Introducing Cellular Network Layer into SUMO for Simulating Vehicular Mobile Devices' Interactions in Urban Environment

Siim-Toomas Marran, Artjom Lind and Amnir Hadachi

ITS Team, Institute of Compute Science, University of Tartu, Ülikooli 17, 51014, Tartu, Estonia

Keywords: Simulation, Mobility Big Data, Call Detail Records, Cellular Networking, Urban Road Traffic, GPS Devices, Mobile Devices, SUMO.

Abstract: During the last decade researchers have been demonstrating the importance of mobile data or CDR data in depicting the human mobility patterns. However, this type of data is not easy to get access to from mobile operators. Besides, in order to make this type of data available and enable their usage for the scientific communities the process can face many constraints that can constitute obstacle. From this perspective, this paper introduces a way to produce realistic real-life mobility logs through the traffic simulation tool SUMO, which has been enhanced with a cellular network layer to mimic cellular networking behavior.

1 INTRODUCTION

The mobility data is a broad connotation that refers to a various types of data, which describes people's movement and activities. Some of the common types of the mobility data are Call Detail Records (CDR) (Hadachi et al., 2014), Global Positioning System (GPS) logs (Hadachi et al., 2013), Vehicular ad hoc network (VANET) logs (Karnadi et al., 2007), Radio-frequency identification (RFID) system logs (Finken-zeller and Müller, 2010), WiFi access points related data (Bulut and Szymanski, 2013), etc. Moreover, with all the advancement in information and tele-communication technologies (ICT). Mobile data has a great potential in sensing the urban mobility dynamics. From this perspective, it is clear that mobile data or CDR data are valuable. In general, CDR are data records, produced inside of the cellular network between the user equipment and the base station, documenting various telecommunication transactions. Besides, it is estimated according to (Statista, 2017) that 4.77 billions mobile users produce daily unimaginable amount of data, with restricted access. The mobility big data has been used for developing many different algorithms and applications such as triangulation and trilateration, Kalman filter for mobile positioning (Lind et al., 2017) , handling the crowd management (Pan et al., 2013) , deploy real-time incident detection solutions (Zaldivar et al., 2011) , manage and avoid congestions in the traffic ,develop Vehicle-To-Infrastructure (V2I) appli-

cations , bind the mobility with various Internet of Vehicles (IoV) and Geographic Information Systems (GIS) to achieve new systems upgraded with the human activity related functionality. As a consequence, there occur moments when it is not feasibly easy to acquire real data for specific purpose, we rely on simulating and generating synthetic data based on mathematical models. For example, we have open source SUMO simulator (DLR, 2017) that provides the ability to simulate real-life urban traffic at microscopic level. This open source simulation package provides realistic simulation due to the OpenStreetMap (OSM) (OSM, 2017) map import functionality and in-built configurable behavior models. The routable entities can be equipped with various devices. In this paper, we are integrating we are integrating SUMO with mobile and GPS devices. Hence, we introduced also a cellular network layer with all its functionalities linking the routable devices and the network. This allows us to generate from the microscopic traffic simulator the associated CDR and GPS logs, which can be later used for the training and testing various machine learning models in the real life context.

2 RELATED WORK

SUMO is a suite of applications to prepare and to simulate various real-life traffic scenarios. Therefore, it has been used for many intelligent transport systems

related research topics. For example, the project *TrafficOnline* is a good illustration of the use of SUMO for simulating the usage of mobile phone data, where the real-world GSM data was used to develop a telephony model (consisting of the following properties: call start, call duration), to determine the travel times (Krajzewicz et al., 2012). However, the dynamic properties, e.g. cell size variations, cell changing mechanics, of the GSM network were not considered.

