

# SimMobility Short-term: An Integrated Microscopic Mobility Simulator

## Carlos Lima Azevedo

Massachusetts Institute of Technology  
Cambridge, Massachusetts 02139, USA  
Phone: +1 6174522482  
Fax: +1 6172531130  
email: [cami@mit.edu](mailto:cami@mit.edu)

email: [katarzyna@u.nus.edu](mailto:katarzyna@u.nus.edu)

## Harold Soh

National University of Singapore  
21 Lower Kent Ridge Rd, Singapore  
Phone: +65 65167367  
email: [hsoh@comp.nus.edu.sg](mailto:hsoh@comp.nus.edu.sg)

## Neeraj Milind Deshmukh

Singapore MIT-Alliance Research &  
Technology  
1 Create way, 138602 Singapore  
Phone: +65 66011634  
Fax: +65 6684 2118  
email: [neeraj@smart.mit.edu](mailto:neeraj@smart.mit.edu)

## Kakali Basak

Singapore MIT-Alliance Research &  
Technology  
1 Create way, 138602 Singapore  
Phone: +65 66011634  
Fax: +65 6684 2118  
email: [kakali@smart.mit.edu](mailto:kakali@smart.mit.edu)

## Balakumar Marimuthu

Singapore MIT-Alliance Research &  
Technology  
1 Create way, 138602 Singapore  
Phone: +65 66011634  
Fax: +65 6684 2118  
email: [balakumar@smart.mit.edu](mailto:balakumar@smart.mit.edu)

## Tomer Toledo

Technion - Israel Institute of Technology  
Haifa, Israel  
Phone: +972 48293080  
Fax: +972 48295708  
email: [toledo@technion.ac.il](mailto:toledo@technion.ac.il)

## Simon Oh (\*Corresponding author)

Singapore MIT-Alliance Research &  
Technology  
1 Create way, 138602 Singapore  
Phone: +65 66012828  
Fax: +65 6684 2118  
email: [simonoh.phd@gmail.com](mailto:simonoh.phd@gmail.com)

## Li-Shiuan Peh

Massachusetts Institute of Technology  
Cambridge, Massachusetts 02139, USA  
Phone: +1 6173248428  
Fax: -  
email: [peh@csail.mit.edu](mailto:peh@csail.mit.edu)

## Katarzyna Marczuk

National University of Singapore  
21 Lower Kent Ridge Rd, Singapore  
Phone: -  
Fax: -

## Moshe E. Ben-Akiva

Massachusetts Institute of Technology  
Cambridge, Massachusetts 02139, USA  
Phone: +1 6172535324  
Fax: +1 6172531130  
email: [mba@mit.edu](mailto:mba@mit.edu)

Word count: 5999 (Manuscript) + 5 Figures + 1 Tables = 7499 words

1    **ABSTRACT**

2    In this paper, we present the development of an integrated microscopic mobility simulator, SimMobility  
3    Short-Term (ST). “Integrated” as its models, inputs and outputs, simulated components and code-base are  
4    integrated within a multi-scale agent- and activity-based simulation platform capable of simulating  
5    different spatial-temporal resolutions and account for different levels of travelers’ decision making.  
6    “Microscopic” as both the demand – agents and its trips – and the supply –trip realization and movements  
7    on the network – are microscopic, i.e. modeled individually. Finally, “Mobility”, as it copes with the  
8    multimodal nature of urban networks and the need for the flexible simulation of innovative transportation  
9    services such as on-demand and smart mobility solutions. This paper follows previous publications that  
10   describe SimMobility’s overall framework and models. SimMobility is an open-source multi-scale  
11   platform that considers land-use, transportation, and mobility-sensitive behavioral models. SimMobility  
12   ST aims at simulating the high-resolution movement of agents (traffic, transit, pedestrians and goods) and  
13   the operation of different mobility services and control and information systems.

14   This paper presents SimMobility ST modelling framework and system architecture, reporting its  
15   successful calibration for Singapore and its use in several scenarios of innovative mobility applications.  
16   We also show how detailed performance measures from SimMobility ST can be integrated with a daily  
17   activity and mobility patterns simulator. Such integration is crucial to accurately model the impact of  
18   different technologies and service operations at the urban level, as the identity and preferences of  
19   simulated agents are maintained across temporal decision-scales ensuring the consistency and accuracy of  
20   simulated accessibility and performance measures of each scenario.

21   **KEYWORDS**

22   microscopic simulation, traffic simulation, activity-based models, mobility, communication networks

23

24

## 1 1 INTRODUCTION

Microscopic traffic simulation applications are now part of the daily transportation planning and operation realities. In order to explore and evaluate complicated future transportation scenarios, we need to conduct experiments. Physical experiments can be conducted in order to understand how the different options work together on a small scale. For the entire urban area, however, a simulation model is the only viable option. Microscopic traffic simulation models have been widely used to test different road network and ITS solutions. They aim at replicating detailed vehicle motions and interactions by modelling agent decisions such as route choice, accelerations, decelerations, and lane-changes. These models are implemented as synchronous applications that update the kinematic parameters of each entity (driver-vehicle units, public transportation, management systems and even pedestrians) at every simulation time step. Similar to other transportation simulators, the design of microscopic models is based on a demand and supply equilibrium representation. Traffic demand input is formulated either by defining it in terms of input flows and turning proportions at intersections or, for larger networks, in terms of origin–destination (OD) matrices that will rely on route-choice models for network assignment [1]. An example of these simulators are AIMSUN [4], VISSIM [5], Q-Paramics [6], Tansmodeler [7], ARTEMIS [8], CORSIM [9], DRACULA [10], HUTSIM [11], Integration [12], MITSIMLab [13], SUMO [14], Cube DynaSIM [15]. The first four simulation tools belong to the short group of integrated platforms available for fast implementation and that have been successfully used in a variety of transportation projects, accounting for a share of more than 70% of practitioners and researchers' preference [16].

In microscopic traffic simulation, the supply implementation relies on the specification of the network configuration, the traffic management algorithms and the driving behavior model. From the initial models developed in the 50's for car-following behavior [2], traffic microscopic simulation models now include multiple detailed behaviors and have reached a high level of maturity not only among the research community but also at the practitioners regular [1]. For a comprehensive review on all driving behavior components used in simulation, the reader should refer to [1], [3].

Commercial simulation tools have devoted a large share of their new features to enhanced interfaces, visualizations and, sometimes, calibration frameworks. At the same time, they have managed to integrate dedicated traffic control modules, public transportation and pedestrian simulation [4], [5], into their core architecture. On the other hand, the simulation research stream has proposed several innovative driving behavior models [11],[24] integration of communication technologies or even emissions models [20]. Some of these features can also be found in case studies using commercial software, but typically by means of coupling external modules to the main simulation tool, eventually compromising computational performance and interactive behaviors. Furthermore, several recent efforts in the research community can be found in the development of sophisticated activity-based modelling frameworks and their integration in simulation platforms that focus on individual's entire day activity pattern. Comprehensive agent-based modelling structures developed so far can be listed as TRANSIMS [17], MATSim [18] and FEATHERS [19]. These applications model the multiple choices and behaviors of a single agent during a day and have shown to better represent the interactions and dependencies of individual mobility. However, computational efficiency of such platforms are always been a major concern as they usually deal with the entire population of an area which is synthetically generated.

The above development streams have raised several challenges in the integration of complex mobility and transportation models within microscopic simulation. The next generation of simulators should include activity-based frameworks, integrated formulations with higher level, consideration of alternative modes such as on-demand mobility and autonomous vehicles, advanced and flexible driving behavior models, the possibility to easily integrate innovative transportation services such as vehicle-to-vehicle communication or logistic services.

