

High-Precision Design of Pedestrian Mobility for Smart City Simulators

Piergiorgio Vitello[‡], Andrea Capponi[†], Claudio Fiandrino^{*}, Paolo Giaccone[‡],
Dzmitry Kliazovich[♦], and Pascal Bouvry[†]

[‡] Dip. di Elettronica e Telecomunicazioni, Politecnico di Torino, Italy,

[†] FSTC-CSC, University of Luxembourg, Luxembourg, ^{*} IMDEA Networks Institute, Madrid, Spain, [♦] ExaMotive, Luxembourg

E-mails: [‡] piergiorgio.vitello@studenti.polito.it, paolo.giaccone@polito.it, [†] {firstname.lastname}@uni.lu,

^{*} claudio.fiandrino@imdea.org, [♦] kliazovich@ieee.org

Abstract—The unprecedented growth of the population living in urban environments calls for a rational and sustainable urban development. Smart cities can fill this gap by providing the citizens with high-quality services through efficient use of Information and Communication Technology (ICT). To this end, active citizen participation with mobile crowdsensing (MCS) techniques is a becoming common practice. As MCS systems require wide participation, the development of large scale real testbeds is often not feasible and simulations are the only alternative solution. Modeling the urban environment with high precision is a key ingredient to obtain effective results. However, currently existing tools like OpenStreetMap (OSM) fail to provide sufficient levels of details. In this paper, we apply a procedure to augment the precision (AOP) of the graph describing the street network provided by OSM. Additionally, we compare different mobility models that are synthetic and based on a realistic dataset originated from a well known MCS data collection campaign (ParticipAct). For the dataset, we propose two arrival models that determine the users' arrivals and match the experimental contact distribution. Finally, we assess the scalability of AOP for different cities, verify popular metrics for human mobility and the precision of different arrival models.

I. INTRODUCTION

The development of smart cities has become a major focus for researchers. The world population living in cities is witnessing an unprecedented growth. While only 10% of the population lived in cities during 1900, nowadays this percentage corresponds to 50% and is projected to increase reaching 66% by 2050 [1]. Hence, a sustainable development is a pillar component for municipalities to support city growth. Currently, urban environments occupy nearly 2% of the world's surface and cities contribute to 80% of global gas emission, 75% of global energy consumption and 60% of residential water use [2]. With the aim of improving citizens' quality of life, significant research efforts are undergoing to provision citizens innovative and sustainable solutions for public services such as healthcare and wellbeing, safety and smart transportation among the others. The application of the Internet of Things (IoT) paradigm is regarded as a key component for making real the smart city vision [3], [4]. Mobile crowdsensing (MCS) includes citizens in the loop and has been proven as a win-win strategy [5]. Mobility and intelligence of human participants guarantee higher coverage and better context awareness, if compared to traditional sensor networks. In addition, users maintain by themselves the mobile devices and provide periodic recharge.

MCS has become one of the most prominent paradigms for urban sensing [6]. Users¹ contribute data generated from sensors embedded in mobile devices, including smartphones, tablets and IoT devices like wearables. Accelerometer, gyroscope, magnetometer, GPS, microphone and camera are a representative set of sensors which are commonly employed to operate a number of applications in several domains such as health care, environmental and traffic monitoring and management [7], [8]. To illustrate with a few examples, HazeWatch [9] relies on active citizen participation to monitor air pollution and is currently employed by the National Environment Agency of Singapore on a daily basis. Creekwatch [10] is an application for smartphones developed by the IBM Almaden research center. It allows the monitoring of the conditions of watershed through crowdsensed collected data. Garbage Watch [11] employs citizens to monitor the content of recycling bins to enhance the recycling program.

As for proper operation MCS systems require the contribution from a large number of participants [12], [13], the development of large scale testbeds is often not feasible [14]. CrowdSenSim [15], the first simulator for MCS systems, was designed to fill this gap by providing the researchers a tool capable of performing large scale simulations over realistic urban environments. For example, its effectiveness has been demonstrated to evaluate performance of city-wide solutions for public street lighting [16], energy efficient data collection [17] and user recruitment in crowdsensing campaigns [18]. To perform effective simulations in complex environments, such as modern cities, the simulation platform has to be scalable while providing at the same time precise and detailed information.

