Efficient RSU Placement Schemes in Urban Vehicular Ad hoc Networks

S.F. HWANG, W.C. CHEN, C.R. DOW*, AND N.L. NGUYEN
Department of Information Engineering and Computer Science
Feng Chia University, Taichung, 407 Taiwan
E-mail: sfhwang@mail.fcu.edu.tw; spadea21410@hotmail.com;{crdow; lam}@mail.fcu.edu.tw

Roadside units play a vital role in vehicular ad hoc networks. Essential benefits of roadside units include providing information about traffic jams, accidents, and emergency messages to drivers in real-time. Because roadside units are expensive, developing a method to deploy them cost-effectively is pertinent. In this paper, we propose a roadside unit placement method using a limit number of roadside units to cover the intersections in an urban vehicular ad hoc network. Our strategy focuses on identifying potential candidate locations to place roadside units and minimizing the number of roadside units to deploy. Simulation results show that the proposed method outperforms existing methods in terms of the number of roadside units being used and the communication coverage.

Keywords: vehicular networks, roadside unit deployment, isolated intersection, vertex cover

1. Introduction

With the rapid development of wireless communication technologies, Vehicular ad hoc networks (VANETs) are getting more attention. A VANET mainly consists of vehicles and roadside facilities. Each vehicle is equipped with wireless communication equipment for communicating with other vehicles [1] and roadside facilities [2]. Their applications have been more widely used [3, 4, 5, 6, 7, 8] to make people's lives safer and more convenient, such as providing drivers real-time traffic information, commercial advertisements, and parking information.

Generally, communication modes in a VANET can be divided into two categories: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I). V2V enables communication between vehicles with a slow transmission rate and short transmission distance at a low cost. V2I, however, is a communication model in which vehicles communicate with roadside facilities such as Roadside Units (RSUs) using high transmission speed, long transmission distance, and topological fixedness. In the current automotive environment, these two models are combined to offset each model's deficiencies.

In VANETs, RSUs are crucial road facilities. The primary function of RSUs is to communicate with vehicles within their transmission range via wireless communication. RSUs either provide information to the drivers or gather vehicles information [9]. The coverage of RSUs is pertinent for enabling drivers to obtain information smoothly while driving. Although each road junction should be covered by an RSU, it is costly to deploy and maintain RSUs. Thus, the purpose of this study is to minimize the number of RSUs being used in a VANET and guarantee that all intersections are covered.

* Contact Author

The contributions of our scheme are three-fold. First, we consider each intersection as a candidate location due to its high traffic intensity and large number of events. We then calculate the impact values of candidate locations. The impact value of a candidate location indicates how important it is among others. Finally, we identify the isolated intersections to reduce the number of RSUs being used. Experimental results show that our scheme outperforms existing schemes in terms of the number of RSUs being used and communication coverage.

The rest of this paper is organized as follows: Section 2 discusses current approaches for RSUs deployment problem; Section 3 details our proposed scheme for deploying RSUs; the simulation and results are discussed in Section 4; our conclusions are presented in Section 5.

2. Related Work

Generally, the objective of deploying RSUs is to find the most appropriate locations, in which the goals of minimizing transmission delay, lowering packet loss rate, improving network connectivity and transmission range are guaranteed. Patil et al. [10] proposed a Voronoi diagram–based algorithm for deploying RSUs. This algorithm takes transmission time, packet delay, and loss rate as inputs to draw the boundaries of polygons in a Voronoi diagram. Each polygon is considered as a region in which an RSU needs to be deployed. Average number of cars per hour, flow rate, average number of vehicles per square mile, and average speed is calculated to determine the polygon boundaries in a Voronoi diagram. This scheme is based on both V2V and V2I. A car that enters a polygon may not be able to make direct contact with RSUs. In such a case, messages must be obtained by making contact with other cars. However, the messages may be missed if the traffic flow at that time is low.

To improve network connectivity, Lee et al. [11] proposed a method for arranging RSUs by using the historical position of vehicles. This information is used to calculate the number of vehicles at each intersection, which is considered as a candidate position for RSU. Considering maintenance cost of RSUs, Vageesh et al. [12] proposed a setting method that takes energy consumption into account. Liu et al. [3] proposed an RSU placement scheme based on content downloading. In their study, the authors used the time for downloading files, the probability of success, and the number of vehicles covered in a VANET to determine RSU locations. Yan et al. [13] proposed two setting methods, with and without traffic information, for selecting the intersections to install RSUs. Considering traffic information, the authors assumed that they knew the traffic information at the intersections. The authors then adopted a greedy algorithm to select the locations to install RSUs.

The coverage of RSUs and the location of intersections are also the key factors that need to be considered for deploying RSUs. Rizk et al. [14] proposed a method for selecting the locations to deploy RSUs. This approach is based on the weight of candidate locations. The final set of RSUs is obtained by choosing the candidate locations with the higher order of weight. Because the authors only consider intersections as candidate locations, certain road segments may not be covered by RSUs when applying this method.

