Ref. Ares(2026)8303464 - 31/08/2026

INTEGRATED MULTIMODAL SIMULATION AND PREDICTION FRAMEWORK

Project deliverable D5.1

HORIZON Research and Innovation Action | 101203465 |
Call HORIZON-CL5-2023-D6-01

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

DELIVERABLE ADMINISTRATIVE INFORMATION

TITLE OF THE DELIVERABLE	D5.1 - Integrated Multimodal Simulation and Prediction Framework
WORK PACKAGE	WP5
TYPE OF DELIVERABLE	R — Document, report
DISSEMINATION LEVEL	PU - Public
STATUS – VERSION, DATE	Final – V3.0, 25/08/2026
DELIVERABLE LEADER	NTUA
CONTRACTUAL DATE OF DELIVERY	31/08/2026
SUBMISSION DATE	31/08/2026
KEYWORDS	Lightweight multimodal simulation, Large-scale traffic modelling, Macroscopic / mesoscopic traffic simulation, Activity-based travel demand modelling, Agent-based simulation, Emerging mobility services, Mobility-as-a-Service, Mobility controllers

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

LIST OF CONTRIBUTORS

Name	Organisation
Emmanouil Kampitakis	NTUA
Eleni Vlahogianni	NTUA
Aditya Paul	DTU
Shadi Haj-Yahia	DTU
Carlos Lima Azevedo	DTU

QUALITY CONTROL

	Name	Organisation	Date
Peer review	Kevin Riehl, Julius Schlapbach	ETH	13.08.2026
Peer review	Eider Irigoyen, Harbil Arregui	VICOM	18.08.2026
Quality review	Roozbeh Mohammadi	ERTICO	31.08.2026

VERSION HISTORY

Version	Date	Author	Summary of changes
V1.0	28/05/2026	NTUA	First draft
V2.0	11/08/2026	NTUA, DTU	Consolidated draft circulated for internal peer review
V3.0	25/08/2026	NTUA, DTU	Revised after ETH and VICOM peer review

LEGAL DISCLAIMER

Funded by the European Union's Horizon Europe Research and Innovation Actions under grant agreement No. 101203465. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them. The information in this document is provided "as is", and no guarantee or warranty is given that

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

TABLE OF CONTENTS

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

LIST OF FIGURES

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

LIST OF TABLES

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

LIST OF ABBREVIATIONS AND ACRONYMS

Acronym	Meaning
AI	Artificial Intelligence
DMSF	Density-Modulated Saturation Flow
EMA	Exponential Moving Average
MFD	Macroscopic Fundamental Diagram
OD	Origin-Destination
PCE	Passenger-Car Equivalent
S&F	Store-and-Forward
SUMO	Simulation of Urban Mobility
AMoD	Automated Mobility on Demand
MaaS	Mobility as a Service
EV	Electric Vehicle
DAS	Day Activity Schedule
PT	Public Transport
AM	Morning (peak period)
PM	Evening (peak period)
MSP	Mobility Service Provider
MT	Mid-Term (module of SimMobility)
MoD	Mobility on Demand
TAZ	Traffic Analysis Zone
PCU	Passenger Car Unit
MRT	Mass Rapid Transit
TCS	Tradable Credit Scheme

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

DELIVERABLE EXECUTIVE SUMMARY

Deliverable D5.1 "Integrated Multimodal Simulation and Prediction Framework" reports the first implementation of the FEDORA Multi-modal Simulation Space, the environment in which the project's network management concepts are modelled and assessed at network-wide scale. The deliverable consolidates the progress of the first three tasks of WP5, i.e., the multimodal simulation engines integrated framework (T5.1), the agent-based models for complex user behaviour simulation (T5.2), and the simulation controllers for emerging multimodal transport services (T5.3), delivering all together a network-supply simulation engine, a demand-side behavioural representation, and a service and policy control layer that underpin the further development and demonstration of the FEDORA solutions.

Under Task 5.1, a lightweight Python-based multimodal macroscopic traffic simulator has been designed and implemented. It combines a horizontal-queue Store-and-Forward core with a multi-class passenger-car-equivalent overlay, dynamic turn-ratio routing, several discharge-side extensions, a trip-age completion model, and an explicit representation of car, bus and truck classes with bus stops, dispatch and route following. The simulator was compared with a SUMO microsimulation of a realistic sub-network of the Athens city centre (1152 directed links, 561 intersections, 44137 planned trips, morning peak). The macroscopic simulator reproduces peak accumulation to within 5.3%, peak production to within 3%, time-to-peak accumulation to within 1.5 minutes and final trip completions to within about 100 vehicles out of approximately 44000. Moreover, the shape of the Macroscopic Fundamental Diagram (MFD) across the whole accumulation range is successfully reproduced, while it runs approximately three orders of magnitude faster than the microscopic reference. The combination of its speed and aggregate fidelity allows for its use as a training and transfer environment for AI-assisted control strategies under Task 5.5.

Under Task 5.2, the demand-side modelling framework for the Greater Copenhagen Area has been developed on the basis of the open-source, agent- and activity-based SimMobility platform. A complete day activity schedule is generated by the Mid-term Pre-day module for every agent from a hierarchy of day-pattern, tour and intermediate-stop choices. The generated schedules are executed and adapted by the Within-day module through an event-driven chain of pre-trip and en-route behaviour models. The entire demand chain has been tested in SimMobility's controlled Virtual City environment. In this test environment, the Pre-day module generates schedules for a synthetic population of approximately 165000 travellers, 467000 tours and 1.17 million trips on a list of twelve modes including shared and demand-responsive services. The model pipeline and its calibration for Copenhagen will be carried out in the next phase.

Under Task 5.3, a modular controller suite has been developed which has been integrated into SimMobility across three functional pillars, more precisely, service supply, demand management and market regulation. This suite includes a generic Mobility Service Provider controller covering driver lifecycle, trip matching and fleet rebalancing for Mobility on Demand (MoD)/ Automated Mobility on Demand (AMoD) fleets, dedicated fixed-route Bus and Train controllers, a MaaS controller integrating agent choice sets with operators, and a Pricing and Travel Credits controller implementing a regulator, a market platform, dynamic tariffs and tradable credit schemes. The design, implementation and documentation have been completed for all controllers. Further debugging and testing in the Virtual City and deployment on the Copenhagen network are active and remain to be completed for several of them.

The objective of this deliverable is to document the methodology, implementation status and preliminary evaluation of the above-mentioned advancements, as well as to provide the description that would act as a reference for partners developing applications that will be applied in the simulation space, to evidence the compliance of our achievements with the descriptions provided in the Grant Agreement regarding the deliverables T5.1-T5.3 and the Expected Innovations EI4.1-EI4.3, and also lay down the technological baseline for the subsequent deliverable D5.3 as this is an initial version and all the components remain under active development. At present, the implementation of all three components of the FEDORA Simulation Space that are included in this deliverable has progressed considerably and are functioning as expected in a controlled environment. Dynamics at the network level are satisfactorily reproduced at a much lower computation cost than what a well-tested microscopic simulator needs. Disaggregate

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

behavioural demand can be generated for a full synthetic population, and the principal supply-side and policy controllers are implemented and partly tested.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

1 INTRODUCTION

The objective of this deliverable is to present the progress achieved in the development of the first three tasks of Work Package 5 (WP5): Evolutionary Simulation Space for Network Wide Foresight Analysis. Specifically, it reports the work carried out on the development of a lightweight multimodal traffic simulation engine (Task 5.1), an agent-based user behaviour simulation framework (Task 5.2), and a suite of simulation controllers for emerging multimodal transport services (Task 5.3). These developments constitute the initial implementation of the simulation technologies envisaged in WP5.

The developments presented in this deliverable represent the current progress and contribute to the overall objectives of FEDORA by advancing the project's Multi-modal Simulation Space which provides the simulation environment needed to model multimodal transport systems, traveller behaviour and emerging mobility services. In order to achieve these objectives, several simulation components have been developed in the three tasks that are included in this deliverable. Task 5.1 presents a lightweight multimodal macroscopic traffic simulator for efficient large-scale traffic simulation. Task 5.2 develops an open and flexible agent-based framework, based on SimMobility, for modelling heterogeneous traveller behaviour, activity generation and behaviour adaptation. Task 5.3 develops a set of generic controllers for emerging multimodal transport services, such as Mobility Service Providers (MSPs), Mobility as a Service (MaaS), public transport as well as pricing and incentive mechanisms.

1.1 MAPPING FEDORA OUTPUTS

FEDORA GA COMPONENT TITLE	FEDORA GA COMPONENT OUTLINE	RESPECTIVE DOCUMENT CHAPTER(S)	JUSTIFICATION
DELIVERABLE			
D5.1 Integrated Multimodal Simulation and Prediction Framework	*Simulation framework that combines multimodal simulation engines, agent-based demand prediction, and simulation controllers (T5.1-T5.3)*	Chapter 2-5	Chapter 2 covers the multimodal simulation engines integrated framework (T5.1); Chapter 3 covers the agent-based models for complex user behaviour simulation (T5.2); Chapter 4 covers the simulation controllers for emerging multimodal transport services (T5.3); Chapter 5 consolidates the research and scientific innovation
TASKS			
T5.1 Multimodal simulation engines integrated framework	Development of a framework with tools for lightweight, efficient and easy-to-calibrate simulation engines to support simulation-to-reality tasks for	Chapter 2	Presents the design and implementation of the lightweight multimodal simulation framework, including its architecture,

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

	advanced AI-assisted multimodal traffic management strategies.		modelling components, and validation, supporting AI-assisted simulation-to-reality applications.
T5.2 Agent-based models for complex user behaviour simulation	Development of a flexible and open-source behaviour simulation framework with capabilities to model demand shifts from a wide range of mobility modes and operations and innovative configurations under network wide systems in a complex multimodal urban environment and for heterogenous users.	Chapter 3	Describes the development and initial validation of the open-source agent-based user behaviour simulation framework based on SimMobility, including demand modelling, Within-day behavioural adaptation, and integration with multimodal mobility management strategies.
T5.3 Simulation controllers for emerging multimodal transport services	Generation of a set of generic controllers that allow FEDORA to validate and demonstrate the potential to enhance the overall performance of the transport, reduce and the overall transport energy usage.	Chapter 4	Addresses the design and implementation of mobility controllers for multimodal transport services, including MSP, MaaS, public transport, pricing, and travel credits, together with their integration and preliminary validation within SimMobility.

Table 1. Adherence to FEDORA GA Deliverable & Tasks Descriptions

1.2 DELIVERABLE OVERVIEW AND REPORT STRUCTURE

The developments reported in this deliverable constitute the first implementation of the FEDORA Multi-modal Simulation Space and provide the basis for the subsequent WP5 activities. This deliverable is organised into six chapters. Following this introductory chapter, the remaining chapters present the technical developments achieved within the first three tasks of WP5, discuss their scientific contribution, and summarise the main conclusions.

Chapter 2 presents the work that is conducted in Task 5.1 where the design, implementation, and validation of the lightweight multimodal macroscopic traffic simulator developed within FEDORA is described. The chapter introduces the simulator architecture, the initially adopted modelling approaches, the implemented multimodal functionalities, and the validation studies to demonstrate its capability to efficiently simulate large-scale transport networks.

Chapter 3 describes the developments of Task 5.2, focusing on the agent-based user behaviour simulation framework. The chapter presents the adopted SimMobility-based modelling framework, the representation of heterogeneous traveller behaviour, activity generation, behavioural adaptation mechanisms, and the current implementation status of the developed models.

Chapter 4 presents the work performed in Task 5.3 where the simulation controllers developed for emerging multimodal transport services are introduced and the proposed controller architecture is described together with the implemented Mobility Service

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

Provider (MSP), Mobility as a Service (MaaS), public transport, pricing and incentive mechanisms, illustrating their integration within the simulation framework.

Chapter 5 discusses the research and scientific innovations of the developments that are presented in this deliverable, highlighting their contribution to the advancement of multimodal macroscopic simulation and agent-based simulation and their relevance within the overall FEDORA framework.

Finally, Chapter 6 summarises the main achievements of the reported work and outlines the next steps for the continued development of the WP5 simulation space.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

2 MULTIMODAL SIMULATION ENGINES INTEGRATED FRAMEWORK

2.1 TASK DESCRIPTION

The objective of this task is the development of a framework with tools for lightweight, efficient and easy-to-calibrate simulation engines to support simulation-to-reality tasks for advanced AI-assisted multimodal traffic management strategies. The primary focus is on delivering a multimodal macroscopic simulator which is fast and whose objective is to reproduce the network-level dynamics that emerge from a microscopic simulation. This will provide a basis for investigating how AI systems can be trained in simulation and for improving our understanding of the requirements and challenges associated with their transfer to real-world applications, for example, learning-based perimeter control (Task 5.5). In many modern operational analyses of urban traffic networks, high-fidelity microscopic simulators are utilised such as SUMO, AIMSUN, VISSIM etc. These types of simulators follow a bottom-up approach meaning that each vehicle is modelled individually using various car-following and lane-changing models from which the macroscopic network dynamics emerge, i.e., the Macroscopic Fundamental Diagram (MFD). One drawback is the computational cost which restricts their use inside optimisation loops, model predictive controllers and reinforcement-learning environments. On the other hand, macroscopic simulators can model traffic at the link-level and compute aggregated quantities at the link-scale, e.g., vehicle counts, mean speeds, mean densities etc. A comparison of these two modelling approaches shows that macroscopic simulators trade off resolution for speed, thus it is challenging to design a macroscopic model whose emergent behaviour, in terms of the MFD shape, the timing of the onloading and offloading branches, the spatial concentration of congestion and the overall trip completion trajectory, is faithful to its microscopic reference.

In this chapter, we describe our attempt to create a Python-based multimodal macroscopic traffic simulator, which we call simpi, that reproduces the network-level dynamics that emerge from a microscopic simulation (i.e., SUMO) at approximately three orders of magnitude faster, as demonstrated on a realistic urban traffic network. The basic theoretical model is based on three published works:

- Aboudolas et al. (2009) where they provide the base Store-and-Forward (S&F) network model with per-link queue dynamics.
- Tsitsokas et al. (2021) where they extend the S&F model (Aboudolas et al., 2009) with a horizontal-queue formulation (finite link storage capacity that spills back into upstream links), and they embed the single-class S-model moving/waiting decomposition of Lin et al. (2011, 2012).
- Portilla et al. (2020) where the multi-class extension of the classical macroscopic S-model (Lin et al., 2011, 2012) is introduced. The contribution of this work is a per-class passenger-car-equivalent (PCE) formulation that distinguishes between two kinematic regimes on every link (i.e., a free-flow pool and a queue) and derives a class-specific PCE in each regime, together with a corridor-delay formulation of the transition from free flow to queue.
- Tsitsokas et al. (2023) where a periodic dynamic turn-ratio update framework is introduced in which each routing tick recomputes per-origin shortest paths on a link graph whose edge costs are derived from window-averaged Edie speeds. This dynamic update replaces the static turn-ratio matrix of the original S&F formulation with a slow-time-scale rerouting mechanism and is the basis of the routing module that is used in simpi.

To be more precise, our work combines three ingredients from these works, i.e., the multi-class overlay of Portilla et al. (2020), the Store-and-Forward network embedding proposed by Aboudolas et al. (2009) and extended by Tsitsokas et al. (2021) and the dynamic turn-ratio update in Tsitsokas et al. (2023). Moreover, on top of this baseline several extensions are added that were found necessary to close the gap between simpi and the microscopic simulator.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

This chapter is organised as follows. Section 2.2 presents the methodology: the horizontal-queue Store-and-Forward core with the multi-class PCE overlay, the multimodal representation, the discharge-side extensions, the trip-age completion model and the dynamic routing module. Section 2.3 reports preliminary results on a realistic sub-network of the Athens city centre, where simpi is compared against a SUMO microsimulation, and Section 2.4 concludes with the ongoing developments.

2.2 METHODOLOGY

2.2.1 HORIZONTAL-QUEUE STORE-AND-FORWARD CORE WITH MULTI-CLASS PCE OVERLAY

The proposed simulation core combines horizontal-queue Store-and-Forward dynamics with a multi-class traffic representation through passenger-car-equivalent (PCE) factors. It provides a computationally efficient representation of link-level traffic dynamics while accounting for heterogeneous vehicle classes. The following sections describe its main modelling components and associated formulations.

2.2.1.1 BIMODAL LINK STATE

The per-link state is divided into a moving pool state and a discharge-side waiting queue state. Portilla et al. (2020) define a per-class, per-outgoing-movement state layout $n^{free}_{u,d,c}, n^{queue}_{u,d,o,c}$ within their cycle-averaged multi-class S-model. Our contribution here is to combine this with the fine-step horizontal-queue dynamics of Tsitsokas et al. (2021, 2023).

Let $G = (\mathcal{L}, \mathcal{J})$ denote a directed graph whose vertices $\mathcal{J}$ are intersections and whose edges $\mathcal{L}$ are directed road links. For each link $i \in \mathcal{L}$ and vehicle class $c \in \mathcal{C}$, we track two compartments. A *free pool* that carries $n_i^f(c)$ vehicles of class c moving at the current free-flow speed $v_i^f(c)$ (different for each class) and a *queue* which is structured into one slot per outgoing movement $k \in K(i)$. Slot k contains $n_{i,k}^q(c)$ stationary vehicles of class c targeting the corresponding downstream link and is thus a tensor over links, slots and classes while the free pool is a matrix over links and classes. Note that the raw per-class vehicle counts are stored, and the passenger-car-equivalent conversion is applied on demand in downstream computations via class-specific PCE factors that will be defined later in this chapter.

It is now worth explaining in more detail the slot structure inside the queue and make an important distinction. Each queued vehicle belongs to exactly one slot $k \in K(i)$, where $K(i)$ is the set of outgoing movements from link i. In that sense, the slot to which the vehicle is assigned is determined by the downstream link j_k the vehicle intends to enter next. Slots have also a physical counterpart, i.e., vehicles in slot k queue in the lane(s) assigned to that movement. Usually real-world lane assignments are *many-to-many* meaning that a single physical lane can serve multiple movements (e.g., a shared-through lane on which cars can go straight or turn right), and a single movement can use multiple lanes (e.g., a through movement using all lanes of a wide road). Simpi encodes this mapping with two arrays:

- $\mathrm{state.slot\_lane\_share}[l, k]$: fraction of link l's total lane count that movement k receives (e.g., 1/4 for a left-turn pocket on a 4-lane approach).
- $\mathrm{state.lane\_to\_slots}[l, \ell, k]$: whether physical lane ℓ of link l serves movement k where non-zero entries define the many-to-many mapping.

These tables serve as the saturation-flow allocations per lane and when the discharge rule discharges movement k, it uses the lane capacity assigned to slot k. Note also that the slot partition is orthogonal to the vehicle-class partition where class c indexes what type of vehicle it is (car, bus, truck etc.) and slot k indexes which destination it will go to next. A car and a bus that are in the same slot discharge together from the same queue cohort when their corresponding movement receives green and only the sub-

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

population is class-decomposed. This is important for the discharge rule shown later, where we process one movement at a time and recover the per-class outflow from the per-class shares in each slot.

2.2.1.2 PASSENGER-CAR EQUIVALENT PER REGIME

The PCE of a class differs between the free and queued regimes (Portilla et al., 2020). In the queue, only the gross stopping distance (physical length plus minimum stopped gap between vehicles) matters:

$$pce_{queue}(c) = \frac{\ell_c}{\ell_{ref}} \tag{1}$$

where ℓ_c is the gross stopping distance of class c and ℓ_{ref} is that of a reference passenger car. In the free regime, the PCE additionally accounts for the headway-induced spacing at the class's free-flow speed as follows:

$$pce_{free}(c) = \frac{\ell_c + h_c v_c^{free}}{\ell_{ref} + h_{ref} v_{ref}^{free}} \tag{2}$$

where h_c is the reaction-time-based time-headway of class c and h_{ref} is that of a reference passenger car. The PCEs of different vehicle classes differ in a non-trivial way when they have different gross stopping distances, free-flow speeds, or headways.

2.2.1.3 STORAGE CAPACITY AND FREE-TO-QUEUE TRANSITION

The storage capacity of link i in PCE units is given by:

$$C_i^{PCE} = \frac{L_i \cdot N_i^{lanes}}{\ell_{ref}} \tag{3}$$

where L_i is the length of link i, N_i^{lanes} is the number of lanes of link i and ℓ_{ref} is the gross stopping distance of a reference passenger car. A free-pool vehicle progresses toward the downstream queue at the class free-flow speed over a corridor whose length shrinks as the queue grows. The corresponding time-to-queue-tail is the corridor-delay quantity proposed by Portilla et al. (2020) and Tsitsokas et al. (2021):

$$\tau_i^{ttq}(c) = \frac{\left(C_i^{PCE} - \sum_k \sum_{c'} n_{i,k}^q(c') \cdot pce_{queue}(c')\right) \cdot \ell_{ref}}{N_i^{lanes} \cdot v_c^{free}} \tag{4}$$

For modelling the free-flow regime, simpi uses a *fluid-limit reformulation* of the shared continuous quantity $\tau_i^{ttq}(c)$. Note that Portilla et al. (2020) and Tsitsokas et al. (2021) follow a different discretization logic and more precisely the former uses a two-tap fractional-delay FIR while the latter employs a ceil-quantized cumulative-inflow shift. Different from these works, simpi treats the free pool as a well-mixed reservoir that drains at rate $1/\tau_i^{ttq}(c)$:

$$\Delta n_i^{f \to q}(c) = n_i^f(c) \cdot \frac{\Delta t}{\max\left(\tau_i^{ttq}(c), \Delta t\right)} \tag{5}$$

This is a modelling *trade-off* and the motivation for this choice is vectorization and memory. More precisely, since the reservoir-drain form requires no per-link inflow-history buffer (no bookkeeping), from an implementational point of view it can be fused into a single Numba-compiled pass and captures sub-step traversals on short links which is crucial when modelling realistic traffic networks. Of course, as this is a trade-off it comes with a cost. The fluid form is memoryless with respect to when vehicles entered the link thus, we sacrifice arrival-timing cohort identity, platoon-shape preservation, exact step-to-step mass conservation, and structural non-negativity. The max operator in Eq. (5) prevents unbounded transfer when the queue reaches the most upstream part of the link, at the cost of a coarse-Δt near-jam artefact where a large fraction of the free pool can transfer in a single step. Nevertheless, if Δt is small such that $\tau_{ttq} \gg \Delta t$ it avoids this regime in practice. As the queue grows, τ_{ttq} decreases, $\Delta n_i^{f\to q}$ increases and when the link is jammed the free pool empties.

