
Modelling Altitude Information in Two-Dimensional Traffic Networks for Electric Mobility Simulation

Diogo Santos¹, José Pinto, Rosaldo J. F. Rossetti, Eugénio Oliveira

University of Porto, PORTUGAL

*Correspondence to: diogo.ribeiro@fe.up.pt

ABSTRACT

Elevation data is important for electric vehicle simulation. However, traffic simulators are often two-dimensional and do not offer the capability of modelling urban networks taking elevation into account. Specifically, SUMO - Simulation of Urban Mobility, a popular microscopic traffic simulator, relies on networks previously modelled with elevation data as to provide this information during simulations. This work tackles the problem of adding elevation data to urban network models - particularly for the case of the Porto urban network, in Portugal. With this goal in mind, a comparison between different altitude information retrieval approaches is made and a simple tool to annotate network models with altitude data is proposed. The work starts by describing the methodological approach followed during research and development, then describing and analysing its main findings. This description includes an in-depth explanation of the proposed tool. Lastly, this work reviews some related work to the subject.

Keywords

intelligent vehicle software,
eco-driving,
energy-efficient vehicles,
vehicle simulation systems,
urban networks,

DOI: [10.20897/lectito.201635](https://doi.org/10.20897/lectito.201635)

INTRODUCTION

This work was motivated by a presentation with the theme "Assessing the Performance of Electric Buses: a study on the impacts of different routes" by Deborah Perrotta, based on some of the author's previous work (Perrotta et al., 2012, 2014). The presentation alerted to the importance of route elevation in electric vehicle simulation and the general lack of this information on many simulations, giving rise to the main issue tackled by this work: How can an elevation parameter be added to SUMO-based urban network simulations? Elevation data is important when simulating electric vehicles due to this type of vehicle's dependency on road topology, when considering its efficiency. In general, a vehicle ascending a road will put the engine under more stress than when driving in a slope-less road or in a downward direction, increasing energy consumption. In addition, electric vehicles are often equipped with energy recuperation mechanisms which can partially recharge the battery when braking or otherwise descending a road.

In particular, this problem was approached having urban networks from Porto, Portugal, as basis. Given the background of the problem, it is possible to contextualize this work in three major themes: Electric vehicles, intelligent transportation systems and simulation and modelling.

The main motivation of this work is then to improve transport systems simulation (in particular in the SUMO - Simulation of Urban Mobility (DLR - Institute of Transportation Systems, 2015d) platform) by adding elevation data to these simulations. Once completed, this project is expected to play a part in simplifying and extending the process of modelling elevation data in these simulations, simultaneously providing more accurate metrics. As a direct result, it is expected some enhancement to the precision of electric vehicle simulation in urban networks, as increased energy consumption and recuperation events related to road topology should be more precisely modelled.

In order to comply with such expectations, certain objectives were taken into consideration. The main goal considered was to find a pragmatic way of adding elevation data to strictly two-dimensional transportation systems simulations. More specifically, the work entitles some research on already existing approaches to this problem. In

addition, it was set as another objective to examine the viability of generating topographic information using manually acquired GPS data. Finally, the elevation data needs to be integrated with SUMO.

Section II describes in detail the approach taken in developing this work. Section III details the main findings during development along with some analysis. Next, Section IV makes a short description of some of the most important work found relating to this project. Lastly, Section V reviews some of the main points and contributions of this work, along with relevant future work to be developed.

Methodological Approach

In general, the approach followed for this project started by the research and investigation on how the problem of a third dimension was solved in other work that used simulations based on SUMO. This research had the objective of analysing what kind of techniques could be used on this work. After this, it was important to select an appropriate region of Porto to use as the base model to find altitude data for. Then come phases related to retrieving the actual altitude data using any of the approaches found during the first phase, modelling that data and integrating it with the network model in SUMO. Part of the approach also involved the setup and configuration of the simulation platform, comprised of SUMO, the High Level Architecture and Simulink (these last items described in section IV-A).

The following subsections describe in further detail what each of the phases of the work entailed.