In this paper, we model pedestrian mobility with high level precision for use in crowdsensing smart city simulators. We propose to use a procedure to augment the OSM precision (AOP): it builds upon the existing OSMnx library [19] (an extension of OSM²) and has been included in CrowdSenSim. We describe an exact solution (V-AOP) based on Vincenty's formula [20] and an approximated one, based on linearized version (L-AOP).

Then, we analyze two different users' arrivals models.

¹In the remainder of the paper, we use the terms citizen, participant and user interchangeably.

²Available on: <https://www.openstreetmap.org>

In U-MOB, user arrivals are uniformly distributed over the simulation period. In D-MOB, user arrivals are based on the ParticipAct dataset [21] that provides profiles in terms of the number of per hour user contacts. For the sake of anonymity, such information is provided in generic fashion to obfuscate the details of the original user trajectories. This significantly complicates the reverse engineering process of computing the overall number of users that generated a given number of contacts. Indeed, from two overlapping user trajectories, multiple contacts can occur.

Through performance evaluation conducted with CrowdSenSim, we verify the effectiveness of the adopted AOP approach and we evaluate the accuracy obtained by the two arrival models by measuring metrics commonly employed for pedestrian mobility [22], such as the distribution of the contact rate and of the contact stability.

Our main findings are as follows:

- AOP procedure allows to obtain the street network graph at any desired level of precision.
- For medium and small cities, both V-AOP and L-AOP versions for AOP require the same computational time. L-AOP becomes faster only for very large urban environments.
- The U-MOB arrival model approximates very well the D-MOB one. As a consequence, for simulations with large population, uniform user arrivals provide a good approximation to mimic realistic contacts between users.

II. BACKGROUND AND MOTIVATION

Urban planners commonly rely on spatial distribution of citizens or locations to inspect the complex dynamics of urban environments. For example, through analysis of Foursquare checkpoints, Daggit et al. [23] explore how cities grow and highlight that spatial correlation is one of the most important growth factors. To determine the most prominent location to open a new activity, Jenses [24] proposed a mathematical model that characterizes the potentiality of locations. For example, the candidate locations to open a bakery are those that mimics the features of the average locations of existing bakeries. Another example is [25], where the authors analyze how the growth and positions of craft breweries is related with social and demographic evolution. Such relation is captured by analyzing the incidence of the neighborhood change over time.

In urban planning, the sole information of the location of specific points of interest over time such as Foursquare check-ins can be sufficient. Vehicular traffic simulators like SUMO [26] base the mobility models on spatial vectors and rely on tools like NETCONVERT, which turns a OSM map into a set of edges and nodes defining respectively roads and road intersections. In order to be scalable and to obtain meaningful statistical results [15], MCS simulators require the complete path of all users participating and non-participating in the campaign to be known in advance. For example, relying on spatial vectors would increase the amount of computation performed over runtime to determine user positions with fine time precision, lowering the scalability. This is a key element

to correctly compute the precise amount of collected data and its associated energy cost under specific policies for data collection [27]. CrowdSenSim defines the trajectory x of each user as a sequence of n steps x_1, \dots, x_n with $n > 1$, where each of the x_i steps uniquely defines the location in terms of time and spatial references, i.e., latitude and longitude. This definition is in the spirit of previous research on human mobility [28]. The vertices of street network graphs have to be close one with each other to ensure high precision of the sequence of n steps defining user trajectories. Furthermore, CrowdSenSim knows about the precise location of access points, e.g., 3G and WiFi, for communication purposes [27] and additional on-demand information typical of the application under investigation like the location of lampposts for public street lighting [16].

OSMnx [19] is a Python package that allows the researchers to easily download and analyze street networks of any location in the world from OSM. OSMnx allows to download walking/biking/driving maps, e.g., walking maps, define each sidewalk, footpath and road crossing in a city. Hence, combining CrowdSenSim with OSMnx would result in a win-win strategy to enhance the flexibility of the simulator. To illustrate, OSMnx-based maps would distinguish the two sidewalks of a street, while the previous version of the simulator only sees the street.

Unfortunately, OSM street nodes are inconsistent for direct use in CrowdSenSim because they include dead-ends, intersections and all the points in a segment when streets curve. OSMnx automatically simplifies and corrects topology through an algorithm, removing those points and unifying each resulting set of sub-edges into single edges. The approach is commonly used in different research domains to limit the amount of data to be analyzed [29]–[31]. However, for MCS purposes the resulting topology lacks of a sufficiently fine-grained level of detail. First, the distance between two nodes in the graph can be excessively large, making user trajectories to be captured with few intermediate points. Second, any statistical information on the amount of sensed data would aggregate in a limited number of locations, making the results not very informative.