2.2.1.4 DISCHARGE DYNAMICS

The discharge follows the standard Store-and-Forward triple-min form (queue ∧ sending ∧ receiving) and the per-class PCE weighting of the sending and the receiving factors is based on Portilla et al. (2020). Note that the triple-min here is computed as a *two-pass structure* (Figure 1). More precisely, Pass 1 computes the min(queue, sending) per movement and Pass 2 aggregates the resulting desired flows at each downstream link and applies a proportional receiving-allocation cap. This was done for vectorization efficiency. The kernel operates in the two passes as follows. Pass 1 computes each movement's desired flow against a frozen snapshot of the state:

$$\Phi_{i,m\to j}^{des}(c) = \min\left(n_{i,k(m)}^{q}(c), q_c \cdot \Delta t \cdot g_{i,m}(t) \cdot \psi_{i,m} \cdot \delta_{i,m}(t) \cdot \mu_{DMSF}(i,t) \cdot \gamma_{i,m}(t) \cdot \sigma_{i,m,c}\right) \tag{6}$$

where

- q_c is the class saturation flow rate
- $g_{i,m}(t) \in \{0,1\}$ is the signal green mask
- $\psi_{i,m} \in (0,1]$ is a per-movement saturation-flow factor whose value depends on the turn directions of the movement
- $\delta_{i,m}(t)$ is a capacity drop multiplier
- $\mu_{DMSF}(i,t)$ is a density modulated saturation-flow multiplier
- $\gamma_{i,m}(t)$ is a gap-acceptance factor active on minor movements at *priority* intersections
- $\sigma_{i,m,c}$ is a queue share or per lane share

Pass 2 follows and applies the downstream receiving-space cap. The PCE-weighted desired flows for each downstream link j are aggregated over all upstream movements and all classes, and each movement's per-class flow is scaled by the (class-independent) ratio

$$a_{i,m,j} = \min\left(1, \frac{R_j}{\sum_{(i',m'):j'=j}\sum_{c'}\Phi_{i',m'\to j}^{des}(c')\cdot pce_{queue}(c')}\right), \quad \Phi_{i,m\to j}(c) = a_{i,m,j}\cdot\Phi_{i,m\to j}^{des}(c) \tag{7}$$

where

- The denominator is the total PCE demand at the downstream link (summed across all upstream movements (i', m') heading to j and all classes c'),
- $a_{i,m,j}$ is a single scalar per triple $(i, m. j)$ which is applied uniformly to every class's desired flow through that movement,
- R_j is the receiving-space capacity of the downstream link,

$$R_j = \max\left(0, C_j^{PCE} - \sum_{c'}\left(n_j^f(c')\cdot pce_{free}(c') + \sum_{k'} n_{j,k'}^{q}(c')\cdot pce_{queue}(c')\right)\right) \tag{8}$$

As one can observe, this two-pass structure implements a proportional-demand receiving-space allocation. The form is closest to that of Portilla et al. (2020) but here we use the actual per-step desired-flow shares Φ^{des} in place of nominal β-weighted terms. Moreover, the snapshot decouples the sender and receiver roles that a single link plays inside one step and guarantees that the discharge is invariant to the order in which links are processed and $\sum_{i,m}\Phi_{i,m\to j}(c) \le R_j$ is satisfied automatically.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

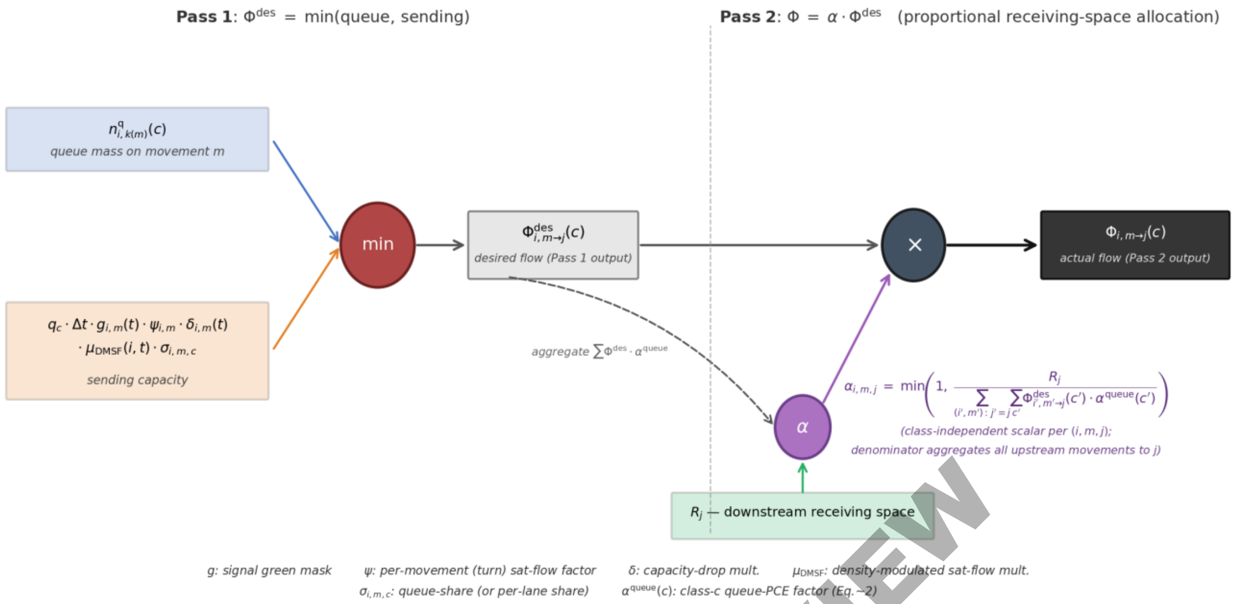


Figure 1. Store-and-Forward discharge on movement $(i, m \rightarrow j)$ for class c.

2.2.2 MULTIMODAL REPRESENTATION

The multimodal representation extends the simulation framework to explicitly account for different vehicle classes and their distinct operational characteristics. In addition to class-specific traffic dynamics, it incorporates public transport operations through bus stops, dispatching and route following, while supporting dedicated infrastructure for specific vehicle classes. The following sections describe these components in detail.

2.2.2.1 VEHICLE CLASSES

Each vehicle class has its own set of kinematic constants. That is:

- Physical length ℓ_c (the gross stopping distance).
- The queued PCE factor $pce_{queue}(c)$.
- A free flow speed factor φ_c that multiplies the link's posted free-flow speed to obtain $v_c^{free} = \varphi_c \cdot v_i^{free}$. By convention $\varphi_c \leq 1$ although any positive value can be permitted.
- A reaction-time-based headway h_c (this is consumed in pce_{free}).
- An optional per-class saturation-flow multiplier $\frac{q_c}{q_{ref}}$ applied to the class's sending capacity.

An example of three canonical vehicle class definitions is given in Table 2.

Table 2. Canonical vehicle-class definitions.

CLASS	LENGTH (M)	pce_{queue}	φ_c	$h_c\ (s)$	q_c/q_{ref}
CAR (reference)	7.5	1.00	1.00	1.5	1.00
BUS	15.0	2.00	0.85	1.8	1.00

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

TRUCK	13.5	1.80	0.90	2.0	0.85

2.2.2.2 BUS-STOP TYPOLOGIES AND LANE BLOCKAGE

The bus-stop model recognizes two typologies, i.e., BAY and IN_LANE. A BAY stop is an off-carriageway pull out, and no lane is blocked when a bus stops. In an IN_LANE stop the bus stops on the lane and blocks it during dwelling. For a stop s let dwelling(s) denote the (fluid-limit) bus mass currently in the stop's dwell pool which is a real-valued quantity so a value of dwelling(s) = 0.7 represents 0.7 bus-body's worth of mass currently blocking the stop's lane. The fractional values come from the fluid-limit discretization of the continuous arrival process. In simple words, only a fraction $\approx \Delta t \cdot \frac{v_c^{free}}{L_i}$ of the eligible free-pool mass reaches the stop's position over one step, so a single bus can be admitted over several steps. Let $B(s) \in \{0,1\}$ denote the block indicator, i.e., if dwelling in that stop blocks or not traffic. To model the blockage induced by dwelling buses we reduce the effective lane count. More precisely, the effective number of lanes at link i is the nominal lane count reduced by the maximum blockage over all bus stops that are placed within a link:

$$N_i^{lanes,eff} = \max\left(0, N_i^{lanes} - \max_{s \in i} B(s) \cdot \min\left(1, \text{dwelling}(s)\right)\right) \tag{9}$$

The clip $\min(1, \ \cdot)$ ensures that if many buses simultaneously dwell at a bus stop, they still block at most one lane. Similar to that, the max operator ensures that if many bus stops are present in the link (we make the assumption that the bus stops are always placed at the rightmost lane) the total blockage is no more than one lane.

2.2.2.3 BUS DISPATCH POLICIES

Buses are inserted into the simulation according to a per-route dispatcher. The available dispatchers are the following:

- The *constant-headway* dispatcher emits a continuous fluid rate of $\frac{\Delta t}{H}$ fractional buses per step over a service window $[T_0, T_1)$.
- The *schedule* dispatcher accepts an explicit list of departure times $\{t_1, t_2, \dots\}$ and inserts one bus per schedule departure falling within $[t, t + \Delta t)$.
- The *rate* dispatcher accepts a time-varying rate profile $\lambda(t)$ (buses per hour) and inserts $\frac{\lambda(t)\cdot\Delta t}{3600}$ buses per step as a fractional value added directly to the free-pool bus mass.

2.2.2.4 BUS ROUTE FOLLOWING

A bus route r is defined by its path which is a fixed ordered sequence of links $\pi_r = \left(l_0, l_1, \dots, l_{L_r}\right)$ and a subset of stops $S_r \subseteq \pi_r$. Note that bus stops are read as edges along the path where the bus should stop. When a bus of route r is dispatched, it is tagged with its route identity. Note that buses do not follow the turn-ratio table that the other modes follow. Their next link is fully determined by the route:

$$n\left(l_j, r\right) = l_{j+1}, \quad j = 0,1, \dots, L_r - 1 \tag{10}$$

and on the terminating link l_{L_r} the bus is credited as completed. Buses obey the same free-flow travel time as cars; they traverse the free-flow corridor at speed $\varphi_{bus} \cdot v_i^{free}$ and then join the movement queue but the queue they join is the one that their route prescribes. To achieve that (including knowledge where the bus should stop), per-route bookkeeping is maintained throughout every simulation step and more precisely two counters are updated:

- $n_{l,r}^{free}$ (free pool per link, per route)
- $n_{l,k,r}^{q}$ (queue per movement, per route)

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

2.2.2.5 CLASS-DEDICATED LANES

The design described in this subsection is currently under development. It concerns only corridors that carry a lane restricted to one or more vehicle classes and corridors without any such restriction continue to be modelled as described above.

For corridors that carry a class-dedicated lane (e.g., a dedicated bus lane) the network representation splits the physical corridor into two parallel links:

- A *shared* link L_i^A carrying the general-purpose lane count N_i^{shared} that admits every class.
- A *class-dedicated* link L_i^B carrying the restricted-lane count N_i^r which admits the classes that belong to $\mathcal{C}_i^B$ (the set that defines which vehicle classes are allowed to enter the class dedicated lane).

Both links connect the same upstream and downstream junctions in parallel. Each is an ordinary Link carrying its own storage, sending and corridor formulas. Some consequences of this representation are the following:

- A single admitted vehicle on L_i^B discharges at $q_c \cdot \Delta t \cdot N_i^r$ per step, matching the physical infrastructure.
- A lane-blocking event on the dedicated lane blocks L_i^B without blocking the whole corridor.
- L_i^A and L_i^B may carry different free-flow speeds and have their own storage cap etc.

2.2.3 DISCHARGE-SIDE EXTENSIONS

The baseline Store-and-Forward discharge formulation is extended with additional mechanisms to better capture the operational characteristics of real-world road networks. These extensions account for movement-specific capacities, congestion effects, lane interactions, priority-controlled intersections, and signal operations. The following sections describe each extension and its integration into the discharge formulation.

2.2.3.1 PER-MOVEMENT SATURATION-FLOW FACTORS

Empirical measurements at signalized intersections show that turning movements discharge at a lower saturation flow when compared to through movements because of the geometric radius of the manoeuvre. We capture this with a per-movement, class independent multiplier $\psi_{i,m} \in (0,1]$ (Transportation Research Board, 2016) inside the sending side of the discharge (see Eq. (6)):

$$\psi_{i,m} = \begin{cases} 1.00 & \text{through/straight,} \\ 0.85 & \text{right turn,} \\ 0.70 & \text{left turn,} \\ 0.50 & \text{U-turn.} \end{cases} \tag{11}$$

2.2.3.2 CAPACITY DROP

Empirical observations at oversaturated freeway bottlenecks have shown that the effective saturation flow of a link drops by approximately 5–15% once the queue on that link has persisted for a sufficient duration (Daganzo, 1999; Cassidy and Rudjanakanoknad, 2005; Cassidy and Bertini, 1999). This "capacity drop" is introduced via a multiplicative factor $\delta_i(t) = \delta_{i,m}(t) \in [0,1]$ different for each link i and scales down the sending capacity in Eq. (6) uniformly for all movements. When the total queue on link i has been continuously non-empty for at least T^{drop} seconds, δ_i drops from 1 to a factor δ_0 (typically 0.90, corresponding to a 10% throughput loss) and that drop remains fixed while the queue persists and is released back to 1 once the queue clears:

$$\delta_i(t) = \begin{cases} \delta_0 & \text{if link } i\text{'s queue persistence} \geq T^{drop} \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

2.2.3.3 DENSITY-MODULATED SATURATION FLOW

The Store-and-Forward baseline holds the saturation flow constant over the whole density range meaning that as soon as a movement has a green phase and a non-empty queue, it discharges at q_c. The empirical fundamental diagram's flow-density curve

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

has a descending branch above a critical density ρ_c. To that end, we introduce a multiplicative sending-side throttle that reproduces that up to an extent. The mathematical shape follows the classical Greenshields-derived flow-density curve (Greenshields, 1935), applied as a scalar multiplier on the queue-based discharge. Let $\rho_i = \left(n_i^{f,PCE} + n_i^{q,PCE}\right)/C_i^{PCE}$ denote the normalized link density. Then, the multiplier is

$$\mu_{DMSF}(i,t) = \begin{cases} 1 & \rho_i \le \rho_c \\ \max\left(\mu_{min}, \frac{\rho_i(1-\rho_i)^a}{\rho_c(1-\rho_c)^a}\right) & \rho_i > \rho_c \end{cases}, \qquad \rho_c = \frac{1}{a+1} \tag{13}$$

where a is a shape parameter of the flow-density curve, and ρ_c is the critical density while the floor μ_{min} prevents deadlock at near-jam density. Note that this definition is piecewise and ensures that DMSF acts as identity below critical density and only reduces sending when the link's density is above the critical density. Note that DMSF and capacity-drop both appear as multiplicative factors in the sending capacity defined in Eq. (6), but each mechanism is triggered by different signals. When both are triggered, their parameters are not separately identifiable from aggregate discharge data, meaning that it would be beneficial to jointly tune them against the empirical target.

2.2.3.4 PER-LANE APPORTIONMENT AND LANE-MERGE CAP

For the per-movement saturation flow share σ appearing in Eq. (6), simpi follows a two-stage split process. *Stage A* allocates the link's saturation flow evenly across the shared physical lanes,

$$s_i^{lane} = \frac{S_i}{N_i^{shared}} \tag{14}$$

where N_i^{shared} is the number of general-purpose lanes on link i. *Stage B* apportions each lane's budget among the movements it serves in proportion to a per-lane queue proxy $\pi_{i,m,c}$ that divides each movement's queue by the number of upstream physical lanes on link i that feed its slot,

$$\pi_{i,m,c} = \frac{n_{i,k(m)}^q(c)}{n_{serving}(m)} \tag{15}$$

$n_{serving}(m)$ is the number of physical lanes on link i from which a driver can legally execute movement m at the stop bar. On a two-lane approach, a right turn served only by the rightmost lane has $n_{serving} = 1$ while the through movement that can be served by both lanes has $n_{serving} = 2$. Note also that the heuristic assumes that each movement's queue is spread uniformly across its serving lanes and per-lane queues are not tracked explicitly.

Worked example. Consider a link with $N_i = 2$ physical lanes which allows two outgoing movements, i.e., a through movement T that is served by both lanes ($n_{serving}(T) = 2$) and a right-turn movement R served only by the rightmost lane ($n_{serving}(R) = 1$). Let n_T^q and n_R^q be their car-class queue masses and suppose that both are simultaneously green. Stage A gives both lanes the budget $\frac{S_i}{2}$. Lane 1 that serves entirely the through movement, contributes $\frac{S_i}{2}$ entirely to T. On lane 2, the per-lane queue proxies are $\frac{n_T^q}{2}$ for T and n_R^q for R, and the lane's $\frac{S_i}{2}$ budget is split in proportion. The totals are

$$\sigma_T S_i = \frac{S_i}{2} + \frac{S_i}{2} \cdot \frac{n_T^q/2}{n_T^q/2 + n_R^q} \tag{16}$$

$$\sigma_R S_i = \frac{S_i}{2} \cdot \frac{n_R^q}{n_T^q/2 + n_R^q} \tag{17}$$

Observe that the $\sigma_R S_i$ is bounded above by $\frac{S_i}{2}$ meaning that this is the maximum saturation flow of the single lane R.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

2.2.3.5 GAP ACCEPTANCE AT PRIORITY INTERSECTIONS

When the classical S&F discharge formulation is applied to unsignalized priority-controlled intersections, Eq. (6) becomes unrealistic because every movement would discharge at its nominal saturation flow regardless of conflicting traffic. To that end, we utilise the Harders/Siegloch macroscopic gap-acceptance capacity formula (Harders, 1968; Siegloch, 1973), in the form

$$c_{minor}(i) = \frac{q_M(i)\cdot\exp(-q_M(i)\cdot t_c)}{1-\exp(-q_M(i)\cdot t_f)} \tag{18}$$

where

- $q_M(i)$ (veh/s) is the summed flow on the conflicting major streams at intersection i.
- t_c (default 5s) is the critical gap (minimum acceptable major-stream headway that a minor driver will accept).
- t_f (default 3s) is the follow-up time minimum inter-vehicle headway for successive minor drivers using the same accepted gap.

The formula is derived under the assumption that major-stream headways are exponentially distributed and returns the effective capacity of the minor stream.

The gap-acceptance capacity enters the discharge as a multiplicative factor (Eq. (6)) and more precisely,

$$\gamma_{i,m}(t) = \begin{cases} \min\left(1,\ c_{minor}\left(\frac{host(i)}{s_{ref}}\right)\right) & \text{if } m \text{ is a minor movement at a PRIORITY junction} \\ 1 & \text{otherwise} \end{cases} \tag{19}$$

Here $host(i)$ denotes the intersection at which link i is an incoming link and s_{ref} (veh/s) is a representative per-lane saturation flow.

Two points should be emphasized. First, the major-stream flows $q_M(i)$ are tracked as an exponentially-smoothed window of the observed discharge rates of the major incoming links at intersection i. Second, in the current implementation, all major approaches at i are summed into a scalar $q_M(i)$ and no per-conflict set of movements is considered which means that different minor movements that belong to the same intersection have the same $\gamma_{i,m}(t)$ value.

2.2.3.6 SIGNAL CONTROL

Each signalized intersection is described by a fixed-time cyclic plan. This signal plan has n phases and each phase φ has a green duration g_φ. The cycle also includes yellow, all-red and startup lost times and the runtime state machine of an intersection has four states, i.e.,

- Green
- Yellow
- All-red
- Startup

The state transitions are triggered by the modular cycle position:

$$t_{local} = (t + \text{offset}) \bmod T_{cycle}, \quad T_{cycle} = \sum_{\varphi=0}^{n-1} g_\varphi + T_{lost} \tag{20}$$

The offset shifts the cycle globally, allowing green-wave coordination along an arterial for example.