This paper presents SimMobility Short-term (SimMobilityST), a new open-source microscopic mobility simulator integrated in a multilevel simulation platform. Designed using an agent-based framework, SimMobilityST aims at simulating the movement of agents (traffic, transit, pedestrians and goods) and the decisions and operation of control centers, within one day. It considers individual travel

1 behavior in detail using an activity-based formulation. In the next two sections both the modelling  
 2 framework and the system architecture are presented. Its modular structure is described in more detail  
 3 with a focus its multiple components along with the recent applications that showcase its flexibility in the  
 4 simulation of different and innovative mobility scenarios. Finally, the successful calibration process of the  
 5 demand-supply parameters of SimMobilityST for the city of Singapore is described.  
 6

## 7 **2 SIMMOBILITY**

### 8 **2.1. Overall Framework**

9 SimMobility is designed as three primary modules segmented according to timeframes [27]. The short-  
 10 term model functions at the operational level; it simulates movement of agents at a microscopic  
 11 granularity (within day) and is presented in more detail in the next section. The mid-term (day-to-day)  
 12 simulator handles transportation demand for passengers and goods; it simulates agents' behavior  
 13 including their activity and travel patterns; and it shares several mobility decisions with the short-term  
 14 level (e.g., with the route-choice model) [38]. The long-term (year-to-year) model captures land-use and  
 15 economic activity, with special emphasis on accessibility. It predicts the evolution of land use and  
 16 property development and use, determines the associated life cycle decisions of agents, and accounts for  
 17 interactions among individuals and firms. The high-level design of SimMobility is shown in **FIGURE 1**.

18 SimMobility must therefore include all the key mobility-related decisions that people make in  
 19 their everyday life. These decisions may be personal decisions of households or the commercial decisions  
 20 of firms [26]. To support this level of representation, SimMobility is based on the concept of agent-based  
 21 or micro-simulation. Representation of individuals as agents in the model is necessary to simulate how  
 22 people will react in the future to new infrastructures, new technologies, innovations in system  
 23 management and policy changes.

24 The SimMobility framework is fully modular in the sense that each of the levels can run  
 25 independently and only access the other levels when necessary. The key to multi-scale integration in  
 26 SimMobility is a single database model that is shared across all levels. Every agent exists and is  
 27 recognized at all levels simultaneously, and information is used according to each level's needs; in this  
 28 way, behaviors will remain consistent and, even if run separately, the impacts from one level's model will  
 29 be propagated to the others gracefully.

30 In previous work, these models have not been fully integrated. While there is limited interaction  
 31 of outputs, there is no internal coherence. SimMobility is unique in that the same pool of agents is used  
 32 across all timeframes. For further details on SimMobility's overall framework and Mid-Term and Long-  
 33 Term Simulators, the reader is referred to [27],[38] and, [50] respectively.  
 34

### 35 **2.2. SimMobility Short-term Framework**

36 SimMobilityST is an agent-based, multimodal microscopic simulator where agents' movements are  
 37 captured at a very fine resolution (up to 100 milliseconds). SimMobilityST comprises three main  
 38 components. The Microscopic Movement module is responsible for advancing drivers, pedestrians and  
 39 goods on the transportation network according to their respective behavioral and decision models. The  
 40 Control and Management module simulates the control centers, such as traffic and parking control, bus  
 41 control, rail control, autonomous fleet control, logistic control, etc. The outcomes of these control actions  
 42 will influence an agent's movement decisions, path choices and other related decisions in the movement  
 43 simulator. Within the Control and Management Module, different control centers may be considered and  
 44 replicated. At the current state of the simulator, the service controller and the traffic management  
 45 controller are operational as described below. On-going efforts are being made in the development of the  
 46 freight controller using detailed freight and logistics data [47]. The third component is the

1 Communication network simulator, which simulates agent-to-agent communications. The information can  
 2 be passed from one agent to another via the mobile communication or via vehicle-to-vehicle  
 3 communication or maybe via vehicle-to-infrastructure communication. The Communication network  
 4 simulator is responsible for simulating the physical communication network (for example, a wireless  
 5 network), and agents simulated within microscopic traffic network will use this simulated network to pass  
 6 information between them. This will help the agents to get the realistic communication network, which  
 7 will handle the message delivery delay or coverage.  
 8

### 9 **3 MODULES**

#### 10 **3.1 The Microscopic Traffic Simulator**

11 The structure of the Microscopic Movement module is detailed in **FIGURE 2**. The virtual world is  
 12 populated during the initialization phase, after which the simulation receives the control  
 13 information/action plan at every time step. Two kinds of behaviors are simulated: high level (travel)  
 14 decisions, such as route choices. Lower level (movement) decisions, such as car-following and lane-  
 15 changing, occur while the agent is in movement. While the agent's position is updated at every time step,  
 16 the movement-related decisions only takes place when specific events occur.

#### 17 **Demand input**

18 Instead of the traditional Origin-Destination matrix definition used in the demand formulation of traffic  
 19 microscopic simulation models, SimMobilityST (and its higher level counterparts) uses an activity-based  
 20 demand formulation in the form of activity-schedules. In such approach, trip chains are generated by  
 21 individual daily schedules instead of aggregated traffic specific matrices. Such data can be obtained  
 22 directly using mobility and goods survey data or using a pre-day model, such as the one integrated in the  
 23 SimMobility mid-term framework [38]. The pre-day model consists of an activity-based modelling  
 24 system formulated as interconnected discrete choice models representing choices at distinct dimensions.  
 25 This pre-day model development follows the Day Activity Schedule approach [25], [26] which focuses  
 26 decisions related to daily activity and mobility. There are three different hierarchies in the system: day  
 27 pattern level, tour level and intermediate stop level. Each level consists of several models, such as mode  
 28 choice or departure tie choice. For the full specification of the pre-day in SimMobility Mid-term model,  
 29 the reader is referred to [27]. The output is an activity schedule and the trip chains for each agent in the  
 30 simulation.

31 Within SimMobilityST, agents are then moved as per the planned trip chain. However, the  
 32 realized trips can be changed during the simulation if specific circumstances such as high congestion,  
 33 incidents, public transportation interruptions or any control or information provision are observed.

34 For each agent's sub-trip (a multimodal trip can have several sub-trips) generated in the  
 35 simulation its role is assigned (pedestrian, passenger and driver) and its role-specific characteristics are  
 36 generated (e.g., aggressiveness, look-ahead distance, reaction times, etc). For each vehicle-based trip, an  
 37 individual vehicle is generated. On-going work is being carried out to allow vehicle ownership and  
 38 parking models to integrate a consistent vehicle generation model with unique identifiers. The generated  
 39 vehicles are then assigned vehicle attributes (e.g., type and drive train) based on configurable  
 40 distributions.

#### 41 **Static Supply input**

42 The network in SimMobilityST is composed by a road network layer, a pedestrian network layer and a  
 43 public transportation layer. The road network layer is composed by (1) Nodes, (2) Links, with (3)  
 44 Segments, (4) Polylines, and (5) Lanes. Connectivity attributes is assured by (6) Lane Connectors, (7)  
 45 Turning Groups, (8) Turning Paths and (9) Conflict Points.