The closest to our work is [32]. The authors propose an axial line methodology, defined as the minimum number of relatively straight lines that intersect along urban streets. The methodology captures the urban morphology with arbitrary precision, but the street networks are not defined in terms of a detailed set of latitude and longitude coordinates.

III. THE AOP PROCEDURE

OSM provides the graph of the street network $G_{\text{OSM}} = (V, E)$, where V is the set of vertices or nodes and E the set of edges. Each node consists of different attributes and the main ones are the following: a unique identifier called *osmid*, the latitude (y), the longitude (x). Each edge consists of the following attributes: access, bridge, highway, lanes, maximum speed, name, oneway, osmid, service, tunnel, u , v , width, where u and v are the osmid of the adjacent nodes of

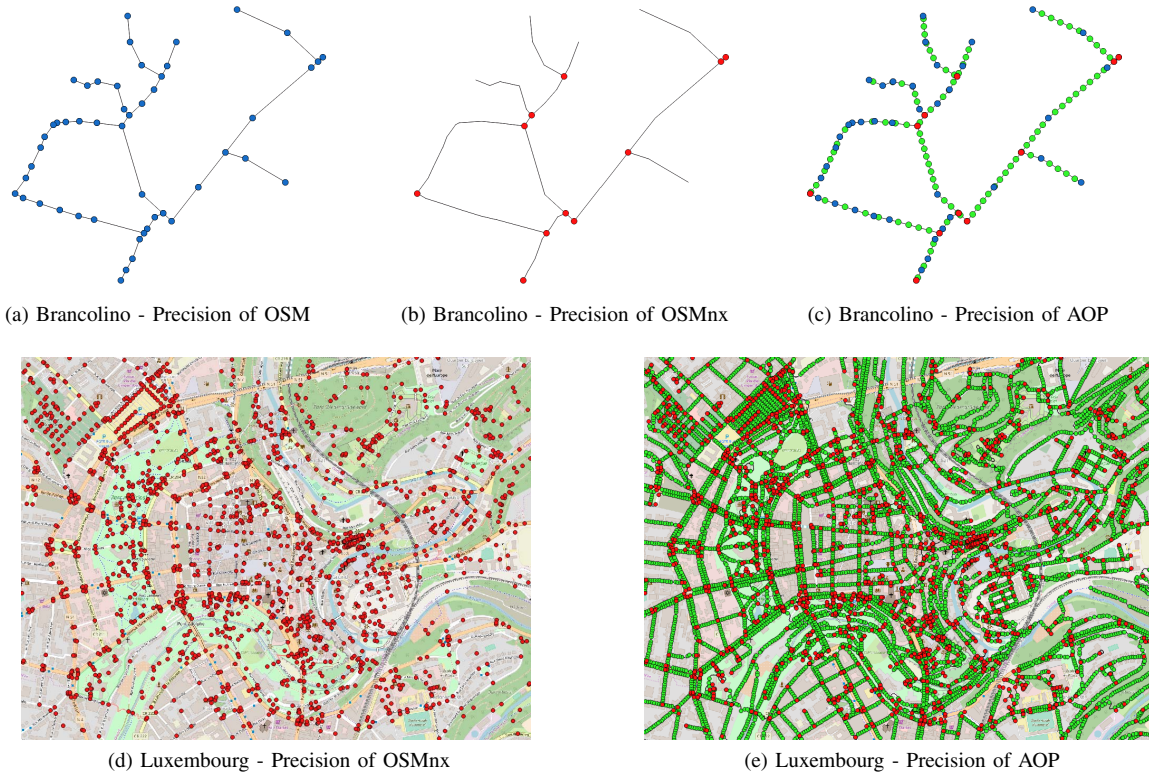


Fig. 1. Granularity for different cities. Blue circles denote the vertices provided by OSM, red ones by OSMnx and green circles those created by AOP.

an edge. Let $d(u, v)$ be defined the spatial distance between two generic nodes, being $u, v \in V$.