Three lost-time effects are supported per intersection:

- *Yellow* intervals of duration y_i inserted at the end of each green phase. During yellow the per-movement green mask $g_{i,m}(t) \in \{0,1\}$ that appears in Eq. (6) is forced to zero.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

- *All-red* intervals of duration r_i inserted between the yellow of one phase and the green of the next phase. The green mask is again forced to zero in this case.
- *Startup lost time* of duration s_i following each transition into GREEN. During this window the discharge capacity of the newly-green movements ramps linearly from zero to its nominal value,

$$r_i^{start}(t) = \min\left(1, \frac{t - t_i^{green}}{s_i}\right) \quad (21)$$

where t_i^{green} is the time at which intersection i last entered its GREEN state. This models the observed acceleration of the queue tail after the light turns green and it is applied as a per-link multiplicative factor in Eq. (6).

2.2.4 TRIP-AGE COMPLETION MODEL

The Trip-Age Completion Model extends the simulation framework with a mechanism for representing trip progression and completion based on the distance travelled through the network. It aims to provide greater control over the resulting trip-length distribution while retaining the computational efficiency of the aggregate traffic representation. Its formulation and current implementation are described below.

2.2.4.1 MOTIVATION

The formulation proposed by Tsitsokas et al. (2021) is an anonymous-walker exit-ratio formulation in which trips are completed according to a per-link Bernoulli trial. We can view this model as an absorbing Markov chain on the link graph where the exit-ratio plays the role of the state-dependent absorbing probability. Although this approach is analytically tractable, the number of link traversals until absorption follows a geometrical distribution which is monotonically decreasing (mode at the minimum number of link traversals) and it has a right tail extending to infinity. The shape of the distribution is determined solely by the exit rate, making it difficult for the classical model to reproduce the trip-length distribution prescribed by the input demand. It can systematically over-produce short trips and simultaneously long-wandering trips. The trip-age completion model presented in this subsection, although it is still under development, replaces the memoryless Bernoulli rule with an approach that is inspired by age-structured population dynamics (McKendrick, 1925; Webb, 1985). Rather than evolving the full McKendrick–von Foerster distribution, each compartment tracks only the total mass and total remaining budget. Note that the within-compartment budget distribution is closed by a deterministic (Dirac) rule (Figure 2).

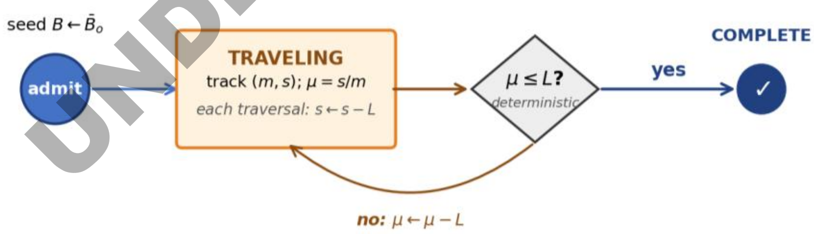


Figure 2. Trip-age completion lifecycle.

2.2.4.2 STATE AND DETERMINISTIC-CREDIT CLOSURE

Each compartment (free pool of link i, or slot k of the queue of link i) tracks two moments per class:

- Total mass m (in PCE) and,
- Total remaining-distance budget pool $s = \sum_a B_a$, where a indexes the vehicle class c in the compartment.

The cohort's mean remaining budget is derived on demand:

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

$$\mu_i(c) = \frac{s_i^f(c)}{m_i^f(c)}, \qquad \mu_{i,k}(c) = \frac{s_{i,k}^q(c)}{m_{i,k}^q(c)} \tag{22}$$

For the multi-class case, s is PCE-weighted so that $\mu = s/m$ has units of budget of PCE. At admission, the pool is incremented by $(\text{admitted mass}) \cdot \bar{B}_0$, where $\bar{B}_0$ is either a global mean trip length or a per-origin planned trip length. Every link traversal consumes L_i meters of budget from the pool of the traversing mass. Mass and pool are extensive and the mean μ is derived and thus preserved under proportional splits (transfers, queue inflows, discharges).

At each discharge event, the traveling cohort has two possible transitions. Either its mean budget μ is below or above the link length. If it is below the link length the whole cohort becomes *ready,* otherwise the cohort continues its trip with mean budget $\mu' = \mu - L$:

$$\text{cohort becomes ready} \Leftrightarrow \mu \leq L \qquad \text{else } \mu' = \mu - L \tag{23}$$

This is a deterministic-credit closure approach, and it corresponds to a Dirac within-cohort distribution meaning that every car in a cohort is assumed to have the same remaining budget as the cohort mean. Note that the trip-age model is still under development and extension.

2.2.5 DYNAMIC ROUTING

In simpi there is also a rerouting module. When this module is activated, the turn ratios are no longer static. Every τ seconds the routing block recomputes for each origin o, the shortest-path tree on the link graph with edge costs that are derived from the previous window's observed link speeds. The newly admitted demand, together with carried-over trips (ongoing) that have remained from the previous tick, are loaded into their calculated shortest path which are then aggregated over each traversed connection. The connection values are then row-normalized into the new turn ratio matrix T^{dijk} which replaces the old one. This dynamic turn-ratio update procedure is based on Tsitsokas et al. (2023).

2.2.5.1 LINK COST

The window-averaged mean speed on link i over an interval τ is estimated as

$$v_i = \min\left(v_i^{free}, L_i \cdot \frac{\sum_{t\in\tau} q_i(t)}{\Delta t \sum_{t\in\tau} n_i(t)}\right) \tag{24}$$

where $q_i(t)$ is the outflow and $n_i(t)$ is the accumulation of link i at step t. Eq. (24) gives an estimate of the mean traversal speed, clipped at the link's free-flow speed and Δt is the simulation timestep. The Dijkstra cost of link i is the classical travel time

$$c_i = w_i \cdot \frac{L_i}{\min(v_i, v_{min})} \tag{25}$$

where v_{min} prevents the case $v_i \to 0$ when a link is jammed and w_i is a weight parameter. When $w_i > 1$, routing through link i is discouraged, and when $w_i < 1$ it is encouraged. The weight parameter is optional and defaults to 1. The weighting is useful to reflect the empirical observation that drivers do not divert from main corridors easily.

2.2.5.2 PER-ORIGIN SHORTEST PATHS AND CONNECTION COUNTING

First the cost graph $\{c_i\}$ is created. For each origin o, Dijkstra produces the per-destination shortest paths $\pi_o(d)$. Note that connection here means a (in-link, out-link) pair at a shared intersection. Each admission volume $v(o,d)$ accumulated over the previous τ-second window is loaded onto its shortest path $\pi_o(d)$ and contributes to every connection (i,j) that it traverses. Moreover, OD pairs with nonzero demand from o contribute to the counter. A second contribution loads onto the same counter at each tick, from the *ongoing-trip stack*. At the previous tick, any trip whose Dijkstra-estimated duration exceeded τ is split at the first link ℓ where cumulative traversal time crossed τ. The pre-split portion contributes to that tick's connection weights, and the post-split portion is pushed onto a per-origin stack where we store also the split link ℓ, destination d and volume v. At the current tick, before the fresh-admission pass runs, the stack is re-processed and each entry treats ℓ as a fresh origin carrying the already-realised

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

volume v, runs Dijkstra from ℓ using the freshest costs, and walks the shortest path from ℓ to d, depositing weight on every traversed connection just as the fresh-admission pass does.

Connection weights are accumulated per (in-link, out-link) pair from every OD's shortest-path load,

$$w[i.j] \mathrel{+}= v(o,d) \quad \forall (i,j) \in \pi_o(d) \tag{26}$$

and then row-normalized per link to produce the fresh turn-ratio matrix,

$$\text{turn}(i,j) = \begin{cases} w[i,j]/\sum_{j'} w[i,j'] & \text{if } \sum_{j'} w[i,j'] > 0, \\ \text{turn}_{\text{prev}}(i,j) & \text{if } \sum_{j'} w[i,j'] = 0 \text{ and a ratio exists,} \\ 1/|\mathcal{O}(i)| & \text{otherwise (uniform over valid outlets } \mathcal{O}(i)) \end{cases} \tag{27}$$

The volume $v(o,d) = r(o,d) \cdot A(o)/D(o)$ combines the nominal OD demand rate $r(o,d)$ with the origin's realised admission scaling, $A(o)$ is the cumulative realised admissions at origin o over the routing window, and $D(o) = \sum_{d'} r(o,d')$ is the corresponding nominal total across o's destinations.

2.2.5.3 SMOOTHING AND EXTENSIONS

Several optional extensions have been developed and are applied to T^{dijk} before replacing the previous turn-ratio matrix. A soft exponential moving average with weight $a \in (0,1]$ blends the fresh Dijkstra matrix with the previous one:

$$T_{new}(i,j) = a \cdot T^{dijk}(i,j) + (1-a) \cdot T_{prev}(i,j) \tag{28}$$

The EMA rate-limits how fast turn ratios can shift between corridors of similar cost, thus limiting oscillatory behaviour. Moreover, it has a behavioural reading. A fraction a of the population routing through each link takes the fresh Dijkstra recommendation, and the remaining $1 - a$ sticks to the previous turn ratio matrix. As individual re-routings are not tracked the blending can describe only the aggregate effect. An in-flight recommit allows a fraction (default 1.0) of already-queued vehicles (based on previous turn ratios) to re-target on the newly-updated ratios. This models the physical constraint that not all queued vehicles can change lanes to a new outlet.

2.3 PRELIMINARY RESULTS

The following experiments test whether the developed macroscopic simulator can reproduce the aggregate traffic dynamics that emerge from a well-resolved microscopic model. Simpi resolves the network at the link level using Store-and-Forward dynamics with a multi-class PCE overlay and dynamic routing. SUMO is the microscopic reference that we utilise which resolves individual vehicles' car-following and lane-changing per-vehicle. The two simulators are compared on aggregate observables that a macroscopic model is expected to reproduce, i.e.,

- the Macroscopic Fundamental Diagram,
- network accumulation, and
- completed-trip time series.

The test scenario is a realistic road network that replicates part of the Athens city centre (Figure 3) under morning peak demand. It consists of 1152 directed links and 561 intersections (a mix of signal-controlled and priority-controlled) and the demand consists of 44137 planned vehicle trips. SUMO is treated as the empirical ground truth (microscopic reference) and simpi is calibrated to match its aggregate behaviour. All parameters were chosen and calibrated in a step-by-step fashion to close the gap to the microscopic reference on the sub-network under study. Note that an automated calibrator is under development which will utilise rich simulation data in order to tune the simulator to match the microscopic reference.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

Figure 3. Simulated sub-network of the Athens city centre used in the experiments.

Figure 4 shows the three main comparisons: (a) the accumulation trajectory $N(t)$; (b) the cumulative trip completion $C(t)$; (c) the Macroscopic Fundamental Diagram (production per bin against accumulation). In each panel the microscopic reference is shown as a grey mean with a ±1σ envelope, and the simulator's output as a coloured line (accumulation and cumulative completion) or scatter (MFD). The accumulation trajectory shows a slightly higher peak than the microscopic reference (5.3% above). The cumulative-completion curve is essentially indistinguishable from the microscopic reference for most of the horizon and finishes within about a hundred vehicles of the microscopic mean (98 difference). The MFD scatter shows a good overlap for the loading and draining branches of the two simulators across the whole accumulation range. Moreover, the descending branch is present and has approximately a satisfying slope.

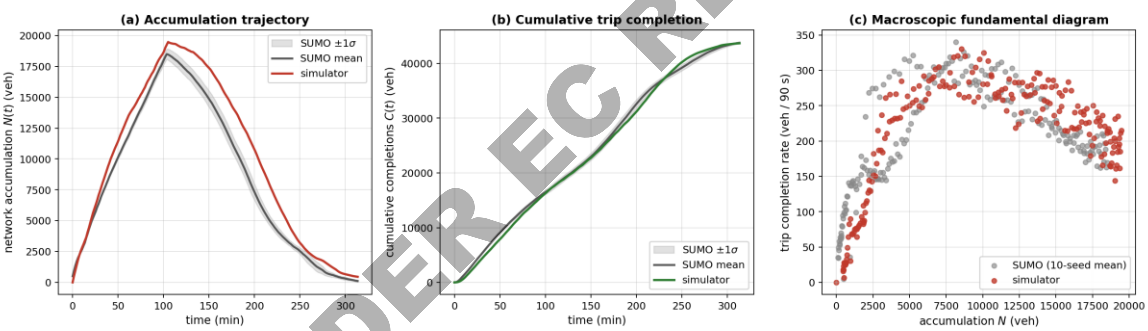


Figure 4. (a) Accumulation trajectory $N(t)$, (b) Cumulative trip completion $C(t)$, (c) Macroscopic Fundamental Diagram (MFD).

Table 3 summarises five macroscopic indicators. The peak accumulation is matched to within 5.3%, the peak production to within 3%, the time to peak accumulation to within 1.5 minutes, the completion ratio to within 0.4%, and the final completed count to within about a hundred vehicles out of about 44000. These matches are simultaneous for a single configuration and represent the strongest match we have obtained so far on this sub-network.

Table 3. Comparison of five macroscopic indicators between the microscopic reference and the calibrated macroscopic simulator.

METRIC	MICROSCOPIC REF.	SIMULATOR	RATIO
Peak accumulation	18495 veh	19476 veh	1.053
Peak production / 90s	340.1 veh	330.3 veh	0.971
Time-to-peak accumulation	103.5 min	105.0 min	1.014

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Completion ratio (final/generated)	0.994	0.990	0.996
Final completed count	43782 veh	43684 veh	0.998

Regarding computational performance, on a single core of a modern workstation (12th Gen Intel® Core™ i7-1255U processor, 16 GB RAM), the simulator processes the Athens network in approximately four seconds which is approximately 1000 simulator steps (Δt=5s) per second and approximately 5000 simulated seconds per real second. This is about three orders of magnitude faster compared to a SUMO run.

2.4 CONCLUSIONS

This chapter presented the current development status of simpi, a Python-based multimodal macroscopic traffic simulator designed to bridge the gap between computational efficiency and realistic network-level traffic dynamics. The simulator combines a horizontal-queue Store-and-Forward formulation with a multi-class passenger-car-equivalent representation, dynamic routing, and a series of discharge-side extensions to better reproduce the aggregate behaviour of detailed microscopic traffic simulations while maintaining computational efficiency suitable for optimisation and AI training environments.

The preliminary evaluation on a realistic sub-network of the Athens city centre demonstrated encouraging results. The developed simulator was able to reproduce the key macroscopic characteristics of the reference SUMO microsimulation, including the evolution of network accumulation, trip completion dynamics, and the shape of the Macroscopic Fundamental Diagram. The close agreement observed across multiple aggregate performance indicators suggests that the proposed modelling framework captures the dominant mechanisms governing large-scale urban traffic dynamics while operating at substantially lower computational cost.

Although these initial results are promising, the simulator remains under active development. Several components are currently being extended and refined, including the implementation of class-dedicated lane modelling, further improvements to the trip-age completion framework, enhanced dynamic routing capabilities, and additional calibration mechanisms. In parallel, an automated calibration framework is being developed to systematically tune the simulator parameters using rich microscopic simulation data, reducing manual calibration effort and improving transferability across different networks and traffic conditions.

The work presented in this chapter establishes the methodological foundations for a lightweight multimodal simulation environment that will support the remaining activities of the project, particularly the training, testing, and validation of AI-assisted traffic management strategies. Future work will focus on extending the validation to networks that incorporate additional transport modes (e.g., buses) and operational features and further improving the fidelity of the macroscopic model while preserving its computational advantages. Finally, alignment with Task 5.5 will be ensured in order for the under-development simulator to be appropriate for use in the sim-to-real framework.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

3 AGENT-BASED MODELS FOR COMPLEX USER BEHAVIOUR SIMULATION

3.1 TASK DESCRIPTION

The objective of this task is the development of an open-source, flexible simulation framework for modelling user behaviour in a multimodal urban environment populated by heterogeneous travellers. The framework must capture how demand shifts across a wide variety of mobility services, such as Automated Mobility on Demand (AMoD), shared micromobility, and electric-vehicle fleets, and how it responds to system-level mechanisms such as Mobility as a Service (MaaS), dynamic pricing, and artificial currencies. To this end we build upon SimMobility, an agent-based, activity-driven, open-source platform[1], and extend its demand models for the Greater Copenhagen Area, implementing and evaluating enhancements to its planning and Within-day components using data from selected pilot studies.

Modelling travel demand at this level of behavioural detail requires a departure from the aggregate methods that have historically dominated transport planning. The state of the art has advanced over the last decades from the traditional four-step model, through tour-based models, to activity-based models (Bowman and Ben-Akiva, 2000). Four-step models treat trips as independent units and reproduce origin-destination flows without an explicit representation of the traveller, which makes them ill-suited to policies that act on individual decisions and on the sequencing of activities across the day. Activity-based models instead suggest that demand for travel is derived from the demand to perform activities: a trip is undertaken only when the utility of an activity and its associated travel exceed the utility of remaining in place (Bowman and Ben-Akiva, 2000). This premise is precisely what FEDORA requires, because the solutions under study (pricing, tradable credits, shared and automated services) reshape behaviour at the level of the individual traveller and the daily schedule, not at the level of anonymous aggregate flows. An agent-based, activity-driven representation also makes it possible to analyse FEDORA's solutions across diverse user groups, including low-income, low-accessibility, and elderly populations, and its open architecture supports replication and commercial transfer, as demonstrated in prior applications.

The demand modelling framework adopted here is the Day Activity Schedule (DAS) approach of Bowman and Ben-Akiva (2000) and Bowman (1998), as operationalised within the SimMobility Mid-term simulator (Adnan et al., 2016; Lu et al., 2015). The same framework has been estimated and implemented for Singapore (Li, 2015), for the Greater Boston Area (Viegas de Lima et al., 2018), and, most relevant to the present work, for Denmark (Monteiro et al., 2022). FEDORA's contribution is not a re-derivation of this framework but its application and extension to the Greater Copenhagen Area: the Danish activity-based model set is adopted and re-calibrated, its demand-generation and Within-day execution flows are enhanced to represent emerging services such as automated mobility on demand, shared micromobility, and electric-vehicle fleets, and they are coupled with the dynamic mobility controllers developed in Task 5.3.

This chapter is organised as follows. The Methodology first introduces the SimMobility platform and the structure of its Mid-term simulator, then presents the activity-based modelling framework and the two demand-side components at the heart of Task 5.2, the Pre-day module, which generates each agent's daily activity and travel schedule, and the Within-day module, which executes and adapts that schedule in response to network conditions. It then describes the data and synthetic population that drive the models,

[1] https://github.com/smart-fm/simmobility-prod

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

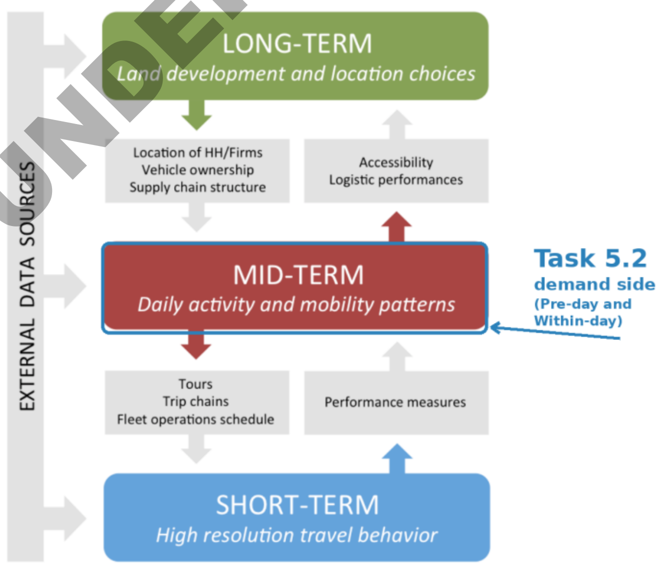
the estimation and calibration procedure, and the application of the framework to the Greater Copenhagen Area together with the enhancements introduced for FEDORA. The Preliminary Results report the current status of the Copenhagen implementation, and the chapter closes with the ongoing developments.

3.2 METHODOLOGY

3.2.1 THE SIMMOBILITY SIMULATION PLATFORM

SimMobility is an open-source, integrated agent- and activity-based simulation platform developed by the Massachusetts Institute of Technology (MIT) and the Singapore-MIT Alliance for Research and Technology (SMART) (Adnan et al., 2016). It was designed around two principles that make it well suited to FEDORA: first, that travel demand and network supply should be simulated simultaneously, so that behavioural responses and network performance are mutually consistent; and second, that a single synthetic population of agents should be carried consistently across very different temporal horizons, so that long-run, day-to-day, and Within-day decisions are represented in one coherent behavioural chain rather than in disconnected models. In this respect SimMobility sits alongside other large-scale agent-based transport simulators such as MATSim (Horni et al., 2016) and POLARIS (Auld et al., 2016). Its particular relevance here is its tight integration of an activity-based demand model with network supply across the long-, mid- and short-term horizons.

The platform is accordingly organised into three modules that operate at different time scales. The Long-term module simulates year-to-year dynamics, capturing the evolution of land use, vehicle ownership, and the economic decisions that shape where agents live and work. The Mid-term module simulates day-to-day dynamics, each agent's daily activity and travel pattern, and its interaction with the transport network. The Short-term module simulates Within-day dynamics at microscopic granularity, reproducing the second-by-second movement of individual agents through the network. The three modules share a common representation of the agent and of the network, so that the outputs of one horizon become the inputs of the next: the long-term module supplies the population and its residential and workplace locations, and the mid-term module supplies the daily schedules that the short-term module executes in detail (Figure 5).



Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

Figure 5. The SimMobility multi-scale platform: the Long-term, Mid-term, and Short-term modules, the external data sources that feed them, and the information passed between successive time scales. Task 5.2 develops the demand side of the Mid-term module (outlined). Source: Adnan et al. (2016)

Task 5.2 operates within the Mid-term module, which is the level at which daily travel demand is generated and adapted. The behavioural model specifications are held in editable Lua scripts (Lu et al., 2015; Viegas de Lima et al., 2018), so that the choice-model coefficients, utility specifications, and availability rules can be inspected, and re-calibrated for a new region without recompiling the simulator. This keeps the transfer of the adopted model set to the Greater Copenhagen Area tractable.

3.2.2 THE MID-TERM SIMULATOR AND THE ACTIVITY-BASED MODELLING FRAMEWORK

The Mid-term simulator consists of three interacting components (Lu et al., 2015). The Pre-day component generates, for every individual in the synthetic population, a complete daily activity and travel plan (the Day Activity Schedule) from a system of estimated discrete-choice models. The Within-day component executes those plans over the course of the simulated day, determining departure times and route choices and allowing agents to re-plan in response to real-time network conditions. The Supply component represents the transport network and its services, both public and private, and returns the experienced travel times and costs that the demand side reacts to. Pre-day and Within-day are the two demand-side components that Task 5.2 develops. The demand side interacts with the network through the Supply component, and it is at this point that the coupling to the Task 5.3 controllers takes place (Figure 6).

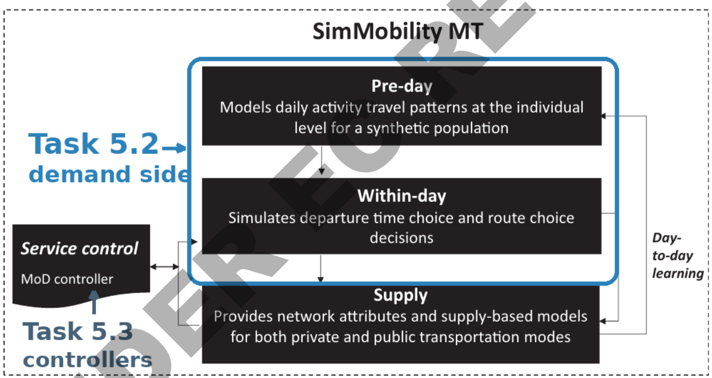


Figure 6. The SimMobility Mid-term simulator: the Pre-day, Within-day, and Supply components with the day-to-day learning loop. The Pre-day and Within-day components (outlined) are the demand side developed in Task 5.2, and the Service control block is the interface to the Task 5.3 controllers. Source: Nahmias-Biran et al. (2019)

Both demand-side components rest on the same behavioural foundation: the Day Activity Schedule approach (Bowman and Ben-Akiva, 2000; Bowman, 1998). Under this approach a traveller's day is represented not as a set of independent trips but as a structured schedule of activities, organised into home-based tours (sequences of trips that begin and end at home) which are themselves combined into a single daily plan. A tour is built around a primary activity and may contain secondary activities, or intermediate stops, performed along the way. Representing the day as a schedule rather than as isolated trips is what allows the model to account for the constraints and opportunities that link decisions across the day: the limited time budget, the need to return home, the competition between activities for the same hours, and the trade-offs an individual makes between travelling and staying in place (Bowman and Ben-Akiva, 2000).

Each decision in the schedule (whether to travel, which activities to pursue, where and when to perform them, and by which mode) is represented as a discrete choice made under the principle of random utility maximisation. A traveller is assumed to associate a utility with each available alternative, composed of a systematic part that depends on observed attributes and a random part that

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

captures everything the analyst does not observe. The alternative with the highest utility is chosen. Under the standard assumptions this yields a logit probability for alternative i in a choice set C_n for individual n:

$$P_n(i) = \frac{e^{V_{in}}}{\sum_{j \in C_n} e^{V_{jn}}} \tag{29}$$

where V_{in} is the systematic utility. The choices are not independent but hierarchical: higher-level decisions (whether and for what purpose to travel) are made conditional on the expected outcomes of the lower-level decisions they enable (where, when, and how). This coupling is carried by the logsum, the expected maximum utility of a choice set:

$$\Lambda_n = ln \sum_{j \in C_n} e^{V_{jn}} \tag{30}$$

which measures the overall attractiveness, or accessibility, of the alternatives available at a lower level and enters as an explanatory variable in the level above. Conditionality therefore flows downward through the hierarchy while accessibility, in the form of logsums, flows upward, so that, for example, the decision of whether to make a shopping tour at all is informed by how attractive the available shopping destinations and modes actually are for that specific individual. The individual models differ in the attributes that enter their systematic utilities and in their alternatives, but all share this common logit-and-logsum structure (Figure 7).

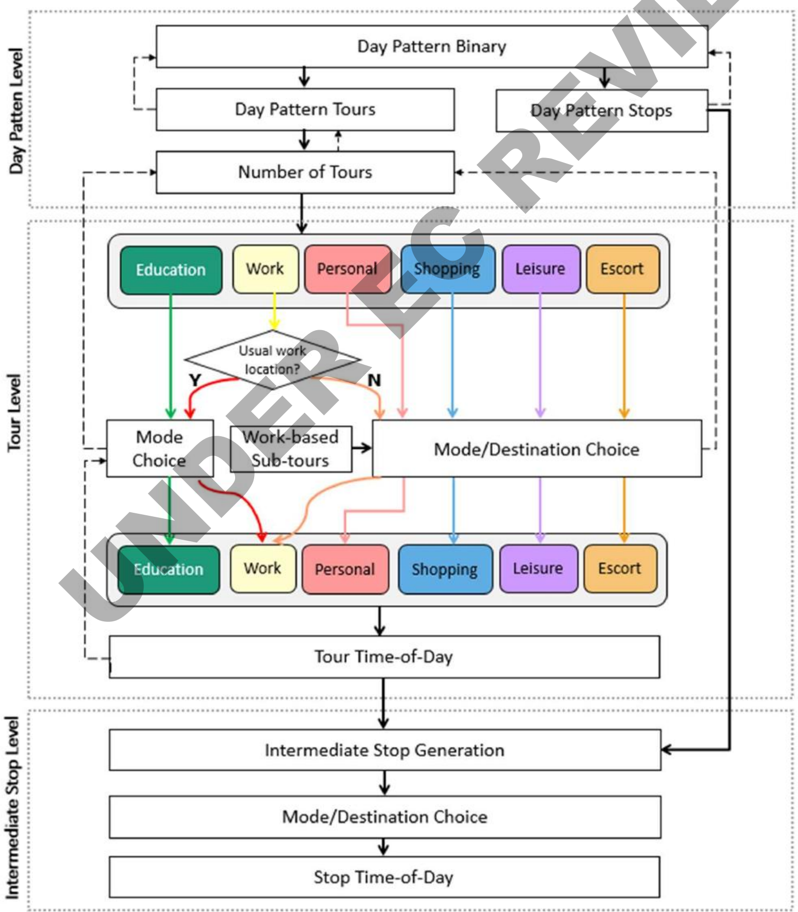


Figure 7. The activity-based model system: the day-pattern, tour, and intermediate-stop levels, with the six activity purposes and the choice models at each level. Conditionality flows downward (solid) and accessibility (logsum) flows upward (dashed). Source: Monteiro et al. (2022).

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

This activity-based Mid-term framework is not newly created for FEDORA: it was developed and demonstrated in earlier research, including its original application to Singapore (Li, 2015) and its transfer to the Greater Boston Area (Viegas de Lima et al., 2018). Task 5.2 reuses that established framework rather than rebuilding it, bringing the Pre-day and Within-day components together with the adopted Danish model set into a single, combined codebase for the Greater Copenhagen Area.

3.2.3 THE PRE-DAY MODULE

The Pre-day module is the demand-generation engine of the Mid-term simulator. For every individual in the synthetic population it produces a complete Day Activity Schedule by evaluating a hierarchical system of discrete-choice models, structured into three levels (the day-pattern level, the tour level, and the intermediate-stop level) in which lower-level choices are conditional on upper-level ones and feed accessibility logsums back to them. This three-level structure is the established SimMobility Pre-day specification, set out in detail in its earlier applications to Singapore (Li, 2015) and the Greater Boston Area (Viegas de Lima et al., 2018). In the Copenhagen implementation this system comprises twenty-nine estimated behavioural models, ten of which are activity-generic (the day-pattern, usual-workplace, work-based sub-tour, and intermediate-stop models) and nineteen of which are specific to the individual activity purposes (Monteiro et al., 2022), together with a shared logit engine that turns each model's utilities into a probability and draws a choice from it.

Execution proceeds in two phases over the same population. In the first phase the module computes, for every individual and without committing to any choice, the logsum of each tour-level model and of the day-pattern models, that is, the expected utility of each activity's mode and destination options evaluated across all zones and modes. These logsums are accessibility measures: they tell the upper-level models how attractive and how feasible each activity is for that specific person before any decision is taken, and they are stored for reuse. In the second phase the module simulates the schedule itself, walking down the hierarchy and drawing each choice by Monte Carlo from its logit model, consuming the first-phase logsums wherever an upper-level model needs to know the accessibility of the levels beneath it. The description that follows traces this second phase from the top of the hierarchy to the assembled schedule. Table 4 sets out the complete model system, level by level.

Table 4. The Pre-day model system: for each level, the choice models, what each decides, the number of alternatives, and the activity purposes to which each applies

LEVEL	CHOICE MODEL: what it decides	ALTERNATIVES	APPLIES TO
Day-pattern	Day-pattern binary: does the person travel at all today?	2	all
Day-pattern	Tour purposes: which purposes are toured today	64	all
Day-pattern	Stop purposes: which purposes may appear as stops	64	all
Day-pattern	Number of tours: per active purpose	2-3	each toured purpose
Tour	Usual workplace: fixed workplace vs. other	2	work
Tour	Tour mode: fixed destination	6	work, education
Tour	Tour mode + destination: chosen jointly	≈28000	shop, leisure, personal, escort, non-usual work
Tour	Tour time-of-day: arrival × departure windows	1176	all toured purposes

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

LEVEL	CHOICE MODEL: what it decides	ALTERNATIVES	APPLIES TO
Tour	Work-based sub-tour: generation / mode+dest. / time-of-day	2 / ≈28000 / 1176	work
Stop	Stop generation: next stop purpose or stop (looped, ≤3)	7	each half-tour
Stop	Stop mode + destination: chosen jointly	≈56000	each generated stop
Stop	Stop time-of-day: half-hour window	48	each generated stop

3.2.3.1 DAY-PATTERN LEVEL

The day-pattern level fixes the overall shape of the day before any individual tour is detailed, and it is the level that most clearly distinguishes an activity-based model from a tour-based one, because it reasons about the day as a whole. It begins with a binary decision (whether the person travels at all on the modelled day or stays at home) taken as a function of the individual's socio-economic attributes and of the day-pattern accessibility logsums. A person who does not travel receives an empty schedule and the chain ends. For a person who does travel, a combinatorial choice then determines which of the six activity purposes (work, education, shopping, leisure, personal business, and escort) are pursued through at least one tour that day; because any subset of the six may be chosen, this is a choice among sixty-four alternatives, informed by the per-activity logsums so that purposes with poor accessibility for that individual are less likely to be selected. A parallel combinatorial choice determines which purposes are permitted to appear as intermediate stops during the day, and its outcome sets the availability conditions later consulted by the stop-generation model. Finally, for each purpose that will be toured, a dedicated model draws the number of tours of that purpose, one or two for most purposes, and up to three for leisure. The availability of purposes is conditioned throughout on the individual's characteristics: education tours, for instance, are not available to non-students. The day-pattern level thus outputs the skeleton of the day: whether the person is mobile, the primary purposes they will travel for, how many tours of each, and which purposes may be inserted as stops.

3.2.3.2 TOUR LEVEL

The tour level details each tour created by the day-pattern level, deciding where it goes, by which mode, and when. For work tours a preliminary choice first establishes whether the individual travels to their usual, fixed workplace or to another work-related location, since this determines whether a destination has to be modelled at all. When the destination is fixed (the usual workplace, or the known location of a school for education tours) only the travel mode is chosen, from the six modes available in the base Copenhagen configuration: public transport, car as driver (drive-alone), two shared-car options corresponding to car passengers in vehicles of different occupancies, cycling, and walking. The twelve-mode menu shown later for the Virtual City demonstration (Figure 10) is a finer set that additionally includes the on-demand and pooled services central to FEDORA, which are added to the Copenhagen mode set as the demand-side enhancements mature. When the destination is not fixed, which is the case for shopping, leisure, personal, and escort tours and for non-usual work tours, the mode and the destination are chosen jointly: the model evaluates every combination of a candidate destination zone and a mode, which in the Copenhagen zone system of 4697 zones amounts to more than twenty-eight thousand alternatives per tour. The systematic utilities of these alternatives draw on the individual's attributes (licence holding, vehicle availability, income), on the level-of-service between the home zone and each candidate destination (in-vehicle time, waiting and walking time, fare or operating cost, for the outbound and return periods), and on the attraction of each destination zone (its employment for work, retail floorspace for shopping, or population for the

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

discretionary purposes). Mode availability is constrained by the individual's licence and household vehicle holdings and by distance thresholds, walking and cycling are unavailable beyond their respective distance caps.

Once a tour's mode and destination are set, a time-of-day model places the tour in the day by selecting jointly its arrival and departure half-hour windows from the forty-eight half-hours of the day. Because every feasible pairing of an arrival and a later departure window is an alternative, this is a choice among more than eleven hundred possibilities, evaluated against travel times that vary by time of day for the chosen mode and origin-destination pair. Selecting both endpoints jointly fixes not only when the tour occurs but also how long the primary activity lasts. Finally, for work tours, a work-based sub-tour model determines whether the individual makes an additional excursion from the workplace during the working day (a lunch trip, for example) and, if so, models that sub-tour's mode, destination, and timing as a small tour nested within the work activity.

3.2.3.3 INTERMEDIATE-STOP LEVEL

The intermediate-stop level inserts secondary activities into the tours. Working outward from the primary activity along each half-tour (the outbound leg from home to the primary activity and the inbound leg back) a stop-generation model repeatedly decides whether to add a further stop and, if so, of which purpose, drawing from the purposes that the day-pattern level permitted as stops and from a stop option that terminates the insertion; up to three stops are allowed on each half-tour. For each stop that is generated, a joint mode-and-destination model chooses where the stop is made and how it is reached, with the mode constrained to be consistent with the tour it belongs to, and a time-of-day model places the stop in one of the forty-eight half-hour windows consistent with the time remaining. The intermediate-stop level is what turns simple out-and-back tours into the realistic trip chains (home to childcare to work or work to shop to home) that characterise observed travel.

3.2.3.4 TOUR ORDERING AND TIME-OF-DAY SCHEDULING

The models above decide the composition of the day, but the order in which a person's tours are placed in time is set by a fixed priority convention rather than by a further choice model: the primary anchoring activities, work and education, are scheduled first, followed by the discretionary purposes. Scheduling then proceeds greedily and in this order, with each tour claiming a contiguous block of time and the time-of-day models for later tours seeing only the windows that earlier tours have left free. This mechanism enforces two behavioural realities (that the day is consumed in sequence and that higher-priority activities get first claim on it) and guarantees that the resulting schedule is internally consistent, with no two activities overlapping in time. It also means that a person's remaining time budget tightens as the day is assembled, so that low-priority tours may ultimately be dropped if no feasible window remains.

3.2.3.5 THE DAY ACTIVITY SCHEDULE

The output of the Pre-day module is the Day Activity Schedule: for each individual, an ordered list of activities and the trips between them, resolved to half-hour arrival and departure times, with each activity located at a specific zone and each trip assigned a primary mode. For example, one agent's day might comprise a work tour made by car that includes an escort stop on the outbound leg (leaving home around 07:30, dropping off a passenger on the way, working from 08:00 to 16:30, and returning home by 17:00), followed by a short shopping tour made by bicycle in the early evening (out around 18:00, shopping from 18:30 to 19:00, and home by 19:30), with the remaining hours spent at home. In the final assembly step the chosen zones are converted to specific network nodes and the schedule is written out as one record per stop, carrying the activity type, its location and timing, the mode used to reach it, and the preceding stop from which the traveller came. This schedule is the direct interface to the operational simulation: it is the plan that the Within-day module executes and adapts, and, through it, the demand that the Supply component must serve. Figure 8 illustrates, for a single synthetic agent, how the three levels combine into one day's schedule.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

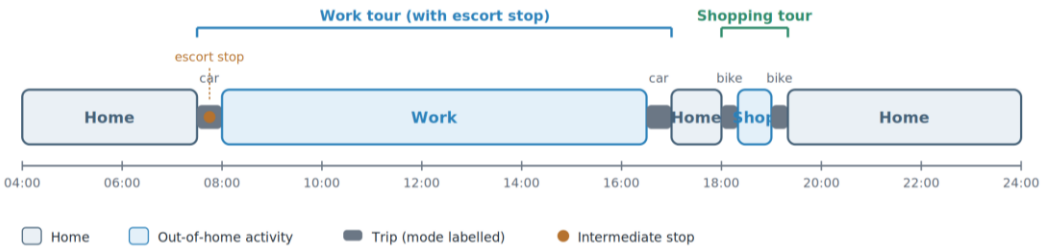


Figure 8. Worked example of one synthetic agent's Day Activity Schedule: a work tour containing an escort stop and a later shopping tour, resolved into a timed, located, and mode-labelled sequence of activities and trips.

3.2.4 INPUTS AND THE SYNTHETIC POPULATION

The Pre-day models are driven by a set of ready-made inputs that are loaded once and held constant through the demand simulation. The foundation is the synthetic population: a full roster of individual agents, each carrying the socio-economic attributes that the choice models read, age, gender, occupational status, income, driving-licence and public-transport-pass holding, work-schedule flexibility, home and usual-activity locations, household composition, and vehicle availability. From these primary attributes the simulator derives the specific variables the models consume, such as income bands, age bands, and the household-level vehicle-availability category that gates the drive-alone option.

Alongside the population, the models read a description of the urban environment and of the cost of moving through it. Zone-level land-use data provide, for each of the 4,697 zones of the Greater Copenhagen system, the attraction variables that make destinations desirable: employment, retail provision, population, enrolment, parking, and central-area indicators. Level-of-service matrices provide, for every origin-destination pair and for the morning-peak, evening-peak, and off-peak periods, the travel times and costs of each mode, including in-vehicle, waiting, and walking times, fares, and car operating and parking costs. Time-dependent travel-time profiles give, at half-hour resolution, how car and public-transport travel times vary over the day, which is what allows the time-of-day models to trade off congested against uncongested departure windows. A final set of scenario parameters fixes the modelling conventions: the six travel modes, the six activity types, the definition of the peak periods, the maximum number of stops per half-tour, and the walking and cycling distance thresholds. Table 5 summarises these inputs and the role each plays in the models.

Table 5. Principal Pre-day inputs: the synthetic population and its attributes, zone land-use variables, origin-destination level-of-service matrices, time-dependent travel-time profiles, and scenario parameters

INPUT	CONTENT	ROLE IN THE MODEL
Synthetic population	Individual agents with socio-economic attributes: age, gender, occupation, income, licence, PT pass, work-schedule flexibility, household composition, vehicle availability, home and activity locations	Defines who travels and the attributes the choice models read
Zone land-use data	Per-zone attraction variables for the 4697 zones: employment, retail, population, enrolment, parking, central-area indicators	Makes destinations attractive; enters destination choice
OD level-of-service matrices	Mode-specific travel times and costs by OD pair for the AM-peak, PM-peak, and off-peak periods	Supplies the cost of travel to the mode/destination and time-of-day models

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

INPUT	CONTENT	ROLE IN THE MODEL
Time-dependent travel-time profiles	Half-hourly car and public-transport travel times over the day	Lets time-of-day models trade off congested vs. uncongested windows
Scenario parameters	Modes, activity types, peak-period definitions, maximum stops per half-tour, walk/bike distance thresholds	Fixes the modelling conventions and availability rules

For the Greater Copenhagen application, the population and its attributes are constructed from the Danish National Travel Survey and associated register and land-use data, and the level-of-service matrices are derived from the Danish transport network; the full national synthetic population comprises on the order of six million agents (Monteiro et al., 2022). Assembling this population in a form the simulator can consume is itself a substantial task, because each attribute must be recoded into the exact categories the estimated models expect, and it is one of the preparatory activities required for the Copenhagen application.

3.2.5 MODEL ESTIMATION AND CALIBRATION

The discrete-choice models are not specified by hand but estimated from observed behaviour. For Denmark the utility specifications were estimated by maximum likelihood on the Danish National Travel Survey (several tens of thousands of individual travel diaries recording the activities and trips performed on a surveyed day, together with the socio-economic characteristics of the respondents) combined with the level-of-service matrices of the national model, using the Biogeme estimation software (Bierlaire, 2020; Monteiro et al., 2022). Availability conditions are imposed during estimation and simulation so that alternatives enter a choice only when they are genuinely available to the individual: public transport is always available, walking and cycling only within their distance thresholds, and the drive-alone mode only for licence-holders in car-owning households.

FEDORA adopts this Danish model set rather than re-estimating it, and the work of Task 5.2 is to transfer, verify, and re-calibrate it for the Greater Copenhagen application. Transfer requires that every model in the chain be made consistent with the Copenhagen population coding, the Copenhagen zone system, and the six-mode configuration; verification requires that each model's implemented specification faithfully reproduce its estimated one; and calibration requires that the alternative-specific constants and scaling terms be adjusted so that the simulated aggregate outcomes (mode shares, tour rates by purpose, and time-of-day profiles) reproduce the observed ones. Because the models are held in editable Lua scripts, this re-calibration can be carried out and repeated without modifying the simulator itself, which is what makes iterative re-calibration practical.