Another related research topic is VANET simulations. The researcher tried to create an integration between the traffic simulator SUMO and third party network simulators, such as OMNet++ (OMNeT++, 2017), where they have created bidirectionally coupled hybrid simulators and Viens (Sommer et al., 2008). Besides, the list is long for example there are also NS-2 (ns 2, 2017) and NS-3 (ns 3, 2017) that have been used for VANET system's for detailed and realistic performance evaluation. Similar to Veins, TraNS (Traffic and Network Simulation Environment) based on NS-2, was introduced to combine two disjointly developed research simulators into one in order to deal with VANET performance evaluation (Piorkowski et al., 2008). The NS-3 was introduced as replacement to NS-2 and it was used in the VANET crossroad scenario to evaluate the performance of HWMP, OLSR and DD routing protocols, while using IEEE 802.11p standard and Two-RayGround Propagation Loss Model, to send multiple Constant Bit Rate (CBR) flows over UDP between 20 source-destination pairs. (Kolici et al., 2015)

Furthermore, we have the emergence of the vehicular networking concept, where the vehicle is connect to everything. Therefore, the birth of many type of communications related to VANET such as Vehicle-To-Everything (V2X), vehicle-to-vehicle (V2V), and vehicle-to-infrastructure (V2I). The main purpose of introducing the connectivity into the vehicle to solve issues related to traffic control and management via the share or collection of reliable information from the vehicles. For this purpose, many simulators have been created such as VSimRTI (Wedel et al., 2009). Obviously VANET solutions aren't only viable solutions to tackle the road traffic estimation problem. Due the widespread of the mobile communications we can estimate the traffic density on the bigger and more important roads with the help of cellular network, since many commuters are carrying mobile devices with them. Such approach has proved to be accurate and very capable to detect and quantify state changes (Bolla and Davoli, 2000).

3 SYSTEM DESIGN AND ARCHITECTURE

The purpose of this simulation is to produce CDR-like activity logs with the help of microscopic road traffic simulator SUMO, where we added an extra layer – cellular networking. This integration of a new layer introduces to SUMO's core and support packages many changes and new features.

3.1 Models

The most important step in the process of integrating the cellular network into SUMO is to define the models of each new component. The new models can be divided into two main categories: Cellular network layer and devices. The cellular network layer contains the following elements:

Cellular Tower has been added to the cellular network layer with following properties: the Cartesian coordinates; real life Geo-spatial information; list of cellular antennas attached to the tower and keeps the list of connected mobile devices' references. **Cellular Antenna** is also known in the cellular network as a transmitter. Antennas are entities attached to the cellular towers. The model has the following properties: coverage as a polygon shape (e.g.hexagon, custom polygons are also supported); signal strength; frequency, radius; reference to its tower and an id value, which represents Common Gateway Identifier (CGI) value. **Mobile Event Data** is a data structure model to describe a mobility event for a certain routable entity. Concerning the devices added to the simulator can be resumed to the GPS and mobile devices and they are defined as follows: **GPS Device** is, in the simulator context, a device which can be attached to the routable entity like pedestrian, vehicle, bicycle etc. **Mobile Device** is a device that generate the mobile event data, attachable to the routable entity.

3.2 Core Mechanics of the Simulation

The core mechanics of the road traffic simulation SUMO has been gathered into one class which contains the road network and orchestrates the simulation. This class is initialized during the start of the program and is given to the network builder as a reference to create the infrastructure for the simulation. During the road network building, the cellular network is being loaded into the simulation network. Then, when the simulation performer has all the vital dependencies, then the simulation cycle starts. The core of the simulation contains of multiple event controllers, which are being ran after every tick. The si-

mulation have been divided into four components according to their event controller action types and intermediate smoothing and they are as follows:

Begin of time step events is the original event controller in SUMO, which, as an example, takes care of traffic light events, trigger type events(lane speed, rerouting), routing device, and pedestrian move events. **Intermediate actions**, depends on the previous event control cycle and smooths the network for the next event control cycle. During this process, the system is detecting regularly for collisions, checking the traffic lights, checking if the edges are active, plans the vehicles movement, executing the vehicles movement, run lane changes, load new routes and mobile events for the cellular simulation, insert new vehicles to the network, insert new events, etc. During the execution of vehicles movement, there is a case when the vehicle has arrived to the destination and, immediately afterwards, the vehicle shall be removed from the simulation. This function has been altered to have checks for existing mobility devices (GPS, mobile) to run their last events immediately and then invalidating those events to stop them from recurring. This entire code alternation prevents losing event data due the premature removal of the vehicle. **End of time step events** is an event running cycle which runs the events dependent on the updated location of the routable elements. There are also step by step logger events, vehicular devices specific events (Bluetooth, GPS, mobility management related state update) and vehicle flow calibrator events. **Mobility events** is a newly added event running cycle entirely for the purpose of mobility event simulation. The events are created by the mobile devices from the data, which has been attached to the vehicle after it has been loaded into the simulation during the intermediate actions.