The objective of AOP is to increase the precision of G_{OSM} to achieve a given *target distance* D between any two adjacent nodes. The procedure guarantees that D is also the maximum distance between two adjacent nodes, thus setting smaller D augments the number of nodes in the graph and consequently its granularity. D can be as low as 1 m. By taking as input G_{OSM} and D , AOP generates a new graph G_{AOP} . Note that $|V_{\text{OSM}}| \leq |V_{\text{AOP}}|$. Higher precision means higher granularity, hence the cardinality of $|V_{\text{AOP}}|$ increases. We consider two algorithms to interpolate the position of the added nodes. V-AOP is based on the exact distance between two nodes in the OSM, and L-AOP is based on a linear approximation of the distance, reducing the computation time.

Fig. 1 compares the street network graph of different cities, Brancolino and Luxembourg. Specifically, Brancolino is a small town in Italy, while Luxembourg is a medium-size city and capital of the homonym country. The figure highlights the different precisions of the street network graph obtained with OSM (see Fig. 1(a)), OSMnx (see Fig. 1(b) and Fig. 1(d)) and AOP (see Fig. 1(c) and Fig. 1(e)). AOP is general enough to be applied to any situation where the original precision of OSM is not sufficient.

IV. PEDESTRIAN MOBILITY MODELS

This section details the two user arrival models employed in our numerical results. In the first model, the user arrivals are uniformly distributed over the simulation period (U-MOB).

The second mobility model derives user arrivals from a dataset (D-MOB). Specifically, it exploits real traces obtained from the well-known MCS ParticipAct framework [21]. The duration of each user trajectory and the user walking speed are uniformly distributed between [1, 40] minutes and [1, 1.5] m/s [15] respectively. The active users move in a random walk fashion over the street graph G generated by the AOP procedure. We leave as future work the possibility of defining on a per-user basis the direction of the movement and the start and the end-point of the trajectory.

A. The U-MOB Mobility Model

With U-MOB, user arrivals are uniformly distributed over the simulation period, i.e., the average total number of users that start walking over a certain time window is constant. Typically the time window is 1 hour long for comparison with D-MOB (Subsec IV-B), as ParticipAct traces are given with such granularity. Users who start and end their walking period in different hours are counted only once.

B. The D-MOB Mobility Model

Chessa et al. [21] compare different human mobility datasets. In this work, we focus on the ParticipAct dataset, originated from a MCS campaign of around 170 students in the Emilia Romagna region (Italy). Without having at disposal the dataset, we extracted the profile of the average number of contacts during 7 days and used as a reference to determine the user arrivals for D-MOB. Specifically, given the total simulation period in days, we subdivide the period into hours and we estimate the minimum number of users

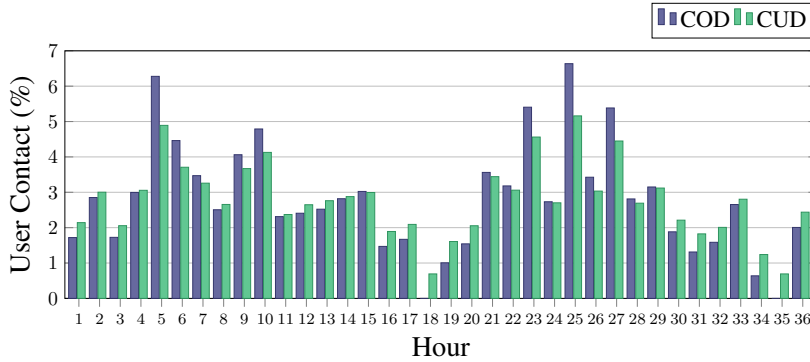


Fig. 2. Dataset-based arrivals according to the profile of Fig. 3 of [21]

to be allocated so that the average user contact follows the ParticipAct profile. Note that a user contact occurs when two users are within a certain distance R . For simplicity, we count unique contacts even when users trajectories intersect multiple times. For D-MOB, the following two methodologies have been designed:

- The Contact-Only-Distribution (COD) assigns a number of users per hour to reach the corresponding average number of contacts from the dataset. Once this preliminary phase is completed, the remaining not assigned users are divided proportionally to the per hour number of contacts. The method favors hours with high number of contacts.
- The Contact and User-Distribution (CUD) method is hybrid. It operates similarly to COD for the first phase, but in the second phase it assigns the remaining number of users proportionally to the number of hours of the simulation period. This method favors equally hours with high and low number of contacts.

Fig. 2 shows an example of COD and CUD arrivals over a 36 hours period with 50,000 users.

V. EVALUATION OF AOP PROCEDURE

For performance evaluation, we first evaluate the scalability of the AOP procedure and the precision of the dataset-based methods. Then, we compare the average per-user number of contacts and contact stability.