3.2.6 THE WITHIN-DAY MODULE

Where the Pre-day module answers what an agent intends to do over the day, the Within-day module governs how that intention unfolds in real time, and it is the dynamic bridge between commuter demand and transport supply. It takes each agent's Day Activity Schedule and executes it as an event-driven process: rather than replaying the plan rigidly, it exposes the agent to the events that occur as the simulated day progresses (a change in network conditions, the arrival or delay of a service, a control signal from an operator) and lets the agent adapt. To this end each agent is equipped with behaviour models that are triggered by these events.

From the traveller's perspective the day alternates between two states: performing an activity and undertaking a trip. The distinction matters because different behaviour models apply in each. While an agent is performing an activity, incoming events are handled by pre-trip behaviour models, which govern decisions taken before a trip begins (most importantly, when to depart for the next activity, and by which mode). Once the agent is under way, events are handled by en-route behaviour models, which govern decisions taken during the trip, such as whether to divert to a different route in response to congestion. In the context of Task 5.2 it is precisely these

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

real-time adjustments (shifting a departure time, switching a mode before setting out, or re-routing mid-trip) that let the demand side respond to the interventions FEDORA studies.

The mechanism that makes this possible is an event chain. The module maintains the connections between events and behaviour models, so that when an event reaches an agent the appropriate model is invoked, and the agent's resulting decision itself generates new events, a changed departure time alters when the agent enters the network, which changes the load the network experiences, which may trigger further adaptations by other agents. This chaining is what allows the simulation to reproduce the complex, mutually dependent interactions among travellers, mobility services, and system-wide interventions, and to capture how demand shifts as the state of the network evolves. Because these adaptations are fed back to the Supply component and re-emerge as updated network conditions, the Within-day module closes the loop between the schedules generated by Pre-day and the network that must serve them. Figure 9 depicts this event-driven cycle and the behaviour models triggered in each state.

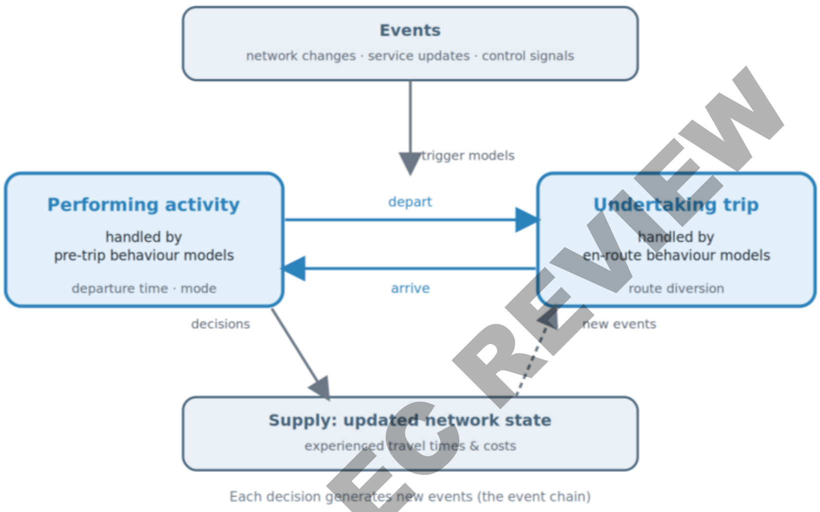


Figure 9. The Within-day event chain: the activity/trip state cycle, the pre-trip and en-route behaviour models triggered in each state, and the feedback of adapted decisions to the network.

3.2.7 APPLICATION TO THE GREATER COPENHAGEN AREA AND FEDORA EXTENSIONS

The framework described above is applied to the Greater Copenhagen Area in three stages and is then extended to represent the services and mechanisms that are specific to FEDORA. The starting point for the application is the Danish activity-based model set estimated and implemented by Monteiro et al. (2022). Task 5.2 adopts that model set, re-calibrates it for Greater Copenhagen, and consolidates it within a single FEDORA codebase, with the Pre-day and Within-day components and the Task 5.3 controllers.

The first stage is **validation in a controlled environment**. Before the model set is exercised on the full regional network, its demand-side functionality and the enhancements made to it are tested inside SimMobility's synthetic "Virtual City", a small, fully controlled network in which edge cases can be isolated, the event-chain mechanics of the Within-day module can be exercised, and baseline supply-demand interactions can be checked without the computational overhead of a full-scale run. This sandbox is where the transferred model chain is first made to run end to end and where structural errors are caught.

The second stage is **calibration for Greater Copenhagen**. The econometric choice models (destination, mode, activity-schedule, and time-of-day) are calibrated against empirical travel-survey and land-use data specific to the region, so that the synthetic population's attributes, the spatial accessibility of destinations, and the observed local travel behaviour are faithfully reproduced. This is the stage at which the adopted Danish model set is tuned into a credible representation of Copenhagen travel.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

The third stage is **synchronisation with the Task 5.3 controllers**. The demand-generation and Within-day execution flows are coupled with the dynamic mobility-controller modules developed in Task 5.3 (such as tradable-credit schemes and adaptive mobility-on-demand algorithms) so that demand-side responses feed into, and react to, supply-side control signals in real time. This two-way coupling is what turns the demand model from a stand-alone forecasting tool into a component of a closed-loop policy simulation: a pricing or credit signal issued by a controller changes the utilities the Within-day behaviour models evaluate, the agents adapt their departure times, modes, and routes, and the resulting change in demand is returned to the controller.

3.3 PRELIMINARY RESULTS

Work in the current phase has focused on assembling the demand-side modelling environment and exercising it end to end in SimMobility's controlled Virtual City environment. The points below report that first demonstration and set out the baseline generation and calibration that the next phase will produce, together with the integration with Task 5.3 that proceeds in parallel. Table 6 summarises where each of the main demand-side activities stands and the step planned next.

End-to-end demand run in the Virtual City: The demand chain has been exercised end to end in SimMobility's controlled Virtual City environment, a small synthetic network in which the behaviour models can be run in isolation and their output inspected before the full regional network is used. In this setting the Pre-day module already generates a complete Day Activity Schedule for the synthetic population, on the order of 165000 travelling agents, 467000 tours and 1.17 million individual trip legs, each assigned a mode, an origin and destination zone, and a time of day. Figure 10 shows that the mode-choice models populate the full twelve-mode menu, including the on-demand and pooled services that are central to FEDORA, and Figure 11 shows the tour structure the module produces: how many tours each agent makes and how many stops each tour chains together before returning home. Because this is the Virtual City synthetic scenario and not the Greater Copenhagen network, the shares are illustrative of the pipeline rather than calibrated demand. Establishing a reliable behavioural baseline for Copenhagen, and calibrating it against local travel data, remain the next-phase tasks.

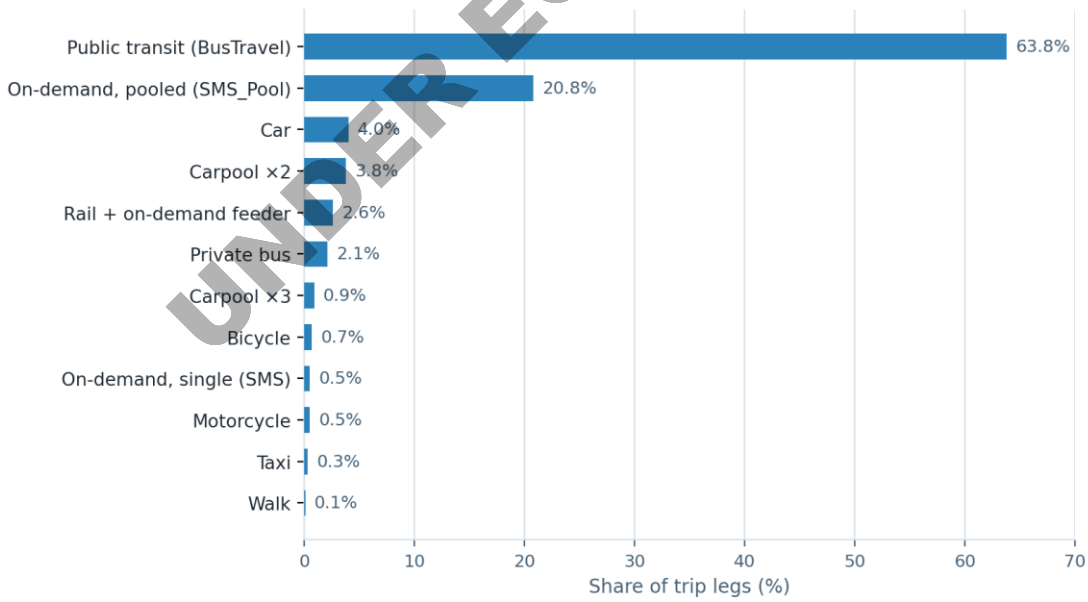


Figure 10. Virtual City demonstration: the mode-choice models assign every trip leg one of twelve modes, spanning conventional, shared and on-demand services. Shares are from the Virtual City synthetic scenario and are illustrative of the pipeline, not calibrated.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

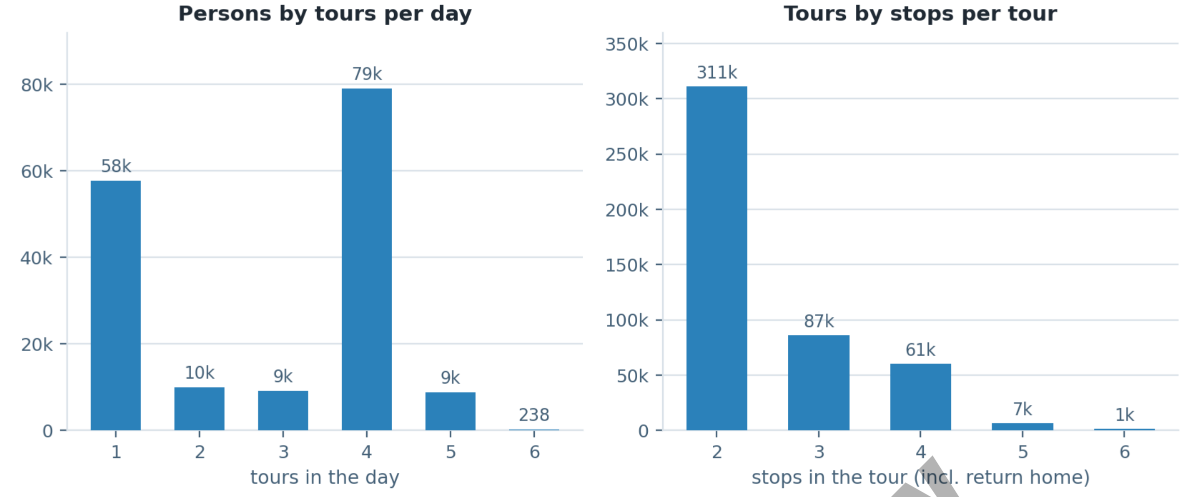


Figure 11. Virtual City demonstration: the tour structure produced by the Pre-day module, the number of tours each agent makes (left) and the number of stops chained within each tour, including the return home (right). Virtual City synthetic scenario.

Behavioural validation and calibration. Building on the end-to-end run reported above, the next step is the quantitative validation of the demand-side models in this controlled Virtual City environment. This small synthetic network allows the behaviour models and their enhancements to be exercised in isolation, edge cases to be identified, and the baseline supply-demand interactions to be checked without the computational overhead of the full regional network. Establishing a reliable behavioural baseline in this controlled setting is treated as a precondition for scaling to the Copenhagen application.

Baseline activity generation. A central objective of the next phase is to generate a full-day, disaggregate baseline of activity patterns for the Copenhagen population. By pairing the synthetic population's socio-economic characteristics with the zonal land-use data, this baseline is intended to reproduce the structure of observed daily schedules and to provide an un-disrupted reference against which the effect of FEDORA's policy scenarios can later be measured. The transfer and calibration of the model set required to reach it are the main demand-side tasks still to be undertaken.

Integration with the Task 5.3 controllers. In parallel, the demand-side components are being aligned with the mobility-controller logic developed under Task 5.3, so that demand-side adaptations can in due course be co-simulated alongside the supply-side intervention mechanisms. Ensuring this cross-framework interoperability is a necessary step towards the closed-loop policy experiments that the framework is ultimately intended to support.

Table 6. Status of the main demand-side activities and the step planned next for each

DEMAND-SIDE ACTIVITY	STATUS	PLANNED NEXT STEP
Demand-side modelling environment (model chain and inputs)	Being set up	A runnable Copenhagen demand configuration
Synthetic population and attribute coding	In progress	Population recoded into the categories the models expect
End-to-end demand run in the Virtual City	Demonstrated	Quantitative validation and a verified behavioural baseline before regional use

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

DEMAND-SIDE ACTIVITY	STATUS	PLANNED NEXT STEP
Transfer and verification of the model set	Next phase	The chain made consistent with the Copenhagen coding and zones
Baseline activity generation for Copenhagen	Next phase	A full-day, disaggregate reference for policy comparison
Calibration for Greater Copenhagen	Next phase	Simulated mode shares, tour rates and time-of-day matched to observed data
Integration with the Task 5.3 controllers	In progress	Cross-framework interoperability for closed-loop co-simulation
Enhancements for emerging services	Planned	AMoD, micromobility, EV fleets, MaaS, pricing and credits represented in the choice framework

3.4 CONCLUSIONS

This chapter presented the demand-side modelling framework developed under Task 5.2 and its current application to the Greater Copenhagen Area. The framework is built on SimMobility, an open-source, agent- and activity-based platform whose Mid-term simulator generates and adapts individual daily travel demand through a system of estimated discrete-choice models. Its Pre-day module constructs, for every agent, a complete Day Activity Schedule from a hierarchy of day-pattern, tour, and intermediate-stop choices linked by accessibility logsums, and its Within-day module executes and adapts those schedules in real time through an event-driven chain of pre-trip and en-route behaviour models. Together they provide the disaggregate, behaviourally rich representation of demand that FEDORA's solutions require.

The work reported here has established the demand-side modelling framework for the Greater Copenhagen Area, set out its Pre-day and Within-day components together with the model system and inputs they require, and defined the approach by which the models will be validated and calibrated. The transfer of the model set to Copenhagen, the generation of a baseline of daily activity schedules, and the calibration of the models against local data are the central demand-side tasks of the next phase.

The framework remains under active development. The choice models are still being refined and prepared for calibration against Copenhagen data, and the enhancements that will represent automated mobility on demand, shared micromobility, electric-vehicle fleets, Mobility as a Service, dynamic pricing, and artificial currencies are to be introduced into the choice framework. A key subsequent step is the two-way coupling of the demand side with the dynamic controllers of Task 5.3, which will allow demand responses and supply-side interventions to be co-simulated in a closed loop, the capability on which the evaluation of FEDORA's solutions ultimately depends.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

4 SIMULATION CONTROLLERS FOR EMERGING MULTIMODAL TRANSPORT SERVICES

4.1 TASK DESCRIPTION

This task aims to build a suite of generic mobility controllers and integrate this suite within the SimMobility framework to model, manage, and optimise emerging multimodal transport services. The primary objective of this task is to evaluate and validate how coordinated supply-side dispatching and demand-side management strategies can enhance overall transport network performance and reduce total energy consumption. By establishing comprehensive control architectures, the task establishes a simulation environment capable of modelling complex interactions between conventional public transit, on-demand services, and regulatory policy mechanisms. This task is intended to work in conjunction with Task 5.2, which focuses on the demand-side concerns of the agents within the SimMobility framework. The implementation methodology directly realises the task objective by organising the controller suite into three core functional pillars – (a) service supply, (b) demand management, and (c) market regulation.

Supply-side service execution is established through the generic Mobility Service Provider (MSP) controller. The MSP controller will model a diverse range of MSPs, including conventional and automated Mobility-on-Demand (MoD/AMoD) services. It will support multiple provider instances, each representing a specific operator (e.g., ride-hailing, on-call services, car sharing) with distinct operational characteristics. The MSP controller architecture is designed to handle a variety of driver modes, trip proposition and matching between drivers and passengers, as well as fleet rebalancing. Alongside the MSP controller, there are also dedicated fixed-route Bus and Train Controllers to simulate conventional public transport that capture physical road network interactions, vehicle capacities, and stop-level passenger dynamics.

On the demand management side, the Mobility as a Service (MaaS) controller bridges passenger choice models with service operators, thereby acting as an integration layer between the demand side (agents) and the supply side (MSPs). The MaaS controller provides two functionalities. With front-end integration, it provides a unified user interface, handling journey planning, ticketing, and subscription management. It aggregates information from various active MSPs to present a consolidated set of options to the traveller, which will feed into the available choice set of the agents and influence their travel demand, route choice and service utilisation decisions. With back-end strategy, it implements intelligent demand management strategies. By analysing the active state of all participating MSPs in the system, it optimises resource allocation and service availability, ensuring the unified platform operates effectively across diverse transport modes.

Finally, the methodology establishes market-based policy regulation levers via the Travel Credits Controller, incorporating a Regulator Agent and Market Platform to simulate dynamic congestion tariffs, budget allocations, and tradable credit schemes (TCS). The regulator sets pricing policies and travel credit allocation mechanisms. These are implemented within the simulation by augmenting the attributes of specific links (e.g., area-based or cordon-based pricing), forcing agents to account for monetary costs in their route and mode choice models. On the other hand, the market platform processes transactions, coordinates interactions with travellers, and evaluates pricing and incentive schemes. These interactions are recorded in real-time, allowing the system to evaluate the effectiveness of the incentive schemes and adjust pricing or credit distribution strategies dynamically.

Through this unified supply-demand-regulatory framework, Task 5.3 creates the necessary digital infrastructure to simulate multimodal transport systems and assess system-wide impacts of applied intervention policies.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

This chapter is organised as follows. Section 4.2 begins with a design description of the supply-side simulation mechanics and the controllers used in SimMobility. Section 4.2.1 details the generic MSP controller. Section 4.2.2 and Section 4.2.3 delve into the Bus and Train Controllers respectively, for public transport. Section 4.2.4 explores the MaaS controller and its integration into the overall SimMobility framework. Section 4.2.5 provides insights into the Travel Credits Controller responsible for demand management and policy regulation. Section 4.3 depicts Preliminary Results and Section 4.4 concludes with ongoing developments.

4.2 METHODOLOGY

Task 5.2 provides a fundamental description of the demand-side behaviour of the SimMobility platform, specifically the MT module. This section provides a description of the supply-side mechanics underlying SimMobility and subsequently outlines the design of the controllers which are to be integrated into the SimMobility framework.

SimMobility relies on mesoscopic modelling to simulate traffic in a network (Lu et al., 2015). Instead of tracking individual vehicle acceleration or signal-level interactions, mesoscopic models track individual vehicle entities as platoons or cohorts and evaluate their movement dynamics via aggregate, macroscopic traffic flow relationships like speed-density curves. The road network in SimMobility MT is organised into a hierarchy - namely links, segments, lane groups, and individual lanes. The basic computational processing unit is the road segment, which is split into a moving area and a queuing area (Figure 12). Vehicles in the moving area travel at a uniform speed dictated by a macroscopic speed-density curve based on segment traffic density. When arrival rates exceed the segment's sending capacity, a horizontal queue forms in the queuing area. Heterogeneous vehicle types (such as buses or trucks) are integrated into these continuous flow calculations using Passenger Car Units (PCUs) to dynamically adjust density and capacity consumption. The idea of a PCU is to convert a larger vehicle such as a bus or a truck into an equivalent number of passenger cars to analyse the larger vehicle's impact on traffic metrics.

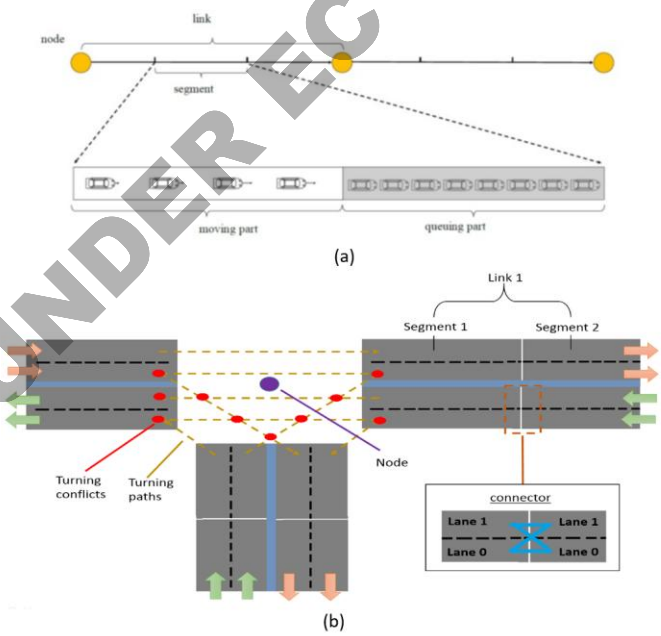


Figure 12. Link and segment structure depicting (a) moving and queuing areas and (b) nodes and turning points.