In [15], the authors proposed the Hybrid Algorithm (HA), which is a combination of the Greedy algorithm (GA) and the Dynamic Algorithm (DA). The GA scheme selects candidate locations based on their priority. It purely chooses intersections with higher priorities to place RSUs until all intersections in a given map are covered. This limitation of the GA scheme has the high overlap ratio between adjacent RSUs and more RSUs being

deployed. In the DA scheme, each round of the RSU selection strategy, the selected RSUs in previous rounds can be replaced by the lower priority RSU to minimize the overlap ratio. The DA scheme has a limitation that important intersections could be ignored. Rizk et al. [14] proposed the Overlap based Greedy Method (OGM) as part of the PRONET project created by the ITS Research Group at the German University. In this method, the distribution of RSUs is based on the overlap rate ratio and the prioritization of Sites of Interest. Other RSU deployment schemes are based on evolutionary algorithms (EA). Ota et al. [16] proposed an EA for optimizing energy consumption and communication connectivity. In this study, the city map is divided into grids. For a given number of RSUs, the optimal distribution of RSUs on the grid map is determined by the EA. Similarly, Moura et al. [17] proposed an EA for RSU deployment based on betweenness centrality and community detection method. In this study, intersections are considered as candidate locations for RSUs. Both algorithms evaluated individuals by a fitness value and terminated when encountering a given number of generations. In contrast, Fogue et al. [18] developed an RSUs deployment method based on a generic algorithm called GARSUD to minimize the warning notification time.

Some other studies do not consider intersections as candidate locations. In [19], the authors modeled an urban area as a grid map and consider each center point of squares is a candidate location. The limitation of this assumption is that the center point of an area is not always a possible location to install RSUs. For instance, this location can be surrounded by buildings and obstacles in which the radio wave propagation can be disturbed or even RSU cannot be deployed. In [20, 21], the authors proposed a self-organizing network approach that parked cars are leveraged as candidate locations for RSUs. In this scheme, the parked cars act as temporary RSUs or relays for the existing fixed RSUs. Because cars may be parked at different time and space, this parking inconsistency could affect network connectivity and reliability of the system. Another issue that must be concerned is the battery life of the cars which can result in RSU power off.

3. Roadside Unit Deployment

The RSU placement problem involves finding an optimal placement on an urban street map to minimize the number of RSUs to deploy while maximizing their coverage. In this section, we describe our approach to solve this problem. Our method consists of three steps. First, we identify the candidate locations from a street map. Next, we select the optimal locations to place the RSUs based on the candidate locations set. Finally, we identify the isolated intersections in which deploying RSUs is not pertinent.

3.1 Candidate Locations for Deploying RSUs

The first step of our method is to identify the candidate locations in which RSUs will be established. We choose the intersections as the potential locations to install RSUs due to the high traffic intensity and a large number of events at these locations [22, 23]. In addition, we consider the middle of the road as a candidate location when the length of the road L meets the following condition:

$$R < L \leq 2R \tag{1}$$

where R is the transmission range of RSU. Installing an RSU at the middle of the road can improve the reliability of communications. Moreover, this can reduce the number of RSUs being deployed at the two ends of the roads. As shown in Fig. 1, O, P, and Q denote

the middle points of the road segments AB, DE, and GL, respectively. These segments meet the condition 1, thus its middle points are the candidate locations. Considering the road segment AB, we have two subcases:

1. Having a candidate location at O: O is the only location to deploy an RSU.
2. Not having a candidate location at O: A and B are the two locations to deploy RSUs.

Thus, fewer RSUs are used if we consider the middle of the road as an RSU candidate location. Fig. 1 presents an example of the candidate locations for a given map.

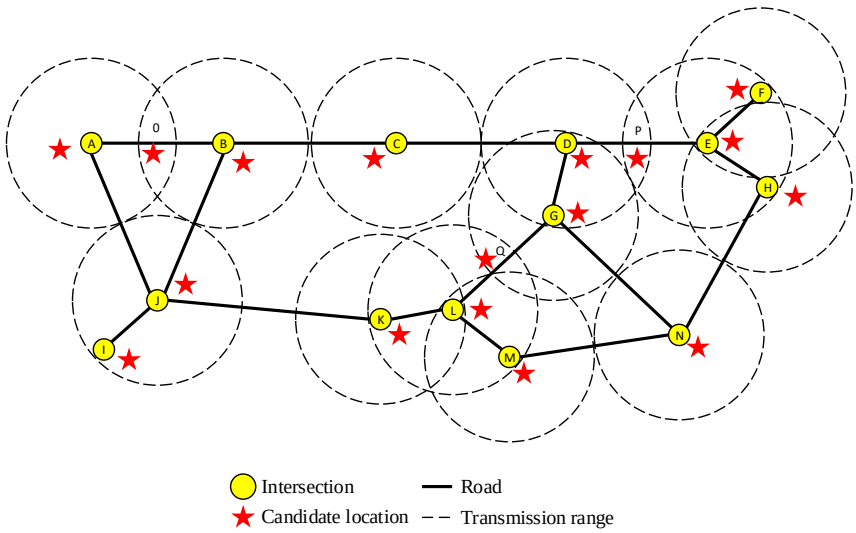


Fig. 1: RSU Candidate Locations

3.2 Determining RSU Locations

3.2.1 Impact Value of Candidate Locations

After obtaining a set of candidate locations, we will then determine which candidate locations are the best locations for RSUs. In our scheme, we focus on the traffic intensity and the number of events at intersections. Let $N = n_1, n_2, .., n_m$ and $A = a_1, a_2, .., a_m$ be the sets of traffic intensity and the number of events of m intersections, respectively. We use these values to compute the weight of each candidate location. As the values of traffic intensity and the number of events are in different ranges, this can affect the weight values of candidate locations. Thus, we normalize these values to the range of $[0,1]$. Two factors α and β are used as the coefficients, where $0 \leq \alpha, \beta \leq 1$ and $\alpha + \beta = 1$. The values of α and β depend on which parameters are more important, the traffic intensity or the number of events at intersections. α and β can be adjusted according to user needs. For instance, if we consider the traffic intensity is more important than the number of events then we set the value of α greater than that of β; if we only consider the traffic intensity, then we set $\alpha = 1$ and $\beta = 0$. If the candidate location is not an intersection, we set its weight value to 0. We calculate the weight value of the candidate location i,(W_i), using the following

equation:

$$W_i = \alpha \frac{n_i - min(N)}{max(N) - min(N)} + \beta \frac{a_i - min(A)}{max(A) - min(A)} \tag{2}$$

where n_i and a_i are the traffic intensity and the number of events at the candidate location i, respectively. Fig. 2 illustrates the candidate locations with their corresponding weight values.

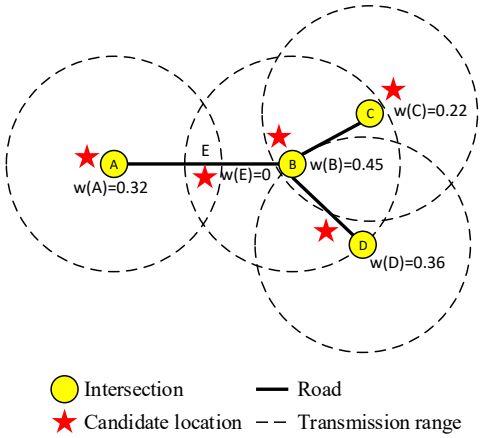


Fig. 2: Candidate Location with Weight Value

To minimize the number of required RSUs, we compute the impact value of each candidate location based on the weight values. The impact value of a candidate location indicates how crucial it is to its adjacent candidates. The impact value I_i of the candidate location i is calculated as follows:

$$I_i = W_i + \sum_{k \in V_i} W_k \tag{3}$$

where V_i is a set of candidate locations which are in the transmission range of the RSU placed at candidate location i; k indicates the candidate location k in the coverage of candidate i; W_k is the weight value of candidate location k. The candidate location k is supposed to be in the transmission range of candidate location i iff the distance between the candidate locations i and k is less than or equal to R. Supposed that all RSUs have the same coverage range of R. Fig. 3 shows the impact value of each candidate location presented in Fig. 2 by applying equation 3. For instance, the candidate locations E, C, and D are in the coverage ratio of candidate location B, thus the impact value of B, $I(B) = W(B) + W(E) + W(C) + W(D) = 1.03$. Table 1 lists the important notations used in this paper.

Algorithm 1 describes the pseudocode for calculating the impact values of candidate locations. A set of candidate locations C is taken as input and the impact value of each candidate in the set is computed through a loop.

1. (Lines 1–2) At each iteration, we compute the impact value of each candidate location c_i. Initially, the impact value of candidate location i is assigned as 0.

2. (Lines 3–5) For each candidate location c_k in V_i, we add its weight to p_i.

3. (Line 6) The algorithm will be terminated when all the impact values of candidate locations in C are calculated.

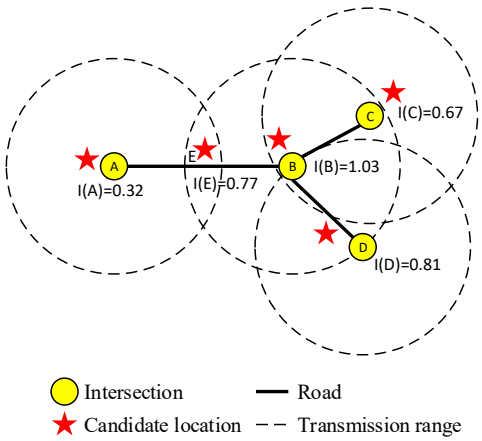


Fig. 3: Candidate Location with Impact Value

Table 1: Summary of Notations

Notation	Descriptions
C	Set of candidate locations.
P	Impact values of candidate locations, where each element p_i is the impact value of candidate location i, $i \in C$.
R	Transmission range of RSU.
W	Weight values of candidate locations, where each element w_i is the weight value of candidate location i, $i \in C$.
I	Set of isolated intersections.
V_i	Set of candidate locations in the coverage of the RSU installed at candidate location i, $i \in C$.
$LocRSU$	Set of candidate locations where RSUs will be installed.

3.2.2 Determining RSU Locations

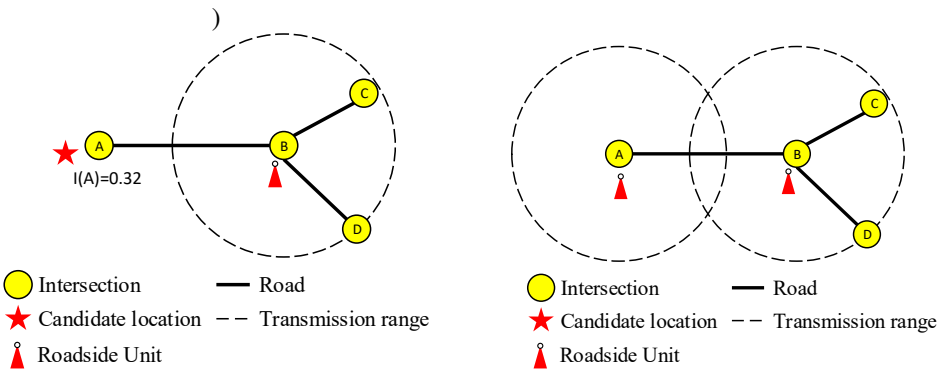
The impact value of a candidate location indicates how valuable it is for installing an RSU. In this paper, the impact value is considered to be the gold standard for selecting the locations to install RSUs. Therefore, each round of determining RSU location strategy, we select the candidate location which has the largest impact value. If there are multiple candidate locations with the same impact values, we consider the candidate location which has the largest degree. The degree of a candidate location is the number of candidate locations which are adjacent to it. If a candidate location is selected to place an RSU, other candidate locations within its transmission range will then be removed to reduce the number of RSUs being used. This operation may cause the impact values of the adjacent candidate locations of the removed candidates being changed. In this case, we recalculate their impact values. We repeat these steps until all the intersections are covered.

As shown in Fig. 3, we can find that, among the candidate locations denoted as A, B, C, D, and E, location B has largest impact value. Therefore, we first select B to place an RSU. As shown in Fig. 4a, locations C, D, and E are in the coverage radius of B, so that we remove them from the candidate locations set. Because the remaining location A is not affected by the others, its impact value is unchanged. Location A is also the last candidate location, thus we select A as a location for RSU. Fig. 4b shows the locations where RSUs will be deployed. The strategy for selecting candidate locations is presented

Algorithm 1 Candidate Location Impact Values

Input: $C = \{c_1, c_2, c_3, \ldots, c_m\}$
Output: $P = \{p_1, p_2, p_3, \ldots, p_m\}$
1: **for each** $c_i \in C$ **do**
2: $\quad p_i \leftarrow 0$
3: $\quad$ **for each** $c_k \in V_i$ **do**
4: $\quad\quad p_i \leftarrow p_i + w_k$
5: $\quad$ **end for**
6: **end for**

in Algorithm 2. This algorithm takes a set of candidate locations with impact values described in Algorithm 1 as input. The output is a list of intersections where RSUs will be deployed.



(a) Process of determining RSU locations (b) RSU location graph

Fig. 4: Determining RSU Locations

1. (Lines 1 and 2) First, variable *Temp* is used to store the candidate location set C to ensure that C has not changed. Initially, the *LocRSU* set is empty.

2. (Lines 3–5) For each iteration, we find the best candidate location c_i having the largest impact value. Noted that if there are some candidate locations having the same impact values, the candidate location with the largest degree is selected. The selected candidate locations in this step are considered as the best locations to install RSUs.

3. (Lines 6–9) When a candidate location is selected to place an RSU, we identify the candidate locations within its transmission range and remove these candidates from the candidate location set *Temp*. Next, we update the impact values of the remaining candidate locations.

4. (Lines 10 and 11) Finally, we obtain an RSU location set.

Algorithm 2 RSU– Locations

Input: C, P
Output: $LocRSU$
1: $Temp \leftarrow C$
2: $LocRSU \leftarrow \varnothing$
3: **repeat**
4: $i = findBest(Temp)$
5: $LocRSU = LocRSU \cup c_i$
6: **for each** $c_k \in V_i$ **do**
7: $Temp \leftarrow Temp \backslash c_k$
8: **end for**
9: update the impact values of the items of $Temp$
10: **until** $Temp = \varnothing$
11: **return** $LocRSU$

3.3 Isolated Intersections

In Section 3.2, we present the strategy for selecting candidate locations to deploy RSUs. We also identify some candidate locations to reduce the number of RSUs being used. This kind of candidate locations is called isolated intersections. The isolated intersections are the intersections that their adjacent intersections are already covered by some RSUs. Because the intersections that are nearby the isolated intersections are covered, we consider removing the RSUs deployed at isolated intersections to reduce the deployment cost.

As shown in Fig. 5, the intersection B is an isolated intersection. This is because the intersection C within its transmission range is covered by other RSUs, and the intersections A and D, which are adjacent to B, are also covered by other RSUs. The pseudocode for marking intersections as isolated intersections is presented in Algorithm 3.

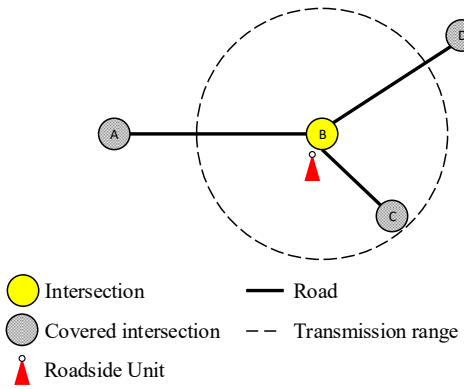


Fig. 5: Isolated Intersection

1. (Lines 1 and 2) First, the algorithm takes an RSU location set, which is the output of the algorithm 2, as input and initiates a loop.

2. (Lines 3–5) For the intersection which is not covered by other RSUs, we find its adjacent intersections. A $LocRSU_k$ is considered as an adjacent intersection of $LocRSU_i$ if there is a road which directly connects $LocRSU_k$ and $LocRSU_i$.