Research for possible approaches

The main objective of this research phase was to gain further knowledge regarding simulation on the SUMO platform, how to configure and use it correctly and how the problem of three-dimensional simulation was addressed by other users. The main finding was that SUMO already supports three dimensional information to a certain degree: if this information is provided in the network file, simulations take the elevation data into consideration, including the capability of displaying in the graphical user interface information regarding inclination and height for each road segment. The main issue with support for elevation data on SUMO is that this data must be somehow provided when creating the network file, in order for the network to take that information into account. So there still remains a problem to solve: how to retrieve that information and provide it in a way that can be used by SUMO?

The following step consisted on solving this issue. Research brought to attention three possible approaches: manually capturing altitude information using a GPS device, using a combination of *OpenStreetMap* maps and data from the Shuttle Radar Topography Mission to retrieve altitude for our region of interest and, lastly, using the Google Maps Elevation API, by Google. More information about these possible approaches can be found in section II-C.

Selection of the urban network

Contemplating the possibility of manually capturing GPS data, it was of interest to select a region easy to visit if needed, but also with relevant elevation features to be captured, such as multiple ascents and descents of varying degrees of inclination. With this in mind, the urban network of the Porto down-town area, in particular the Aliados zone was chosen. Some previous work by Macedo et al. (Macedo et al., 2013a, 2013b, 2013c) contemplated the modelling of this zone into a SUMO network file ready to be loaded into the simulator, but lacking any sort of elevation data already integrated into it. As such, the present work made use of that network file.

Elevation data retrieval

This phase contemplated the exploration of three different approaches to retrieving elevation data. This subsection intends to define how each of these approaches can be used. In section III-A, a general comparison of pros and cons for each of these methods is given.

Manual GPS information capture: The main idea behind this approach is to use a commercial GPS device to capture GPS data, which includes altitude information. Similarly, some other attempts have tried to infer route geometry using GPS logs (Freitas et al., 2010). Since the work of this paper is related to electric vehicles - in particular, electric buses - it would be interesting to capture this GPS data along common public transportation routes in the Aliados network.

However, capturing a significant amount of GPS points for a good modelling of the elevation along a route can be a time-consuming task. In addition, consumer-grade GPS devices are considered to be inaccurate at determining altitude information. For both these reasons, despite being considered, this approach ended up not being used for the work, albeit being something to consider exploring in the future.

OpenStreetMap and the Shuttle Radar Topography Mission: OpenStreetMap (OSM) (OpenStreetMap Foundation, 2015) is a free, editable world map being collaboratively built by volunteers, offered to the general public with an open-content license. Community efforts have made this map very accurate and detailed for many parts of the world, being a good source of information related to urban networks. As such, it is possible to import OSM map files into SUMO, in order to automatically generate accurate and detailed networks for the simulations. However, the OSM maps also lack accurate altitude information for most of the world. Nonetheless, the community has developed several tools to gather altitude information. In particular, a plugin allows linking the map to NASA's

Shuttle Radar Topography Mission (SRTM) (NASA Jet Propulsion Laboratory, 2015) data and retrieve precise altitude information for most regions of the world. The SRTM is a mission conducted since the year 2000 to obtain elevation data for most of the world with high precision and resolution and making it available to the general public.

Despite using OSM and the SRTM plugin and then directly importing the data to SUMO being simple to do and accurate regarding three-dimensional information, this approach does not allow adding the altitude value to pre-existent SUMO network files. Fully modelling an urban network to use in SUMO can be a time demanding task and disregarding old network files and starting anew in order to use an OSM file for altitude purposes may not be viable. Network files generated by OSM file conversion are often created with issues that require user maintenance to patch, such as inaccurate road representations, which translate into an added effort to use this method of network creation. As such, it is important to have an alternative to retroactively, and automatically, add the altitude information to previously existing and maintained network files.

Google Maps Elevation API: The Google Maps Elevation API (Google, 2015) allows the retrieval of precise altitude information for any point of the world whose data Google possesses. The API receives (longitude, latitude) coordinates and retrieves the altitude for them. It allows querying for single points, multiple ones in bulk or along a path defined by a start and ending point.

Using this API has the advantage of directly accessing Google's data, which is known for its precision and availability all around the globe. Nonetheless, a disadvantage of the API is related to free access limits. Free access to the API is limited to 2500 requests per day, a maximum of 512 locations per request and a total of 10 requests per second, which may make the usage of tools resorting to this API not viable for certain situations.