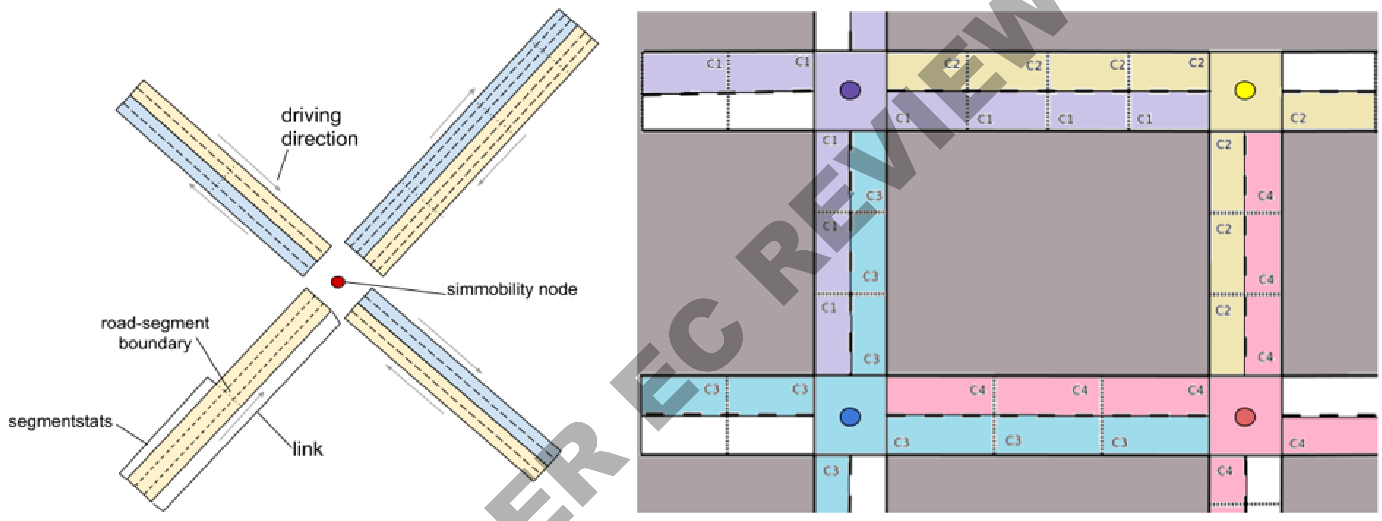
To simulate network dynamics, SimMobility groups upstream links entering a single node into a fundamental management entity called the **conflux** (Figure 13). A conflux is a data structure centred on a node in the road network. All links which *flow into the node* belong to the conflux of that node (yellow links in Figure 13Figure 12a). Each conflux acts as both a spatial data structure and an

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

updating agent. A conflux is thus composed of intersecting links, and each of these links is in turn composed of segments. Each road segment is represented by one or more data structures called **SegmentStats**. SegmentStats provide a data overlay on the original road segment and maintain a reference to it, allowing access to segment properties such as length and number of lanes. A segment is split into multiple SegmentStats wherever a potential queuing point exists, such as a bus stop. For each conflux link, the corresponding SegmentStats of each segment on that link are maintained in order of increasing distance from the conflux node, opposite to the link's direction of flow.

Each SegmentStats further contains a **LaneStats** structure for every lane in that segment, which tracks the vehicles currently occupying the lane and the number that are moving or queuing. Each SegmentStats also contains an additional LaneStats structure called **lane infinity**, which does not represent a physical lane. Instead, it acts as a loading queue for vehicles starting their trips. Finally, each conflux maintains a virtual queue for each of its links. This queue temporarily stores vehicles waiting to enter the link from an upstream link. Network execution relies on the layered SegmentStats and LaneStats structures, which track order, speed, density, and counts of moving versus queuing agents per lane.



(a) (b)

Figure 13. Illustration of a Conflux entity – (a) single conflux construction and (b) collection of confluxes.

Controllers are one of the central components driving SimMobility's supply side abstraction. The MT simulator framework allows for the modular implementation of different controllers, each in charge of some transport mode(s). SimMobility users can define a diverse range of settings for controllers in the simulation configuration XML file. This allows for the configuration of numerous multi-modal transport scenarios. Controllers run in parallel and are individually configurable by the user. In theory, individual agents have access to any of the enabled controllers. In practice, however, agent communication with controllers is largely limited to trip requests, which are sent to the controller associated with the predetermined mode specified for the trip. However, it should be noted that this behaviour changes with the integration of MaaS, as MaaS users will query information from all associated controllers through the MaaS controller.

4.2.1 MOBILITY SERVICE PROVIDER (MSP) CONTROLLER

To accurately represent emerging multimodal transport within urban environments, the simulation framework utilises a generic architecture of supply-side MSP Controllers. The fundamental architecture applies across human-driven (MoD) as well as (AMoD) service providers, and the underlying simulation logic and operational dynamics may differ slightly between the two modes (Nahmias-Biran et al., 2019). The simulation platform itself does not distinguish between MoD and AMoD service operators, but the

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

simulated operational characteristics determine which category the service provider falls under. Managed centrally by an entity called the MobilityServiceControllerManager, these MSP controllers orchestrate operations across multiple autonomous or human-driven service provider instances. In the remainder of this chapter, simulation entities (such as the MobilityServiceControllerManager) are identified using capitalized compound words (a format known as PascalCase), following standard object-oriented class naming conventions. The generic MSP controller architecture abstracts supply-side interactions through three mechanisms.

4.2.1.1 OPERATIONAL MODES AND DRIVER LIFECYCLE DYNAMICS

The system differentiates between two overarching driver operational modes. On-Hail Services operate through decentralized street hailing where drivers operate independently without centralized scheduling. The OnHailTaxiController actualizes on-hail services and serves primarily as an authorization manager for participating drivers. Separately, On-Call Services operate via centralized dispatch, and the OnCallController processes structured request queues, computes optimised trip sequences, and periodically dispatches commands to modify driver schedules. Drivers dynamically join or leave service providers via explicit subscription messages (driverSubscribeMsg, driverUnsubscribeMsg). The operational lifecycle of a driver is modelled as a finite state machine depicted in Figure 14.

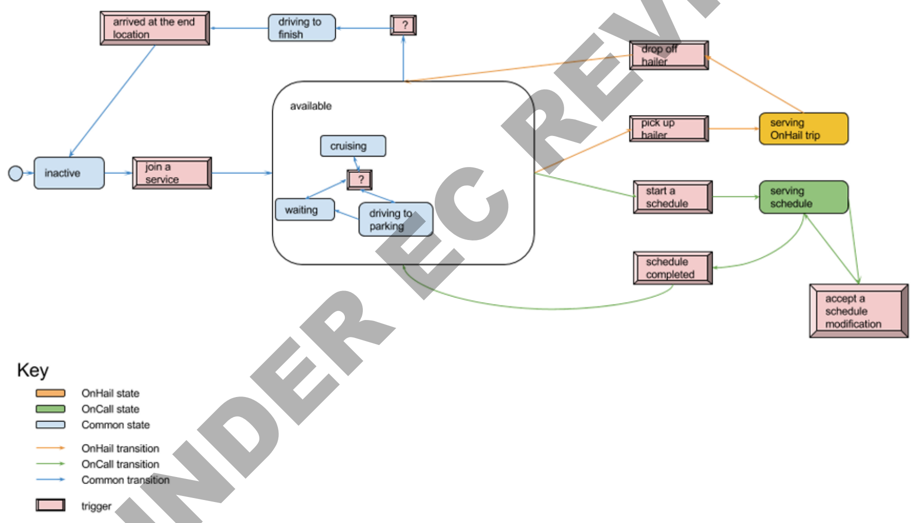


Figure 14. State transitions for MobilityServiceDriver moving across states.

The Inactive state is the initial state post-instantiation where the driver performs no actions. In the aggregated Free state - entered upon subscribing to a service provider - a free driver transitions between three sub-states: cruising (moving along the network links), waiting (parked at a stand or roadside), and Driving to Parking. The Serving state is activated when executing a trip. For On-Hail services, this involves picking up street hail passengers (serving OnHail trip). For On-Call services, this involves following controller-assigned schedules (serving the schedule) and carrying single or pooled passengers across pickup and dropoff sequences. The Driving to Finish state is entered when a shift ends or operation terminates. The driver routes to a designated end location (e.g., depot or garage) before unsubscribing and, if relevant, exiting the simulation.

4.2.1.2 TRIP PROPOSITION AND DRIVER-PASSENGER MATCHING

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

For OnHail trips, trip communication is limited to the driver and passenger, with the OnHailTaxiController simply aggregating drivers under a specific service provider via driver subscription. For OnCall trips, the sequence of interaction between users, controllers, and drivers follows a structured messaging protocol between the MobilityServiceDriver, MobilityServiceUser and MSP Controller entities (Figure 15). When a user issues a TripRequestMsg, the request includes operational parameters such as origin/destination TAZs, time windows, passenger counts, and an explicit evaluation procedure. As part of a two-step evaluation procedure, the controller evaluates potential schedules and issues a TripPropositionMsg containing specific pickup/dropoff time windows and dynamic pricing. The user choice model evaluates the offer before formally accepting (ACCEPT=1) or rejecting (ACCEPT= 0) the proposition. The default setting is direct assignment, where the user implicitly accepts any valid schedule matching their constraints, allowing immediate request-to-vehicle assignment. To satisfy these requests, the OnCallController iteratively executes generic or specialized matching algorithms within a configurable computation interval defined as scheduleComputationPeriod. The default matching algorithm is Greedy Assignment, whereby the controller services non-shareable, single-passenger requests by allocating the nearest available driver in the network.

Other matching algorithms are also available to be configured when instantiating the MSP controllers. The **Shared Assignment** algorithm uses graph-based request pooling to match up to two passengers to the same vehicle (Santi et al., 2014). It only considers vehicles that are completely idle (i.e., carrying no passengers). The algorithm first constructs a feasibility graph, where each node represents a request and an edge connects two requests if they can feasibly share a trip. Feasibility is determined by first estimating each request's desired travel time, defined as the travel time from pickup to dropoff assuming a dedicated vehicle is instantly available at the pickup location. The algorithm then evaluates possible pickup and dropoff sequences for the two requests (e.g., o1, o2, d1, d2). For each sequence, it computes the additional travel time experienced by each passenger relative to their desired travel times. A sequence is feasible if the extra delay for both passengers is below the configurable parameter extraTripTimeThreshold. If at least one feasible sequence exists, an edge is added between the two requests. After evaluating all request pairs, the algorithm applies maximum matching to the feasibility graph to select the set of request pairs to share vehicles. Each selected pair is then assigned to the idle vehicle closest to the first pickup location in the chosen sequence.

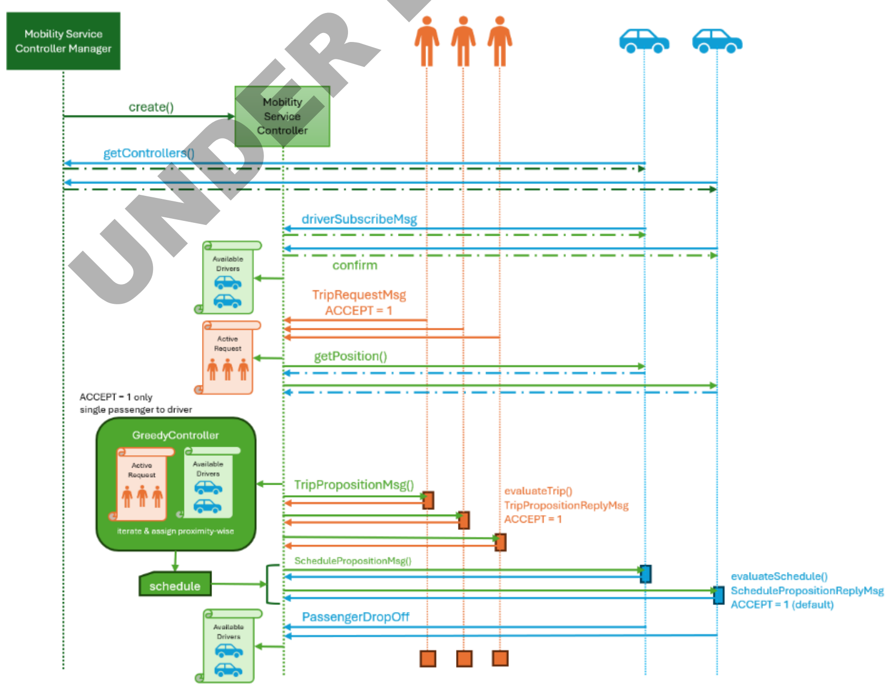


Figure 15. Message Protocol for OnCallController concerning Trip Proposition and Matching under Greedy Assignment.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

The **Frazzoli Assignment** algorithm (currently under testing) is another graph-based request pooling method to match drivers with passengers (Alonso-Mora et al., 2017). It evaluates pairwise shareability among incoming requests $\mathcal{R}$ and current vehicle statuses $\mathcal{V}$. A request r_1 and r_2 share an edge if an empty vehicle could serve both within pickup waiting time bounds (Ω) and maximum allowable travel delay constraints (Δ). Similarly, an edge $e(r, v)$ connects a request r to a vehicle v if v can pick up r without violating the delay tolerances of passengers already onboard. The algorithm explores the cliques of the RV-graph to assemble candidate trips T, where a trip represents a combination of requests. The RTV-graph builds edges $e(r, T)$ between requests and feasible trips, and edges $e(T, v)$ between candidate trips and vehicles capable of serving them. Computational efficiency is maintained by generating higher-order trips only if all constituent subtrips $T \setminus \{r\}$ are already determined to be valid in the graph. Finally, the batch assignment problem minimizes total trip delay and penalties for unserved requests. Initialized with a fast greedy solution Σ_{greedy}, the ILP formulation is solved iteratively:

$$\min_{\mathcal{X}} \sum\nolimits_{e(T_i,v_j)} c_{i,j}\, \epsilon_{i,j} + \sum\nolimits_{r_k \in \mathcal{R}} c_{\text{ko}}\, \chi_k \tag{31}$$

subject to the constraints:

$$\sum_{i \in \mathcal{I}^T_{V=j}} \epsilon_{i,j} \leq 1 \quad \forall v_j \in \mathcal{V}$$

$$\sum_{i \in \mathcal{I}^T_{K=k}} \sum_{j \in \mathcal{I}^V_{T=i}} \epsilon_{i,j} + \chi_k = 1 \quad \forall r_k \in \mathcal{R}$$

where $\epsilon_{i,j} \in \{0,1\}$ indicates whether vehicle v_j is assigned to trip T_i, $\chi_k \in \{0,1\}$ marks an unassigned request r_k, $c_{i,j}$ represents the total travel delay for trip T_i, and c_{ko} is a large constant penalty for ignored requests. Figure 16 illustrates the Frazzoli Assignment algorithm.

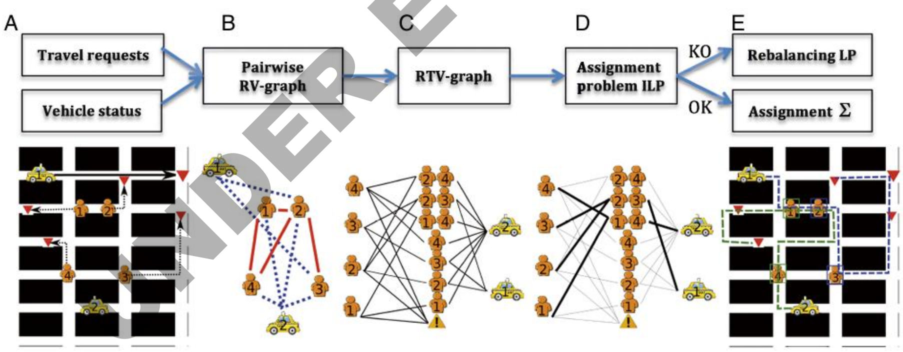


Figure 16. Overview of the Frazzoli Assignment algorithm

The **Incremental Sharing Assignment** algorithm also supports matching up to two passengers per vehicle. Unlike the Shared Assignment algorithm, it incrementally assigns new requests to vehicles that may already be serving passengers by updating their schedules. It also supports both shareable and non-shareable requests, as specified by the user via a parameter in each TripRequestMsg. Unserved shareable and non-shareable requests are stored separately. Drivers are classified into three groups - available (no assigned passengers), partially available (serving shareable requests and not yet at capacity), and unavailable (at capacity or serving a non-shareable passenger). At each computation interval defined by scheduleComputationPeriod, the algorithm executes three stages:

1. Shareable → Partially Available: Each unserved shareable request is considered for insertion into the schedule of each partially available driver. A request is assigned only if a feasible schedule exists that satisfies the waiting time and extra delay constraints for both the new and existing passengers.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

2. Non-shareable → Available: Remaining available drivers are matched with unserved non-shareable requests if they can reach the pickup location within the value specified by the maxWaitingTime parameter.
3. Shareable → Available: Any shareable requests still unmatched after Stage 1 are assigned to available drivers using the same waiting time criterion as Stage 2.

When all requests are non-shareable, only Stage 2 is executed. When all requests are shareable, Stage 2 is skipped.

Requests where the passenger rejects the controller's matching (ACCEPT=0) are processed after the main matching algorithm. To support these requests, the controller continuously monitors its performance, including the request assignment rate and queue length, and uses these to estimate the expected waiting time before a request is assigned. This estimate is returned to the user in a new TripPropositionMsg. If the user accepts the proposed wait, they resubmit the request and participate in the queue. After the main matching process, some ACCEPT=1 requests may still remain unassigned. Each such request maintains an age, which increases every scheduling cycle it remains in the queue. Once its age exceeds a threshold, the request is moved to an emergency queue for priority handling. Before processing ordinary requests, the controller first attempts to serve all requests in the emergency queue. It repeatedly runs the matching algorithm while progressively relaxing the maxWaitingTime and toleratedExtraTime constraints until either all emergency requests are assigned or no vehicles remain available.

4.2.1.3 FLEET REBALANCING

Rebalancing refers to proactively relocating available vehicles to areas where future trip requests are expected, reducing passenger waiting times. In an OnHail service, drivers independently decide where to wait. In an OnCall service, the controller can instead direct idle vehicles to selected locations. Every OnCallController owns a Rebalancer, which encapsulates the rebalancing logic. Whenever a new trip request is received, the controller allows the rebalancer to collect historical demand data for future decisions, in case this is needed by the rebalancing algorithm. The Rebalancer is called at specified intervals, defined by the parameter rebalancingInterval. Once the current frame reaches or exceeds nextRebalancingFrame, rebalancing is performed, after which nextRebalancingFrame is updated by adding rebalancingInterval to its current value. A frame here refers to a discrete simulation or processing step, with frames advancing as the system runs. The default rebalancingInterval is 720 frames (corresponding to 1 hour).

The framework supports three kinds of rebalancing – passive, reactive and predictive. Passive rebalancing essentially means that rebalancing is absent or deactivated. In reactive probabilistic rebalancing, the controller relocates idle vehicles to active Traffic Analysis Zones (TAZ) upon request arrival, with a defined probability. Whenever a trip request is received at a TAZ, the controller dispatches a vehicle to serve it and, with a specified probability (e.g., 50%), also instructs a randomly selected available driver to relocate to that TAZ by sending a CruiseTAZ item to the available driver.

In prediction-based rebalancing (currently under testing), the controller solves a linear program (LP) to minimize total travel costs when redispatching idle vehicles from surplus zones to predicted high-demand TAZs using CruiseTAZ schedule items (Marczuk et al., 2016). A demand prediction mechanism identifies zones where many requests are expected, and the controller proactively sends CruiseTAZ messages to available drivers to reposition them in those demand zones. The excess demand at station, $\hat{d}_i$, is defined as the difference between the demand and the number of vehicles at that station: $\hat{d}_i = d_i - v_i$. A negative value of $\hat{d}_i$ indicates a surplus of vehicles that can be rebalanced to other stations. The cost of relocating a vehicle from station to station is denoted by c_{ij}, which corresponds to the travel time between the two stations. Since travel times vary over time, these costs are continuously updated by SimMobility. The decision variable r_{ij} represents the number of empty vehicles rebalanced from station *i* to station *j*. For simplicity, the time index is omitted, yielding a time-invariant formulation:

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

$$\min_{r_{ij}} \sum_{i,j} c_{ij}\, r_{ij} \tag{32}$$

subject to:

$$\sum_j (r_{ji} - r_{ij}) \geq \hat{d}_i \quad \forall i$$

$$\sum_j r_{ij} \leq v_i \quad \forall i,$$

$$r_{ij} \geq \ 0 \quad \forall i, j$$

The objective minimizes the total rebalancing cost. The first constraint ensures that enough vehicles are relocated to satisfy the excess demand at each station, while the second prevents more vehicles from being dispatched from a station than are available.

4.2.2 BUS CONTROLLER

SimMobility MT exhibits an innovative simulation of bus stops (Lu et al., 2015). Because buses must enter bus stops and interact with passengers, their simulation logic differs from that of other vehicles. In the MT system, bus stops are modelled using a logic unit called a **virtual bus stop**, which is attached to the end of a road segment and encapsulates the relevant bus stop parameters (Figure 17). To place virtual bus stops at the correct locations in the network, the original road segments are split into multiple segments at each bus stop location. Bus movement through bus stops is modelled in four consecutive steps: lane selection, bus stop entry, passenger boarding and alighting, and rejoining the traffic flow. Interactions between buses and the surrounding traffic are also captured, including effects such as queueing spillback from the bus stop into the traffic segments.