Scalability of AOP. The scalability of the AOP procedure depends on the topology of the considered urban environment and on the initial precision provided by OSM. Both versions of AOP procedure are implemented in python. The simulations are carried out using a Linux laptop with Ubuntu 16.04, equipped with an Intel®Core™ i7-4710HQ with 2.50 GHz x8 CPU and a system memory of 7.7 GiB. For the experiments, the target distance D is set to 3 m.

The smaller the city, the quicker will be the responsiveness of AOP. Given two cities with similar size, the one with higher initial precision from OSM will make the procedure to run longer. Note that the city population is totally irrelevant. For evaluation purposes, we choose a set of cities with growing size and latitude and applied AOP in its exact (V-AOP) and linearized (L-AOP) versions (see Tab. I). The computational time depends on the number of initial known vertices and edges and the average edge length. AOP's computational time

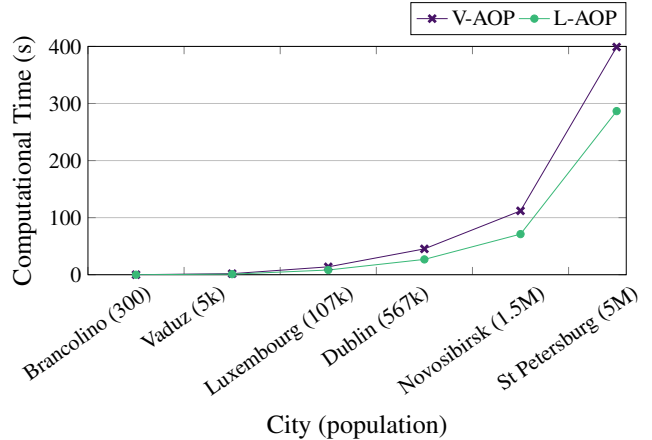


Fig. 3. Computational time and relative accuracy of V-AOP and L-AOP is small for cities with high initial number of edges and short average edge length than cities with low initial number of edges, but long average edge length. To illustrate, AOP converges faster for cities like Edinburgh (171,271 initial edges, average length 17.54 m) or Genoa (166,479 initial edges, average length 15.31 m) than Novosibirsk (133,556 initial edges, average length 42.69 m).

Fig. 3 compares the computational time of V-AOP and L-AOP. Unsurprisingly, for medium and small cities, the time difference is negligible. As expected, the computational time increases considerably with the size of the city and the initial number of nodes G_{OSM} provides. As expected, L-AOP converges faster than V-AOP. To illustrate, for St Petersburg, the time convergence reduction is in the order of 100 s.

Human mobility metrics. We employ two metrics to assess U-MOB and D-MOB arrival models. For both metrics we simulated a population of 50,000 users distributed over a period of 48 hours. We assume the time is slotted, with a timeslot equal to 1 minute. Two users are defined as neighbors if during timeslot j their distance is below a given radius R . For the experiments, R is set equal to 50 m, which is a reasonable distance for D2D communications based on WiFi-Direct. The per-User Average number of Contacts (UAC), specifically evaluated for user i , is defined as follows:

$$UAC_i = \frac{1}{T_i} \cdot \sum_{j=1}^{T_i} n_{j,i}, \quad (1)$$

where T_i is the amount of time user i is active, measured in

TABLE I
COMPARISON OF POPULATION, NUMBER OF EDGES, AVERAGE INITIAL EDGE LENGTH OF EACH EDGE AND NODES FOR DIFFERENT CITIES

CITY	POPULATION	EDGES G_{OSM}	AVG_LEN G_{OSM}	NODES G_{OSM}	NODES G_{L-AOP}	NODES G_{V-AOP}
Brancolino	365	50	24.54 m	50	383	387
Vaduz	5,391	4,098	23.94 m	3,889	30,444	30,664
Luxembourg City	107,247	31,298	24.07 m	28,266	217,805	235,545
Dublin	567,000	98,375	23.20 m	89,900	643,858	711,631
Novosibirsk	1,567,087	133,556	42.55 m	113,349	1,549,102	1,827,703
St. Petersburg	5,279,299	634,561	28.69 m	539,211	4,899,025	5,752,357