Nonetheless, with the objective of pre-processing and annotating SUMO network files with altitude data before running the simulations, the API was chosen to be used for the purpose of this work.

Network altitude model creation and integration of the model with SUMO

Altitude data is combined with the original SUMO network file in order to create a new network file enhanced with three-dimensional information. These recreated files can be considered the altitude model for the urban networks, as the simulation can extract needed altitude information from it. Since SUMO simulations rely on the network files to function, that part of the integration is set by default.

However, the developed work takes into account the possibility of, in the future, interacting with SUMO externally. As such, single point lookup capabilities were implemented and questions addressing how to store data for future queries were thought about. Conclusions regarding future work related to this issue are described in section V. Although the possibilities mentioned in the future work are not yet implemented, the development was made with future extensibility in mind.

Urban network simulation

To allow for future testing and integration of the altitude data into more complex simulation systems, the simulation architecture proposed by Macedo et al. (Macedo et al., 2013b, 2013c) was installed and configured. This architecture comprises in the interconnection of the SUMO simulator with Simulink models, through a High Level Architecture (in the case of this work, Pitch pRTI (Pitch Technologies, 2015)).

The configured system seemed to run properly after setup. However, since the simulations did not implement usage of altitude information from SUMO yet, no testing of its performance was made.

RESULTS & ANALYSIS

General comparison between methods

Section II mentioned that several elevation data retrieval approaches were taken into account and analysed. Table 1 summarizes the main advantages and disadvantages for each of the methods, found during research.

The elevation data retrieval tool

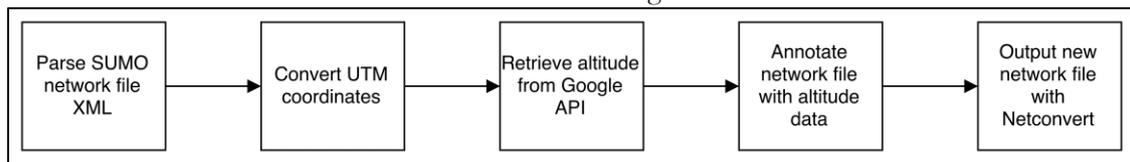
The main practical output of this work was the development of a tool to automatically retrieve altitude data given coordinates in SUMO's projection space. Specifically, this tool is able to either read a previously created SUMO network file and create a new one, annotated with altitude data, or receive a single two-dimensional (x, y) coordinate and retrieve the corresponding elevation data.

The architecture of the tool is simple and modular. The parser / writer module parses the original SUMO network files in order to extract coordinate information and is responsible to recreate the file with altitude information. A geographic projection converter manages the conversion of coordinates in the network files to usable latitude and longitude projections. A basic *http* module queries the Google API for the needed altitude information. Lastly, our work interacts with SUMO's native *NETCONVERT* tool to merge the altitude data to the network file, in the correct format.

General Comparison Of Elevation Retrieval Approaches

	Pros	Cons
Manual GPS	+ Keep data for offline access; + Adapt to region of interest; + Possible path reconstruction.	- Time consuming; - May need extra complex manipulation; - Inaccurate.
OSM + SRTM	+ OSM is free, open content; + Highly detailed networks; + Can extract just a region of the world; + Fast processing for big networks.	- Cannot integrate directly with SUMO; - Cannot be used to extend old network files.
Google API	+ Accurate data; + Near-global availability; + Ease of use.	- Limited free access; - Slow for real time; - No offline access.

At the time of writing, the execution flow of the tool for network file handling, in general, is as follows, summarized in Fig. 1:

**Summary of the tool's execution flow.**

Parse the SUMO XML network file to extract the two-dimensional (x, y) coordinates of the relevant nodes, as well as information about the geometric projection used by SUMO (in particular, the “offsets” for the 'x' and 'y' coordinates (DLR - Institute of Transportation Systems, 2015c));