46 Nodes represent intersections or source/sink for trip chains. They are used for Link definition in

route-choice and for the detailed characterization of intersections. Links are directional roads that connect Nodes and are composed by Segments. The latter are road sections with uniform geometric characteristics (speed limit, design speed, grade). Each segment as a fixed number of Lanes, each with its specific lane rules (lane-changing regulation and use privilege). Polyline determine the shape of the Segment and lane connectors define the connectivity between Segments. At the Node level, Turning Groups, Turning Paths and Conflict Points can be defined. Turning groups define the connectivity between Links, while the Turning Paths Additionally, (10) Road Items are point specific features that can be added to the network to represent items on the road to which drivers must respond (e.g.: traffic lights, bus stops, etc). On-going work is being carried out to extend this framework with parking infrastructure.

## Driving behavior

The core traffic model of SimMobilityST is based on MITSIM, an open-source microscopic traffic simulation application developed by the Massachusetts Institute of Technology. MITSIM moves vehicles according to route choice, acceleration and lane-changing models. The acceleration model captures drivers' response to neighboring conditions as a function of surrounding vehicles motion parameters. The lane-changing model integrates mandatory and discretionary lane-changes in single model. Merging, drivers' responses to traffic signals, speed limits, incidents, and tollbooths are also captured. The driving behavior models implemented in MITSIM are those estimated by [22] and [21]. The MITSIM lane-changing model was later enhanced by [24], for the specific purpose of integrating latent plans in the lane selection process, namely in urban arterials and in freeways with a large number of lanes.

Several additional enhancements were made to the MITSIM original driving behavior: an enhanced reaction time formulation capable of explicitly model reaction time and perception delays for each person in a detailed and flexible manner as introduced (see [28] for further details); lateral movement during lane-change was also included. For the current implementation, the lateral speed is kept constant during the lane-change, but the implementation of a sine function for lateral acceleration similar to the one proposed in [30] has been initiated.

Finally, the design of a dedicated intersection behavior model, based on the conflicts technique has also been implemented. The intersection behavior starts once the intersection is visible to the vehicle. First, the driver identifies the intersection regime (no rules, priority, or controlled). If the intersection is not controlled, the subject vehicle identifies the neighboring vehicles and the conflicting vehicles, and proceeds with a gap-acceptance based model that accounts for intersection-specific priorities (if any).

## Travel behavior

Within SimMobilityST, changes in planned trip chains have to be considered. As the simulation is running, the agents need to find the routes for their trips and transform the activity schedule into effective decisions and execution plans. Agents may get involved in a multitude of decisions, not constrained to the planned set of destination, mode, path and departure time depending upon the network and their state in the simulation cycle [38]. In the current implementation agents can re-route (as drivers or public transportation passenger) in the presence of congestion or the provision of control and information. Route choices are based on a probabilistic model that captures the impact of travel times and biases toward routes that use freeways over urban streets. The impact of real-time information on routing decisions is captured by a route-switching model in which informed drivers re-evaluate their pre-trip route choices based on the traffic conditions observed en-route. For perfect model integration with higher level simulators, the route-choice model used in SimMobility ST is the same as the one used in the SimMobility Mid-Term framework and can its details be found [38], [27].

## Pedestrian movements

The pedestrian behavior model focuses on the problem of how a pedestrian makes crossing-related decisions at different levels and at different points of times, when she/he walks along a given path to

1 destination. Specifically, a crossing choice module is designed to determine where a pedestrian crosses  
2 the road, along a given path. A crossing timing decision module is designed to control when a pedestrian  
3 starts crossing the road, once she/he reaches a crossing point.

4 **Commodity movements**

5 The movement of freight vehicles is typically considered in microscopic traffic simulation models by  
6 means of adapting driving behavior parameters for heavy vehicles [31] or by coupling dedicated external  
7 applications with the simulator [32]. Teo et al. [33] extended these traditional approaches by integrating  
8 the simulation of freight movements, logistics decisions and traffic within an agent-based simulation.  
9 SimMobilityST is the first microscopic simulator that integrates commodity specific movements with  
10 detailed traffic models. Similar to individual trip-chain input, SimMobilityST allows commodity specific  
11 shipments. The commodity entity was specified for this purpose and freight drivers are assigned tours  
12 based on the commodities to deliver during the simulation period. A default tour generation model was  
13 developed but this will be relaxed and linked to a freight operator controller, which typically represents a  
14 carrier. Freight vehicles and drivers will then be assigned to a specific freight operator and a set of  
15 delivery stops specified as Road Items in the network. Decisions on the freight vehicle tours can be made  
16 by the freight operator controller or the driver himself. The design and integration of all these entities  
17 within the core models of SimMobilityST is still under development at the stage of writing this document.  
18

19 **3.2 Control and Operation Systems**

20 **Traffic Management Controller**

21 The Traffic Management Controller mimics the traffic and information control system in the network  
22 under consideration. A wide range of traffic control and route guidance systems can be simulated. These  
23 include intersection controls, ramp control, freeway mainline control, lane control signs, variable speed  
24 limit signs, portal signals, variable message signs and in-vehicle route guidance. The Traffic Management  
25 Controller can represent different designs of such systems with logic at varying levels of sophistication  
26 (pre-timed, actuated or adaptive) by means of a flexible configuration input.

27 Control devices can be either link-wide (such as Variable Speed Limits) or lane specific (e.g.;  
28 lane use regulation). They are represented by Road Items and are characterized by their location, type and  
29 visibility distance. Their logic is implemented directly by the Traffic Management Controller and the  
30 analyst will need to code its logic through external scripting files (in Lua language). An example of tested  
31 implementation in the current state of SimMobilityST development is the SCATS-like algorithm for  
32 traffic signals [29] and an innovative time slot-based algorithm for the coordinated management of  
33 intersections for autonomous vehicles [45].

34 **Service Controllers**

35 Service controllers are the central control point responsible for operation a specific mobility service. It  
36 holds relies on static information as well as real-time information and communicates with the simulated  
37 vehicle operators/drivers to send them instruction at various situations. In the current development of  
38 SimMobilityST, the bus transportation framework and controller are already implemented. We currently  
39 have two setting of the controller: the Bus Controller and Smart Mobility Controller.

40 The Bus Controller is responsible for the scheduling and dispatching of buses, for keeping track  
41 of individual arrival times, and for deciding on transit control strategy. Along with the Bus controller,  
42 other features were implemented in SimMobilityST: the Bus Driver Agent will be responsible for routing  
43 the bus on fixed route, the bus movement near the bus stop, real time passenger count and dwell time  
44 calculation. The bus movement for example, includes mandatory lane-changing maneuvers in order to  
45 reach the lane where the stop is, depending on a distance-to-stop threshold. When a bus is far than bus-to-

stop visibility the driver agent may make discretionary lane-changes according to the same logic that applies to other drivers with a preference toward the lane that contains the bus stops; finally, the Passenger/Pedestrian Agents will also interact in the bus framework as they are responsible for realizing their individual bus route choice, boarding choice and alighting choice.

SimmobilityST also provides mechanisms for simulating emerging technologies that are yet to be widely available, for example, mobility-on-demand services. Such feature was achieved with the development of uniform interfaces between a Smart Mobility Controllers and the different models in SimMobilityST. The Smart Mobility Controller relies on third-party code that can run separately from SimMobilityST and mimics the operation of fleets, from regular taxi to Uber-like systems. This code interacts with agents and entities in SimMobility in runtime, and a few features have been made accessible to it.

Two application of the proposed Smart Mobility Controller can be found already in the literature: an Autonomous Mobility on Demand (AMOD) service, which provides one-way car-sharing with self-driving electric vehicles and has emerged as a promising solution for autonomous urban transportation [48]; and Flexible Mobility on Demand (FMOD), which provides personalized and optimized services to travelers in real-time with flexibilities both on the operator and traveler side [49]. These technologies are designed to deal with the recent trends that emphasize more flexibility through the use of shared-ride services and integration of multimodal mobility options.