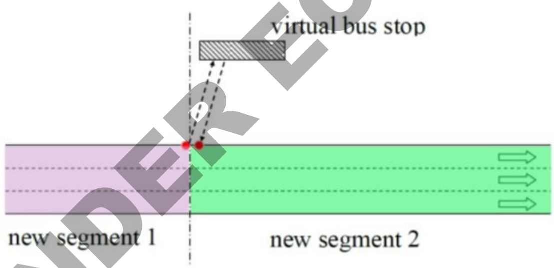


Figure 17. Segment split at bus stop location

Bus operations are coordinated by a central BusController, which dispatches bus entities onto the network based on predefined schedules and route topologies. When activated in the simulation configuration, the controller influences the bus fleet in three ways – it defines bus routes by specifying road segment sequences, it defines the visit schedule of the bus stops along the specified route and finally determines the bus dispatch frequency by specifying start/end windows and headway intervals. Based on these controller decisions, buses are generated at scheduled headway intervals to execute fixed spatial routes.

As previously mentioned, to capture physical dwell dynamics without disrupting the continuous road hierarchy, passenger stops are integrated directly into the supply network by splitting original road segments into new sub-segments. The supply engine accounts for PCU equivalents to reflect the physical impact of buses on traffic density, queue lengths, and segment sending capacities. Parallelized pedestrian agents walk along shortest paths at configurable speeds to reach virtual stops, enabling complete passenger-transit interaction loops.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

4.2.3 TRAIN CONTROLLER

The simulation of Mass Rapid Transit (MRT) services on a railway network is managed by the **TrainController** in SimMobility (Koh et al., 2017). Unlike buses, which operate on the same road network as other vehicles, MRT services require a separate rail network. Consequently, the TrainController manages a dedicated rail supply network for rail-based transport, while the other controllers operate on the road network. The use of separate supply networks effectively results in two separate but interconnected simulation environments within SimMobility: the rail simulator and the road-based simulator.

The rail simulator consists of 6 entities: pools, blocks, stations, platforms, schedules, and lines. A *pool* represents an area such as a depot where out-of-service trains wait. Trains are classified as active or inactive, with active and inactive pools maintained as ordered lists to enforce dispatching order. A *station* connects the rail simulator to the broader road-based multimodal supply network, allowing commuters to be transferred between the two. A *platform* permits passengers to board and alight trains and is associated with a train service in a specific direction. A *block* represents a section of track with specific geometric and operational properties and connects platforms. Blocks are the equivalent of segments in the rail network. A *line* is an ordered sequence of platforms and blocks through which trains operate. Each physical train service is represented by two lines, one for each direction, while a train's *schedule* specifies the subset of platforms it must serve and their expected arrival times. Stations define passenger walking-time parameters, while blocks specify train acceleration rates and maximum train speeds, and platforms specify minimum and maximum train dwell times. All these entities, except for schedules, are illustrated in Figure 18.

The rail simulator contains 2 agent types: *passengers* and *trains*. At each decision timestep, passengers and trains make decisions based on their attributes, the attributes of relevant entities, and instructions from the TrainController. For example, the TrainController can impose an additional speed limit on a train to maintain a safe distance from another train. Passenger movement follows a largely linear process. A passenger enters the rail simulator when transferred from the road-based simulator after tapping in at a station. A station-specific walking time is then assigned for reaching the platform. Passengers queue in FIFO order and board the first train with sufficient capacity; if capacity is unavailable, boarding is denied and the passenger waits for the next train. Once aboard, the passenger's movement between stations is controlled by the train agent. At the destination, the passenger is assigned a walking time to either the station exit gates or the next platform when transferring. The train sub-trip is completed once the passenger exits the system.

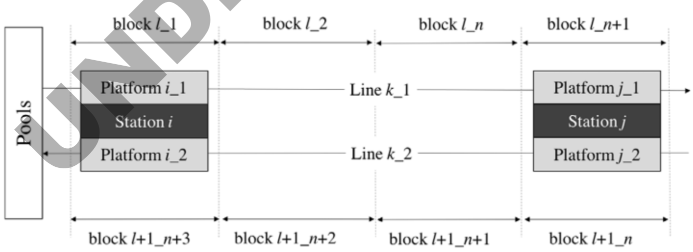


Figure 18. Entities in the Rail Supply Simulator

Train movement is more complex and is coordinated by the Service Controller. A train is activated by moving it from the inactive to the active pool and is assigned a schedule containing its departure time, service stations, and expected arrival times. Upon reaching a station, the train reports to the Service Controller, which checks any user-specified instructions, manages passenger boarding and alighting, calculates dwell time, and determines the train's subsequent behaviour. While the train travels between stations, its position relative to other trains and stations is monitored and its speed adjusted according to train movement models. At the end of a line, the train is typically assigned a new schedule, usually for the return journey, or, when service frequency decreases, is returned to the inactive pool.

Funded by the European Union

Project funded by Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra Swiss Confederation Federal Department of Economic Affairs, Education and Research EAER State Secretariat for Education, Research and Innovation SERI

A signalling system maintains safe distances between trains by ensuring sufficient braking distance and operating clearance. On shared tracks, it also prevents conflicting trains from occupying the track simultaneously. The simulator implements the signalling framework proposed by Keiji et al. (2015). Block entities provide the track-specific properties required by the model, including the effect of gradients on maximum acceleration. Train movement between stations is approximated using a trapezoidal speed-time profile, consisting of acceleration, constant-speed, and deceleration phases, with the corresponding kinematic equations implemented in Lua. These continuous equations are evaluated at discrete timesteps. Large timesteps can produce unrealistic deceleration due to position-estimation errors, so trains may be instructed to begin decelerating one timestep earlier when necessary. Stations may have several operational properties: *U-Turn*, allowing trains to reverse direction; *Bypass*, allowing trains to overtake, wait, or avoid obstructing other trains; *Interchange*, allowing passenger transfers between lines; and *Shared Track*, allowing multiple lines to use the same track and trains to switch lines. When a train arrives, its schedule and any TrainController instructions determine its departure behaviour, such as continuing to the next station, switching lines, or waiting. Passenger lists for the train, station, and platform are then updated according to the station's origin-destination matrix, and the resulting occupancy change is used to calculate dwell time. Additional delays are then incorporated, including those caused by operational constraints such as U-turns or user-defined holding policies, before the train departs.

4.2.4 MOBILITY-AS-A-SERVICE (MAAS) CONTROLLER

The MaaS controller bridges supply and demand by presenting users with available multimodal transport options and communicating their choices to the corresponding mobility controllers. Unlike other SimMobility service providers, MaaS acts as an aggregator rather than a direct mobility provider, sitting between users and the individual mobility service controllers. Its functionality therefore depends on communication with both users and service providers, requiring its integration into the existing supply infrastructure and the development of appropriate communication channels (Figure 19).

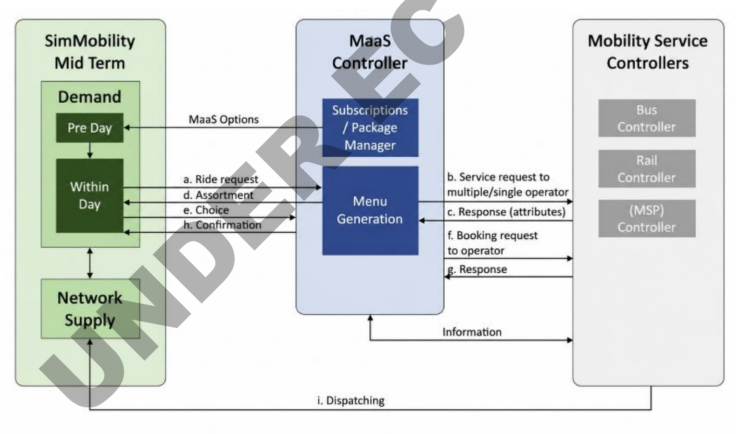


Figure 19. Integration of the MaaS controller in SimMobility

MaaS plans consist of collections of *discounts* and *allowances* bundled together on a per-mode basis, where each plan can feature distinct discounted prices and usage limits. An agent's MaaS account tracks their specific subscription plan and current allowance balance. Subscription decisions are determined during the Pre-day simulation of Task 5.2, which directly influences Within-day trip choices. When a subscribed agent requests trip options, the MaaS controller verifies their remaining mode allowance; if sufficient allowance exists, the corresponding discount is applied to the menu cost, whereas non-subscribers or agents with zero allowance receive standard, undiscounted costs. To maintain computational scalability across large-scale agent populations, MaaS accounts are managed via SQL tables using an on-demand loading policy, querying database entries only when an agent has an upcoming trip.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Menu construction begins once the MaaS controller receives a passenger's trip query message (MSG_TRIP_QUERY). The controller queries relevant service controllers, compiles incoming options into a personalized menu, updates costs according to the user's MaaS plan allowance, and applies a customizable Lua filtering function—such as filtering for the cheapest option per mode—to eliminate redundant routes before dispatching the menu (MSG_MENU_RESPONSE) to the user. To perform mode-choice cost estimation efficiently, origin-destination (O/D) pair cost metrics are precomputed in database tables. Public transit and multimodal modes utilise route-specific tables, whereas on-demand modes like taxis and AMOD use tables which are partitioned by time of day (morning peak, evening peak, and off-peak). Service controllers estimate wait times, travel times, and total costs using user-defined cost functions—incorporating variables such as base fares (B), distance rates ($fare_d$), time rates ($fare_t$), toll fees, and capacity sharing—and sum these attributes across individual legs for multimodal routes.

To facilitate these interactions across SimMobility's multi-threaded architecture, new message types and bi-directional communication channels connect passengers, the MaaS controller, and individual mobility service controllers over a shared message bus (Figure 20). Passengers initiate the process by sending an MSG_TRIP_QUERY message containing their origin and destination to the MaaS controller. The MaaS controller forwards this query to enabled mobility service controllers, which compute trip attributes and respond with MSG_MENU messages. After assembling and transmitting the final menu back to the passenger, who evaluates the options and replies with an MSG_MENU_RESPONSE message indicating the selected mode. For on-call services like MOD or taxi, the MaaS controller forwards the booking request to the appropriate supply controller, which assigns a driver, returns booking confirmations, and receives an MSG_TRIP_COMPLETE notification upon trip completion to update the user's account records.

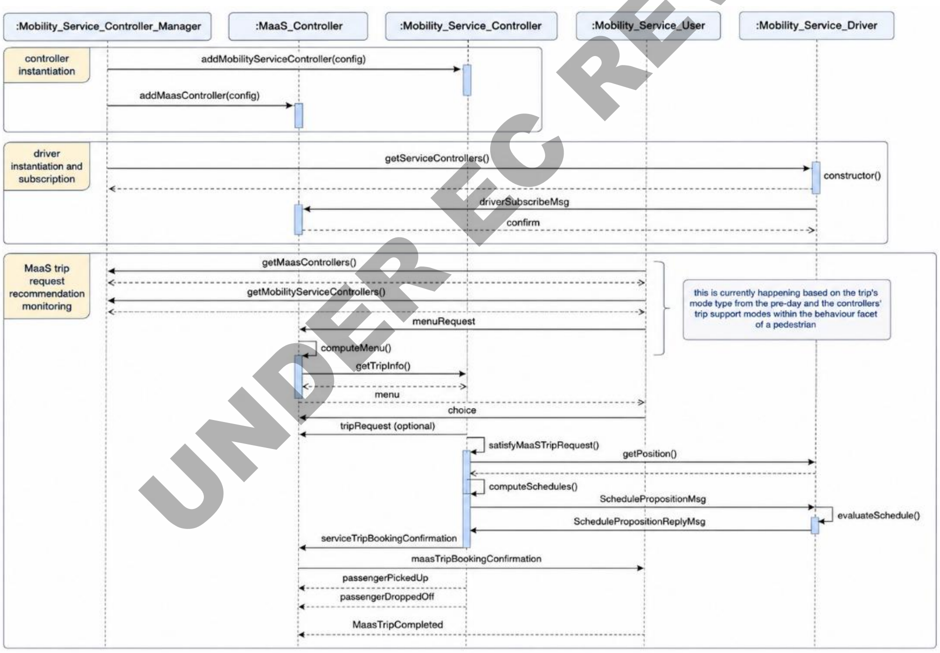


Figure 20. MaaS Controller message protocol

4.2.5 TRAVEL CREDITS CONTROLLER

The Tradable Credit Scheme (TCS) is a market-based demand management strategy that manages road congestion through dynamic mobility credits rather than monetary tolls, ensuring revenue neutrality and equity (ensuring that the scheme does not disproportionately burden certain groups of travellers). The system operates via three main entities: the regulator, travellers, and a market platform. To prevent market speculation and trading spikes, the regulator continuously distributes credits to travellers at a fixed rate r, enforcing a maximum lifespan l per credit that caps an individual account's balance at $l \cdot r$. Travellers use these credits

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

to pay dynamic upfront tolls $g(t)$ at trip departure. Those lacking sufficient credits must purchase the exact deficit immediately, while those holding a surplus may sell their entire balance directly to the regulator to lower peer-to-peer transaction costs, though simultaneous buying and selling is prohibited. Meanwhile, the central market platform logs all transactions, processes regulatory approvals, and updates the daily credit price p_d based on excess daily credit consumption. See Figure 21 for an illustration of interactions between these components.

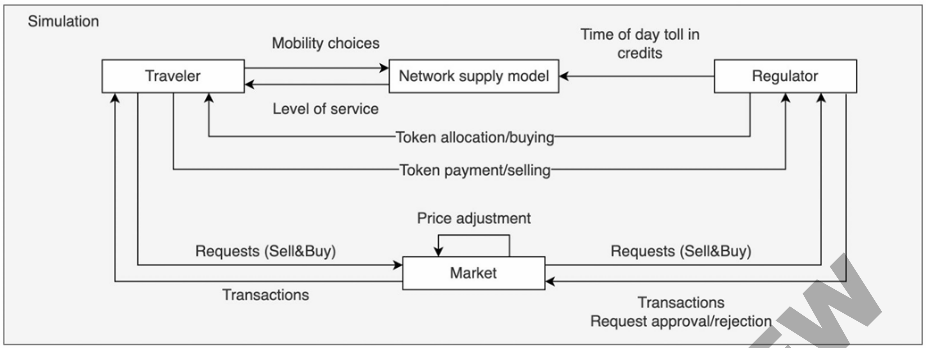


Figure 21. Interactions between travellers, regulator, and market

The implementation of this operational TCS system in SimMobility MT relies on 5 main object-oriented software components extending the platform's extensible base class structure (Liu et al., 2026). First, the generic *Account* base class encapsulates core attributes including account ID, account type, and owner ID. Second, the derived *TCS Account* class additionally maintains active credits and transaction logs as vectors, together with operational variables such as used, bought, and sold credit counters, transaction counts and types, and predicted individual selling profits. Third, *Transaction* objects are instantiated whenever credits are allocated, traded, or redeemed. Each transaction records its type (buy, sell, use, or allocation), buyer ID, seller ID, and the vector of credit items involved. Each *Credit* object is generated by the regulator and maintains its birth time and lifetime, which are continuously tracked during simulation to enforce automatic credit expiration. The *Regulator* class, which inherits from SimMobility's base Agent class, maintains a dedicated TCS account, enforces the continuous allocation rate, sets transaction fees, collects operational revenue, and approves or rejects traveller trade requests. Similarly, the *Market* class inherits from Agent and monitors transactions between travellers and the regulator, maintains transaction records, and adjusts daily credit prices based on aggregate supply and demand. Finally, each Person MT traveller (SimMobility agent) maintains an individual TCS account, with its balance updated following credit allocations, trip toll payments, purchases, sales, and expirations.

Communication between these components occurs through the SimMobility messaging system (Figure 22). At each simulation time step, transactions are processed sequentially. First, when the simulation reaches a predefined allocation interval, the Regulator distributes credits to all traveller accounts and sends the allocation records to the Market. Account balances are updated immediately, and credit expiration is checked within the same interval. When a trip begins, the traveller's account is checked against the departure toll. If insufficient credits are available, a purchase request is sent to the Regulator. Once approved, the required credits are issued, the Market records the purchase, and the traveller and regulator accounts are updated. For toll payments or credit sales, credits are transferred to the Regulator, and the transaction is recorded by the Market. Finally, at the end of each day, the Market aggregates the day's transaction data and adjusts the unit credit price for the following day.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

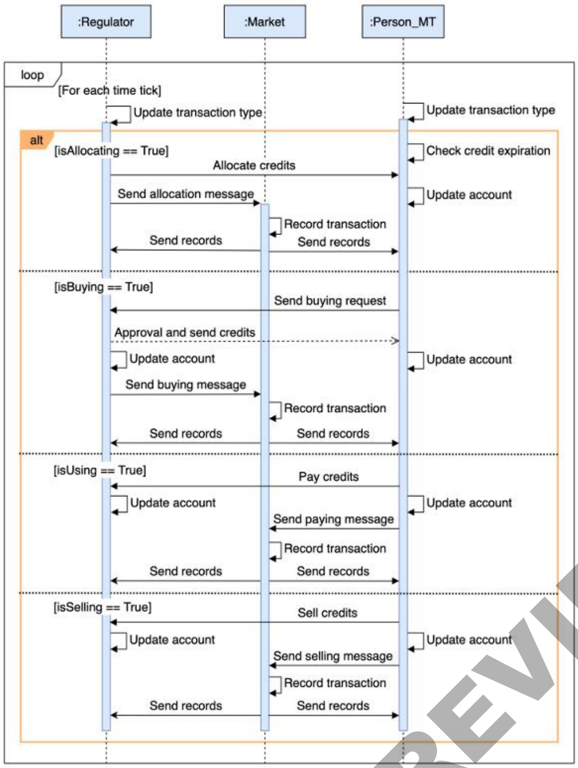


Figure 22. Message Protocol for the TCS framework in SimMobility

4.3 PRELIMINARY RESULTS

The progress of controller development is summarised in Table 7. The columns represent the successive stages of development, with progress generally flowing from left to right, from initial design through implementation, documentation, debugging, testing, and finally deployment in Copenhagen. The MSP Controller has completed design, implementation, documentation, and debugging in C++, with testing in SimMobility's Virtual City and deployment in Copenhagen remaining as ongoing work. The controller supports the generic MSP architecture, including driver management, trip matching, and fleet rebalancing. The MaaS Controller has likewise completed design, implementation, documentation, and debugging in C++. Its integration in Virtual City with the existing mobility service controllers and passenger demand-side processes has been implemented, while deployment in Copenhagen remains ongoing. The Travel Credits Controller has completed its design, implementation, and documentation, with debugging currently in progress. The controller implements the regulator, market, traveller accounts, credit allocation and expiration, transaction processing, and dynamic credit pricing mechanisms described in Section 4.2.5. The Bus Controller has completed all development stages listed in the table, including design, implementation, documentation, debugging, and testing. Its implementation supports fixed-route bus operations, virtual bus stops, passenger boarding and alighting, and interactions between buses and the surrounding road traffic. The Train Controller has similarly completed design, implementation, documentation, debugging, and testing. It provides the rail-specific simulation infrastructure required for MRT operations, including train schedules, station and platform interactions, signalling, and train movement. Deployment of both the Bus and Train Controllers in Copenhagen remains ongoing.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Table 7. Progress made for Task 5.3 Controllers

CONTROLLER	DESIGN	IMPLEMENTATION	DOCUMENTATION	DEBUGGING	TESTING IN VC	DEPLOYMENT IN CPH
MSP Controller	✓	✓	✓	✓	WIP	—
MaaS Controller	✓	✓	✓	✓	WIP	—
Pricing & Travel Credits Controller	✓	✓	✓	WIP	—	—
Bus Controller	✓	✓	✓	✓	✓	—
Train Controller	✓	✓	✓	✓	✓	—
MOD Controller	✓	✓	✓	✓	✓	—

To demonstrate SimMobility's capability to analyse and evaluate multi-modal transport systems, past research conducted on the Virtual City is presented next. As a benchmark for SimMobility's multimodal capabilities, key Virtual City findings are referenced originally produced by Basu et al. (2018); these historical results serve as a comparative baseline as we work to replicate and validate them within our consolidated and feature-expanded codebase.

Figure 23 depicts the various transport mode networks in Virtual City. Three scenarios are designed and simulated separately for comparison. The *Base Case* represents conditions without smart mobility services and includes single-occupancy cars, carpooling with two or three passengers, bus, MRT, traditional taxi, motorcycle, and walking (Walk), with modal availability determined by the study area. In the *Without Mass Transit* scenario, AMoD is introduced in place of Bus and MRT. AMoD behavioural preferences are assumed to match those of taxis, while its price is reduced by 40% relative to taxi fares, following trends found in the literature. Network travel times are provided to the day-to-day learning module and subsequently fed back to the Pre-day model to update individual choices. The *With AMoD* scenario is constructed by adding AMoD to all modes available in the Base Case. In addition to conventional door-to-door service, AMoD is used to provide first- and last-mile connectivity to MRT, limited to trips with origins or destinations at MRT stations.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

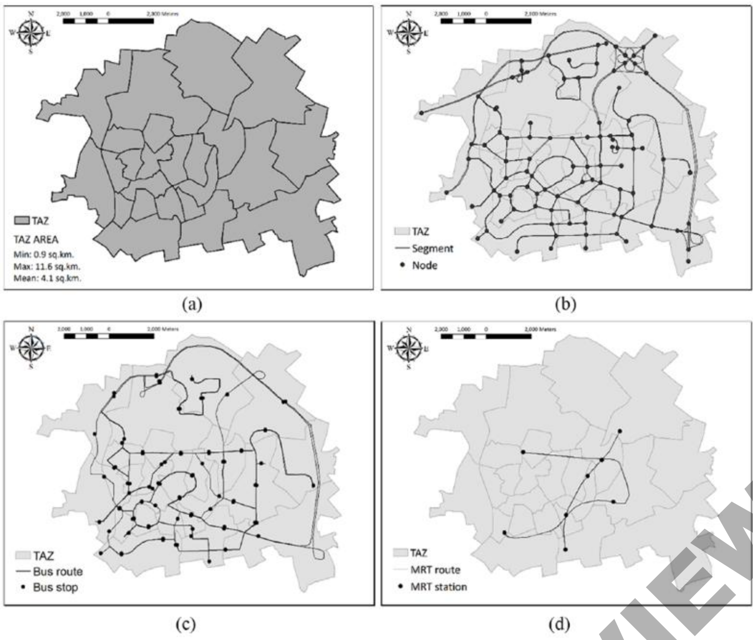


Figure 23. Maps of the Virtual City showing: (a) TAZ boundaries, (b) road network, (c) bus network, and (d) MRT network.