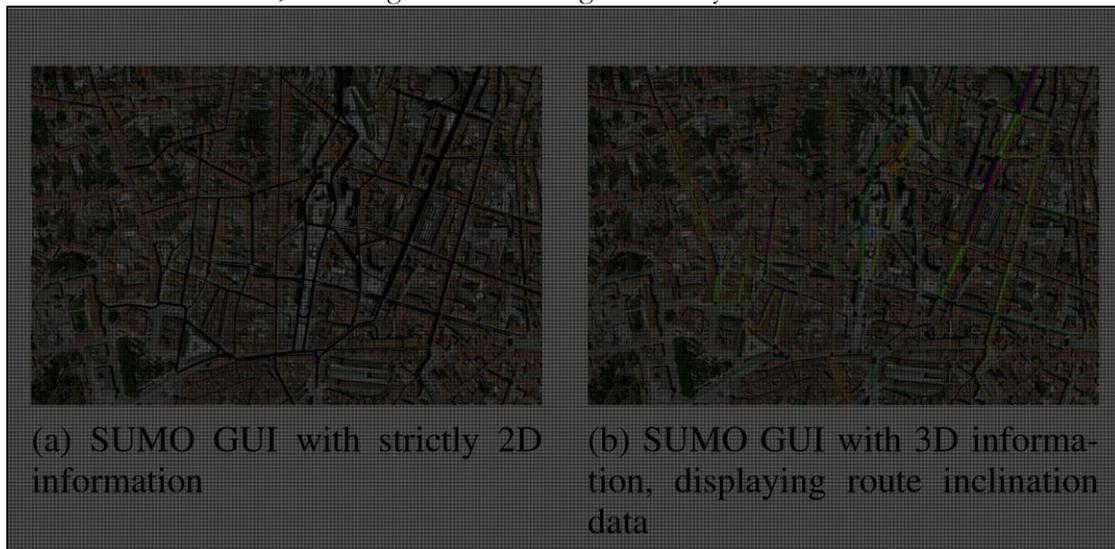
Convert each of the obtained coordinates to a latitude and longitude based projection. Currently, the tool supports converting from Universal Transverse Mercator (UTM) coordinates (MapTools, 2015), as it is the projection found in the network files used for testing during development. The Java library Proj4J (OSGeo Foundation, 2015) is used to implement the geographic projection conversions;

Each one of the converted points is bundled into a single query to the Google Maps Elevation API, mentioned in section II-C.3, which retrieves the elevation data for every point. A single query is performed instead of a query per point, in order to save on the number of API calls executed for each converted file;

A new, temporary, XML network file is created, with the same information as the original file, but with a 'z' dimension added to the previously two-dimensional nodes, containing the elevation data;

Lastly, SUMO's NETCONVERT tool (DLR - Institute of Transportation Systems, 2015b) is used on the aforementioned temporary network file. NETCONVERT uses the three-dimensionally annotated node information to generate a new network file with all the expected elevation data.

Fig. 2 shows a comparison of the SUMO GUI without any kind of three-dimensional data and with provided elevation data, colouring the network edges relatively to their inclination ratio.



General comparison of elevation retrieval approaches.

It is also possible to use the tool to retrieve data for a single coordinate, without providing any SUMO network files. In this case, the user needs only provide a UMT coordinate and the corresponding “offsets” for the 'x' and 'y' coordinates. The execution flow is similar to the aforementioned one, but in this case only the coordinate conversion step is performed before querying the API with the single point. Then, the retrieved elevation data is displayed to the user in the system console. Such an approach can be useful to extend multi-paradigm, multi-resolution, and multi-perspective simulation, integrating different models as proposed by other literature (Rossetti & Bampi, 1999; Rossetti et al., 2008; Timóteo et al., 2010; Pereira & Rossetti, 2012).

Google Maps Elevation API access times

In order to explore a possible real-time usage of the developed tool integrated with a SUMO simulation, the average API access times, over 10 attempts, for different number of points were measured. Real-time usage could be interesting to explore in order to, instead of previously annotating the complete network file, query for the information as needed, during the simulation. The results are shown in Table II. It is possible to observe that the access times remain approximately constant regardless of the number of points sent in the query. However, these access times are close to 0.8 seconds, which can be considered slow.

Google Maps Elevation API Access Times

Number of points	Round trip time (milliseconds)
1	730
100	710
150	780
200	725
250	777
300	738

Related Work

This section summarizes two of the main works this paper is based on.

A HLA-based multi-resolution approach to simulating electric vehicles in simulink and SUMO

This work defines a High Level Architecture (HLA) to interconnect different simulation systems (Macedo et al., 2013c). In specific, the HLA integrates a Matlab / Simulink (The MathWorks Inc., 2015) model of an electric bus subsystem with a microscopic traffic simulation using SUMO.

Using the referred system it is possible to make a real-time analysis of electric vehicle performance under dynamic conditions. The work was tested on a scenario based in the Porto Aliados bus network, in Portugal. This is the same urban network used for the experimentations on the present work.

Despite the importance of elevation data for precise electric vehicle simulation, it does not take into consideration this dimension. The search for a way to correct that shortcoming motivated this work.

Besides using the same urban network for the simulations, the developments of this work were implemented with recourse to a similar system, in particular the Simulink + SUMO over HLA approach. This paper was found to be relevant to the present work due to the implemented HLA system. Currently, the developed tool mentioned in section III-B does not make use of the system's functionality as it interacts with SUMO on a network file pre-processing basis. However, future work contemplates the usage of the tool as a real-time standalone altitude retrieval, to communicate both with SUMO and Simulink to provide altitude information when needed (and not already available on the SUMO simulation). If successful, such an improvement could be considered an important complement to the work of Macedo et al. (Macedo et al., 2013c), providing a more robust electric vehicle simulation platform.

Electric vehicle simulator for energy consumption studies in electric mobility systems

In this work (Maia et al., 2011), an electric vehicle model implemented directly in SUMO is suggested, as opposed to the Simulink model mentioned in the previously. At the time of writing of that paper, SUMO was lacking a direct way to integrate electric vehicles in its simulation capabilities. As such, the paper proposed an extension to the simulator's capabilities by adding electric vehicle models, validated with a series of tests. It is worth noting, though, that as of version 0.24, released in 2015, SUMO offers native electric vehicle simulation capabilities, along with electric charging stations data (DLR - Institute of Transportation Systems, 2015a). As of yet, advantages of using SUMO's native capabilities in place of the approach proposed by R. Maia et al. remain to be analysed by the authors.

Despite the aforementioned electric vehicle simulation capabilities in the current SUMO versions, the paper was still found to be relevant regarding the addition of elevation data in the simulations. In particular, it proposes the integration of an independent altitude information file with the SUMO network using SUMO's

NETCONVERT tool, which was already mentioned in section III-B. The work showed that SUMO already provided some native capabilities to integrate elevation data and motivated the usage of SUMO's NETCONVERT for the tool mentioned in section III-B. It is worth mentioning, however, that this work did not specify how that altitude information was obtained, how the file was structured nor exactly what kind of integration NETCONVERT did with the file. In contrast to the work developed by R. Maia et al., the current work and the developed tool intended to offer a more extensive explanation of how elevation data is obtained and integrated into SUMO, as well as provide a way to retroactively extend SUMO network files with altitude data, without any pre-prepared altitude information files.

CONCLUSIONS

Relating to the compared elevation data retrieval approaches, it was possible to see that all three have their advantages and disadvantages. In particular, the Google Maps API can be considered fast to process large bulks of data, however no clear advantage in API call time is obtained by querying for a single point. Since the average call time reaches near 0.8 seconds, this approach may not be suitable for real-time usage. However, the data obtained using the API is the same used in Google Earth / Google Maps which, in general, means good quality data. In addition, the data availability is high, with most of the world being mapped and automatic data interpolation being made in cases of unavailable points.

Regarding the *OpenStreetMap* integrated with the Shuttle Radar Topography Mission, one can consider it to be a good option to retrieve altitude data, especially because the SRTM data is very accurate. However, in order to use this approach, it is necessary to start by extracting a mapped region from *OpenStreetMap*, retrieve the altitude data using SRTM and only then generate the SUMO network file. This means that this approach cannot annotate previously existing network files. Since these files can be time-consuming to create, disregarding previous work in order to start anew using OSM + SRTM may not be viable. In addition, real time usage is not directly possible.

Lastly, when considering manual GPS altitude data capturing, one must take into account that consumer-grade GPS devices are considered to be inaccurate when acquiring altitude information. In addition, to minimize the necessity of interpolating data, which leads to even further inaccuracy, it is necessary to capture a large number of points. Since this capture needs to be done in-site, it can be a very time-consuming task. Nonetheless, keeping a local database of GPS data minimizes the need for online tools, which reduces access times and reliability issues, in turn making it a great alternative for real-time usage.

