

Voronoi-based Placement of Road-side Units to Improve Dynamic Resource Management in Vehicular Ad Hoc Networks

Prithviraj Patil and Aniruddha Gokhale

*Vanderbilt University, Dept of EECS

Nashville, TN 37212, USA

Email: {prithviraj.p.patil,a.gokhale}@vanderbilt.edu

Abstract—Vehicular Ad-hoc Networks (VANETs) illustrate mobile P2P networks, which hold significant promise in improving traffic safety and alleviating traffic congestion. Reliable VANET-based services require dynamic resource management due to limited and often fluctuating network connectivity of VANETs that stem from the wireless and mobile nature of vehicle-to-vehicle (V2V) communications. To address these needs, a collaboration with Road-Side Units (RSU) have been proposed to complement V2V communication by providing event and data brokering capability in the form of Vehicle-to-Infrastructure (V2I) communications. Deploying RSUs involves upfront investment and maintenance costs, and hence solutions are needed that maximize the benefit of RSUs by placing them effectively in accordance to existing and projected traffic density, and the types of services planned for VANETs. To address these challenges, this paper proposes a novel Voronoi diagram-based algorithm for the effective placement of RSUs using packet delay and loss as a criteria. This approach has two-fold advantages: a significant reduction in the number of RSUs required to cover a geographic region, and increase in the logical coverage area of each RSU irrespective of the dynamic vehicular traffic conditions thereby improving reliability of communications. This algorithm has been evaluated in the context of a road network and traffic conditions for an urban area. When compared with other baseline placement algorithms, communication reliability stemming from our Voronoi diagram-based placement algorithm results in less packet delay and lesser packet loss both of which are important to realize the different VANET-based services.

Keywords—Dynamic resource management in vehicular networks; collaboration via road-side units; placement using Voronoi diagrams.

I. INTRODUCTION

Vehicular Ad-hoc Networks (VANETs) are an exemplar of mobile P2P networks that can support a variety of services to help in alleviating traffic congestion via timely rerouting and control of traffic signals, timely evacuation during emergencies, improving safety during lane changing and maintaining minimum braking distance depending on road conditions, and even supporting multimedia applications such as streaming and multi-player games (played by passengers).

Despite the promise offered by VANETs, for these services to indeed be useful to the society and to scale to larger regions and vehicular traffic, VANETs must be able to overcome key challenges stemming from the frequent disruptions in the availability of network connectivity caused in large part to the highly dynamic nature of the vehicle to vehicle (V2V)

network, and bandwidth loss caused due to high and variable speeds of vehicles moving in different directions. For example, since the nodes (*i.e.*, vehicles) in a VANET move very fast – no less than 40 miles/hour on highways – a stable and sustained connection between any two given nodes is highly improbable or at least cannot be assured. Even with a communication range as high as 250 meters between any two vehicles, at such speeds, the vehicles can establish a effective connection for at most two seconds(not accounting for initial connection setup delay), particularly when they are heading in opposite directions.

Such a short time duration to maintain communication capabilities makes it hard to realize the range of services outlined above – particularly when the timeliness and quality of the disseminated data is paramount. To overcome these challenges, it has been suggested to deploy road-side units (RSUs) as a means to provide a collaborative mechanism for dynamic resource management in VANETs and provide the QoS assurances to applications. The collaboration between vehicles and RSUs is enabled through a vehicle-to-infrastructure (V2I) network.

Despite many technical advances and standardization in this realm, a number of challenges still remain unresolved. Of particular importance to us in this paper is the issue of effective placement of RSUs. RSUs help to overcome the inherent unreliability in V2V communications by providing an alternate, high availability channel for communication and serving as brokers for data exchange thereby promoting dynamic resource management among vehicles, and the vehicles and the wired network. However, an *ad hoc* placement of RSUs will not address the original problem. Moreover, a solution comprising RSUs requires an upfront investment, and hence their placement must be planned in accordance to several factors including (a) present and projected traffic patterns and vehicular density, (b) the variety of services that are emerging and their communication profile, and (c) the technical advances as well as limitations of the underlying communication mechanisms. All the while, the cost of placement and maintenance must be kept low for it to be an economically viable solution.

In this paper we address these requirements by proposing a novel RSU placement algorithm based on the notion of a

Voronoi Diagram [1]. We evaluate the effectiveness of RSU placement resulting from our algorithm in the context of a road network and traffic conditions for representative urban area. When compared with other baseline placement algorithms, the reliability and availability of vehicular communications stemming from our Voronoi diagram-based placement algorithm illustrates significantly less packet delay and lesser packet loss both of which are indicators of higher reliability and availability, and are important to realize the different VANET-based services.

The remainder of the paper is organized as follows. Section II discusses related research in RSU and V2V communications, comparing them with our work; Section III describes the Voronoi-based algorithm and the methodology for efficient placement of RSUs; Section IV describes a range of experiments using simulation to test the efficacy of algorithm providing results of these simulations; and finally Section V provides concluding remarks highlighting the lessons learned and provides directions for future work.

