This is because cyclists, as well as motorized traffic, tend to follow the main corridors through the city (and especially the bottlenecks within the network).

TABLE 1 RL model estimates and distance-equivalents (DE) for different population subgroups.

group	male				female			
	bike		e-bike		bike		e-bike	
	est.	DE	est.	DE	est.	DE	est.	DE
length [m]	-0.0236	1	-0.0230	1	-0.0240	1	-0.0232	1
bike path [m]	0.0017	-0.07	0.0020	-0.09	0.0018	-0.08	0.0023	-0.10
bike lane [m]	0.0032	-0.13	0.0026	-0.11	0.0030	-0.12	0.0026	-0.11
speed 30kmh [m]	0.0013	-0.05	0.0009	-0.04	0.0015	-0.06	0.0013	-0.05
slope 2-6% [m]	0.0007	-0.03	0.0021	-0.09	0.0014	-0.05	0.0011	-0.05
slope 6-10% [m]	-0.0012	0.05	0.0032	-0.14	-0.0004	0.02	-0.0010	0.04
slope >10% [m]	-0.0177	0.75	-0.0129	0.56	-0.0072	0.30	-0.0152	0.65
AADT >10k [m]	0.0026	-0.11	0.0020	-0.09	0.0013	-0.05	0.0027	-0.11
u-turn [-]	-1.5369	-	-1.7132	-	-1.4425	-	-1.6339	-

Note: all parameters, except the one for slope 6-10% for female regular bike riders, are significant at the 1% level.

The ability to generate accurate predictions can be evaluated by applying the models to their respective validation sets. Considering the current routing methods in MATSim, comparing the RL with SP-based routing is obviously interesting. We further consider a stochastic SP router (SSP, see Section 5.3), as this allows us to mimic the probabilistic nature of generated routes while avoiding the efforts related to the RL from a user perspective (model estimation and computational overhead at runtime when used for prediction). The SP and SPP routing are performed for the unweighted graph, as well as for a DE-weighted version. We compare these three routing strategies by computing the first preference recovery (FPR) (see Meister et al. (*28*)) for each route in the validation set. The FPR represents the share of routes in the validation set for which the model makes a correct prediction. In a route-choice context, the FPR is typically computed and plotted for different threshold values, i.e., how much share of a certain route needs to be predicted for the prediction to be considered successful. To account for the probabilistic/stochastic behavior of the RL/SSP, we generate 100 routes for each validation OD pair and compute the respective average FPR. The resulting curves are shown in Figure 2. One can see that the RL outperforms the other SP-based routing strategies and that the SP-based routing performs slightly better on the DE-weighted graph than the unweighted one. The results are, again, similar to previously conducted studies for this area (*17, 28*). Compared to the literature, the obtained values are on the lower range which is mostly due to the comparably high network density.