As mentioned in previous sections, a strong suit of SimMobility is the flexibility to adapt demand models as a part of the Pre-day simulation component (that generates the day activity schedule). Therefore, we can observe the changes in mode choice while comparing the Pre-day simulation results for the three scenarios. Figure 24 compares Virtual City segment density during morning peak and off-peak periods across all three scenarios. While congestion increases moderately in the With AMoD scenario relative to the Base Case, removing mass transit leads to severe network degradation. Furthermore, off-peak traffic clears as expected under both the Base Case and With AMoD scenarios, whereas network congestion continuously deteriorates without mass transit.

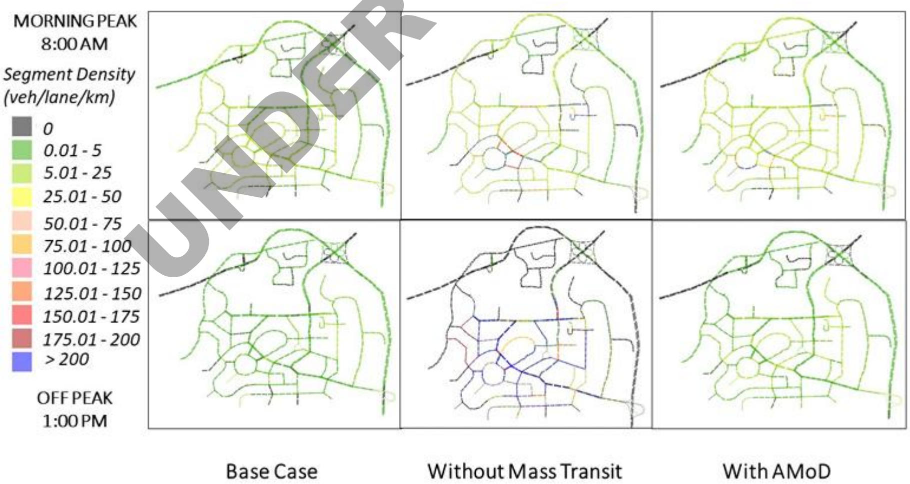


Figure 24. Congestion maps for peak and off-peak periods

Figure 25 highlights that removing mass transit drives the most significant surge in AMoD mode share. Across both future scenarios, lower tariffs make AMoD more attractive to travellers than traditional taxis. Additionally, AMoD adoption directly absorbs the decline in bus trips while simultaneously boosting MRT usage, underscoring how effective first- and last-mile connectivity strengthens mass

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

transit demand. The consistently high walk share aligns with Land Transport Authority (LTA) benchmarks across key Asian transit hubs where walking accounts for 20–30% of trips.

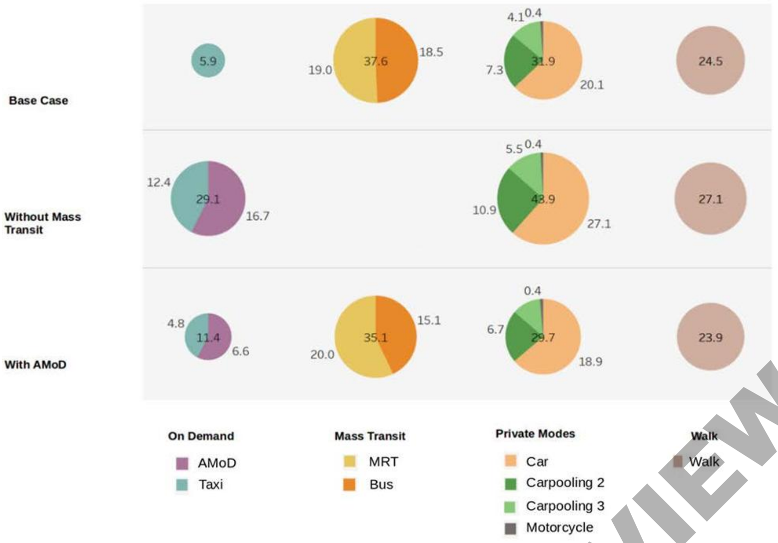


Figure 25. Changes in modal shares across the three scenarios

4.4 CONCLUSIONS

This chapter presented the design and implementation of the supply-side, demand-management, and market-regulation components being developed under Task 5.3. The work extends SimMobility's existing controller architecture with a generic Mobility Service Provider (MSP) framework, dedicated Bus and Train Controllers, a MaaS Controller, and a Travel Credits Controller. Together, these components provide mechanisms for modelling on-demand and conventional public transport services, multimodal service selection and coordination, fleet rebalancing, and market-based demand management. The implementation also establishes the necessary communication interfaces between passengers, mobility service providers, and regulatory components, while integrating the separate road and rail supply networks within the broader simulation framework.

These ongoing developments establish the digital infrastructure required by Task 5.3 to simulate complex interactions between transport supply, traveller demand, and regulatory interventions. The implemented controllers provide a flexible foundation for evaluating coordinated dispatching, multimodal service provision, and policy mechanisms within a common agent-based environment, thereby supporting the task's objective of assessing their impacts on overall network performance and energy consumption. The preliminary implementation status indicates that the core design and development have been completed for the principal controllers, while testing, debugging, and deployment activities remain ongoing for selected components.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

5 RESEARCH AND SCIENTIFIC INNOVATION

This chapter highlights the main methodological advances introduced in Tasks 5.1–5.3 and positions them with respect to the current state of the art in multimodal traffic simulation as well as agent-based behavioural modelling and simulation of emerging mobility services.

Task 5.1 advances the state of the art in multimodal traffic simulation and contributes to Expected Innovation EI4.1 through the development of simpi, a lightweight multimodal macroscopic traffic simulator designed to bridge the gap between computational efficiency and behavioural fidelity. Microscopic simulators provide detailed representations of traffic, but the computational cost often limits their use in optimisation, reinforcement learning and simulation-to-reality approaches. In contrast to that, simpi unifies theory of macroscopic traffic modelling and newly developed methodological components like multimodal vehicle representation, dynamic routing, realistic queue propagation, improved discharge dynamics and trip length modelling for replication of aggregate level network behaviour of microscopic simulations. The developed framework features a simulation engine that is computationally efficient for large-scale multimodal transport analyses, and it directly addresses the objectives of FEDORA to support future Sim2Real applications for AI-assisted traffic management. The initial validation against a large-scale, realistic SUMO-based model for the case study area of Athens city centre shows that significant computational savings (approximately 3 orders-of-magnitude faster) can be obtained while accurately replicating crucial network-level traffic behaviour as captured by the Macroscopic Fundamental Diagram, network accumulation and trip completion trajectories.

Task 5.2 advances the state of the art in agent-based, activity-based travel-demand modelling and contributes to Expected Innovation EI4.2, a suite of agent-based models for replicating the behaviours of different types of users that leads to accurate demand generation and calibration. Conventional four-step and trip-based models represent travel as anonymous aggregate flows and cannot capture how individual travellers reorganise their daily activity schedules in response to new services or pricing signals. Building on the open-source, agent- and activity-based SimMobility platform, Task 5.2 develops a flexible behaviour-simulation framework in which each traveller's day is represented as a coordinated schedule of activities and tours, and extends it with fundamental demand models sensitive to emerging mobility modes and operations (Automated Mobility on Demand, shared micromobility and electric-vehicle fleets) and to network-wide mechanisms (Mobility as a Service, dynamic pricing and artificial currencies), for heterogeneous users including low-income, low-accessibility and elderly travellers. By combining route, departure-time and mode-menu choice with trip and activity-planning models in a fully activity-based formulation, the framework can assess dimensions of user impact that aggregate models cannot. Its distinctive methodological contribution is the two-way coupling of demand with the dynamic supply-side controllers of Task 5.3, so that individual behavioural responses and control signals interact in a closed loop. Applied to the Greater Copenhagen Area on the open-source SimMobility platform and built for an open architecture that others can replicate, this work provides the behaviourally rich, agent-level demand representation on which the network-wide assessment of FEDORA's multimodal solutions depends.

Task 5.3 realises the core objectives of Expected Innovations EI 4.2 (Agent-based demand representation models) and EI 4.3 (Passenger Controllers for demand/supply simulation) within the SimMobility environment. By extending SimMobility's baseline architecture, Task 5.3 establishes a modular control suite structured across three functional pillars — service supply, demand management, and market regulation — to simulate complex interactions between multimodal transport services, traveller choices, and policy mechanisms. Alignment with EI 4.2 is achieved through the integration of the MaaS and Travel Credits Controllers, which interface directly with Task 5.2's agent choice models. The MaaS controller's front-end aggregates real-time multi-operator supply data to update agents' available choice sets, influencing route, mode, and service decisions. Concurrently, the Travel Credits Controller introduces market-based levers — including TCS and dynamic tariffs — that modify network link attributes, forcing individual agents to evaluate monetary and credit costs within their behavioural decision processes. To fulfil EI 4.3, Task 5.3 deploys

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

a suite of dedicated supply-side controllers that actively govern the dynamic interaction between passengers and transport services. The generic MSP controller handles trip matching, driver modes, and fleet rebalancing for MoD/AMoD fleets. Complementing this, fixed-route Bus and Train Controllers model physical road-rail interactions, capacity constraints, and stop-level passenger dynamics. Together, these passenger-service controllers provide the digital infrastructure needed to evaluate coordinated supply dispatching and demand-side management, directly supporting system-wide optimisations in network performance and energy efficiency.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

6 CONCLUSION

This deliverable presented the first implementation of the FEDORA Multi-modal Simulation Space through three tasks. Chapter 2 (Task 5.1) provided a lightweight multimodal traffic simulator in Python using a queue Store-and-Forward core together with a multi-class passenger-car-equivalent overlay, a dynamic turn ratio routing scheme, several discharge-side extensions and a trip age completion model, all benchmarked against a realistic sub network of the Athens city centre simulated by SUMO, and which reproduces its network accumulation, trip completion and Macroscopic Fundamental Diagrams while running roughly three orders of magnitude faster. Chapter 3 (Task 5.2) introduced the agent- and activity-based demand framework on the open-source SimMobility platform, with the demand chain exercised end to end in the controlled Virtual City environment, where the Pre-day module generates complete Day Activity Schedules for the synthetic population across the full mode menu, and with the transfer and calibration of the model set for the Greater Copenhagen Area identified as the next-phase priority. Chapter 4 (Task 5.3) delivered a modular controller suite organised across service supply, demand management and market regulation, comprising the generic Mobility Service Provider controller, the Mobility-on-Demand controller, dedicated Bus and Train controllers, the MaaS controller and the Pricing and Travel Credits controller, all designed, implemented and documented, with several already tested in the Virtual City.

These outputs contribute to Objective 4 of the FEDORA project, the realisation of a multi-modal simulation space that enhances the optimisation and decision-making capabilities of the services space solutions and enables future mobility scenarios to be created and assessed. As set out in Chapter 5, they contribute to Expected Innovations EI4.1, EI4.2 and EI4.3 and consolidate into Expected Results ER4.1 and ER4.2. More broadly, the simulation spaces can provide the environment in which network optimisation services and demand-management measures can be assessed at network scale. The agent-based demand models reported here establish the demand representation required for that assessment. In parallel, the computational efficiency of the supply simulation engine provides the basis for the training and transfer of AI-assisted control strategies.

Several improvements are expected to be implemented for the components reported in this deliverable. On the macroscopic simulation-engine side, class-dedicated lane modelling is being developed, the trip-age completion framework and the dynamic routing module are being extended and refined, and an automated calibration framework is going to be developed that will tune the simulator from microscopic simulation data in order to reduce manual calibration effort. Moreover, validation will be extended to networks incorporating additional transport modes (e.g., buses). On the demand side, the model set will be transferred and verified against the Copenhagen coding and zoning system, a full-day disaggregate baseline of activity patterns generated, and the models calibrated against local travel data so that simulated mode shares, tour rates and time-of-day distributions match observed behaviour. The choice framework will then be enhanced to represent automated mobility on demand, shared micromobility, electric-vehicle fleets, Mobility as a Service, dynamic pricing and artificial currencies. On the controller side, the remaining debugging and Virtual City testing will be completed and all controllers deployed on the Copenhagen network.

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

7 REFERENCES

Chapter 2

1 Aboudolas, K., Papageorgiou, M., & Kosmatopoulos, E. (2009). Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. Transportation Research Part C Emerging Technologies, 17(2), 163–174. https://doi.org/10.1016/j.trc.2008.10.002

2 Cassidy, M. J., & Bertini, R. L. (1999). Some traffic features at freeway bottlenecks. Transportation Research Part B Methodological, 33(1), 25–42. https://doi.org/10.1016/s0191-2615(98)00023-x

3 Cassidy, M. J., & Rudjanakanoknad, J. (2005). Increasing the capacity of an isolated merge by metering its on-ramp. *Transportation Research Part B Methodological*, *39*(10), 896–913. https://doi.org/10.1016/j.trb.2004.12.001

4 Daganzo, C.F. (1999). Remarks on Traffic Flow Modeling and Its Applications. In: Brilon, W., Huber, F., Schreckenberg, M., Wallentowitz, H. (eds) Traffic and Mobility. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-60236-8_7

5 Transportation Research Board (2016). Highway Capacity Manual, 6th Edition: A Guide for Multimodal Mobility Analysis. Washington, DC: The National Academies Press.

6 Greenshields, B.D. (1935) A Study of Traffic Capacity. Highway Research Board, 14, 448-477.

7 Harders, J. (1968). *Die Leistungsfähigkeit nicht signalgeregelter städtischer Verkehrsknoten*. Bonn: Bundesministerium für Verkehr, Abteilung Straßenbau.

8 Lin, S., De Schutter, B., Xi, Y., & Hellendoorn, H. (2011). Fast Model Predictive Control for urban road networks via MILP. IEEE Transactions on Intelligent Transportation Systems, 12(3), 846–856. https://doi.org/10.1109/tits.2011.2114652

9 Lin, S., De Schutter, B., Xi, Y., & Hellendoorn, H. (2012). Efficient network-wide model-based predictive control for urban traffic networks. Transportation Research Part C Emerging Technologies, 24, 122–140. https://doi.org/10.1016/j.trc.2012.02.003

10 McKendrick, A. G. (1925). Applications of mathematics to medical problems. Proceedings of the Edinburgh Mathematical Society, 44, 98–130. https://doi.org/10.1017/s0013091500034428

11 Portilla, C., Espinosa, J., & De Schutter, B. (2020). A multi-class urban traffic model considering heterogeneous vehicle composition: An extension of the S model. Transportation Research Part C Emerging Technologies, 115, 102613. https://doi.org/10.1016/j.trc.2020.102613

12 Siegloch, W. (1973). Die Leistungsermittlung an Knotenpunkten ohne Lichtsignalsteuerung. Straßenbau und Straßenverkehrstechnik, Heft 154. Bonn: Bundesminister für Verkehr, Abteilung Straßenbau.

13 Tsitsokas, D., Kouvelas, A., & Geroliminis, N. (2021). Modeling and optimization of dedicated bus lanes space allocation in large networks with dynamic congestion. Transportation Research Part C Emerging Technologies, 127, 103082. https://doi.org/10.1016/j.trc.2021.103082

Funded by the European Union

Project funded by
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Swiss Confederation
Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

14 Tsitsokas, D., Kouvelas, A., & Geroliminis, N. (2023). Two-layer adaptive signal control framework for large-scale dynamically-congested networks: Combining efficient Max Pressure with Perimeter Control. Transportation Research Part C Emerging Technologies, 152, 104128. https://doi.org/10.1016/j.trc.2023.104128

15 Webb, G. F. (1985). Theory of Nonlinear Age-Dependent Population Dynamics. CRC Press.

Chapter 3

1 Adnan, M., Pereira, F. C., Azevedo, C. L., Basak, K., Lovric, M., Raveau, S., Zhu, Y., Ferreira, J., Zegras, C., & Ben-Akiva, M. E. (2016). SimMobility: A multi-scale integrated agent-based simulation platform. Paper presented at the 95th Annual Meeting of the Transportation Research Board, Washington, D.C.

2 Auld, J., Hope, M., Ley, H., Sokolov, V., Xu, B., & Zhang, K. (2016). POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations. Transportation Research Part C, 64, (pp. 101-116).

3 Bierlaire, M. (2020). A short introduction to PandasBiogeme (Technical report). Lausanne: École Polytechnique Fédérale de Lausanne.

4 Bowman, J. L. (1998). The day activity schedule approach to travel demand analysis (Doctoral dissertation). Cambridge, MA: Massachusetts Institute of Technology.

5 Bowman, J. L., & Ben-Akiva, M. E. (2000). Activity-based disaggregate travel demand model system with activity schedules. Transportation Research Part A, 35 (1), (pp. 1-28).

6 Horni, A., Nagel, K., & Axhausen, K. W. (Eds.). (2016). The Multi-Agent Transport Simulation MATSim. London: Ubiquity Press.

7 Li, S. (2015). Activity-based travel demand model: Application and innovation. PhD thesis, National University of Singapore.

8 Lu, Y., Adnan, M., Basak, K., Pereira, F. C., Carrion, C., Saber, V. H., Loganathan, H., & Ben-Akiva, M. E. (2015). SimMobility mid-term simulator: A state of the art integrated agent-based demand and supply model. Paper presented at the 94th Annual Meeting of the Transportation Research Board, Washington, D.C.

9 Monteiro, M. M., Seshadri, R., & Lima Azevedo, C. M. (2022). Estimating and implementing a Danish activity-based model in SimMobility. Paper presented at the Annual Transport Conference at Aalborg University (Trafikdage), Aalborg, Denmark. Retrieved from https://journals.aau.dk/index.php/td.

10 Nahmias-Biran, B., Oke, J.B., Kumar, N., Basak, K., Araldo, A., Seshadri, R., Akkinepally, A., Lima Azevedo, C., Ben-Akiva, M. (2019). From Traditional to Automated Mobility on Demand: A Comprehensive Framework for Modeling On-Demand Services in SimMobility. Transportation Research Record, 2673(12), 15-29.

11 Viegas de Lima, I., Danaf, M., Akkinepally, A., Lima Azevedo, C., & Ben-Akiva, M. (2018). Modelling framework and implementation of activity- and agent-based simulation: An application to the Greater Boston Area. Transportation Research Record, 2672 (49), (pp. 146-157).

Chapter 4

1 Lu, Y., Adnan, M., Basak, K., Pereira, F. C., Carrion, C., Saber, V. H., ... & Ben-Akiva, M. E. (2015, January). SimMobility mid-term simulator: A state-of-the-art integrated agent-based demand and supply model. In *94th annual meeting of the transportation research board, Washington, DC* (p. 17).

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

2 Nahmias-Biran, Bat-hen, Jimi B. Oke, Nishant Kumar, Kakali Basak, Andrea Araldo, Ravi Seshadri, Arun Akkinepally, Carlos Lima Azevedo, and Moshe Ben-Akiva. "From traditional to automated mobility on demand: a comprehensive framework for modelling on-demand services in SimMobility." *Transportation Research Record* 2673, no. 12 (2019): 15-29.

3 Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S. H., & Ratti, C. (2014). Quantifying the benefits of vehicle pooling with shareability networks. Proceedings of the National Academy of Sciences, 111(37), 13290-13294.

4 Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., & Rus, D. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proceedings of the National Academy of Sciences, 114(3), 462-467.

5 Marczuk, K. A., Soh, H. S., Azevedo, C. M., Lee, D. H., & Frazzoli, E. (2016, July). Simulation framework for rebalancing of autonomous mobility on demand systems. In MATEC Web of Conferences (Vol. 81, p. 01005). EDP Sciences.

6 Koh, K., Azevedo, C. L., Basak, K., & Ben-Akiva, M. E. (2017, October). Simulation of demand and supply of urban rail in a multimodal environment. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC) (pp. 647-654). IEEE.

7 Keiji, K., Naohiko, H., & Shigeru, M. (2015). Simulation analysis of train operation to recover knock-on delay under high-frequency intervals. Case Studies on Transport Policy, 3(1), 92-98.

8 Liu, R., Argyros, D., Jiang, Y., Ben-Akiva, M. E., Seshadri, R., & Lima Azevedo, C. (2026). Assessing the impacts of tradable credit schemes through agent-based simulation. Journal of Intelligent Transportation Systems, 1-23.

9 Basu, Rounaq, Andrea Araldo, Arun Prakash Akkinepally, Bat Hen Nahmias Biran, Kalaki Basak, Ravi Seshadri, Neeraj Deshmukh, Nishant Kumar, Carlos Lima Azevedo, and Moshe Ben-Akiva. "Automated mobility-on-demand vs. mass transit: a multi-modal activity-driven agent-based simulation approach." *Transportation Research Record* 2672, no. 8 (2018): 608-618.

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI