

Efficient Millimeter-Wave Infrastructure Placement for City-Scale ITS

Ioannis Mavromatis, Andrea Tassi, Robert J. Piechocki, and Andrew Nix
Department of Electrical and Electronic Engineering, University of Bristol, UK
Emails: {Ioan.Mavromatis, A.Tassi, R.J.Piechocki, Andy.Nix}@bristol.ac.uk

Abstract—Millimeter Waves (mmWaves) will play a pivotal role in the next-generation of Intelligent Transportation Systems (ITSs). However, in deep urban environments, sensitivity to blockages creates the need for more sophisticated network planning. In this paper, we present an agile strategy for deploying road-side nodes in a dense city scenario. In our system model, we consider strict Quality-of-Service (QoS) constraints (e.g. high throughput, low latency) that are typical of ITS applications. Our approach is scalable insofar that takes into account the unique road and building shapes of each city, performing well for both regular and irregular city layouts. It allows us not only to achieve the required QoS constraints but it also provides up to 50% reduction in the number of nodes required, compared to existing deployment solutions.

Index Terms—ITS, Connected and Autonomous Vehicles, CAV, V2X, 5G, mmWaves

I. INTRODUCTION

Next-generation Intelligent Transportation Systems (ITSs) will require Vehicle-to-Infrastructure (V2I) wireless connectivity. These communication links offer the potential to enhance the road safety and efficiency in urban vehicular environments [1]. Introducing Millimeter-Waves (mmWaves) for dense urban environments will significantly improve the performance of small-cell access networks [2]. As shown, mmWave-V2I links have the potential of enabling gigabit-per-second data rates and ultra-low latency [3], [4].

The communication capabilities of the vehicles are highly dependent on the number of deployed Road-Side Units (RSUs) and their coverage range. RSUs, are costly to deploy and maintain. Therefore, compromises between the coverage provided and the deployment costs have to be made. MmWave RSUs especially, are bounded by their Line-of-Sight (LOS) requirements and their strict propagation characteristics [5]. However, they are a perfect candidate for small-cell vehicular deployments as they can meet the rigid bitrate and latency Quality-of-Service (QoS) constraints needed by next-generation vehicular applications [3].

It is crucial to deploy a number of RSUs in the most suitable locations, to improve the overall network performance. Given the variety of urban environments, it is necessary to find an agile method to obtain the best locations for each street layout and to deploy thousands of RSUs throughout a city. In this paper, we propose a strategy that can automate the RSU placement process for different urban scenarios. We will take into account the unique road and building layout of an urban environment as well as

the strict QoS constraints of vehicular applications. Within a city, buildings may be away from a road or privately owned. Also, city blocks may be empty or covered with vegetation. On the other hand, traffic lights are usually placed at road intersections. Also, street furniture (e.g. lamp posts) are typically equally spread along the sides of a road. In this paper, we assume that our RSUs are deployed on top of lamp posts and traffic lights. Thus, by positioning the RSUs only on the road, easier access for deployment or maintenance is provided. Furthermore, the network efficiency could be improved, avoiding the wall and rooftop blockages [6].

In [7] and [8], the authors presented an automated base station placement algorithm for mmWaves. However, they did not consider any QoS constraints for their optimization algorithm apart from the LOS coverage rate. In our approach, we consider two main Key Performance Indicators (KPIs). The first one is the LOS network coverage achieved after the deployment of all the chosen RSUs. The second one is the Received Signal Strength (RSS) averaged throughout the considered deployment area. A similar work can be found in [9], where the authors proposed a strategy for placing IEEE 802.11p RSUs. They considered the delay tolerance of the warning notifications as their optimization variable and solved their optimization problem utilizing a Genetic Algorithm (GA) [10]. However, mmWave links behave differently compared to IEEE 802.11p ones. In this paper, we will address the problem through a novel approach to find the best RSU locations by taking into account the propagation characteristics in mmWaves environments. Similarly, authors in [11], [12], only considered the distance or the propagation characteristics, but not the shape of the buildings and the roads. In our case, we will utilize tools from Computational Geometry (as in [8]), and consider all the above to find the desirable RSU locations. More specifically, we will investigate the performance of our algorithm under two urban environments, describing their unique characteristics. Our outcome will be compared with solutions obtained by GA and Greedy Construction (GC) [13] algorithms, to strengthen the enhanced performance provided by our approach.

The rest of the paper is organized as follows. In Sec. II we describe our system model. We introduce the tools utilized from Computational Geometry, and how we identify the candidate RSU positions. In Sec. III we formulate our opti-

mization problem based on two QoS constraints. Later, we outline our proposed strategy, solving the above problem. Sec. IV presents our performance investigation where we compare our strategy against the GC and GA approaches described before. Finally, Sec. V summarizes our findings.

II. SYSTEM MODEL

We consider an urban city map \mathcal{M} with dimensions $[\mathcal{M}_x, \mathcal{M}_y]$, measured in meters. Let $\mathcal{C} \triangleq \{1, \dots, C\}$ denote the candidate RSU positions, with all being within the boundaries of \mathcal{M} . For all the above positions we denote as $\mathcal{D} \triangleq \{1, \dots, D\}$ the positions chosen to deploy an RSU, and $\mathcal{R} \triangleq \{1, \dots, R\}$ the rejected ones. We have that $\mathcal{D} \subseteq \mathcal{C}$, $\mathcal{R} \subseteq \mathcal{C}$, $\mathcal{D} \cup \mathcal{R} = \mathcal{C}$, and $\mathcal{D} \cap \mathcal{R} = \emptyset$ hold.

Our RSU placement algorithm works in two steps:

- 1) At first, we identify all locations \mathcal{C} to deploy an RSU employing tools from Computational Geometry.
- 2) Then, we choose a subset $\mathcal{D} \subseteq \mathcal{C}$ in order to maximize the outdoor coverage and the RSS for the entire map.

In our system model, we assume that all the RSUs are mounted at the height of a lamp post or a traffic light. We also assume that all vehicles' antenna will be mounted on their rooftop. By that, we avoid most of the low-level obstacles (for e.g., kiosks, vehicles, trees, etc.). For simplicity, we consider a 2D network planning. Finally, we make use of OpenStreetMap [14] to obtain our city maps.

A. Identifying Potential RSU Locations

The problem of identifying \mathcal{C} can be approached by using tools from Computational Geometry. Similarly to [8], we introduce the notions of *Simple Polygons (SPs)* and *Polygons With Holes (PWHs)* in our system, describing the buildings and the roads, respectively.

A SP is considered a flat-shaped object consisting of straight, non-intersecting line segments, that, when joined pair-wise, they form a closed path. Given a city map, many SPs have adjacent sides, being part of the same city block. Therefore, they can be concatenated using the polygon union operation [15], and their inner holes can be removed, forming a solid object (as in Fig. 1). By that, we can decrease the complexity of our algorithm and consequently the execution time without any loss of accuracy. Furthermore, PWH is a polygon with an irregular shape that contains one or more holes or cutouts in it. Having access to the metadata from OpenStreetMap, we can accurately calculate the different polygons representing each road. Each polygon is later concatenated with the others, having finally a concave PWH that will be used for the RSU placement. The SPs introduced before, will be used to determine if a link is in LOS or not.

Given the generated SPs and PWHs, we can identify \mathcal{C} for a particular map. Our algorithm searches along the sides of PWHs for sharp edges and long straight sections. The edges, being the corner of two roads, are usually the best positions for an RSU, as they can maximize the LOS coverage (as shown in [7]). We also consider the length of

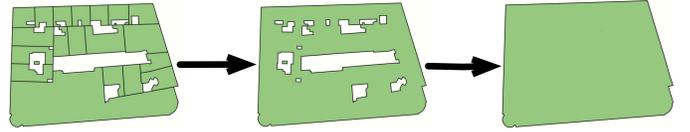


Figure 1. Example of the polygon union operation for a given city block.

a road. A road qualifies as long when the distance between two intersections is greater than a given threshold RSU_l . For any long road, we consider more potential positions, equally spaced between the two intersections. The number of these positions is given as the ceiling function from the division of the length l_i of the road i , over the given threshold, i.e., $\lceil l_i/RSU_l \rceil$. Combining both lists, \mathcal{C} is found.

III. PROBLEM FORMULATION AND SOLUTION

Given \mathcal{C} , the objective is to find $\mathcal{D} \subseteq \mathcal{C}$ to maximize the network coverage and achieve a minimum RSS throughout the network. The LOS coverage rate is modeled by equally spacing $\mathcal{Z} \triangleq \{1, \dots, Z\}$ grid points on the map with equal weights. Each point represents a squared tile having the same RSS throughout its surface. Using a tile-like approach with relatively small tiles, we can decrease the processing power required without having a significant loss of accuracy.

We determine \mathcal{D} by taking into account two different constraints. For a given \mathcal{Z} , we consider a set of \mathcal{N} reference points, with $\mathcal{N} \subseteq \mathcal{Z}$, being the grid points on top of the road polygons. We define the binary variable γ_n , to denote the state of a reference point n at location (x, y) as follows:

$$\gamma_n(x, y) = \begin{cases} 1, & \text{if } n \text{ is covered by at least one RSU} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Our first constraint imposes that a number of tiles, subset of $|\mathcal{N}|$, are in LOS with at least one RSU, i.e.:

$$\sum_{n=1}^{|\mathcal{N}|} \gamma_n(x, y) \geq \tau |\mathcal{N}| \quad (2)$$

where $\tau \in [0, 1]$, is a tolerance factor.

We say that P_{tx} and G_{tx} are the transmission power and antenna gain of each RSU. Also, $L_{LOS}(d)$ signifies our propagation loss at a distance d . Given the above, we can calculate the RSS for each tile as:

$$RSS(d) = P_{tx} + G_{tx} - L_{LOS}(d) \quad (3)$$

where $L_{LOS}(d)$ is the path-loss component and can be calculated as $L_{LOS}(d) = 10\alpha \log_{10}(d) + C_{att}(d)$. $C_{att}(d)$ is the channel attenuation with regard to the distance d . It is defined by the rain and atmospheric attenuation as well as the channel attenuation factor H_{att} for a given mmWave LOS link at 60 GHz in urban environments [16], i.e. $C_{att}(d) = 40d + H_{att}$. Finally, α is the path loss exponent.

For all \mathcal{D} , there is a number of tiles K_i that surrounds it. Each tile can be served by more than one RSU. We define as $\max_{i \in \{1, \dots, \tau|\mathcal{N}|\}} \{K_i^{RSS}\}$ the highest received RSS from all RSUs that serve this tile. The interference between the deployed

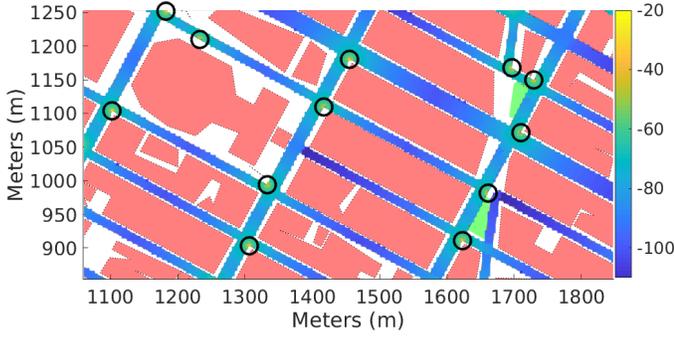


Figure 2. Example of 12 RSUs chosen for Manhattan by our strategy with their corresponding RSS. The street geometry affects the perceived RSS.

RSUs, is not taken into account. For the entire covered area, we sort all tiles with respect to their received RSS value, we take the first $\tau |\mathcal{N}|$ and denote them as $\text{RSS}_{i,k}^{\max}$. $\text{RSS}_{i,k}^{\max}$ is the average for the best $\tau |\mathcal{N}|$ tiles and for a given \mathcal{D}_i . Our second constraint ensures that a target number of tiles has an average RSS that is greater than or equal to a threshold RSS_{th} :

$$\sum_{i=1}^{|\mathcal{D}|} \text{RSS}_{i,k}^{\max} \geq \text{RSS}_{th} \quad (4)$$

A. Problem Formulation

Let e_c be the vector that defines the state of each RSU i in \mathcal{C} . We have:

$$e_i = \begin{cases} 1, & \text{if } i \text{ is deployed} \\ 0, & \text{if } i \text{ is not deployed} \end{cases} \quad (5)$$

In order to find the best RSU locations, our problem is formulated as the minimization of (5):

$$\text{Minimise}_{e_1, \dots, e_c} \sum_{i=1}^{|\mathcal{C}|} e_i \quad (6a)$$

$$\text{subject to: } \sum_{n=1}^{|\mathcal{N}|} \gamma_n(x, y) \geq \tau |\mathcal{N}| \quad (6b)$$

$$\sum_{i=1}^{|\mathcal{D}|} \text{RSS}_{i,k}^{\max} \geq \text{RSS}_{th}, \quad \forall i \in \mathcal{D} \quad (6c)$$

$$e_i \in [0, 1], \quad \forall i \in \mathcal{C} \quad (6d)$$

$$\text{RSS}_{i,k}^{\max} \text{ for } k \text{ in LOS}, \quad \forall i \in \mathcal{D} \quad (6e)$$

where the above are the constraints described before. Also, the RSS is calculated for all tiles being in LOS with an RSU, i.e. a SP does not block the link. An example solution for a small urban area can be seen in Fig. 2. In this particular example, we can observe the effect of the street geometry on the RSS.

B. Proposed Algorithm

To solve (6a)-(6e), we propose a novel algorithm to calculate the list \mathcal{D} . A city-scale RSU placement problem can be computationally expensive. So, we designed our algorithm to operate in three phases, to minimize the execution time. Our algorithm (Alg. 1) works are as follows:

Algorithm 1 Agile RSU Placement

Output: Return lists with RSUs: \mathcal{D} and \mathcal{R}

Phase 1:

- 1: Calculate the tiles k'_i required to achieve the RSS_{th} for each \mathcal{C}
- 2: **while** Constraint (6b) not met **do**
- 3: Find served tiles in the system and remove from lists k'_i .
- 4: Find RSU i with the longest k'_i list and add it in \mathcal{D} .
- 5: **end while**

Phase 2: – Skip if (6b) and (6c) are met.

- 6: **while** Constraint (6c) is not met **do**
- 7: **for all** RSUs in $\mathcal{C} \setminus \mathcal{D}$ **do**
- 8: Calculate number of non-covered tiles that they can serve.
- 9: **end for**
- 10: **if** Non-covered tiles on map **then** \triangleright i.e., (6b) is not maximised
- 11: Find RSU i that covers the most non-covered tiles.
- 12: Add i in list of chosen RSUs \mathcal{D} .
- 13: **else**
- 14: **for all** RSUs in $\mathcal{C} \setminus \mathcal{D}$ **do**
- 15: Calculate the potential mean RSS if RSU is chosen.
- 16: **end for**
- 17: Find RSU i that maximises the mean RSS for the system.
- 18: Add i in list of chosen RSUs \mathcal{D} .
- 19: **end if**
- 20: **end while**

Phase 3:

- 21: **repeat**
 - 22: **for all** RSUs i in \mathcal{D} **do**
 - 23: Find k in $\mathcal{C} \setminus \mathcal{D}$ that improves constraints (6b) and (6c).
 - 24: Replace i with k .
 - 25: **end for**
 - 26: **until** \mathcal{D} cannot be improved more \triangleright i.e., no more swaps can be done
-

1) *Phase 1:* We start by calculating the number of tiles k'_i required to achieve a mean RSS value greater than or equal to RSS_{th} for all \mathcal{C} . Later, we iteratively add to \mathcal{D} the RSU with the most non-served tiles within RSS_{th} found before, until constraint (6b) is met. Thus, we ensure a sufficient amount of coverage in our system in a fast greedy-addition-like fashion. If (6c) is also fulfilled, we proceed to Phase 3. If not, we continue with Phase 2.

2) *Phase 2:* We add more RSUs in the system until both constraints are met. We identify the non-covered areas of the map and prioritize our RSU placement towards them, as this can increase our system performance. Then, we find the areas that are not adequately served by the existing RSUs, and we add RSUs that can fulfill (6c). When both constraints are fulfilled, we have two admissible lists \mathcal{D} and \mathcal{R} and we proceed to the next phase.

3) *Phase 3:* From the above two phases, we may not always achieve an ideal solution for \mathcal{D} . This happens especially when the requirements for a specific scenario are more relaxed (e.g. a low coverage rate is required). To improve the performance at this point, we search in \mathcal{R} , if it exists an RSU that can improve (6b) or (6c). If so, we replace these two. We iterate, until no other RSUs can be swapped, meaning that we have our final \mathcal{D} .

IV. SIMULATION RESULTS

Our strategy is compared against a GC and a GA approach. GC is scalable, but it cannot fulfill the required KPIs, as it does not take into account the RSS threshold. The second one is notoriously computationally expensive but

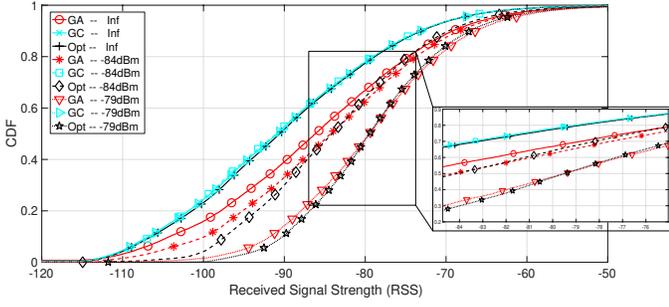


Figure 3. The empirical CDF of the RSS for the map of Manhattan. A tolerance $\tau = 0.99$ and three different RSS_{th} were used for this scenario.

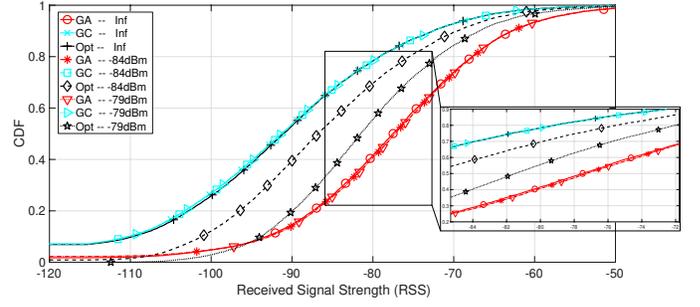


Figure 4. The empirical CDF of the RSS for the map of Paris. A tolerance $\tau = 0.90$ and three different RSS_{th} were used for this scenario.

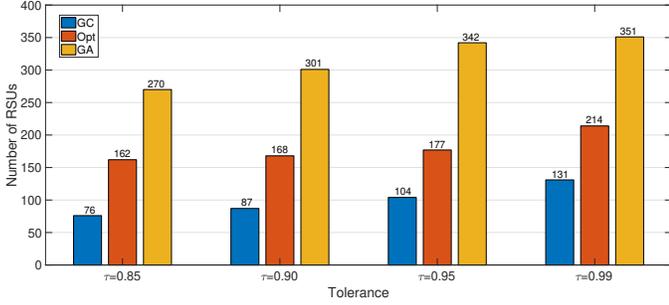


Figure 5. The number of RSUs (Manhattan), given by all algorithms, for all different tolerance parameters. The RSS_{th} is equal to -84 dBm.

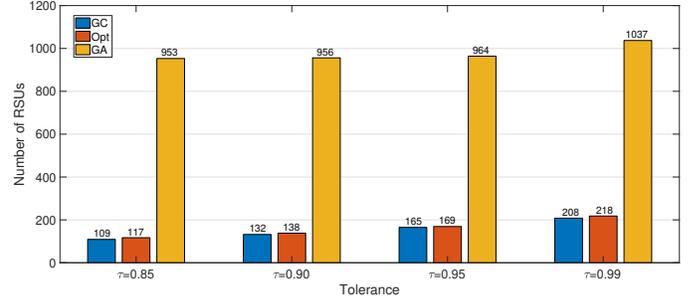


Figure 6. The number of RSUs (city of Paris), given by all algorithms, for all different tolerance parameters. The RSS_{th} is equal to -90 dBm.

Table I
LIST OF AREAS USED AND SIMULATION PARAMETERS.

Urban Area	Manhattan, NY, USA	Paris, FR
Centre	$-73.9841^\circ W, 40.75545^\circ N$	$2.33235^\circ W, 48.875^\circ N$
Number of Maps	2×2	2×2
Map Size $[\mathcal{M}_x, \mathcal{M}_y]$	$1 \text{ km} \times 1 \text{ km}$	$1 \text{ km} \times 1 \text{ km}$
Area of Interest	$900 \text{ m} \times 900 \text{ m}$	$900 \text{ m} \times 900 \text{ m}$
Grid Size	$4 \text{ m} \times 4 \text{ m}$	$4 \text{ m} \times 4 \text{ m}$

Parameter	Symbol	Value
Transmission power	P_{tx}	10 dBm
TX Antenna Gain	G_{tx}	15 dBi
Path-Loss Exponent	α	2.66
Channel Att. Factor	H_{att}	70 dB
Distance Threshold	RSU_t	100 m

generates high-quality solutions. We test all the algorithms in two urban areas from Manhattan (New York, USA) and Paris (FR). An average road lane is 2.9m to 4.6m wide. Therefore, we considered a grid with side equal to 4m, so each tile covers roughly the width of a road lane. The considered areas and the simulation parameters are summarized in Table I.

Each area is 4km^2 and divided into four equal sections. For each section, we consider an area of interest of 810m^2 , to avoid border effects. Four different tolerance parameters are employed, namely, $\tau \in \{0.85, 0.90, 0.95, 0.99\}$, with 90% being an average coverage rate, while 99% being the extreme case. Also, four RSS thresholds are used, that is

$RSS_{th} \in \{\text{Inf}, -90, -84, -79\}$. The first value signifies the case where no RSS threshold is considered. The last one was chosen based on the sensitivity threshold of IEEE 802.11ad, i.e., the minimum RSS (without considering the RX antenna gain) to achieve one-gigabit-per-second data rate. This value is the amount of data estimated to be generated and transmitted from each Connected and Autonomous Vehicle (CAV) [17].

In Fig. 3, we present the RSS results per-tile for Manhattan and all the considered optimization strategies. Also, $\tau = 0.99$ and three different thresholds were considered, i.e. $RSS_{th} \in \{\text{Inf}, -84, -79\}$, to investigate the effect on each approach. We observe that GC always produces similar results, comparable with our scheme (when the RSS constraint is disregarded). That is because GC cannot take into account the RSS threshold. Obviously, as the RSS_{th} is taken into consideration, we observe that both our algorithm and GA perform better than GC. In particular, our strategy achieved a city-wide mean RSS of -83.6 dBm and -78.9 dBm, while GA achieved -84.3 dBm and -79.2 dBm for $RSS_{th} = -84$ dBm and $RSS_{th} = -79$ dBm, respectively.

Similarly, Fig. 4 presents the RSS results for Paris and $\tau = 0.90$. Again, GC algorithm behaves as before, having similar performance for all thresholds and being comparable with our algorithm when RSS_{th} is not taken into account. However, the main difference compared to the Manhattan results is the performance dissimilarity between our algorithm and GA. We observed that, even with relaxed parameters GA always finds an extreme solution (mean

RSS of -78.8 dBm and -78.5 dBm for $RSS_{th} = -84$ dBm and $RSS_{th} = -79$ dBm respectively), while our algorithm behaves as before (mean RSS of -86.2 dBm and -80.8 dBm). This is because of the highly irregular building shapes of Paris, compared to the grid-like shape of Manhattan, making it very difficult for GA to find the best RSU positions in the city and always getting stuck to a local maximum. For both cities, all the algorithms behave similarly to other tolerance parameters and will not be presented here due to the limited space.

In Figs. 5 and 6, we present the number of RSUs required to fulfill the QoS constraints required, for all τ . Fig. 5 refers to the Manhattan scenario, for $RSS_{th} = -84$ dBm. Once more, GC utilizes fewer RSUs, but it does not take into account the RSS_{th} . Comparing the number of RSUs obtained with the proposed approach and with GA, we observe that our scheme ensures a reduction of up to 50% in the number of RSUs compared to GA. From Figs. 3 and 5, we observe that our algorithm achieves comparable results, fulfilling the QoS requirements with a smaller number of RSUs. Moving on to Fig. 6, the difference between the number of RSUs and the two algorithms is greater, having GA solving our optimization problem with almost eight times as many RSUs. Again, as before, this is because of the irregularity of the building and road shapes.

V. CONCLUSIONS

In this paper, we proposed an agile and efficient strategy for city-wide mmWave-RSU placement. We presented a scalable algorithm, able to compute high-quality RSU deployment on a map. In doing so, the proposed strategy takes into account two KPIs for vehicular communications: the coverage rate and the RSS threshold. Our approach is compared against the GC and GA strategies. GC is fast and scalable, but it cannot satisfy the above mentioned KPIs, while GA is computationally expensive. We observed that our strategy meets the target coverage constraints utilizing a smaller number of RSUs. Also, our performance investigation showed that our approach is suitable for both regular and irregular city layouts when other strategies fail to handle non-uniform building shapes. All the above make it a suitable solution for large-scale experimentation and the next-generation ITSs.

ACKNOWLEDGMENT

This work was partially supported by the University of Bristol and the Engineering and Physical Sciences Research Council (EPSRC) (grant ref. EP/I028153/1). This work is also part of the FLOURISH Project, which is supported by Innovate UK under Grant 102582.

REFERENCES

[1] P. Belanovic, D. Valerio, A. Paier, T. Zemen, F. Ricciato, and C. F. Mecklenbrauker, "On Wireless Links for Vehicle-to-Infrastructure Communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 269–282, Jan. 2010.

[2] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!" *IEEE Access*, vol. 1, pp. 335–349, May 2013.

[3] I. Mavromatis, A. Tassi, R. J. Piechocki, and A. Nix, "MmWave System for Future ITS: A MAC-Layer Approach for V2X Beam Steering," in *Proc. of IEEE VTC 2017*, Sep. 2017, pp. 1–6.

[4] A. Tassi, M. Egan, R. J. Piechocki, and A. Nix, "Modeling and Design of Millimeter-Wave Networks for Highway Vehicular Communication," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10676–10691, Dec. 2017.

[5] M. Abouelseoud and G. Charlton, "System Level Performance of Millimeter-Wave Access Link for Outdoor Coverage," in *Proc. of IEEE WCNC 2013*, Apr. 2013, pp. 4146–4151.

[6] S. Friedner, "5G Infrastructure Requirements in the UK," LS telcom UK, Tech. Rep., Dec. 2016.

[7] N. Palizban, S. Szyszkowicz, and H. Yanikomeroglu, "Automation of Millimeter Wave Network Planning for Outdoor Coverage in Dense Urban Areas Using Wall-Mounted Base Stations," *IEEE Wireless Commun. Lett.*, vol. 6, no. 2, pp. 206–209, Apr. 2017.

[8] S. S. Szyszkowicz, A. Lou, and H. Yanikomeroglu, "Automated Placement of Individual Millimeter-Wave Wall-Mounted Base Stations for Line-of-Sight Coverage of Outdoor Urban Areas," *IEEE Wireless Commun. Lett.*, vol. 5, no. 3, pp. 316–319, Jun. 2016.

[9] M. Fogue, J. A. Sanguesa, F. J. Martinez, and J. M. Marquez-Barja, "Improving Roadside Unit Deployment in Vehicular Networks by Exploiting Genetic Algorithms," *Applied Sciences*, vol. 8, Jan. 2018.

[10] M. Mitchell, *An Introduction to Genetic Algorithms*. MIT Press, 1996.

[11] T. J. Wu, W. Liao, and C. J. Chang, "A Cost-Effective Strategy for Road-Side Unit Placement in Vehicular Networks," *IEEE Trans. Commun.*, vol. 60, no. 8, pp. 2295–2303, Aug. 2012.

[12] R. Zhang, F. Yan, W. Xia, S. Xing, Y. Wu, and L. Shen, "An Optimal Roadside Unit Placement Method for VANET Localization," in *Proc. of IEEE GLOBECOM 2017*, Dec. 2017, pp. 1–6.

[13] D. S. Hochbaum and A. Pathria, "Analysis of the Greedy Approach in Problems of Maximum k-Coverage," *Naval Research Logistics (NRL)*, vol. 45, no. 6, pp. 615–627, 1998.

[14] "OpenStreetMap Project," <https://www.openstreetmap.org>, 2017.

[15] F. Martnnez, A. J. Rueda, and F. R. Feito, "A New Algorithm for Computing Boolean Operations on Polygons," *Computers & Geosciences*, vol. 35, no. 6, pp. 1177 – 1185, 2009.

[16] A. Yamamoto, K. Ogawa, T. Horimatsu, A. Kato, and M. Fujise, "Path-Loss Prediction Models for Intervehicle Communication at 60 GHz," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 65–78, Jan. 2008.

[17] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1617–1655, Sep. 2016.