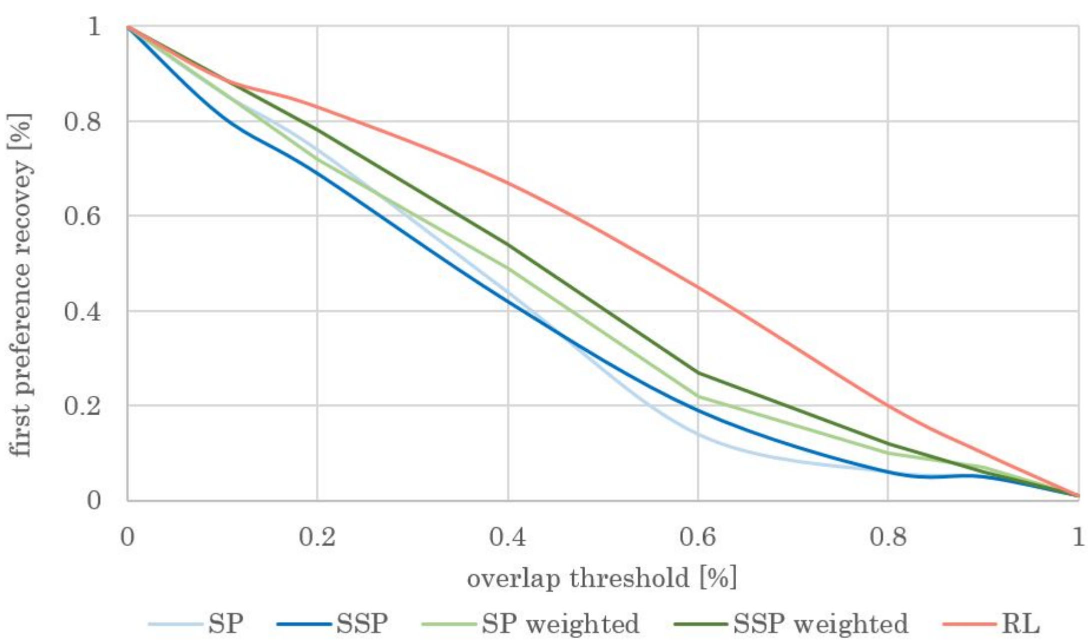

FIGURE 2 FPR-curves for the three considered routing strategies.

RESULTS

Computational efficiency

The following evaluates the computational efficiency of the different routing strategies. We report the average computation times for 10,000 random OD pairs for various SP routing algorithms available in the MATSim environment, the SSP and the RL, for three network sizes. It should be noted that both newly developed routers (SSP and RL) are not specifically optimized for computational performance. The following analysis is meant to provide ball-park numbers and justify further research on the algorithmic efficiency. The considered MATSim routers include Dijkstra, Speedy-Dijkstra (improved through bi-directional, as opposed to uni-directional, network structure), AStarLandmarks router (same concept as Dijkstra but avoiding processing all nodes in the graph using destination-directed heuristics), as well as SpeedyALT, an data-structure optimized version of AStarLandmarks. The SSP is implemented based on the Speedy-Dijkstra class. All computations are performed single-threaded on a personal computer with an Intel Core i7 2GHz CPU. The results are shown in Table 2 below. One can see that the SpeedyALT and AStarLandmarks routers are substantially faster than both other SP routers, especially for larger networks. The introduction of stochasticity in the SPP only slightly increases the computation times over the Speedy-Dijkstra.

TABLE 2 Computational efficiency for random OD pairs and different network sizes.

		Dummy	Zurich	Zurich agglo.
network	links [n]	1,738	47,100	266,631
	extent [km^2]	4.1	351.7	3837.7
SpeedyALT	[ms]	**0.11**	**0.98**	**3.77**
AStarLandmarks	[ms]	**0.19**	**1.59**	**6.61**
Speedy-Dijkstra	[ms]	**0.56**	**10.98**	**51.68**
Dijkstra	[ms]	**0.59**	**14.01**	**69.74**
SSP	[ms]	**0.61**	**11.27**	**54.22**
RL	[ms]	**1.65**	**40.92**	**215.36**
$RL_{earlystop}$	[ms]	**1.25**	**33.49**	**187.44**

Even though theoretically having a smaller time complexity, the RL router is about 3-4 times slower than the (Speedy-)Dijkstra router and about 10-15 times slower than the AStarLandmarks ones. The computation for the RL router includes two components, i.e., calculating the downstream utility and looping through the network while applying the decision rule (see equation (1)) at every link. The latter component represents a negligible share of the overall computation time. Potential optimization strategies must hence address the developed BFS procedure, e.g. by exploiting more efficient data-structures or parallelizing the search, both of which we leave open for future research. Another obvious way to reduce the computational burden of the search is to restrict it only to the relevant parts of the networks. We do this by stopping the search after a specified number of additional iterations once the origin node has been found. Given the breadth-first principle (as opposed to depth-first), this can substantially reduce the search space, especially for short OD pairs relative to the network size. The respective computation times for the evaluated networks ($RL_{earlystop}$) can be reduced by about 15%. It should be noted that this introduces an additional parameter, the number of additional iterations after the origin has been found, that needs to be calibrated given a certain network and model structure. As mentioned in Section 3.2, the models for our use case include a u-turn parameter. Models estimated without such parameters would allow to apply a reversed SP algorithm from the destination towards all possible origins, which would lead to computation times much closer to the actual SP-routers.