### 3.3 The Communication Network Simulator

Many simulators broaden their applicability by allowing customized interactions with third-party components, and SimMobility is no exception. In addition to traditional library-based extension, SimMobilityST provides a TCP socket integration layer that allows other software systems to interact with a running SimMobilityST simulation. This is primarily used in [34] to overlay Android emulators running transit-related applications or “apps” onto existing SimMobility agents, thus providing accurate location information to the apps. In return, the apps provide more realistic within-day re-routing, creating a feedback loop which should optimize the system.

The choice to communicate over TCP sockets has several advantages. First, it requires minimal changes to existing third-party software systems –usually only a small communication module is needed. Second, it facilitates interactions between a larger number of simulators. In the example just discussed, SimMobilityST connects to a running instance of a network simulator (ns-3), which it uses to provide accurate timing and packet loss information for messages sent between Android clients. Finally, the use of TCP sockets provides a stable, cross-platform means of interaction with clearly defined boundaries.

## 4 SYSTEM ARCHITECTURE

### 4.1 Overall framework

SimMobilityST applies several design heuristics to make modelling and development easier for a heterogeneous user base. First, entities are isolated from each other, and can only interact through properties that are shared among them. This isolation is achieved through the use of agent-based simulation techniques. Second, the simulator is location-agnostic regarding agents. In other words, an agent's interface does not change depending on where it is in relation to other networks (except when MPI is enabled). Third, SimMobilityST's time-step is indivisible –agents are assumed to all tick forward at once. Finally, SimMobility is hierarchical and provides sensible defaults. A good example of this behavior is the use of trip-chains, which can be “filled in” with more information as the agent's trip progresses. If an agent does not have a route for a given segment of the trip chain, one can be estimated for it.

SimMobilityST is designed as a hybrid software framework, including both event-driven messaging and discrete time-step simulation. Heterogeneous time-steps are supported, allowing coarse-grained software agents such as traffic signals to interact efficiently with fine-grained agents like drivers.

1 Ultimately, SimMobilityST was designed with accuracy and performance as its two primary goals. To  
 2 this end, it includes a parallel and a distributed component, which will now be described in turn.  
 3

#### 4 **4.2 Parallel computing**

5 SimMobilityST features a robust, straightforward approach to massive parallel scalability that was  
 6 designed to take advantage of the processing power of modern hardware. The majority of computations  
 7 performed by SimMobilityST are done by entities as they update their internal state. This process is  
 8 performed once per entity per time tick. Entities with similar time step resolutions are grouped together  
 9 into “Workers” that manage the update process for a given thread.

10 Entities can generally ignore the Worker that they assigned to, as all communication through  
 11 other agents is done using buffer-backed variables. These variables use an internal buffer to allow lock-  
 12 less communication with any entity on any Worker. In addition, Entities will be automatically added to a  
 13 Worker when their start time of their first trip arrives, and will be removed once the final trip's destination  
 14 has been reached. Typically, entities only interact with their Workers when requesting a manual  
 15 migration. This can occur when the agent crosses an MPI boundary (described in Section 4.3), or if some  
 16 kind of spatial optimization such as the mid-term's “conflux” structure is desired.

17 This use of buffers runs counter to traditional logic of using mutex-based locking for parallel  
 18 communication. Ultimately, the conventional approach exhibits several systemic and non-trivial  
 19 problems. Primarily, it limits the repeatability of the simulation by introducing an unacceptable amount of  
 20 non-determinism. Entities are constantly reading and reacting to the internal state of each other, and  
 21 performing these reads in parallel leads to different orderings for each simulation run. Attempting to solve  
 22 this by ordering entity updates will remove some of the benefits of parallelism, which is particularly  
 23 detrimental to entities that are not sensitive to update order. A secondary issue with lock-based  
 24 synchronization is the heavy toll it places on many-cored systems, especially in traffic simulation with its  
 25 high degree of inter-agent data dependency. As the number of discrete processing units on a system  
 26 increases, more and more performance is lost to overhead.

27 An additional channel for communication is event-driven messaging. Here, entities can register  
 28 for messages to be delivered for a series of given events, such as node arrivals or agent deactivations.  
 29 These messages will be triggered during a time tick, gathered and sent at the end of that time tick, and  
 30 received at the beginning of the next time tick. Thus, messages inherently incur a one-tick delay, although  
 31 it is possible to reduce this to zero if certain conditions hold (e.g., passengers on a bus can receive zero-  
 32 delay messages). Although event processing is inherently single-threaded, it can lead to large  
 33 performance gains by allowing entities to deactivate their update phase until a given message arrives.  
 34 Furthermore, entities can register for events and still maintain their update phase, thus allowing for a  
 35 hybrid of event-driven and discrete time-stepped simulation.  
 36

#### 37 **4.3 Distributed computing**

38 Once simulations encompass a large enough number of entities, it is inevitable that parallel simulation  
 39 will reach a point of diminishing returns. At this point, the simulation must be split and run on different  
 40 machines (“nodes”) through SimMobilityST’s MPI-based distributed computing platform. Although the  
 41 same parallel setup just discussed is still run on each node, the inter-node communication is inherently  
 42 less flexible and requires modelers to abandon the location-agnostic property of the single-node setup. In  
 43 particular, entities must be located in a particular geometric “space”, and various spatial decomposition  
 44 techniques are used to assign different spaces to different physical machines.

45 The global state of the simulation, including the road network and the trip chains, is split and  
 46 distributed to each node, whereupon it is loaded on an as-needed basis (to reduce memory usage). Agents  
 47 are distributed to the node which contains their starting trip, and will be automatically transferred to other  
 48 nodes as they cross the relevant node’s boundary. A node boundary is defined as a line perpendicular to a  
 49 given road segment which divides the upstream and downstream halves of that segment between two

1 nodes.

2 Entities on different nodes can sense each other via mirroring, but they cannot otherwise interact.  
 3 Furthermore, entities that are not in a mirrored region cannot send each other messages unless they are on  
 4 the same node. This is an unfortunate necessity, as it puts a lower bound on the number of nodes that a  
 5 given node must interact with, in turn allowing SimMobilityST to scale efficiently up to arbitrarily  
 6 complex road networks and simulation workloads. In order to work around these limitations, a delayed-  
 7 hopping message protocol is in development which allows messages to be sent to agents on other nodes at  
 8 a time cost of N time ticks, where N is the number of nodes between the originating agent and the  
 9 receiving agent. This system is flexible, and permits the time delay to be further reduced (to at least 1)  
 10 through the use of software-defined relays.

11

12 **4.4 Visualizer**

13 To represent the simulator output graphically an interface is developed in C++ using QT libraries. This  
 14 GUI is used for debugging purpose and to demonstrate of traffic impact through vehicle animation  
 15 (**FIGURE 3**). This application accepts the simulator output file produced in plain text format and then it  
 16 display's vehicle trajectory at each frame tick (in microscopic view mode) or network statistics  
 17 (mesoscopic view mode). It supports zoom-in and zoom-out operation handle large or small area  
 18 visualization, road network entity search functionality to locate object easily within the GUI.

19

20 **4.5 Data management**

21 SimMobilityST supports multiple data interfaces to exchange data with the other level or the data those  
 22 are exogenous to the simulator. It is able to read or write data from/to XML files, PostgreSQL database  
 23 and the CSV files. This will give user greater flexibility to run SimMobilityST with variety of data  
 24 sources depending on the experiment requirement. SimMobilityST data requirement can be grouped into:  
 25 Configuration, In/Output data, Model parameters.