Algorithm 3 Isolated Intersections

Input: $LocRSU = \{LocRSU_1, LocRSU_2, LocRSU_3, \ldots, LocRSU_m\}$
Output: I
1: **for each** $LocRSU_i \in LocRSU$ **do**
2: $\quad I \leftarrow \varnothing$
3: $\quad$ **if** $LocRSU_i$ is not covered by other RSUs **then**
4: $\quad\quad flag \leftarrow true$
5: $\quad\quad Adj \leftarrow getAdjacentList(LocRSU_i)$
6: $\quad\quad$ **for each** $LocRSU_k \in Adj$ **do**
7: $\quad\quad\quad$ **if** $LocRSU_k$ is not covered by other RSUs **then**
8: $\quad\quad\quad\quad flag \leftarrow false$
9: $\quad\quad\quad\quad break$
10: $\quad\quad\quad$ **end if**
11: $\quad\quad$ **end for**
12: $\quad\quad$ **if** $flag = true$ **then**
13: $\quad\quad\quad I \leftarrow I \cup LocRSU_i$
14: $\quad\quad$ **end if**
15: $\quad$ **end if**
16: **end for**

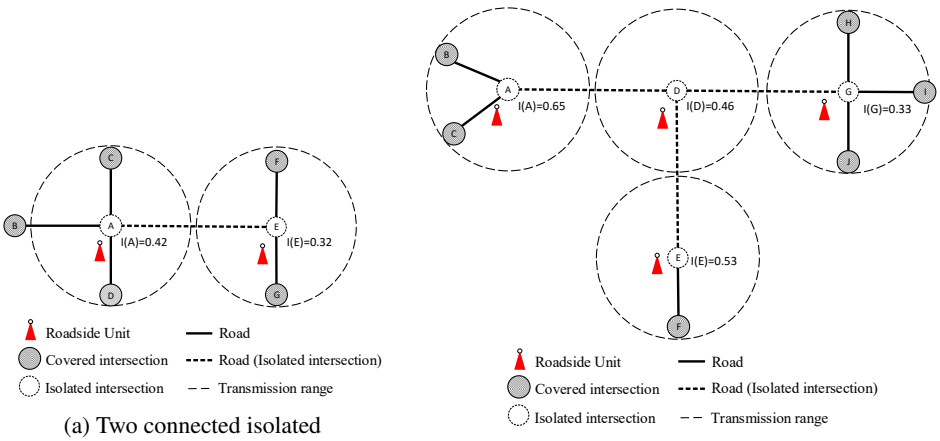
3. (Lines 6–11) To mark $LocRSU_i$ as an isolated intersection, we must ensure that all its adjacent intersections are covered. Thus, if there exists an intersection in the adjacent intersection set of $LocRSU_i$, which is not covered, we ignore the $LocRSU_i$ and proceed to the next iteration.

4. (Lines 12–14) If $LocRSU_i$ is an isolated intersection, we add it into the isolated intersection set I.

5. (Lines 15 and 16) When all the intersections in $LocRSU$ set are checked, the algorithm will be terminated and we will get a list of isolated intersections.

It is unnecessary to deploy an RSU at a single isolated intersection. However, in case there are multiple isolated intersections which are adjacent to each other, without setting up RSUs at these intersections may cause long delay and message loss. For example, as depicted in Fig. 6a, the intersections A and E are connected to each other. If we do not place RSUs at A and E, it does not guarantee that the vehicles within the road segment AE are covered by RSUs. Similarly, as shown in Fig. 6b, the isolated intersections A, D, G, and E are connected to each other. If we do not install RSUs at these locations, the vehicles within these road segments may take a long time to encounter an RSU.

To address this issue, after obtaining the isolated intersection set, we will then construct an isolated intersection graph to determine which isolated intersections must be included in the final RSUs set. For instance, Figs. 7a and 7b present the graphs of isolated intersections acquired from Figs. 6a and 6b, respectively. We then find the minimum set of isolated intersections that RSUs should be installed. We consider each isolated intersection as a vertex and the path connecting two vertices as an edge. Thus, the problem of finding the set of isolated intersections to deploy RSUs becomes the Vertex Cover Problem. As shown in Fig. 7a, either A or E can be in the final RSUs set, however, A has the impact value greater than that of B. Thus, A must be retained in the final RSUs set; E should be removed. Similarly, as shown in Fig. 7b, the isolated intersection D should be kept in the final RSUs; A, E, and G should be excluded. This is because the

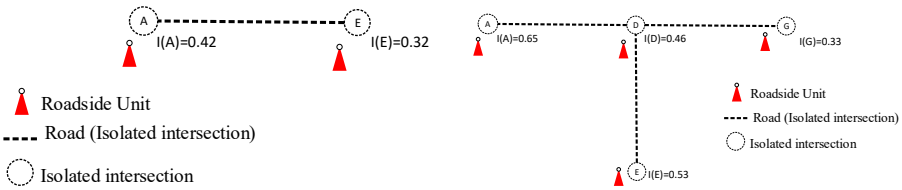


(a) Two connected isolated intersections

(b) Multiple connected isolated intersections

Fig. 6: Connected Isolated Intersections

RSU installing at D can cover the intersections A, E, and G. Figs. 8a and 8b illustrate the final graph obtained after removing the isolated intersections presented in Figs. 7a and 7b, respectively.



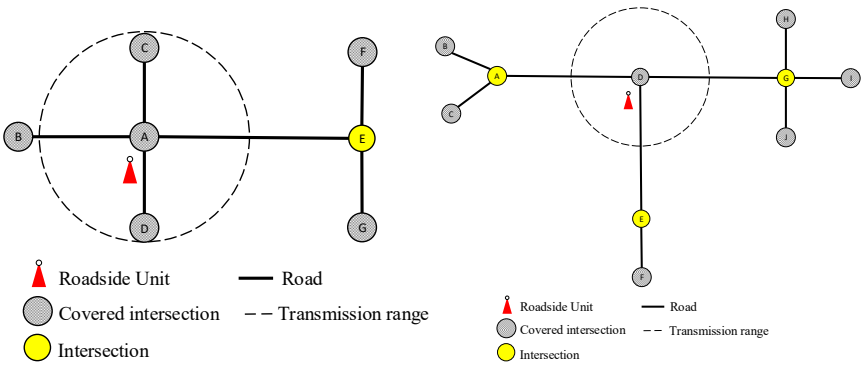
(a) Connected isolated intersections obtained from Fig. 6a