Probabilistic behavior and parameter sensitivity

In the following, we demonstrate the probabilistic behavior of the RL router and comment on its sensitivity to estimated parameters and utility function specifications. Figure 3 shows the heat maps of two sets of 100 generated routes for the same OD pair. The destination is at the eastern end and the shortest path is highlighted in white. The routes in the top figure are generated using a model that only includes the length parameter in its utility function and the ones in the bottom figure have a full utility function specification according to the parameters shown in Table 1. One can see that both sets of routes have a certain overlap with the shortest path. For the full model at the bottom, one clearly sees the effects of the slope parameters on the uphill part, east of the river, towards the destination. One can further observe that the routing generally gets more directed towards the destination the closer the destination is to the current link.

Figure 4 again shows the heat maps of two sets of 100 generated routes for the same OD pair, this time only using the length parameter in the utility function but with variations of the random error scale parameter μ (see equation (2)). Changing the error scale during prediction, while it was fixed during estimation, is not behaviorally consistent. It is, however, still interesting to show how the route generation can be controlled, e.g., for calibration (analog to often performed scaling of utility constants in a mode choice context). The routes in the bottom figure are generated using $\mu = \frac{2}{3}$, while the ones in the top are generated using $\mu = \frac{3}{2}$. One can see that scaling the error down leads to more deterministic routes and vice-versa. The introduced variation is the strongest the further away the destination is. It should be noted that for our use case, using, e.g., a value of $\mu = 5$ results in infinite looping, i.e., fully random behavior.

Comparison to stochastic shortest path

The evaluation in the previous sections showed a substantial computational overhead when using the RL, especially in large networks. Considering the desired heterogeneity in generated routes (and neglecting the behavioral interpretability), an obvious alternative is to apply a stochastic SP (SSP) router, i.e., one that has some sort of randomness included to generate different routes for the same OD pair. This idea has already been proposed by Nagel et al., Ziemke et al. (*4, 5*) in a (cycling) route choice context and is based on the concept of changing the SP weighting parameters by drawing from either normal or log-normal distributions. We use the same concept by applying a random factor $(1+|X|)$ with $X \sim \mathcal{N}(0,\sigma^2)$ to the cost of each link. Applying the random factor to the cost of links instead of the SP parameters is a more general formulation that also works when using unweighted graphs (e.g., only length as a parameter). The resulting variation in generated routes can be controlled by choosing σ. For our case, we manually calibrate to $\sigma = 0.2$, which, relative to the RL, provides the most comparable results w.r.t. aggregated route attributes as well as overlap with the respective shortest path (analog to the FPR metric presented in Section 4.3). Table 3 shows the average attribute distributions for 10,000 routed OD pairs on the Zurich network. The RL only uses a length parameter, the SSP is using the un-weighted graph. One can see that the sets of routes are very similar, except for the general overlap (i.e., correlation, analog to path-size metric) being slightly higher for the RL. These aggregated metrics do not consider the spatial distribution of routes. Still, they generally show that the SSP router can be considered a valid alternative depending on the analysis use case.