26 SimMobilityST stores the configuration data and the model parameters in XML format. It  
 27 provides greater readability for the user to configure SimMobilityST. Simulation input data can be either  
 28 exogenous to the system for example road network, traffic light phases etc. or received from other level of  
 29 SimMobilityST for example trip chains received from mid-term to short-term. SimMobilityST supports  
 30 both XML and Database interfaces for input data and depending on the simulation need user can specify  
 31 the format in the configuration file. Simulation outputs are generated in plain text format that can be used  
 32 for further processing and also the required portion is written in the database for passing to the next level.

33

34 **5 CALIBRATION**

35 **5.1 Demand and supply parameters**

36 Demand parameters are typically calibrated through tuning of the OD flows. As SimMobility use activity  
 37 schedules and trip chains instead of OD matrix, we aggregate the trip chain to generate OD parameter  
 38 which to be calibrated, then, convert the updated OD parameter ( $\hat{\theta}_{k+1}$ ) into trip chains by disaggregating  
 39 through so-called ‘killing-cloning process’ for each iteration ( $k$ ). Thus, each activity schedule will be  
 40 “parameter” to be calibrated (killed or cloned). This means that each individual activity schedule, and  
 41 therefore each agent, will be calibrated. Each activity schedule is considered as fixed, but replicates can  
 42 be generated in the calibration process. The integrated calibration of Mid- and Short-Term simulators  
 43 would relax this assumption by calibrating directly activity based model parameters as the demand.

44 Other parameters in demand side are route choice parameters. Route choice model for private  
 45 transport corresponds to a Path-Size Logit Model [39]. Accordingly, drivers’ route choice decisions are  
 46 captured in a probabilistic manner, and highly likely to choose the route that maximizes his/her utility  
 47 which is characterized by influencing factors including travel-time and distance. On the supply side, all  
 48 driving behavior parameters are considered [21], [40].

1

## 2 5.2 Calibration framework

3 The simultaneous calibration of demand and supply parameters generate a large set. To deal with such  
 4 complexity, we used the Weighted Simultaneous Perturbation Stochastic Approximation, W-SPSA ([36],  
 5 [37]). The algorithm finds the best parameter set by iteratively updating the parameter set to the  
 6 decreasing direction in goodness-of-fit (GoF), which may rely on existing measurements and on prior  
 7 knowledge on demand-supply parameters from previous experiments. Through the ‘killing-cloning  
 8 process’, we run the SimMobilityST to get initial simulation output and assignment matrix. The  
 9 assignment matrix as weight matrix for the measurements in W-SPSA. After initial setting, the calibration  
 10 loop runs until reach to the convergence condition and objective value is within an acceptable level of  
 11 performance. The optimization problem over parameter space during the period of  $H = \{1, 2, \dots, H\}$  can  
 12 be formulated as:

13

$$\min_{x, \beta} z(\theta) = \sum_{h=1}^H [w_c z_c(C_h^0, C_h^S) + w_T z_T(T_h^0, T_h^S) + w_p z_x(x_h, x_h^a)] + w_p z_\beta(\beta, \beta^a) + w_p z_\gamma(\gamma, \gamma^a) \quad (1)$$

14 s.t.  $C_h^S = f_C(x_1, \dots, x_h; \beta; \gamma)$ ,  $T_h^S = f_T(x_1, \dots, x_h; \beta; \gamma)$ ,  $lb_{x_h} \leq x_h \leq ub_{x_h}$ ,  $lb_\beta \leq \beta \leq ub_\beta$ ,  $lb_\gamma \leq \gamma \leq ub_\gamma$

15 where,  $z_c$  and  $z_T$  respectively measures GoF between externally observed (Count:  $C_h^0$ , Travel-time:  
 16  $T_h^0$ ) and simulated measurement ( $C_h^S$ ,  $T_h^S$ );  $z_x$  compares estimated time-dependent OD parameter ( $x_h$ )  
 17 with the seed OD ( $x_h^a$ ) from MT;  $z_\beta$  and  $z_\gamma$  respectively evaluates the estimated parameter set in  
 18 driving behavior ( $\beta$ ) and route choice ( $\gamma$ ) against a priori values ( $\beta^a, \gamma^a$ ); and  $\theta$  is decision vector  
 19 ( $\theta = [x_1, \dots, x_h, \beta, \gamma]$ ). The parameters are bounded upper/lower limits. Note that each evaluation term  
 20 includes weighting coefficient ( $w_c, w_T, w_p$ ), which determined by the reliability on the external  
 21 information.

22 W-SPSA selectively perturbs relevant parameters based on a weight matrix ( $w$ ) which represents  
 23 spatiotemporal correlations between each parameter and measurements. Readers can refer the full  
 24 structure of  $W$  matrix from [43]. To increase applicability in ST calibration, which deals with large  
 25 number of agents for large spatial ranges, a sparse matrix has been generated in this phase. Then, the  
 26 gradient approximation can be formulated as:

27

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{\sum_{j=1}^D w_{ji} [(\epsilon_{kj}^+)^2 - (\epsilon_{kj}^-)^2]}{2c_k \Delta_{ki}} \quad (2)$$

28 where,  $c_k$ ,  $\Delta_{ki}$  indicate the perturbation amplitude and random perturbation vector (following Bernoulli  
 29 process) respectively. Also, note that  $(\epsilon_j^+)^2$  and  $(\epsilon_j^-)^2$  represent deviation vectors measuring distance  
 30 between the observed and the simulated measurement with plus ( $\hat{\theta}_k + c_k \Delta_k$ ) and minus perturbed  
 31 parameter ( $\hat{\theta}_k - c_k \Delta_k$ ) respectively.  $\hat{g}_{ki}$  is the  $i^{th}$  element of the approximation of the gradient vector.  
 32 This gradient provides the amount of movement from current  $k^{th}$  state ( $\hat{\theta}_k$ ) to the next iteration ( $\hat{\theta}_{k+1}$ ).

33 Using Eq. 2, we update  $\hat{\theta}_{k+1}$  which yields less cost on evaluation function:

34

$$\hat{\theta}_{k+1} = \arg \min_{\hat{\theta}_{k+1}} (z(\hat{\theta}_{k+1} = \hat{\theta}_k - \alpha_k \hat{g}_k(\hat{\theta}_k)), z(\hat{\theta}_{k+1} = \hat{\theta}_k + \alpha_k \hat{g}_k(\hat{\theta}_k))) \quad (3)$$

35 where,  $\alpha_k$  is a step size in a gain sequence. Therefore, SimMobilityST calibration includes a backward  
 36 decision process that gives additional chance to consider opposite direction in the decision vector update.  
 37 Also, note that the two different function evaluation on two parts (Eq. 2) conducted on a parallelized way  
 38 its independent each other and the algorithmic parameters are selected as in [41]. Once objective value  
 39 satisfies the internal convergence term, we terminate full calibration process.

### 1    5.3 Experimental setting

2    SimMobilityST is calibrated by running the traffic for the extended Central Business District (CBD) in  
 3    Singapore (**FIGURE 4**). This area contains more than 1200 intersections, which covered by more than  
 4    2000 loop detectors. A smaller sub-network with 10 intersections called Bugis, located inside CBD, was  
 5    also tested for assessing the impact of daily variability in the calibration process. The aggregated demand  
 6    generated by SimMobility MT has 1497 observed ODs pairs and a total of 48,988 trips. These trips  
 7    (demand), 11 route-choice parameters for demand and 112 driving behavior parameters (supply) are the  
 8    set of parameters to calibrate.