(b) Connected isolated intersections obtained from Fig. 6b

Fig. 7: Isolated Intersection Graph

4. Simulation Results and Analysis

In this section, we conduct experiments on Network Simulator 2.35 (NS2) to evaluate the efficiency of the proposed scheme. We also compare our scheme with the HA [15] and the OGM [14] schemes in terms of the number of RSUs to be deployed, the coverage of intersections, the coverage of vehicles, and the transmission delay. These terms are briefly described as follows. 1) Number of RSUs refers as the number of RSUs needed to be deployed to cover all intersections of a given map. 2) Transmission delay is the transmission delay among vehicles and RSUs. 3) Coverage of intersections is the percentage of intersections which are covered by RSUs. 4) Coverage of vehicles indicates the percentage of vehicles which are covered by RSUs.



(a) The final graph after isolated intersections removing strategy obtained from Fig. 6a

(b) The final graph after isolated intersections removing strategy obtained from Fig. 6b

Fig. 8: Final Graph after Isolated Intersections Removing Strategy

4.1 Simulation Environment

We use the Java language and Network Simulator 2.35 (NS2) for the simulations. First, the number of RSUs and their corresponding locations are calculated using a program written in Java. Next, we take the results as inputs to NS2 for the simulations. The map we use is Taichung City, Taiwan, as shown in Fig 9. The map size is 4(km) $\times$ 4(km) and the number of road segments and intersections are 67 and 53, respectively. We assume that the average vehicle speed is 60 km/h and vehicles travel randomly within the map. The remaining parameter set in NS2 are summarized in Table 2.

Table 2: Parameter Settings of Simulation Environment

Parameter	Value
RSUs transmission range	500m, 600m, 700m, 800m, 900m
Number of vehicles	50, 100, 150, 200
α, β	(0.2, 0.8), (0.5, 0.5), (0.8, 0.2)
Simulation time	600s

4.2 Simulation Results

First, we evaluate the impact of RSU transmission range on the number of RSUs to be deployed. As depicted in Fig. 10, the results show that our scheme outperforms other schemes in terms of the number of RSUs being used. In addition, when RSU transmission range is smaller, our scheme is considerably better than the other schemes. The reason is that when RSU transmission range is smaller, more intersections and road segments are considered as candidate locations. If the number of candidate locations is larger, our scheme detects and removes more isolated intersections, whereas the other two schemes do not.

Fig. 11 shows the comparisons of our scheme with the HA and OGM schemes in terms of the coverage of intersections and the number of RSUs being used under 500m, 700m, and 900m RSU transmission ranges. From Fig. 11 we can see that the percentage of intersections covered by using our scheme is considerably higher than that of the HA

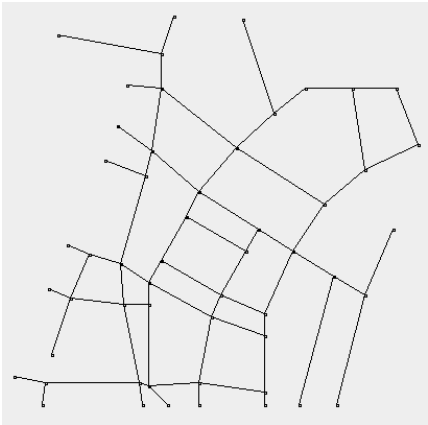


Fig. 9: Simulation Map

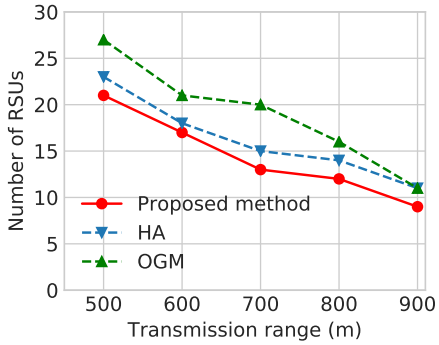
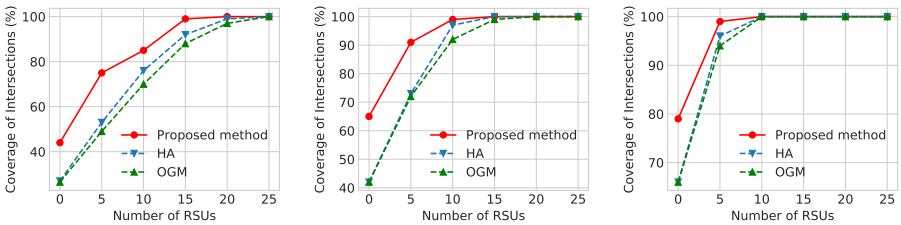


Fig. 10: Transmission Range vs. Number of RSUs

and OGM schemes. For small number of RSUs, shown in Figs. 11a and 11b, our scheme demonstrates the performance up to 20% higher than that of the HA and OGM schemes.

Fig. 12 shows the relationship between RSU transmission range and the number of RSUs being used under different values of α and β. The results show that our scheme works well under variations of α and β. In addition, in the simulation environment, the distribution of vehicle density is uneven and the intersections with high traffic intensity are mostly in the areas having many intersections. When the traffic flow factor α is high, the intersections with high traffic volume will have higher weight values. Thus, the probability of picking these intersections, which can cover more intersections, is higher and the number of RSUs needeed to be deployed is lower.