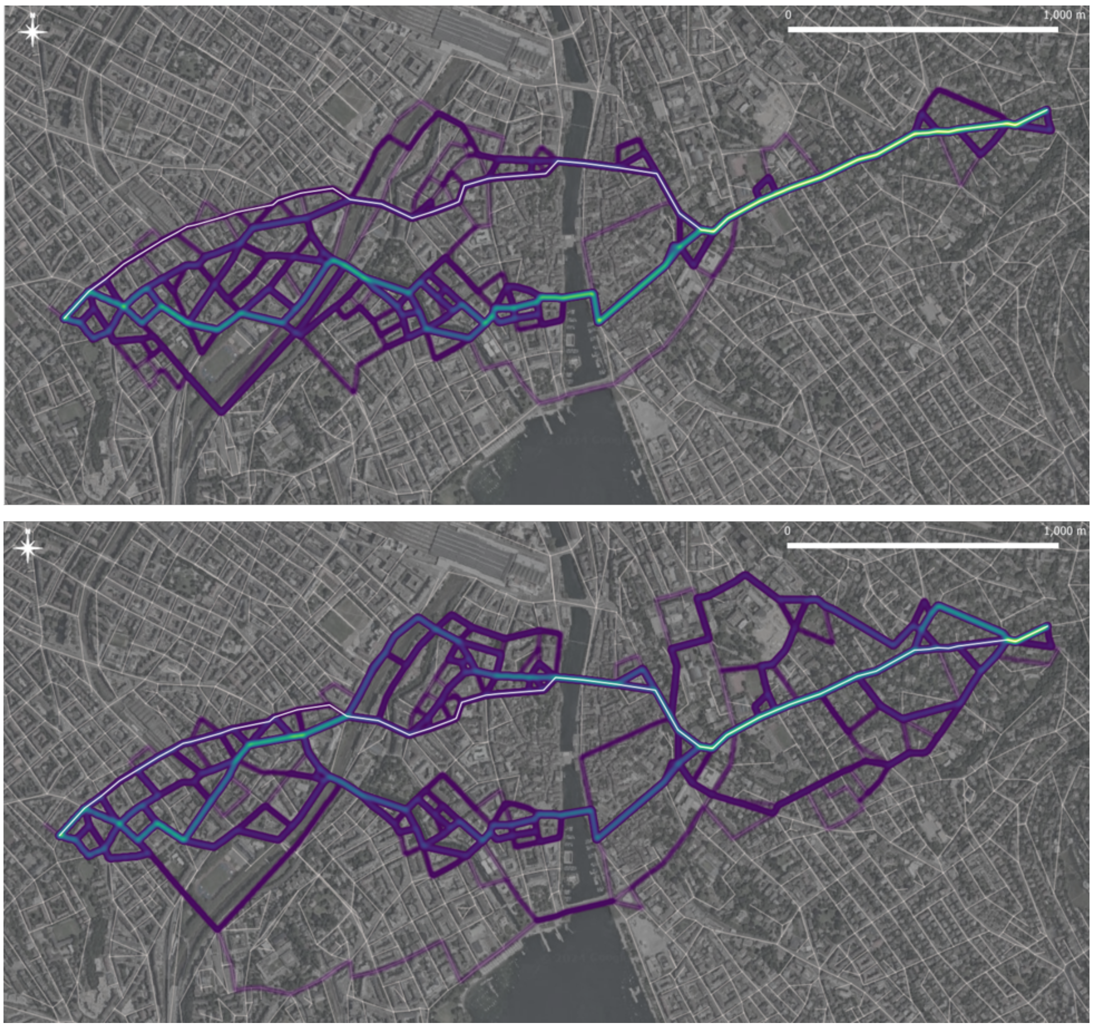

FIGURE 3 Two sets of 100 routes with different utility function specifications for the same OD pair. The top figure shows the results only using length; the bottom one shows results using the full set of available parameters.