3.3 Data Import and Export

The simulator depends on the data which has been prepared beforehand for the simulation and therefore imported into the application through Extensible Markup Language (XML) files. SUMO has in its package multiple programs which help to produce the network and other simulation related data in XML document format to run the simulation properly. For the cellular networking layer, there were created multiple XML data import handlers and their respective XML Schema Definition (XSD) files. First, cellular network related data like antennas and their respective transmitters with coverage areas. Second, mobile events data, which were described in the Section 3.1. For the export there were designed output XML formats for the GPS and mobile devices. When the spe-

cific routable entity finished the navigation in the network then all its devices data were exported into the resulting XML file.

4 SIMULATOR IMPLEMENTATION

The implementation of cellular network log generator and it's integration into SUMO base code was done based on introducing two prerequisite applications: Mobility Event Simulation Generator (MESGEN) and Cell Coverage Area Generator (HEXAGEN).

4.1 MESGEN

Mobility Event Simulation Generator (MESGEN) is a supplementary application required by the main simulator application. MESGEN is responsible for generating the cellular events-time-line for each vehicle (cellular device inside vehicle). SUMO simulator uses the cellular events-time-line to trigger the events in certain cell towers during the simulation when vehicle traverses the corresponding coverage area. Several cellular activity profiles have been defined and are illustrated in Table 1. Each profile is specified by three attributes: frequencies of calls, SMS or and data usage activities. The events generated by MES-

Table 1: User cellular activity profiles (counting amount of events per hour). The numbers in the table correspond to events frequency as follows: none (0), low (1), medium (2) and high (3).

Profile	Calls	SMS	Data
Casual	1	1	1
Only Keep Alive	0	0	0
Business	3	1	1
Teenager	1	3	2
Student	1	1	2
Talkative	2	1	1
Media streamer	1	1	3

GEN are similar to those of cellular event data model (original model was simplified by removing not used events). The MESGEN event types are illustrated in Table 2. The simplified flowchart of the cellular mobility management entity is illustrated in Figure 2. For example MESGEN events *SMS_SEND* and *SMS_REC* occur mimics the sequence of cellular network equipment activities that result in delivering an SMS message (send or receive). MESGEN generates a history of events for each vehicle participating in simulation using the simulation life-time, vehicle activity profiles, road network information and cell

Table 2: MESGEN event type descriptions.

Message Type	Description
SMS_REC	SMS Received
SMS_SEND	SMS Sent
CALL_INIT	Calling out
CALL_REC	Receiving a call
WEB_COM	Uses mobile data

Listing 1: The appearance of the MESGEN output file.

```
<vehicles>
 <vehicle id="0" depart="0.00">
  <mEvents>
   <mEvent begin="114.00" end="137.00" type="CALL_REC"/>
   <mEvent begin="363.00" end="450.00" type="WEB_COMM"/>
   <mEvent begin="490.00" end="516.00" type="CALL_INIT"/>
   <mEvent begin="627.00" end="669.00" type="CALL_INIT"/>
   <mEvent begin="731.00" end="742.00" type="CALL_REC"/>
   <mEvent begin="829.00" end="856.00" type="CALL_INIT"/>
  </mEvents>
 </vehicle>
</vehicles>
```

coverage areas. The result events are associated to a vehicle and have time specified as an offset of SUMO simulation life-time. An example of MESGEN output is illustrated in the Listing 1 having vehicles with associated cellular events. The pseudo code in the Algorithm 1 illustrates the algorithm used in MESGEN generator.

Minimum length and maximum length of different events can be configured through constant variables. It is also possible to define time buffers between occurring events. The values of low and medium frequency events thresholds and random number seed can also be set through the configuration file.