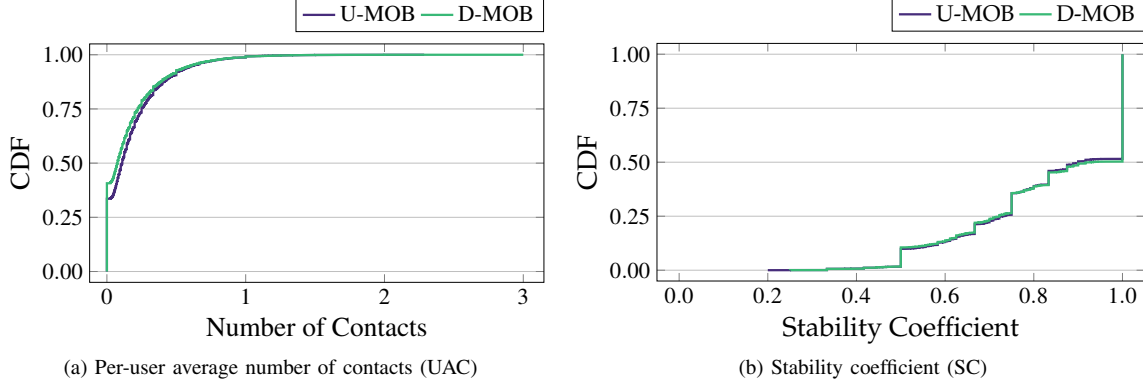


Fig. 4. Analysis of contact distribution and stability of contacts

number of timeslots, and $n_{j,i}$ is the number of neighbors of user i in timeslot j .

The second metric is called Stability Coefficient (SC) [33] and for user i is determined as follows:

$$SC_i = \frac{1}{T_i} \cdot \sum_{j=1}^{T_i} \frac{|n_{j+1,i} \setminus n_{j,i}| + |n_{j,i} \setminus n_{j+1,i}|}{|n_{j,i}| + |n_{j+1,i}|}, \quad (2)$$

where $n_{j,i}$ denotes the set of contacts of user i during timeslot j , $|n_{j+1,i} \setminus n_{j,i}|$ is the number of neighbors that user i loses between timeslots j and $j+1$ and $|n_{j,i} \setminus n_{j+1,i}|$ corresponds to the number of neighbors that users i acquires between timeslots j and $j+1$. This metric determines on how frequent a user changes neighbors, hence is relevant to determine to which degree the user is a valid candidate to become group owner in D2D WiFi-Direct communications.

Fig. 4 illustrates the results in form of CDF (Cumulative Distribution Function) for the distribution of the metrics evaluated across all the users. Fig. 4(a) shows the UAC metric. Interestingly, with the considered high number of users in the system, the U-MOB method with uniform arrivals approximates pretty well the D-MOB arrival distribution, based on the ParticipAct dataset. However, it should be noted that the distribution of UAC metric depends on multiple factors, including the total number of users in the system, the duration of the simulation, the size of the urban environment and the spatial allocation policy, which is uniformly distributed in our case. Fig. 4(b) indicates that half of the users in the system obtains values of SC different from 1, hence between two subsequent timeslots the users remain in contact with their neighbors. On average, for the majority of the users (75%), nearly 30% of the contacts is stable as the SC metric assumes values below 0.7.

Accuracy of the D-MOB arrivals methods. To evaluate the accuracy of the methods employed to generate the user arrivals according to the ParticipAct dataset, we run two different

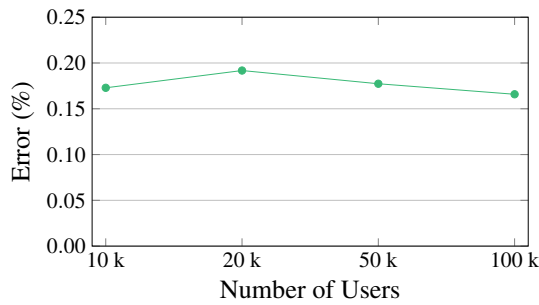
experiments using Luxembourg as city of interest.

In the first experiment, we have four scenarios with increasing number of users in the system, and the simulation period is fixed to 12 hours. Fig. 5(a) shows the relative error while comparing the methods COD versus CUD for D-MOB. Note that the accuracy is high as the error remains lower than 0.25% and is nearly constant for different values of users in the system. Note that the spike achieved for 20,000 users is negligible and due to the different initial spatial allocation.