Regarding the possibility of using a local geospatial database, some considerations have been made. It has been considered integrating one of these databases within the developed tool, in order to store captured GPS coordinates. Besides the previously mentioned offline access capability, it could be possible to reconstruct travelled tracks and integrate that functionality with SUMO networks. In addition, information from other altitude data retrieval approaches could be stored as well, meaning online API accesses would only need to be made once per point and possibly pre-loaded before SUMO simulations, to allow for real-time usage.

At the moment, an initial version of the proposed tool has been implemented, capable of annotating a SUMO network file with altitude information retrieved from the Google Maps Elevation API. The resulting file is directly usable in any future SUMO simulations. As it stands, it is also possible to use the tool to perform altitude queries for single points (as opposed to annotating an entire network file). The tool should be open-sourced and made publicly available soon, integrated in the *MAS-Ter Lab* specification (Rossetti et al., 2002, 2005, 2007). As main contributions from the developed work, the general comparison between altitude information retrieval techniques and the developed tool can be considered. Hopefully, these can help improve the precision of urban network simulation, in particular for electric vehicle simulation cases. However, the tool has other usages in arguably different environments, such as those using serious games and gamification strategies as modelling support tools (Rossetti et al. 2013; Silva et al., 2013), as these can benefit from a more realistic representation of the network.

Regarding possible future work, the main priority would be implementing the local geospatial database, possibly using the *PostGIS* (PostGIS Project Steering Committee, 2015) technology. Using this database, the information retrieved from the Google Maps Elevation API could be stored for future offline usage and, additionally, implement some of the previously mentioned capabilities. In particular, *PostGIS* was considered for its data interpolation and track information storage capabilities “out of the box”.

In addition, it would be important to try and implement a *High Level Architecture Federate* for the developed tool in order to allow real-time usage in conjunction with SUMO. Further experimentation would then be needed to attest the viability of this. Lastly, some direct improvements to the developed tool's code could be made. In particular, some of the geographic projection parameters are “hard-coded” when they could be extracted from the SUMO network file.