9    For the calibration, two types of data were available: loop count and GPS travel-time data from  
 10   probe vehicles, collected in August 2013 by the Land Transport Authority of Singapore and Taxi data  
 11   location respectively. Counts had a resolution of 5min while the GPS data was structured in OD a travel-  
 12   time tables for each 30min interval of the day. Counts were also preprocessed for outlier detection [44]. A  
 13   total of 360 sensors were used in the calibration. The following data was also used in network settings:  
 14   SCATS signal phases, GIS network configuration, Google transit network data for the buses routes and  
 15   schedules, as well as freight (background) traffic data.  
 16

### 17    5.4 Calibration result

18   The result shows the calculated root mean square normalized error (RMSN) between the simulated and  
 19   observed count over all segments and time intervals. **TABLE 1** shows RMSN for multiple days in small  
 20   Bugis network. This shows that the calibration framework is able to calibrate simulator using different  
 21   external data sets, with improvements 40~50% in RMSN. In the extended CBD area and after 250  
 22   iterations, the fit-to-counts has been improved from 0.72 to 0.37 of RMSN (**FIGURE 5 (a)**) and the  
 23   calibrated counts became close to the 45-degree line (**FIGURE 5 (b)**). The calibrated RMSN seems to be  
 24   satisfactory given the above small number of sensors and relatively large number of parameters.  
 25

## 26    6 CONCLUSIONS

27   A microscopic mobility simulator, SimMobilityST, aiming at simulating the movement of agents and the  
 28   decisions and operation of control and logistic centers is presented. SimMobilityST allows for a modular  
 29   integration of specific behaviors associated with new mobility services and transportation modes. It is  
 30   integrated within the SimMobility framework, a multi-scale simulation platform that considers land-use,  
 31   transportation and communication interactions using various behavioral models. The new simulator  
 32   particularly focuses on impacts of innovative transportation services on transportation and mobility  
 33   networks, thereby enabling the simulation of a portfolio of technology, policy and investment options  
 34   under alternative future scenarios. SimMobilityST has been successfully calibrated using external data in  
 35   Singapore. Multiple days supports the replicability of calibration capability as well. This calibrated  
 36   simulator would contribute to increase simulation reliability in evaluation of new scenarios in Singapore  
 37   and elsewhere.

38   The main on-going development efforts have been focusing in integrating further (existing)  
 39   advanced driving behavior models; designing and implementing the urban freight tour-based logic, along  
 40   with its specific behaviors and logistic decisions; and implement further smart mobility services.

41   Finally, the integration of a dedicated framework for simulating electrical vehicles and both  
 42   environmental and safety impacts assessment modules are also three short-term key milestones. The first,  
 43   will allow SimMobilityST to be used in the decision process of the design of the electrical vehicles grid  
 44   and to model the associated changes in the mobility patterns at the city level. The environmental and  
 45   safety impacts assessment modules will allow a more comprehensive evaluation of the technologies and  
 46   services being tested within SimMobility.  
 47

1 **AKNOLEDGEMENTS**

2 We are thankful to Max Zhiyong and Seth Hetu for their contributions to the early development of  
3 SimMobility ST. We also thank the Land Transport Authority (LTA) of Singapore for providing data for  
4 this research study. This research was supported by the National Research Foundation under its CREATE  
5 programme, Singapore-MIT Alliance for Research and Technology (SMART) Future Urban Mobility  
6 (FM) IRG.

7 **BIBLIOGRAPHY**

- 8 [1] Barceló, J. (ed.), 2010. Fundamentals of Traffic Simulation, 1st Edition. Springer.
- 9 [2] Pipes, L. A., 1953. An operational analysis of traffic dynamics. Journal of applied Physics 24 (3),  
10 274–287.
- 11 [3] Hranac, R., Gettman, D., Toledo, T., Kovvali, V., Vassili Alexiadis, 2004. NGSIM Task E.1-1: Core  
12 Algorithms Assessment. Tech. Rep. Federal Highway Administration, USA.
- 13 [4] TSS, 2011. Aimsun Dynamic Simulator Users' Manual. Version 7.0, Barcelona, Spain.
- 14 [5] PTV, 2009. VISSIM 5.20 User Manual. Tech. rep., Planung Transport Verkehr AG, Karlsruhe,  
15 Germany.
- 16 [6] Quadstone Paramics, 2009. The Paramics Manual. Version 6.6. 1. Edinburgh, Scotland, UK.
- 17 [7] Caliper, 2008. Traffic Simulation Software: TransModeller Users' Guide. MA, USA.
- 18 [8] Hidas, P., 1998. A car-following model for urban traffic simulation. Traffic engineering & control 39  
19 (5), 300–305.
- 20 [9] Federal Highway Administration, 2006. CORSIM User's Guide Version 6. FHWA, USA.
- 21 [10] Liu, R., 2010. Traffic simulation with DRACULA. In Fundamentals of Traffic Simulation, pp. 295-  
22 322, Springer New York, USA.
- 23 [11] Koskinen, K., Kosonen, I., Luttinen, T., Schirokoff, A., Luoma, J., 2009. Development of a  
24 nanoscopic traffic simulation tool. Advances In Transportation Studies - An International Journal  
25 Section B 17.
- 26 [12] Rakha, H. 2014, INTEGRATION Rel. 2.40 for Windows - User's Guide, Volume I: Fundamental  
27 Model Features, V. Aerde and Associates, Ltd., Blacksburg, Virginia, USA.
- 28 [13] Ben-Akiva, M., Koutsopoulos, H. N., Toledo, T., Yang, Q., Choudhury, C. F., Antoniou, C., &  
29 Balakrishna, R., 2011. Traffic Simulation with MITSIMLab. In Fundamentals of Traffic Simulation,  
30 145, 233.
- 31 [14] Behrisch, M., Bieker, L., Erdmann, J., and Krajzewicz, D., 2011, Sumo-simulation of urban  
32 mobility-an overview. In SIMUL 2011: The Third International Conference on Advances in System  
33 Simulation (pp. 55-60).
- 34 [15] CubeDynasim, 2013. <http://www.citilabs.com/new-dynasim-4>. CityLabs.
- 35 [16] TU0903-Cost Action, 2012. <http://www.multitude-project.eu/>.
- 36 [17] TRANSIMS, 2014: Transportation Analysis Simulation System. <http://ndssl.vbi.vt.edu/transims-docs.html>. Accessed online June to July 2014.
- 38 [18] MATSIM, 2014. Multi Agent Traffic SIMulation. <http://www.matsim.org>. Accessed online June to  
39 July 2014.
- 40 [19] Bellemans T, Bruno K, Janssens D, Wets G, Arentze T and Timmermans H, 2010. Implementation