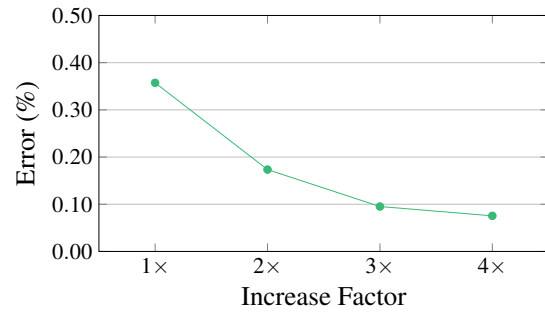
In the second experiment, we evaluate the accuracy by increasing both the number of users and the simulation period. Fig. 5(b) shows the results for different increase factors. An increase factor equal to $1\times$ corresponds to 10,000 users in the system and a simulation period of 24 hours. An increase factor equal to $4\times$ corresponds to have $4\times$ the baseline number of users and simulation period. Hence, $4\times$ corresponds to 160,000 users over 96 hours. The results are pretty accurate and the highest error is nearly 0.4%. Unsurprisingly, the highest error occurs for the lowest number of users in the system as achieving the target number of contacts is more difficult and can only be alleviated by increasing the population in the system.

VI. CONCLUSION

In this paper, we extensively studied models for pedestrian mobility for smart city simulators. First, we apply a procedure, AOP, to augment the precision of the street network provided by OpenStreetMap. We devised two different user arrival models, U-MOB and D-MOB. The latter is based on a well-known real dataset, ParticipAct. For performance evaluation, we assessed the scalability of AOP, the accuracy of the mobility models and verified popular metrics such as the average per-user number of contacts and the stability of the contacts. Interestingly, the U-MOB mobility model approximates very well D-MOB. Thus, for simulations with



(a) Simulation period fixed at 48 hours



(b) Joint increase of users and simulation period

Fig. 5. Accuracy of mobility models

large population and without considering the spatial allocation, modeling user arrivals with uniform distribution is sufficient to mimic realistic human mobility.

ACKNOWLEDGMENT

The authors would like to acknowledge the funding of the joint project UL-ILNAS on Digital Trust for Smart-ICT.