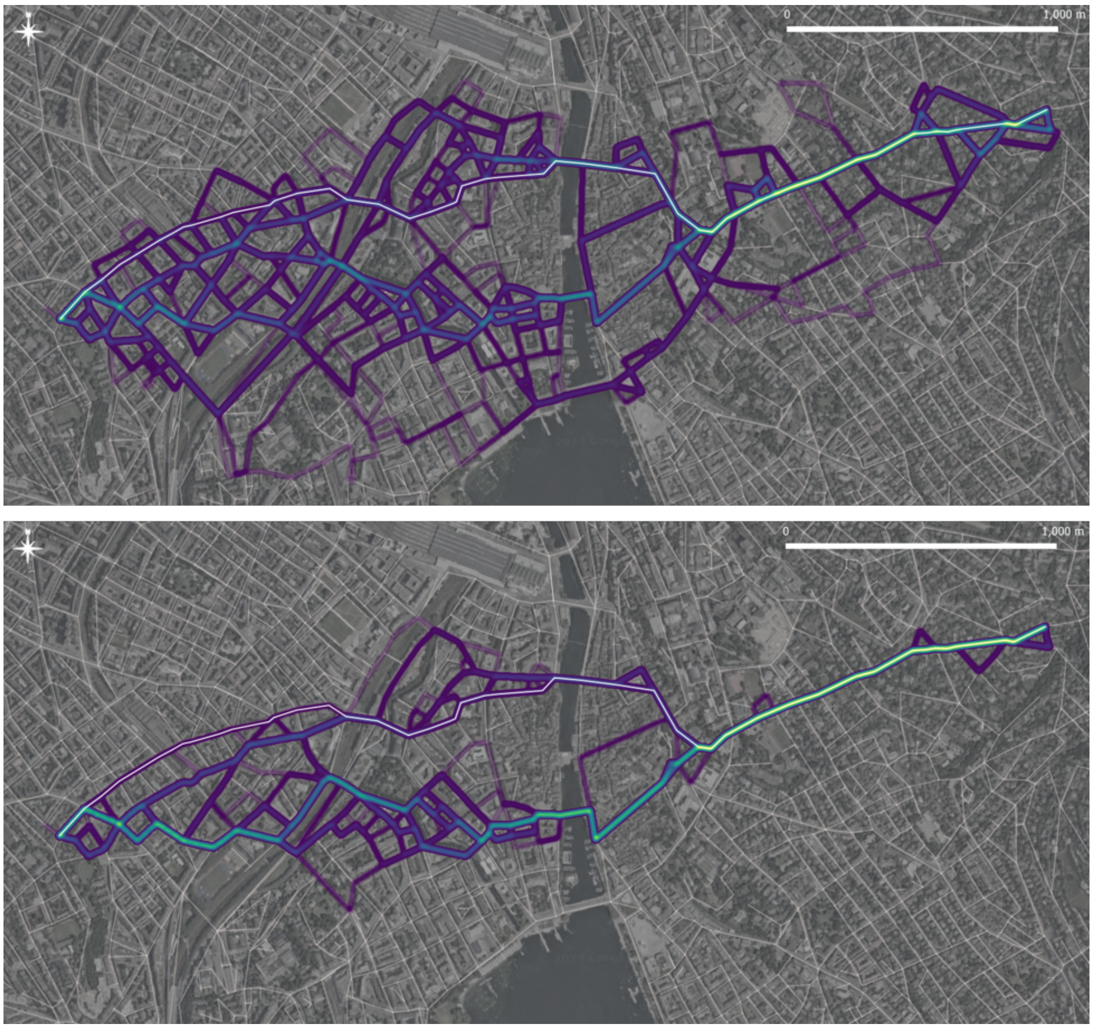

FIGURE 4 Two sets of 100 routes for the same OD pair with different random error scale parameter. Bottom using $\mu = \frac{2}{3}$, top using $\mu = \frac{3}{2}$.

TABLE 3 Comparison of route characteristics for the SP, SSP, and RL router. For both latter, we show average values based on 100 routes for each OD pair.

	SP	SSP	RL
links, med. [n]	79	83	86
length, med. [km]	10.3	10.9	10.7
length, std. [km]	4.9	5.0	5.1
length, 25th/75th [km]	7.0/13.7	6.3/14.5	6.9/14.2
bike path, avg. [%]	3.8	3.2	3.5
bike lane, avg. [%]	24.9	21.6	21.9
speed 30kmh, avg. [%]	24.9	24.8	26.8
slope 2-6%, avg. [%]	13.7	13.7	13.8
slope 6-10%, avg. [%]	5.2	5.4	5.0
slope >10%, avg. [%]	1.4	1.8	1.7
AADT >10k, avg. [%]	4.3	4.4	3.4
SP overlap, avg. [%]	-	73.5	74.7
overlap, avg. [%]	-	66.1	78.4

Cycling counts comparison

As introduced in Section 3.3, we use a Zurich scenario to assess to what extent the different routing strategies succeed in reproducing real observed cycling counts. Count observations are available for the city of Zurich (not the metropolitan area) from 26 counters, typically loop detectors, recorded at 15-minute intervals throughout the year. The count observations for 2023 are temporally aggregated into an average day. They are further spatially aggregated given the local network representation (e.g., at consolidated intersections or merged parallel edges). This results in 17 count stations that are used for the evaluation, see Figure 5. The demand is derived from the relaxed scenario. For this study, we consider it to be static, i.e., we do not model any route-dependency of the mode-choice decisions (the currently available mode-choice model does not consider route characteristics for the cycling mode). The scenario covers an average day that includes around 200,000 cycling trips. The supply is considered static as well, specifically we use average annual daily traffic as variable for the RL, not the actual routed car traffic from the simulation. The average computation time per OD pair is around 10ms, as these are substantially shorter than the random OD pairs used for evaluating the computational efficiency in Section 4.1.