Fig. 13 presents the percentage of vehicles covered under various transmission ranges. It can be seen that our scheme provides better coverage ratio than that of the HA and OGM schemes. The reason is that we take into account the relation of nearby intersections to select locations to deploy RSUs, while the other two schemes do not. In particular, as shown in Figs. 13a and 13b, our scheme outperforms the above-mentioned schemes by up to 20% when the RSU transmission range is small. Since more candidate intersections exist, more isolated intersections are removed. However, when RSU transmission range is larger, the gaps between our scheme and the other two schemes are



(a) Transmission range of 500m

(b) Transmission range of 700m

(c) Transmission range of 900m

Fig. 11: Transmission Range vs. Coverage of Intersections

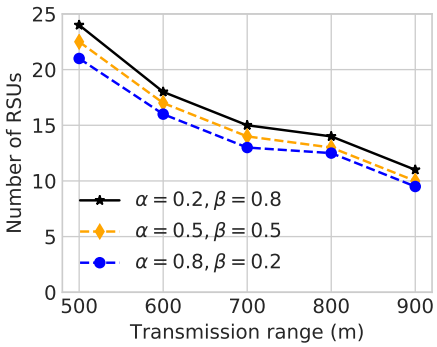


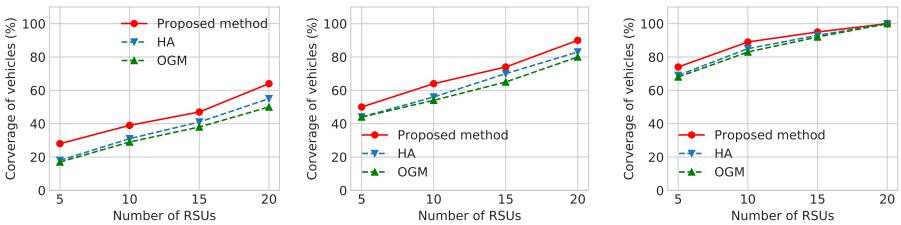
Fig. 12: Impact of Factors α and β on the Number of RSUs being Used under Different Transmission Ranges

reduced, as shown in Fig. 13c. This is because when the RSU transmission range is relatively large, one RSU can cover more intersections. This leads to decreasing the number of candidate intersections and the probability of generating isolated intersections is also lower.

Fig. 14 shows the transmission delay comparisons of our scheme with the HA and OGM schemes under various transmission ranges and different numbers of vehicles. From this figure, we can see that our scheme is not better than the other schemes. This is because, in our scheme, some isolated intersections will be removed if a short delay is allowed to reduce the deployment cost. However, the transmission delay in our scheme is similar to that of the HA and OGM schemes when the RSU transmission range is large.

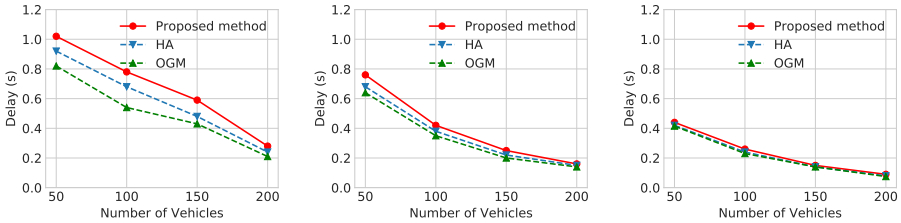
5. Conclusions

This paper proposed an approach for deploying RSUs in an urban VANET environment. In our scheme, intersections and specific road segments are considered as candidate locations for deploying RSUs. Traffic intensity and events are used to compute the weight value and the impact value of each candidate location. These values indicate how important an intersection is, which are used for selecting the locations to deploy RSUs. We also proposed an isolated intersection removing approach to avoid redundant RSUs deployed



(a) Transmission range of 500m (b) Transmission range of 700m (c) Transmission range of 900m

Fig. 13: Transmission Range vs. Coverage of Vehicles



(a) Transmission range of 500m (b) Transmission range of 700m (c) Transmission range of 900m

Fig. 14: Transmission Range vs. Transmission Delay

at some isolated intersections to reduce the deployment cost. To validate our scheme, we conduct a series of simulations with different scenarios. The simulation results show that our scheme considerably outperforms the HA and OGM schemes in terms of the number of RSUs being used and the communication range.

Acknowledgement

The authors would like to acknowledge the Ministry of Science and Technology of the Republic of China for supporting this research under the Grant MOST 104-2221-E-035-022.

REFERENCES

1. E. S. Cavalcante, A. L. Aquino, G. L. Pappa, and A. A. Loureiro, "Roadside unit deployment for information dissemination in a vanet: An evolutionary approach," in *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, 2012, pp. 27–34.
2. Y. Zhang, J. Zhao, and G. Cao, "Service scheduling of vehicle-roadside data access," *Mobile Networks and Applications*, Vol. 15, no. 1, 2010, pp. 83–96.
3. Y. Liu, J. Niu, J. Ma, and W. Wang, "File downloading oriented roadside units deployment for vehicular networks," *Journal of Systems Architecture*, Vol. 59, no. 10, 2013, pp. 938–946.