REFERENCES

- [1] N. B. Grimm, S. H. Faeth *et al.*, “Global change and the ecology of cities,” in *Science*, vol. 319, no. 5864, 2008, pp. 756–760.
- [2] H. B. Dulal and S. Akbar, “Greenhouse gas emission reduction options for cities: Finding the “coincidence of agendas” between local priorities and climate change mitigation objectives,” *Habitat International*, vol. 38, pp. 100 – 105, 2013.
- [3] A. Zanella, N. Bui *et al.*, “Internet of Things for smart cities,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, Feb 2014.
- [4] R. Ranjan, M. Wang *et al.*, “City data fusion: Sensor data fusion in the internet of things,” *Int. J. Distrib. Syst. Technol.*, vol. 7, no. 1, pp. 15–36, Jan 2016.
- [5] G. Cardone, A. Cirri *et al.*, “Crowdsensing in urban areas for city-scale mass gathering management: Geofencing and activity recognition,” *IEEE Sensors Journal*, vol. 14, no. 12, pp. 4185–4195, Dec 2014.
- [6] R. Ganti, F. Ye *et al.*, “Mobile crowdsensing: current state and future challenges,” *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, November 2011.
- [7] W. Khan, Y. Xiang *et al.*, “Mobile phone sensing systems: A survey,” *IEEE Communications Surveys Tutorials*, vol. 15, 2013.
- [8] K. Shilton, N. Ramanathan *et al.*, “Participatory design of sensing networks: Strengths and challenges,” in *10th Anniversary Conference on Participatory Design*, ser. PDC, 2008, pp. 282–285.
- [9] V. Sivaraman, J. Carrapetta *et al.*, “HazeWatch: A participatory sensor system for monitoring air pollution in Sydney,” in *IEEE LCN*, Oct 2013.
- [10] S. Kim, C. Robson *et al.*, “Creek Watch: Pairing usefulness and usability for successful citizen science,” in *SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI. ACM, 2011.
- [11] S. Reddy and V. Samanta, “Urban Sensing: Garbage Watch,” 2011, UCLA Center for Embedded Networked Sensing.
- [12] C. Fiandrino, F. Anjomshoa *et al.*, “Sociability-driven framework for data acquisition in mobile crowdsensing over fog computing platforms for smart cities,” *IEEE Transactions on Sustainable Computing*, vol. 2, no. 4, pp. 345–358, Oct 2017.
- [13] M. Pouryazdan, C. Fiandrino *et al.*, “Game-theoretic recruitment of sensing service providers for trustworthy cloud-centric internet-of-things (IoT) applications,” in *IEEE Globecom Workshops (GC Wkshps)*, Dec 2016, pp. 1–6.
- [14] A. Capponi, C. Fiandrino *et al.*, “Assessing performance of internet of things-based mobile crowdsensing systems for sensing as a service applications in smart cities,” in *IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, Dec 2016, pp. 456–459.
- [15] C. Fiandrino, A. Capponi *et al.*, “CrowdSenSim: a simulation platform for mobile crowdsensing in realistic urban environments,” *IEEE Access*, vol. 5, pp. 3490–3503, Feb 2017.
- [16] G. Cacciatore, C. Fiandrino *et al.*, “Cost analysis of smart lighting solutions for smart cities,” in *IEEE ICC*, May 2017, pp. 1–6.
- [17] A. Capponi, C. Fiandrino *et al.*, “A cost-effective distributed framework for data collection in cloud-based mobile crowd sensing architectures,” *IEEE Transactions on Sustainable Computing*, vol. 2, no. 1, pp. 3–16, Jan 2017.
- [18] C. Fiandrino, B. Kantarci *et al.*, “Sociability-driven user recruitment in mobile crowdsensing internet of things platforms,” in *IEEE GLOBE-COM*, Dec 2016, pp. 1–6.
- [19] G. Boeing, “OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks,” *Computers, Environment and Urban Systems*, vol. 65, pp. 126–139, 2017.
- [20] C. Thomas and W. Featherstone, “Validation of Vincenty’s formulas for the geodesic using a new fourth-order extension of Kivioja’s formula,” *Journal of Surveying engineering*, vol. 131, no. 1, pp. 20–26, 2005.
- [21] S. Chessa, M. Girolami *et al.*, “Mobile crowd sensing management with the ParticipAct living lab,” *Pervasive and Mobile Computing*, vol. 38, pp. 200 – 214, 2017.
- [22] V. Vukadinović, Ólafur Ragnar Helgason *et al.*, “An analytical model for pedestrian content distribution in a grid of streets,” *Mathematical and Computer Modelling*, vol. 57, no. 11, pp. 2933 – 2944, 2013.
- [23] M. L. Daggitt, A. Noulas *et al.*, “Tracking urban activity growth globally with big location data,” *Open Science*, vol. 3, no. 4, 2016.
- [24] P. Jensen, “A network-based prediction of retail stores commercial categories and optimal locations,” *Physical Review E : Statistical, Nonlinear, and Soft Matter Physics*, vol. 74, 2006.
- [25] J. M. Barajas, G. Boeing *et al.*, “Neighborhood change, one pint at a time: The impact of local characteristics on craft breweries,” in *Untapped: Exploring the Cultural Dimensions of Craft Beer*. West Virginia University Press, 2017, pp. 155–176.
- [26] M. Behrisch, L. Bieker *et al.*, “SUMO—simulation of urban mobility: an overview,” in *SIMUL. ThinkMind*, 2011.
- [27] A. Capponi, C. Fiandrino *et al.*, “Energy efficient data collection in opportunistic mobile crowdsensing architectures for smart cities,” in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2017, pp. 307–312.
- [28] A. Chaintreau, P. Hui *et al.*, “Impact of human mobility on opportunistic forwarding algorithms,” *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 606–620, Jun 2007.
- [29] P. Neis, D. Zielstra *et al.*, “The street network evolution of crowdsourced maps: OpenStreetMap in Germany 2007–2011,” *Future Internet*, vol. 4, no. 1, pp. 1–21, 2012.
- [30] T. J. Steiner, G. Huang *et al.*, “Location utility-based map reduction,” in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 479–486.
- [31] H. Zhang and Y. Vorobeychik, “Submodular optimization with routing constraints,” in *Conference on Artificial Intelligence (AAAI)*, 2016.
- [32] X. Liu and B. Jiang, “Defining and generating axial lines from street center lines for better understanding of urban morphologies,” *International Journal of Geographical Information Science*, vol. 26, no. 8, pp. 1521–1532, 2012.
- [33] M. R. Brust, M. Zurad *et al.*, “Target tracking optimization of UAV swarms based on dual-pheromone clustering,” in *3rd IEEE International Conference on Cybernetics (CYBCONF)*, Jun 2017, pp. 1–8.