- 1        Framework and Development Trajectory of the Feathers Activity-Based Simulation Platform,  
2        Transportation Research Record-Journal of Transportation Research Board, No. 2175, pp: 111-119
- 3 [20] Krajzewicz, D., Erdmann, J., Behrisch, M. and Bieker, L., 2012. Recent Development and  
4        Applications of SUMO – Simulation of Urban Mobiity. International Journal on Advances in  
5        Systems and Measurements, Vol 5, N. 3 & 4, pp. 128-138.
- 6 [21] Toledo, T., Koutsopoulos, H., Ben-Akiva, M. E., 2007. Integrated driving behavior modeling.  
7        Transportation Research Part C: Emerging Technologies 15 (2), 96-112.
- 8 [22] Ahmed, K., 1999. Modeling Drivers' Acceleration and Lane Changing Behavior. Ph.D. thesis,  
9        Massachusetts Institute of Technology
- 10 [23] Yang, Q., Koutsopoulos, H. N. and Ben-Akiva, M. E., 2000. "Simulation Laboratory for Evaluating  
11        Dynamic Traffic Management Systems". Transportation Research Record: Journal of the  
12        Transportation Research Board, 1710, 122-130.
- 13 [24] Choudhury, C. F., 2007. Modeling Driving Decisions with Latent Plans. Phd thesis, Massachusetts  
14        Institute of Technology.
- 15 [25] L. Bowman. 2001. Activity-based disaggregate travel demand model system with activity schedules.  
16        Transportation Research Part A 35: 1–28.
- 17 [26] Ben-Akiva, Moshe, John L. Bowman, and Dinesh Gopinath. 1996. Travel demand model system for  
18        the information era. Transportation 23: 241–266.
- 19 [27] Adnan, M., Carrion, C., Basak, K., Hetu, S., Zhiyong, W., Vahid, S., Loganathan, H., Yang, L.,  
20        Pereira, F., Ben-Akiva, M. SimMobility Midterm Framework: A state of the art Integrated Agent  
21        Based Transport Demand and Supply Simulator. Submitted to the 94th Annual Meeting of the  
22        Transportation Research Board, January 2015, Washington D.C., USA.
- 23 [28] Basak, K., Hetu, S., Li, Z., Lima Azevedo C., Loganthan, H., Toledo, T., Xu, R., Peh, L. S., Ben-  
24        Akiva, M., 2013. Modeling reaction time within a traffic simulation model. Annual Meeting of the  
25        Transportation Research Board, Washington D.C., USA.
- 26 [29] Daizong, L., 2003. Comparative evaluation of dynamic TRANSYT and SCATS-based signal control  
27        systems using Paramics simulation. M.Sc. Thesis in Civil Engineering. National University of  
28        Singapore, Singapore.
- 29 [30] Chovan, J., Tijerina, L., Alexander, G., Hendricks, D., 1994. Examination of Lane Change Crashes  
30        and Potential IVHS Countermeasures. Tech. Rep. March, US Department of Transportation,  
31        NHTSA, Washington D.C., USA
- 32 [31] Sarvi, M., 2013. Heavy commercial vehicles-following behavior and interactions with different  
33        vehicle classes. Journal of Advanced Transportation 47 , pp. 572–580.
- 34 [32] Nourinejad, M., Wenneman, A., Nurul Habib, K., Roorda, M. J., 2014. Truck parking in urban areas:  
35        Application of choice modelling within traffic microsimulation. Transportation Research Part A 64  
36        (2014) 54–64.
- 37 [33] Teo, J., Taniguchi, E. and Qureshi, A. G., 2012. Evaluation of Distance-Based and Cordon-Based  
38        Urban Freight Road Pricing in E-Commerce Environment with Multiagent Model. Transportation  
39        Research Record: Journal of the Transportation Research Board, No. 2269, Transportation Research  
40        Board of the National Academies, Washington, D.C., 2012, pp. 127–134.
- 41 [34] Hetu, S., Vahid, S. and Peh, L.S., 2014. Similitude: Interfacing a traffic simulator and network

1 simulator with emulated Android clients. Accepted for presentation in IEEE 79th Vehicular  
2 Technology Conference, 18–21 May 2014, Seoul, Korea.

3 [35] Gao, J. and Peh, L.S., 2014. RoadRunner: Infrastructure-less Vehicular Congestion Control,  
4 Accepted for presentation in the 21st World Congress on Intelligent Transport Systems (ITSWC),  
5 Detroit, Michigan, September 2014.

6 [36] Lu, L., Xu, Y., Antoniou, C., & Ben-Akiva, M. 2015. An enhanced SPSA algorithm for the  
7 calibration of dynamic traffic assignment models. Transportation Research Part C: Emerging  
8 Technologies, 51, 149-166.

9 [37] Antoniou, C., Azevedo, C. L., Lu, L., Pereira, F., & Ben-Akiva, M. 2015. W-SPSA in practice:  
10 Approximation of weight matrices and calibration of traffic simulation models. Transportation  
11 Research Part C: Emerging Technologies, 59, 129-146.

12 [38] Lu, Y., Adnan, M., Basak, K., Pereira, F. C., Carrion, C., Saber, V. H., and Ben-Akiva, M. (2015).  
13 Simmobility mid-term simulator: A state of the art integrated agent based demand and supply model.  
14 In Transportation Research Board 94th Annual Meeting (15-3937).

15 [39] Ramming, M. S. (2001). Network knowledge and route choice (Doctoral dissertation, Massachusetts  
16 Institute of Technology).

17 [40] Ciuffo, B., & Azevedo, C. L. (2014). A sensitivity-analysis-based approach for the calibration of  
18 traffic simulation models. IEEE Transactions on Intelligent Transportation Systems, 15(3), 1298-  
19 1309.

20 [41] Vaze, V., Antoniou, C., Wen, Y., & Ben-Akiva, M. (2009). Calibration of dynamic traffic  
21 assignment models with point-to-point traffic surveillance. Transportation Research Record: Journal  
22 of the Transportation Research Board, (2090), 1-9.

23 [42] Antoniou, C., Azevedo, C. L., Lu, L., Pereira, F., & Ben-Akiva, M. (2015). W-SPSA in practice:  
24 Approximation of weight matrices and calibration of traffic simulation models. Transportation  
25 Research Part C: Emerging Technologies, 59, 129-146.

26 [43] Lu, L., Xu, Y., Antoniou, C., & Ben-Akiva, M. (2015). An enhanced SPSA algorithm for the  
27 calibration of dynamic traffic assignment models. Transportation Research Part C: Emerging  
28 Technologies, 51, 149-166.

29 [44] Lu, X. Y., Varaiya, P., Horowitz, R., & Palen, J. (2008). Faulty loop data analysis/correction and  
30 loop fault detection. In 15th World Congress on Intelligent Transport Systems and ITS America's  
31 2008 Annual Meeting.

32 [45] Tachet, R., Santi, P., Sobolevsky, S., Reyes-Castro, L. I., Frazzoli, E., Helbing, D., & Ratti, C.  
33 (2016). Revisiting street intersections using slot-based systems. PloS one, 11(3), e0149607.

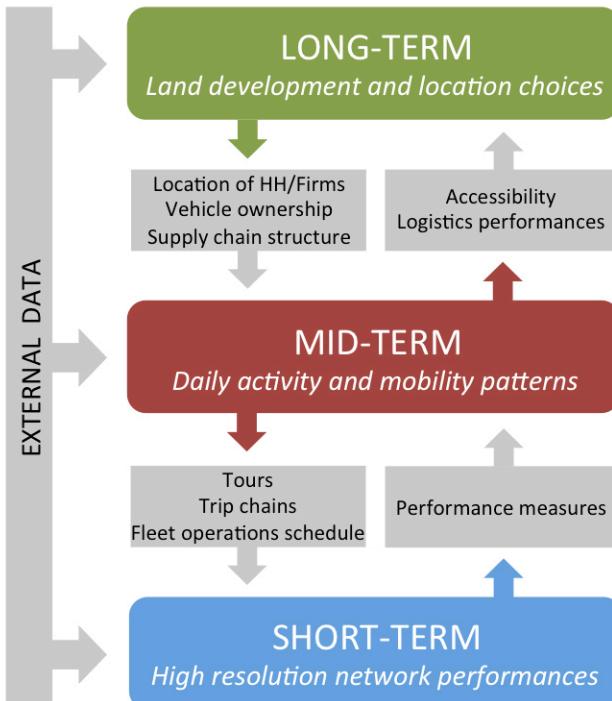
34 [46] Zhu, Y. and J. Joesph Ferreira, Synthetic population generation at disaggregated spatial scales for  
35 land use and transportation microsimulation. Transportation Research Record: Journal of the  
36 Transportation Research Board, , No. 2429, 2014, pp. 168–177.