4. T. H. Luan, L. X. Cai, J. Chen, X. S. Shen, and F. Bai, "Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks," *IEEE Transactions on Vehicular Technology*, Vol. 63, no. 3, 2014, pp. 1419–1435.
5. F. Malandrino, C. Casetti, C.-F. Chiasserini, C. Sommer, and F. Dressler, "The role of parked cars in content downloading for vehicular networks," *IEEE Transactions on Vehicular Technology*, Vol. 63, no. 9, 2014, pp. 4606–4617.
6. T. Yan, W. Zhang, and G. Wang, "Dove: Data dissemination to a desired number of receivers in vanet," *IEEE Transactions on Vehicular Technology*, Vol. 63, no. 4, 2014, pp. 1903–1916.
7. Q. Yu and G. Heijenk, "Abiding geocast for warning message dissemination in vehicular ad hoc networks," in *Proceedings of the IEEE International Conference on Communications*, 2008, pp. 400–404.
8. H. Zhao, H. Gong, K. Liu, M. Liu, and Y. Wen, "Ipad: Intelligent parking-assisted ads dissemination over urban vanets," in *Proceedings of the International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, 2012, pp. 211–218.
9. P. Lin, "Optimal roadside unit deployment in vehicle-to-infrastructure communications," in *Proceedings of the International Conference on ITS Telecommunications*, 2012, pp. 796–800.
10. P. Patil and A. Gokhale, "Voronoi-based placement of road-side units to improve dynamic resource management in vehicular ad hoc networks," in *Proceedings of the International Conference on Collaboration Technologies and Systems (CTS)*, 2013, pp. 389–396.
11. J. Lee and C. M. Kim, "A roadside unit placement scheme for vehicular telematics networks," in *Proceedings of the 2010 International Conference on Advances in Computer Science and Information Technology*, 2010, pp. 196–202.
12. D. Vageesh, M. Patra, and C. S. R. Murthy, "Joint placement and sleep scheduling of grid-connected solar powered road side units in vehicular networks," in *Proceedings of the 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2014, pp. 534–540.
13. T. Yan, W. Zhang, G. Wang, and Y. Zhang, "Access points planning in urban area for data dissemination to drivers," *IEEE Transactions on Vehicular Technology*, Vol. 63, no. 1, 2014, pp. 390–402.
14. R. Rizk, R. Daher, and A. Makkawi, "Rsus placement using overlap based greedy method for urban and rural roads," in *Proceedings of the 7th International Workshop on Communication Technologies for Vehicles (Nets4Cars-Fall)*, 2014, pp. 12–18.
15. J. Chi, Y. Jo, H. Park, and S. Park, "Intersection-priority based optimal rsu allocation for vanet," in *Proceedings of the 5th International Conference on Ubiquitous and Future Networks (ICUFN)*, 2013, pp. 350–355.
16. K. Ota, T. Kumrai, M. Dong, J. Kishigami, and M. Guo, "Smart infrastructure design for smart cities," *IT Professional*, 2018, pp. 1–1.
17. D. L. Moura, R. S. Cabral, T. Sales, and A. L. Aquino, "An evolutionary algorithm for roadside unit deployment with betweenness centrality preprocessing," *Future Generation Computer Systems*, Vol. 88, 2018, pp. 776–784.
18. 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, no. 1, 2018.
19. N. M. Balouchzahi, M. Fathy, and A. Akbari, "A graph-based model for rsus deployment in vehicular networks by considering urban and network limitations and qos

requirements of service advertisement and discovery," *KSII Transactions on Internet and Information Systems*, Vol. 9, no. 5, 2015.
20. A. B. Reis, S. Sargento, and O. K. Tonguz, "Parked cars are excellent roadside units," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, no. 9, 2017, pp. 2490–2502.
21. O. K. Tonguz and W. Viriyasitavat, "Cars as roadside units: a self-organizing network solution," *IEEE Communications Magazine*, Vol. 51, no. 12, 2013, pp. 112–120.
22. M. M. Rahman, M. M. Shuvo, M. I. Zaber, and A. A. Ali, "Traffic pattern analysis from gps data: A case study of dhaka city," in *Proceedings of the IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, 2018, pp. 1–6.
23. S. R. E. Datondji, Y. Dupuis, P. Subirats, and P. Vasseur, "A survey of vision-based traffic monitoring of road intersections," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 17, no. 10, 2016, pp. 2681–2698.

Shiow-Fen Hwang received her B.S., M.S. and Ph.D. degrees in Applied Mathematics from National Chiao Tung University, Taiwan, in 1985, 1987 and 1991, respectively. Currently, she is a Professor in the Department of Information Engineering, Feng Chia University, Taiwan. Her research interests include interconnection networks, mobile computing, and computer algorithms.

Wei-Chen Chen received his B.S. and M.S. degrees in Information Engineering and Computer Science from Feng Chia University, Taiwan, in 2013 and 2015, respectively. Currently, he is an Engineer in Alcor Micro Corporation. His research interest includes vehicular ad hoc network.

Chyi-Ren Dow received the B.S. and M.S. degrees in Information Engineering from National Chiao Tung University, Taiwan, in 1984 and 1988, respectively, and the M.S. and Ph.D. degrees in Computer Science from the University of Pittsburgh, USA, in 1992 and 1994, respectively. Currently, he is a Professor in the Department of Information Engineering, Feng Chia University, Taiwan. His research interests include mobile computing, ad hoc wireless networks, agent techniques, fault tolerance, and machine learning.

Nhut-Lam Nguyen received the B.S. in Computer Science from Can Tho University, Vietnam, in 2005, and the M.S. degree in Computer Science from Griffith College, Ireland, in 2013. He is currently pursuing a Ph.D. degree in the Department of Information Engineering, Feng Chia University, Taiwan. His research interests include ad hoc wireless networks, social media analysis, and machine learning.