Table 4 shows the difference between simulated and observed counts of the different routing methods. For the sake of simplicity, we do not consider intra-day or directional dependencies. For the RL/SSP, we generate 100 routes for each OD pair and weight them accordingly. The sum of observed counts for the average day add up to around 31,500 observations. The RL generates around 19,000 simulated counts, while the SP-based routers only generate around 10,000 simulated counts. Assuming that the synthetic demand is the true demand, the difference in the total sum of counts can be attributed to the actual routing, which in turn would indicate that the RL is better at reproducing observed flows. When looking at the spatial distribution of these counts, one can see that the SP-based routing under-estimates the counts for all but station 9 and fails to route any trips through station 16. For most of the stations, the DE-weighted cost function produces better results than just using length as a parameter. The stochasticity in the SP routers does seem to bring a clear advantage over their deterministic counterparts. The absolute differences between the

FIGURE 5 Spatial distribution of available count stations. The legend represents the range of observed counts for an average day.

different SP-based routers are comparably small. The results do not allow us to make a general statement as to which of the SP-based routers is actually superior in reproducing real counts. The RL generates considerably different flows. It generally also underestimates the counts; however, it slightly overestimates them for five stations. One can see that it outperforms the SP-based routing for all but count station 10, and the relative differences are quite substantial for stations 2, 3, 4, 7, 8, 11, 13, 14, 15, and 17. From these, stations 2, 11, 13 and 17 represent important bottlenecks in the Zurich network (around the lake and across rivers or the main rail aisle).

CONCLUSION

This paper presents the integration of an explicit discrete route choice model into the agent-based framework MATSim. To the best of the authors' knowledge, this has, to date, not been presented

TABLE 4 Observed and simulated counts for the three routing strategies. For the RL and SSP, we show average values based on 100 routes for each OD pair.

count station	observed	SP	SSP	$SP_{weighted}$	$SSP_{weighted}$	RL
1	3,879	2,009	1,973	1,610	1,760	2,384
2	1,542	160	236	148	204	419
3	2,418	680	502	731	590	1,145
4	749	61	46	0	16	158
5	1,527	121	178	326	319	427
6	1,557	124	64	174	123	208
7	1,076	99	146	94	124	694
8	871	439	421	598	644	1,078
9	1,437	1,485	1,471	1,592	1,577	1,592
10	1,109	263	222	349	285	291
11	326	12	27	11	42	347
12	652	760	609	665	665	515
13	2,059	404	421	253	191	2,291
14	1,921	382	434	508	487	759
15	1,622	966	814	1,012	926	1,931
16	1,256	0	0	0	0	38
17	7,702	2,808	2,726	2,763	2,753	4,597
sum	31,710	10,773	10,290	10,834	10,706	18,874

for any of the comparable simulation frameworks. Using explicit discrete route choice models in large-scale micro-/mesoscopic agent-based simulators enables modelers to properly represent heterogeneity in route choices, i.e., to study dynamic and behaviorally consistent stochastic user equilibria. The paper provides technical details of the route choice model, its integration within the MATSim environment, and the developed procedures required to operationalize it (i.e., an adapted BFS algorithm to calculate the downstream utilities across a network). It further evaluates its computational efficiency (being the major requirements from a user perspective) and parameter sensitivity.

The integration is demonstrated using a case study for the city of Zurich. For this case study, the paper presents the route choice model estimation using around 36,000 map-matched GPS trajectories from the 2023 EBIS study in Zurich (*27*), as well as the model validation on a hold-out sample, compared to (stochastic) shortest-path (SP) routing methods. The estimated models are then applied to a synthetic demand from the existing Zurich MATSim scenario and evaluated against local count data from 2023, again compared to SP routers. Doing so, this paper ultimately answers the question of whether the efforts related to integrating and using an explicit discrete route choice model are justified, i.e., provide better results. Our results show that the RL indeed provides better results. Independent of the integration into the MATSim environment, the out-of-sample validation shows that the RL is better at reproducing observed routes than the currently available SP routers. The same holds true when applying the model to the synthetic MATSim demand, where the RL generates considerably different network flows that better align with real count observations. Specifically, the RL generates flows in certain network areas, which the SP routers fail to do completely.