4.2 HEXAGEN

HexagonGen is an accessory level script, written in Python, to generate 120 °sectorized cells for the cellular network behaviour simulation. It takes an input of latitude and longitude coordinate values, size (length of hexagons side), hexagon network width, and height values. The given coordinate pair is the center of the drawn network (tower position). The result of the HexagonGen is an XML file which contains the cell coverage elements and can be fed directly into SUMO.

5 MOBILITY BEHAVIOUR AND MANAGEMENT

5.1 Microscopic Simulation Devices

SUMO provides basic interfaces to communicate with the vehicle and abstract methods, which need to be implemented to develop a working device.

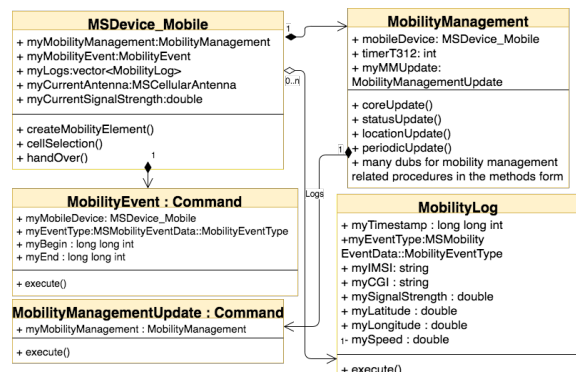**GPS device** (*MSDevice_GPS* class in the code) is a test device, it was created with the objective to study

Algorithm 1: Mobile's events generator.

Precondition: $V \in \mathbb{N}$ set of vehicles, $P \in \mathbb{N}^3$ set of profiles, $T \in \mathbb{N}$ vehicle departure times, l_{call} maximal call duration, l_{sms} maximal interval between sending SMS, l_{data} maximal web surfing duration, E set of event types as was specified in Table 2

```
1:  function TRIGGEREVENT(e, t, l)
2:                                    ▷ Triggers event of type e ∈ E at t, lasting l sec.
3:  end function
4:  function RAND(n_0, n_1)
5:      n ~ U([n_0, n_1])              ▷ random number from (n_0, n_1)
6:      return n
7:  end function
8:  function MESGENVEHICLE(v, t_start, t_end)
9:      n_calls, n_sms, n_data ← P_v    ▷ mobile profile of v
10:     n_all = ∑ P_v
11:     t ← T_v                         ▷ departure time of v
12:     while t_end − t ≤ 0 do
13:         n = RAND(0, n_all)
14:         if n ≤ n_calls then
15:             l = RAND(0, l_call)         ▷ Call duration
16:             if RAND(0, 1) > 0.5 then    ▷ Call direction
17:                 TRIGGEREVENT(CALL_INIT, t, l)
18:             else
19:                 TRIGGEREVENT(CALL_REC, t, l)
20:             end if
21:             t ← t + l
22:         else if n ≤ n_calls + n_sms then
23:             if RAND(0, 1) > 0.5 then    ▷ SMS direction
24:                 TRIGGEREVENT(SMS_SEND, t, 0)
25:             else
26:                 TRIGGEREVENT(SMS_REC, t, 0)
27:             end if
28:             t ← t + RAND(0, l_sms)
29:         else
30:             l = RAND(0, l_data)         ▷ Web Surfing Duration
31:             TRIGGEREVENT(WEB_COM, t, l)
32:             t ← t + l
33:         end if
34:     end while
35:  end function
36:  function MESGENALL(t_start, t_end)
37:      for all v ∈ V do MESGENVEHICLE(v, t_start, t_end)
38:      end for
39:  end function
```

SUMO simulation cycle for the upcoming mobility device. *MSDevice_GPS* has two subclasses called *RouteGPSInfo* and *GPSSignalUpdate*. *RouteGPSInfo* is a data structure that records geographical coordinates and the timestamp. *GPSSignalUpdate* represents GPS signal, it is an extension of SUMO's *Command* class (base SUMO *microsim* event class). During every simulation step, if the vehicle has a device, the event is triggered the vehicles movement.

Figure 1: The class diagram of *MSDevice_Mobile* and its sub-entities.