37 [47] Teo, J., Cheah, L., Lee, Y.J., Marzano, V., Santos, J., Lima Azevedo, C., Zhao, F., Ben-Akiva, M.  
38 (2015). An integrated sensing-based urban freight data collection framework: methodology and pilot  
39 projects in Singapore. URBE conference, Oct 1-2, 2015, Rome, Italy.

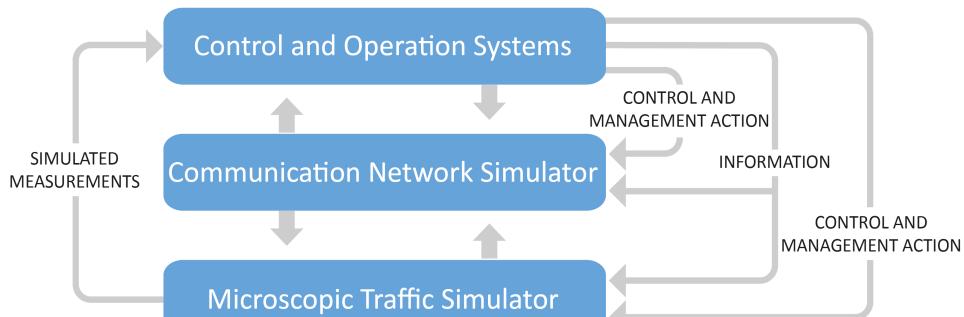
40 [48] Lima Azevedo, C., Marcuk, K., Raveau, S., Soh, H., Adnan, M., Basak, K., ... & Ben-Akiva, M.  
41 (2016). Microsimulation of Demand and Supply of Autonomous Mobility On-Demand. In

- 1           Transportation Research Board 95th Annual Meeting (16-5455).
- 2 [49] Atasoy, B., Ikeda, T. and Ben-Akiva, M. (2015), Optimizing a Flexible Mobility on Demand System,  
3           Transportation Research Record (TRR), Vol. 2536, pp. 76-85.
- 4 [50] Le, D.T., Cernicchiaro, G., Zegras, C., Ferreira, J. (2016). Simulation of synthetic establishments for  
5           modeling firm behavior in SimMobility. International Scientific Conference on Mobility and  
6           Transport Transforming Urban Mobility, mobil.TUM 2016, 6-7 June 2016, Munich, Germany.
- 7
- 8

- 1    **List of Figures**
- 2    **FIGURE 1.** Framework of SimMobility and SimMobilityST
- 3    **FIGURE 2.** SimMobility Short-term Simulator
- 4    **FIGURE 3.** SimMobility Visualizer (Singapore simulation)
- 5    **FIGURE 4.** Extended Central Business District (CBD) in Singapore (shaded area; red dots are  
6       measurement locations)
- 7    **FIGURE 5.** Calibration result for the Singapore extended CBD area
- 8

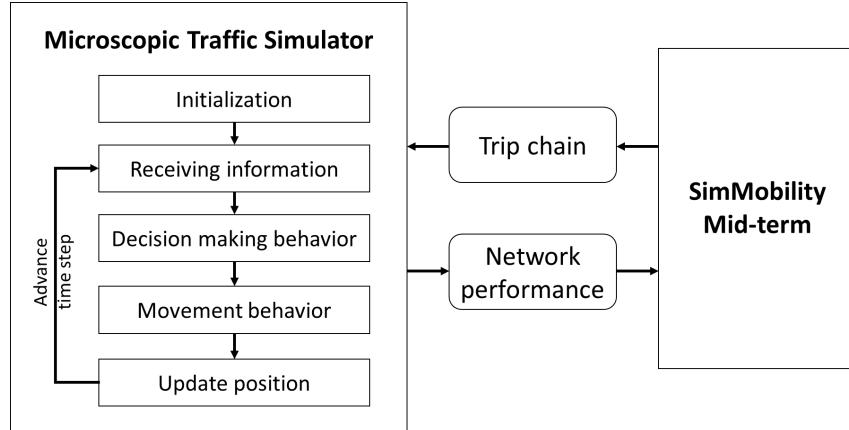


(a) High-level SimMobility Framework



(b) Short-term Framework

**FIGURE 1** Framework of SimMobility and SimMobility ST



**FIGURE 2** SimMobility ST Traffic Simulator

1

2

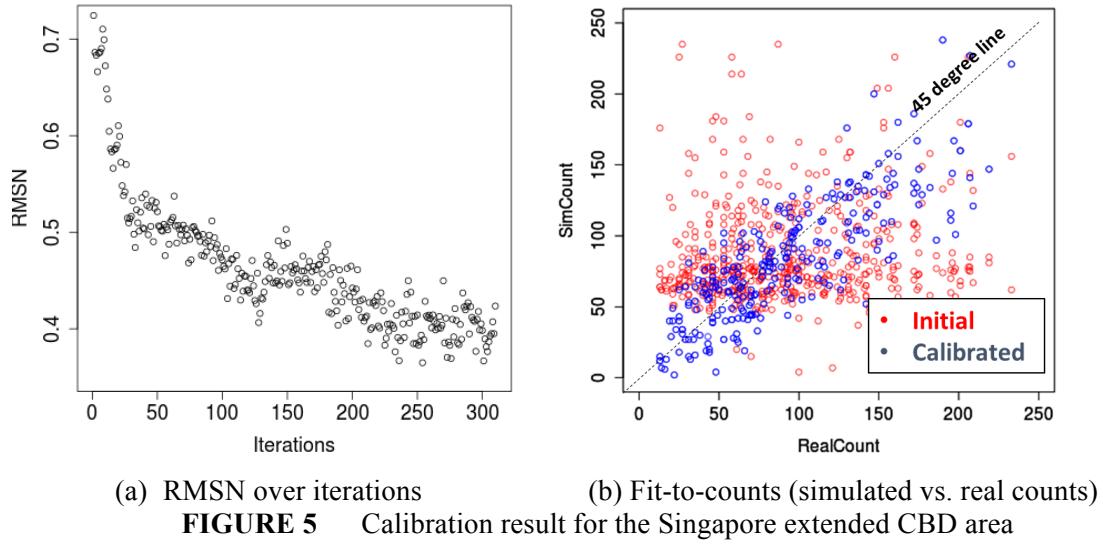


**FIGURE 3** SimMobility Visualizer (Singapore simulation)

1  
2  
3  
4



1  
2 **FIGURE 4** Extended Central Business District (CBD) in Singapore (shaded area; red dots are  
3 measurement locations)  
4  
5  
6



1 **List of Tables**

2 **TABLE 1.** RMSN for multiple days in Bugis network

**TABLE 1** RMSN for multiple days in Bugis network

Date in Aug./2013	6 <sup>th</sup>	7 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>	20 <sup>th</sup>	21 <sup>st</sup>	22 <sup>nd</sup>
Initial	0.58	0.53	0.53	0.54	0.49	0.47	0.51	0.46
Calibrated	0.29	0.29	0.32	0.26	0.25	0.29	0.27	0.25