However, using the RL comes with a trade-off w.r.t. computational efficiency. The (current) average computation time for random OD pairs on the scenario network is about 10-15 times larger than for the A* Landmarks algorithm. For demonstration purposes, we implemented a stochastic SP router that mimics the heterogeneity in generated routes, for which the additional computational burden is negligible. While we could show that such a stochastic router can generate the same distribution of aggregated route attributes as those coming from the RL, the application onto the synthetic demand showed that the resulting flows are still very similar to the deterministic SP router, as are the shortfalls w.r.t. the count data evaluation. However, the authors still consider this a valuable analysis, as such a router might still be more useful for certain modelling use cases. Finally, the RL is fully probabilistic, meaning that unrealistic infinite looping behavior is possible per design (even though this was not a concern for our case study). The probabilistic nature of the router also comes with additional memory considerations, as it is potentially desirable to reproduce the same routes for certain conditions, which in turn requires memorizing seeds used throughout the iterations.

The main limitation of this paper lies in the synthetic demand used for evaluating the count data. The extent to which the synthetic demand actually matches the real demand can be questioned. The development of the scenario did not specifically focus on cycling mode shares, and those have substantially changed in Zurich since the scenario development. Furthermore, the efforts related to the integration and the obtained computational performances are heavily tailored to the MATSim environment. Other comparable simulation frameworks might be more or less suited for such a model integration, and the learnings might only partly be transferable. Generally, this paper should be considered as providing the basis and justification for future research efforts. Specifically, those should include:

- **Synthetic demand:** As just mentioned, it would be desirable to re-evaluate the simulated counts using an updated demand model. Such a model should specifically consider the latest behavioral changes observed for cycling (a mode-calibrated mode choice model that considers cycling-relevant attributes). The related work is currently conducted within our research group.
- **Full scenario optimization:** We used the relaxed demand from the existing Zurich MATSim scenario. Given an updated demand model, it would be desirable to run a complete MATSim optimization run that includes all available modes. This would allow us to analyze dynamic effects on the route choice and examine their respective convergence behavior.
- **Computational performance:** The RL's computational burden mostly relates to calculating the downstream utilities using the adapted BFS. This procedure can potentially be made more efficient, e.g., by using efficient search-space restriction methods (analog to A* Landmarks) or parallelizing the procedure.

AUTHOR CONTRIBUTIONS

The authors confirm contribution to the paper as follows: study design: A. Meister, M. Balac; data collection and preparation: A. Meister; study implementation and result generation: A. Meister, Z. Liang; interpretation of results: A. Meister, Z. Liang, M. Balac; draft manuscript preparation: A. Meister, M. Balac. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Lu, W., D. M. Scott, and R. Dalumpines, Understanding bike share cyclist route choice using GPS data: Comparing dominant routes and shortest paths. *Journal of transport geography*, Vol. 71, 2018, pp. 172–181.
2. Hess, J., A. Meister, V. R. Melnikov, and K. W. Axhausen, Geographic Information System-Based Model of Outdoor Thermal Comfort: Case Study for Zurich. *Transportation Research Record*, Vol. 2677, No. 3, 2023, pp. 1465–1480.
3. Łukawska, M., Quantitative modelling of cyclists' route choice behaviour on utilitarian trips based on GPS data: associated factors and behavioural implications. *Transport Reviews*, 2024, pp. 1–32.
4. Nagel, K., B. Kickhöfer, and J. W. Joubert, Heterogeneous tolls and values of time in multi-agent transport simulation. *Procedia Computer Science*, Vol. 32, 2014, pp. 762–768.
5. Ziemke, D., S. Metzler, and K. Nagel, Bicycle traffic and its interaction with motorized traffic in an agent-based transport simulation framework. *Future Generation Computer Systems*, Vol. 97, 2019, pp. 30–40.
6. Horni, A., K. Nagel, and K. W. Axhausen, *The multi-agent transport simulation MATSim*. Ubiquity Press, London, 2016.
7. Lopez, P. A., M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, Microscopic traffic simulation using SUMO. In *2018 21st International Conference on Intelligent Transportation Systems*, IEEE, 2018, pp. 2575–2582.
8. Auld, J., M. Hope, H. Ley, V. Sokolov, B. Xu, and K. Zhang, POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations. *Transportation Research Part C: Emerging Technologies*, Vol. 64, 2016, pp. 101–116.
9. De Palma, A., F. Marchal, and Y. Nesterov, METROPOLIS: Modular system for dynamic traffic simulation. *Transportation Research Record*, Vol. 1607, No. 1, 1997, pp. 178–184.
10. Hörl, S., M. Balac, and K. W. Axhausen, A first look at bridging discrete choice modeling and agent-based microsimulation in MATSim. *Procedia computer science*, Vol. 130, 2018, pp. 900–907.
11. Ton, D., D. Duives, O. Cats, and S. Hoogendoorn, Evaluating a data-driven approach for choice set identification using GPS bicycle route choice data from Amsterdam. *Travel behaviour and society*, Vol. 13, 2018, pp. 105–117.
12. Ben-Akiva, M., M. S. Ramming, and S. Bekhor, Route choice models. In *Human Behaviour and Traffic Networks* (M. Schreckenberg, ed.), Springer, 2004, pp. 23–45.
13. Cascetta, E., A. Nuzzolo, F. Russo, and A. Vitetta, A modified logit route choice model overcoming path overlapping problems. Specification and some calibration results for interurban networks. In *Transportation and Traffic Theory. Proceedings of The 13th International Symposium On Transportation And Traffic Theory*, 1996.
14. Fosgerau, M., E. Frejinger, and A. Karlstrom, A link based network route choice model with unrestricted choice set. *Transportation Research Part B: Methodological*, Vol. 56, 2013, pp. 70–80.
15. Mai, T., M. Fosgerau, and E. Frejinger, A nested recursive logit model for route choice analysis. *Transportation Research Part B: Methodological*, Vol. 75, 2015, pp. 100–112.

16. Nguyen, J., S. T. Powers, N. Urquhart, T. Farrenkopf, and M. Guckert, An overview of agent-based traffic simulators. *Transportation research interdisciplinary perspectives*, Vol. 12, 2021, p. 100486.
17. Meister, A., Z. Liang, M. Felder, and K. W. Axhausen, Comparative study of route choice models for cyclists. *Journal of Cycling and Micromobility Research*, 2024, p. 100018.
18. Bellman, R., A markovian decision process. *Indiana University Mathematics Journal*, Vol. 6, No. 4, 1957, pp. 679–684.
19. de Freitas, L. M., H. Becker, M. Zimmermann, and K. W. Axhausen, Modelling intermodal travel in Switzerland: A recursive logit approach. *Transportation Research Part A: Policy and Practice*, Vol. 119, 2019, pp. 200–213.
20. Gao, Y. and J.-D. Schmöcker, Estimation of walking patterns in a touristic area with Wi-Fi packet sensors. *Transportation research part C: emerging technologies*, Vol. 128, 2021, p. 103219.
21. Dial, R. B., A probabilistic multipath traffic assignment model which obviates path enumeration. *Transportation research*, Vol. 5, No. 2, 1971, pp. 83–111.
22. Ballo, L. and K. W. Axhausen, Modeling sustainable mobility futures using an automated process of road spac reallocation in urban street networks: A case study in Zurich. *Arbeitsberichte Verkehrs-und Raumplanung*, Vol. 1835, 2023.
23. ARE, *Nationales Personenverkehrsmodell NPVM 2017*. Bundesamt fuer Raumentwicklung, Bern, 2017.
24. Hörl, S. and M. Balac, Synthetic population and travel demand for Paris and Île-de-France based on open and publicly available data. *Transportation Research Part C: Emerging Technologies*, Vol. 130, 2021, p. 103291.
25. Tchervenkov, C., G. O. Kagho, A. Sallard, S. Hörl, M. Balać, and K. W. Axhausen, The Switzerland agent-based scenario. *Arbeitsberichte Verkehrs-und Raumplanung*, Vol. 1802, 2022.
26. BFS, *Mikrozensus Mobilitaet und Verkehr*. Bundesamt fuer Statistik (BFS), Neuchatel, 2023.
27. Heinonen, S., A. Meister, L. Meyer de Freitas, L. Schwab, J. Roth, T. Götschi, B. Hintermann, and K. W. Axhausen, The E-biking in Switzerland (EBIS) study: Methods and dataset. In *102nd Annual Meeting of the Transportation Research Board*, IVT, ETH Zürich, 2023.
28. Meister, A., M. Felder, B. Schmid, and K. W. Axhausen, Route choice modeling for cyclists on urban networks. *Transportation Research Part A*, Vol. 173, 2023, p. 103723.