585

Mobile device (*MSDevice_Mobile* class in th code) is emulating multiple functions, which in real GSM stack are performed by several logical units. Next section will explain how mobile device simulation is performed. Figure 1 illustrates how mobile device component is integrated into SUMO and how does it interact with other modules.

5.2 Mobility Management

During the simulation the state of mobile device is managed by *MobilityManagement*. The state is updated at the end of each simulation step event. Vehicles are initialized based on the parameters file, which specifies what devices are equipped (mobile, GPS, etc). Mobility management entity is only created in case of vehicle being equipped with a mobile device. *MobilityManagementUpdate* is an extension of SUMO *Command* base abstract event class and is responsible for triggering the state update. The new state is assigned based on previous one, the state transitions are illustrated in the Figure 2. Mobility management has a reference to the mobile device to call out mobility related procedures from the lower layers. Then, there are mobility management state related variables; the T312 timer is for reminding a mobile device to make periodic location updates after every user defined steps (by default 10 steps). Some state related extra scenarios, define whether the mobile has been turned on, the SIM card inserted, the IMSI attached, and define whether the mobile has PLMN related information. In the Figure 2 a mobile mobility life cycle is described visually for better understanding. At the start of the mobility management
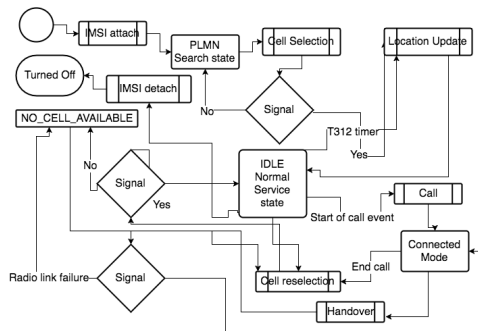

Figure 2: The simple work flow of the mobility management.

update, the creation of the current time step's mobility events for the simulation for this specific device takes place. During the mobility management update event, there will be a call out to the management cores update function which increments instantly the periodic location update with respect to

the previous state that goes through state machine to determine the next steps: a) If the mobility management state is classified as *NULL* the core update will change mobile to the state as it was just physically turned on; b) *MM_IDLE_NORMAL_SERVICE* starts mobile cell selection method on the device level.c) *MM_IDLE_SEARCH_FOR_PLMN* starts mobile cell selection method, but since the device has never been connected to the network, it also waits for the surrounding PLMN information; d) *MM_IDLE_NO_CELL_FOUND* start mobile cell selection, since there is currently no connectivity; e) *MM_CONNECTION_ACTIVE* starts the cell selection process for the handover purposes, if needed.After cell selection, the work-flow will move from the device back into the mobility management, where the state of MM will be updated and location update initiated. If the state machine has done its work, then there is a successive check to determine if the periodic update is needed and, if it is, then it will be executed. The rule with the periodic location update is that if a mobile has not had any location updates during some certain amount of simulation steps, then the mobility management has to step in and remind the transmitter that it still exists and has not been lost during the commute.

5.3 Wave Propagation

During the cell selection, the mobile device has multiple challenges. It has to determine the nearest polygons, determine in which one of them it resides, and determine the signal strength from the base stations transmitter. The signal strength has been envisioned by the free space wave propagation model:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}, \quad (1)$$

Where $P_r(d)$ is the received power, P_t is the transmitter output power, G_t is transmitter gain G_r is antenna gain, d is the distance between the MS and the tower, and L is the system loss factor. (Nishith Tripathi, 2014). This latter indicates the power received by the antenna under ideal conditions. in addition, the free space model predicts the powers decay to be the negative square root of the distance. In our case, we have not included the system loss factor. Besides, the distance in the equation 1 is calculated between the tower and the vehicle based on Haversine distance:

$$d = 2r\, Arcsin\left(\sqrt{sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + f(\phi_1, \phi_2)}\right)$$

$$f(\phi_1, \phi_2) = cos(\phi_1)cos(\phi_2)sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right), \quad (2)$$

where ϕ is the latitude, λ the longitude and r is the Earth's radius in meters (6 371 000 meters).