REFERENCES

- D. Perrotta, B. Ribeiro, R.J.F. Rossetti, J.L. Afonso, "On the potential of regenerative braking of electric buses as a function of their itinerary," *Procedia-Social and Behavioral Sciences*, vol. 54, pp. 1156-1167. 2012.
- D. Perrotta, J. L. Macedo, R. J. Rossetti, J. F. de Sousa, Z. Kokkinogenis, B. Ribeiro, and J. L. Afonso, "Route Planning for Electric Buses: A Case Study in Oporto," *Procedia - Social and Behavioral Sciences*, vol. 111, pp. 1004–1014, feb 2014.
- DLR - Institute of Transportation Systems, "Models/Electric - SUMO - Simulation of Urban Mobility," 2015. [Online]. Available: <http://www.sumo.dlr.de/userdoc/Models/Electric.html>
- DLR - Institute of Transportation Systems, "NETCONVERT - Sumo," 2015. [Online]. Available: <http://sumo.dlr.de/wiki/NETCONVERT>
- DLR - Institute of Transportation Systems, "Networks/SUMO Road Networks - Sumo," 2015. [Online]. Available: http://sumo.dlr.de/wiki/Networks/SUMO_Road_Networks#Coordinates_and_alignment
- DLR - Institute of Transportation Systems, "SUMO - Simulation of Urban Mobility," 2015. [Online]. Available: http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/
- Google, "The Google Maps Elevation API | Google Maps Elevation API | Google Developers," 2015. [Online]. Available: <https://developers.google.com/maps/documentation/elevation/intro>
- I.J.P.M. Timóteo, M.R. Araújo, R.J.F. Rossetti, E.C. Oliveira, "TraSMAPI: An API oriented towards Multi-Agent Systems real-time interaction with multiple Traffic Simulators," in *13th International IEEE Conference on Intelligent Transportation Systems (ITSC 2010)*. IEEE, 2010, pp. 1183-1188.
- J. L. P. Macedo, "An integrated framework for multi-paradigm traffic simulation," 2013. [Online]. Available: <http://repositorio-aberto.up.pt/handle/10216/72541>
- J. Macedo, S. Guilherme, Z. Kokkinogenis, D. Perrotta, and R. J. F. Rossetti, "A Framework for Electric Bus Powertrain Simulation in Urban Mobility Settings: coupling SUMO with a Matlab/Simulink nanoscopic model," in *1st SUMO User Conference 2013*. Deutsches Zentrum für Luft- und Raumfahrt e.V. Institut für Verkehrssystemtechnik Rutherfordstraße 2, 12489 Berlin-Adlershof, 2013, pp. 95 – 102.
- J. Macedo, Z. Kokkinogenis, G. Soares, D. Perrotta, and R. J. F. Rossetti, "A HLA-based multi-resolution approach to simulating electric vehicles in simulink and SUMO," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 2367–2372.
- J.L.F. Pereira, R.J.F. Rossetti, "An integrated architecture for autonomous vehicles simulation," in *Proc. of the 27th annual ACM symposium on applied computing*, 2012, pp. 286-292.
- Jorge F Silva, João Emílio Almeida, Rosaldo JF Rossetti, Antonio Leca Coelho, "Gamifying evacuation drills," in *8th Iberian Conference on Information Systems and Technologies (CISTI 2013)*, IEEE, 2013, pp. 1-6.
- MapTools, "More details about the UTM coordinate system," 2016. [Online]. Available: <https://www.maptools.com/tutorials/utm/details>
- NASA Jet Propulsion Laboratory, "Shuttle Radar Topography Mission," 2015. [Online]. Available: <http://www2.jpl.nasa.gov/srtm/index.html>
- OpenStreetMap Foundation, "OpenStreetMap," 2015. [Online]. Available: <https://www.openstreetmap.org>
- OSGeo Foundation, "Proj4J," 2015. [Online]. Available: <https://trac.osgeo.org/proj4j/>
- Pitch Technologies, "Pitch pRTI Free," 2015. [Online]. Available: <http://www.pitch.se/downloads/pitch-prti-free>
- PostGIS Project Steering Committee, "PostGIS — Spatial and Geographic Objects for PostgreSQL.", 2015. [Online]. Available: <http://postgis.net/>
- R. Maia, M. Silva, R. Araujo, and U. Nunes, "Electric vehicle simulator for energy consumption studies in electric mobility systems," in *2011 IEEE Forum on Integrated and Sustainable Transportation Systems*. IEEE, jun 2011, pp. 227–232.
- R.J.F. Rossetti and S. Bampi, "A software environment to integrate urban traffic simulation tasks," *Journal of Geographic Information and Decision Analysis*, vol. 3, no. 1, pp. 56-63. 1999.
- R.J.F. Rossetti, E.C. Oliveira, A.L.C. Bazzan, "Towards a specification of a framework for sustainable transportation analysis," in *Proc. of the 13th Portuguese Conference on Artificial Intelligence*, Guimarães, Portugal, Dec. 3-4, 2007, pp. 179-190.
- R.J.F. Rossetti, J.E. Almeida, Z. Kokkinogenis, J. Gonçalves, "Playing transportation seriously: Applications of serious games to artificial transportation systems," *IEEE Intelligent Systems*, vol. 28, no. 4, pp. 107-112. 2013.
- R.J.F. Rossetti, P.A.F. Ferreira, R.A.M. Braga, E.C. Oliveira, "Towards an artificial traffic control system," in *11th International IEEE Conference on Intelligent Transportation Systems (ITSC 2008)*. IEEE, 2008, pp. 14-19.
- R.J.F. Rossetti, R. Liu, "An agent-based approach to assess drivers' interaction with pre-trip information systems," *Intelligent Transportation Systems*, vol. 9, no. 1, pp. 1-10. 2005.
- R.J.F. Rossetti, R. Liu, H.B.B. Cybis, S. Bampi, "A multi-agent demand model," in *Proc. of the 13th Mini-Euro Conference and the 9th Meeting of the Euro Working Group Transportation*, 2002, pp. 193-198.

T.R.M. Freitas, A. Coelho, R.J.F. Rossetti, "Correcting routing information through GPS data processing," in *13th International IEEE Conference on Intelligent Transportation Systems (ITSC 2010)*. IEEE, 2010, pp. 706-711.

The MathWorks Inc., "Simulink - Simulation and Model-Based Design," 2015. [Online]. Available: <https://www.mathworks.com/products/simulink/>