During the signal strength calculation, we use the configuration supplied tower transmitter output strength (dBm) and the transmitter's frequency. From the frequency, we calculate the wavelength. Wavelength equals the speed of light divided by the transmitter emitted frequency.

5.4 Location Update, Cell Selection and Reselection, Handover

The procedures are key component of keeping track of the mobility management level state, this can be seen also in the Figure 2.

Cell selection is a mobile device level method for measuring the current signal strength value of the serving antenna and manage handover procedure. During the

(a) The example of R-tree after all the shapes have been inserted into the tree.
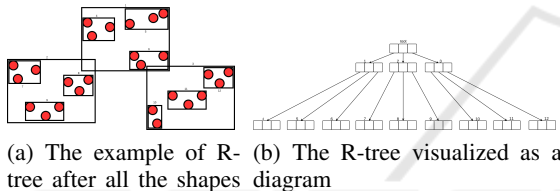
(b) The R-tree visualized as a diagram

Figure 3: Overview of R-tree process.

cell selection, the first step is to find surrounding cellular network cells. Those cells (polygons) are structured into the R-tree entity during the creation of the mobility tower control unit, where they were added for a faster search, as illustrated in the Figures 3(a) and 3(b), R-tree is a tree data structure to handle spatial data efficiently by indexing shapes for the future access. (Guttman, 1984) After searching the nearest polygons to the device, we check through the vector of cells and determine whether the vehicle is in the cell or not using *the winding number algorithm*. The vehicle might be in multiple cells; therefore, we are calculating the best signal strength and accordingly, we pick the transmitter and cell to camp on. In essence, *the winding number algorithm* is an algorithm which counts the number of times the polygon winds around the point of interest. If the point is not inside of the polygon, then the resulting winding number is 0. (Kai Hormann, 2001) During the cell selection, the registration to the tower or the deregistration from the tower will be determined. This is emulating the registering the mobile devices location area into the visitor location register. The final step is the status update of the mobility management.

Location update is the mobility management level

procedure for updating the devices location in the network. Location update resets the periodic update timer (sets the timer back to 0) and updates the mobility management state.

Handover procedure is triggered by the state machine at the level of *MM_CONNECTION_ACTIVE* when there is a better quality cell flagged true.

5.5 Mobility Events and the Simulation Cycle

We have covered the mobility management tied event object called *MobilityManagementUpdate*. In this section, we will discuss the mobile device related event. *MobilityEvent* extends the SUMO *Command* abstract event base class. The mobility events have their own *MSEventControl* container-event queue and the events are being executed at the very end of the simulation step, after the mobility management update events. Each mobility event has a type, which is declared in the enum of *MobilityEventType* in the class of *MSMobilityEventData*. The event has the mobile device reference to let the mobile know about the radio link failure and trigger the log creation about the occurred event. Other attributes of the mobility event are the events start time and end time. Calls and the web communication events depend on the timeframe variables.

5.6 Mobility CDR-like Logs

Call detail record (CDR) is the information about incoming and outgoing mobile activities, e.g., calls or SMS messages. The data in the CDR is about the event originated and the terminated parties (both sides phone numbers), time of connection through the starting time and the call duration, call event type, unique generated id for the record, etc. (Horak, 2008) *MobilityLog* class is a data structure that, emulates the essence of CDR. The attributes of the mobile logs are illustrated in Table 3 (also visible in the Figure 1):

6 RESULTS

The results are illustrated through the outcome of the logs generated by the simulator. The figure 4 is reflecting the interactions between the mobile devices, GPS devices and the cellular network.

6.1 GPS Logs

MSDevice_GPS related logs can be created in two ways: through main SUMO configuration file or

Table 3: Mobility Log Record Attributes.

Attribute	Description
timestamp	Amount of seconds since simulation started
event_type	CALL_INIT, CALL_REC, SMS_SEND, SMS_REC, WEB_COM, KEEP_ALIVE
imsi	subscriber ID (vehicle ID)
cgi	Common Gateway Identifier (CGI) the transmitter id (or simply a cell id)
signal_strength	radio signal strength during the event
latitude	latitude coordinate (vehicle actual location)
longitude	longitude coordinate (vehicle actual location)
speed	vehicle speed during the event (in m/s)

through SUMO route file. Listing 2 illustrates the GPS signal log, it contains timestamp, longitude, and latitude.

Listing 2: GPS coordinates (longitude and latitude) with a timestamp.

```
<gps-output>
 <vehicle id="1">
  <gps-device id="gps_1">
  <signal time="1.00">
   <coordinates latitude="58.29626476" longitude="26.44804043"/>
  </signal>
  <signal time="3.00">
   <coordinates latitude="58.29628805" longitude="26.44810251"/>
  </signal>
  <signal time="5.00">
   <coordinates latitude="58.29622327" longitude="26.44831755"/>
  </signal>
 </vehicle>
</gps-output>
```

6.2 Cellular Network Simulation Logs

The *MSDevice_Mobile* class generates logs in case the corresponding mobile device has configuration values set to true, and the mobile data events have been generated by *MESGEN*. In addition, the cellular network coverage file (XML) should be specified. Otherwise there would be no connectivity for the devices and the events cannot occur.

The configuration of the mobile devices is similar to the GPS devices. One must define the mobile device into the route file or one can set all the vehicles to carry a mobile device in the SUMO configuration file.

In the results, the cellular network behaviour simulation written into XML file are illustrated in Listing 3. The logging entity is defined in the Figure 1 (class *MobilityLog*) and is explained in the Section 5.6.

Listing 3: CDR-like mobility logs.

```
<mobile-output>
 <vehicle id="1">
  <event timestamp="92.00" eventType="KEEP_ALIVE"
   imsi="1" cgi="t-438-1" signalStrength="-21.01"
   latitude="58.29" longitude="26.46292491"
   speed="13.68143750"/>
  <event timestamp="94.00"
   eventType="SMS_REC"
   imsi="1" cgi="t-438-1" signalStrength="-21.70533381"
   latitude="58.29089506" longitude="26.46336419"
   speed="12.64468285"/>
 </vehicle>
 <vehicle id="27">
  <event timestamp="27.00" eventType="CALL_INIT"
   imsi="27" cgi="t-852-1" signalStrength="-21.11894073"
   latitude="58.25022813" longitude="26.43376626"
   speed="0.00000000"/>
  <event timestamp="28.00" eventType="CALL_INIT"
   imsi="27" cgi="t-852-1" signalStrength="-21.21370266"
   latitude="58.25021502" longitude="26.43379995"
   speed="2.45816080"/>
 </vehicle>
</mobile-output>
```
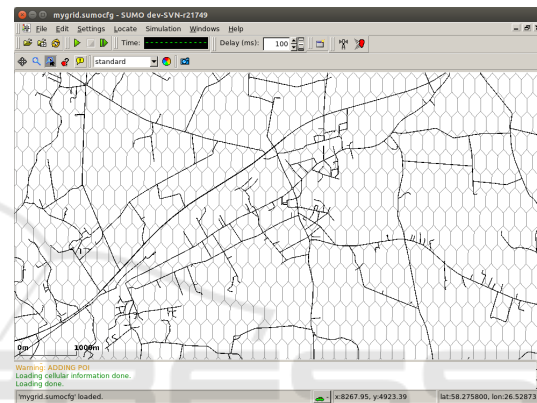

Figure 4: 5G standard proposes use of micro-cells, therefore an example of HexagonGen generated micro-cells in SUMO.

7 CONCLUSION

In this article, we are introducing a new layer into SUMO simulator. Our layer simulates the cellular network behavior and generates the logs of the interactions between the mobile devices and the mobile network. In addition, in order to make the integration with SUMO we created two main components: Mobility Event Simulation generator and mobile cell coverage generator. The results of our simulator is a set of mobile logs similar to CDR data that reflects the communication and the interactions between the vehicular mobile devices in the SUMO's vehicles and the cellular network. In general, the outcome is very encouraging and there is many enhancements that can be added; especially with in regards to the mobile network model for reflecting the mobile network protocols and behavior.

ACKNOWLEDGEMENT

REFERENCES

Bolla, R. and Davoli, F. (2000). Road traffic estimation from location tracking data in the mobile cellular network. In *Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE*, volume 3, pages 1107–1112. IEEE.

Bulut, E. and Szymanski, B. K. (2013). Wifi access point deployment for efficient mobile data offloading. *ACM SIGMOBILE Mobile Computing and Communications Review*, 17(1):71–78.

DLR (2017). DLR - Institute of Transportation Systems - SUMO Simulation of Urban MObility. http://www.dlr.de/ts/sumo. 2017-12-15.

Finkenzeller, K. and Müller, D. (2010). Rfid handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication. *UK: John Wiley & Sons, Ltd.*

Guttman, A. (1984). *R-trees: A dynamic index structure for spatial searching*, volume 14. ACM.

Hadachi, A., Batrashev, O., Lind, A., Singer, G., and Vainikko, E. (2014). Cell phone subscribers mobility prediction using enhanced markov chain algorithm. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 1049–1054. IEEE.

Hadachi, A., Mousset, S., and Bensrhair, A. (2013). Approach to estimate travel time using sparsely sampled gps data in urban networks. *Electronics Letters*, 49(15):957–958.

Horak, R. (2008). *Telecommunications and Data Communications Handbook, 2nd Edition*. Wiley.

Kai Hormann, A. A. (2001). The point in polygon problem for arbitrary polygons. *Computational Geometry*, 20(3):131–144.

Karnadi, F. K., Mo, Z. H., and Lan, K.-c. (2007). Rapid generation of realistic mobility models for vanet. In *Wireless communications and networking conference, 2007. WCNC 2007. IEEE*, pages 2506–2511. IEEE.

Kolici, V., Oda, T., Spaho, E., Barolli, L., Ikeda, M., and Uchida, K. (2015). Performance evaluation of a vanet simulation system using ns-3 and sumo. In *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on*, pages 348–353. IEEE.

Krajzewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. (2012). Recent development and applications of sumo-simulation of urban mobility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138.

Lind, A., Hadachi, A., and Batrashev, O. (2017). A new approach for mobile positioning using the cdr data of cellular networks. In *Models and Technologies for Intelligent Transportation Systems (MT-ITS), 2017 5th IEEE International Conference on*, pages 315–320. IEEE.

Nishith Tripathi, J. H. R. (2014). *Cellular Communications: A Comprehensive and Practical Guide*. Wiley. 1032 pages.

ns 2 (2017). The Network Simulator - ns-2. https://www.isi.edu/nsnam/ns/. 2017-12-15.

ns 3 (2017). ns-3. https://www.nsnam.org/. 2017-12-15.

OMNeT++ (2017). OMNeT++ Discrete Event Simulator - Home. https://www.omnetpp.org/. 2017-12-15.

OSM (2017). OpenStreetMap. https://www.openstreetmap.org. 2017-12-15.

Pan, B., Zheng, Y., Wilkie, D., and Shahabi, C. (2013). Crowd sensing of traffic anomalies based on human mobility and social media. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 344–353. ACM.

Piorkowski, M., Raya, M., Lugo, A. L., Papadimitratos, P., Grossglauser, M., and Hubaux, J.-P. (2008). Trans: realistic joint traffic and network simulator for vanets. *ACM SIGMOBILE mobile computing and communications review*, 12(1):31–33.

Sommer, C., Yao, Z., German, R., and Dressler, F. (2008). On the need for bidirectional coupling of road traffic microsimulation and network simulation. In *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, pages 41–48. ACM.

Statista (2017). Number of mobile phone users worldwide from 2013 to 2019. https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/. 2017-12-15.

Wedel, J. W., Schünemann, B., and Radusch, I. (2009). V2x-based traffic congestion recognition and avoidance. In *Pervasive Systems, Algorithms, and Networks (ISPAN), 2009 10th International Symposium on*, pages 637–641. IEEE.

Zaldivar, J., Calafate, C. T., Cano, J. C., and Manzoni, P. (2011). Providing accident detection in vehicular networks through obd-ii devices and android-based smartphones. In *Local Computer Networks (LCN), 2011 IEEE 36th Conference on*, pages 813–819. IEEE.