II. RELATED WORK

This section describes related research comparing it with our research. Although the core problem we address in this paper is the placement of RSUs to improve reliability of vehicular communications, our technique is developed considering specific models of vehicular mobility and the need to balance data traffic across available communication mechanisms, such as V2V and V2I. Therefore in this section we focus on the following dimensions of prior research: (a) the models used to characterize and simulate traffic, (b) the communication mechanism used, such as V2V or V2I or both, and (c) works that are similar in spirit to ours.

A. Models and Algorithms Used

Studying how applications behave in mobile and wireless environments requires models for mobility, and assumptions about the wireless environment. Earlier experimentation considered different mobility models, such as random mobility or random waypoint, and different traffic distributions like Exponential and Gaussian. These mobility models were used predominantly in research on mobile ad hoc networks (MANETs). VANETs are similar to MANETs, however, with a key difference: vehicles in VANETs cannot move randomly as prescribed by the random waypoint models, and moreover, they move at significantly higher speeds than those considered in MANET research. Thus, our research was required to leverage only those models that are compliant with the VANET characteristics.

Specialized models for VANETs were developed in the past, such as the *car following model*, and *cellular automaton model*. In the car following model [2], [16], [20] vehicles move in a single lane and their behavior depends entirely on the vehicle ahead of it. In the cellular automata model [3], [10], [12], the entire region is divided into a number of cells where the behavior of vehicles in one cell depends on the behavior of vehicles in adjacent cells. Other models like *fluid flowing*

model [4], [8], [9] also exist, where vehicles are considered to be behaving as if they are part of a platoon and hence modeled as a group and not individually. The flow of one platoon is dependent on the flow on the nearest platoon ahead of it.

Since real-world traffic data is of aggregated nature (*e.g.* the number of vehicles crossing a junction), proper simulation of traffic data using mobility models is necessary to obtain meaningful results. Our research leverages the car following model since modeling the V2V communication, which is based on hop-by-hop data traversal among vehicles is better captured when individual vehicles are the primary artifact of any model.

B. Communication Mechanism

Our survey of previous work that aim to provide real-time applications in VANETs can be classified into those that primarily use V2V or V2I but seldom both.

1) *Vehicle to Infrastructure (V2I) communication*: In [22], the probabilistic delay for each RSU has been analytically determined using the maximum speed and total average traffic for the road segment. In [19], a power saving model based on number of RSUs required to achieve a specified connectivity level is presented. The model has also been analytically validated. Game theory is used in [17] to illustrate a distributed approach to obtain an optimum number of RSUs. However, this approach is deemed useful for sparse networks only, where fluctuations in traffic is not substantial. So the work is most applicable to traffic on freeways where the traffic is free flowing and does not incur high fluctuations. In contrast, our work proposes a placement of RSU in an environment which caters to multiple traffic conditions all at once: sparse, dense, stable, and fluctuating. In another approach [11], vehicles do not communicate with RSUs individually but through one leader to reduce the network traffic and to use bandwidth efficiently. The leader will collect all information from other nodes and share it with RSUs.

2) *Vehicle to Vehicle (V2V) communication*: Several approaches have been proposed for vehicles to communicate efficiently with other vehicles using V2V. In [22], vehicles communicate based on the nature of data to communicate, *e.g.*, priority of the data. Only vehicles with higher priorities will get a particular data. An approach where vehicles communicate pairwise in opposite directions is presented in [18]. In another work [5], vehicles communicate based on their locations on particular road. First, the vehicles share their locations with remote nodes, and then decide which information to share. These approaches focus only on V2V communication, which as we see later is not enough for VANET applications.

C. Capacity Planning and Infrastructure Related Work

The possibility of leveraging existing infrastructure like cell phone towers as RSU-equivalents have been tested in prior work. In [14], the authors estimate the number of communicating vehicles per cell tower in the city of Munich. Their results show that at most 600 vehicles can be reached per cell tower. Since cell towers are not deployed considering road network, the authors conclude that cell towers may not be

sufficient for VANET communication. A similar approach has been tested in [6], however, this approach used a combination of different technologies including GPS, WiFi and cellular network to communicate among cars. Another paper [21] uses Voronoi Diagrams for determining the coverage area by cell towers, however, the placement of cell towers is not based on Voronoi Diagrams but only to determine the optimum coverage area of each cell tower. Another approach for placement is evaluated in [13], but it uses only junctions (mostly busiest junctions) for RSU placement.

In all of the above approaches we surveyed, prior work attempts to improve network communication among the vehicles (and infrastructure) but do not consider the fluctuations in traffic patterns which depend upon factors such as time of the day and location. Moreover, these approaches also exclusively consider either V2V or V2I but not both. In contrast, our work achieves dynamic resource management by establishing a collaborations between vehicles (*i.e.*, V2V) and RSUs (*i.e.*, V2I) by making effective placement decisions for RSUs. Moreover, our work also considers fluctuating traffic conditions, time of the day and location information.

III. VORONOI DIAGRAM-BASED RSU PLACEMENT

This section presents our RSU placement algorithm that is based on the idea of a Voronoi Diagram [1]. In a Voronoi diagram,¹ a region is divided into cells that are convex polygons. The formation and shape of each polygon is dictated by the presence of a unique point (sometimes called the generating point) within that polygon such that every other point in that cell is closer to its generating point than any other generating points of other cells formed. Figure 1 depicts a conceptual rendering of a Voronoi diagram.

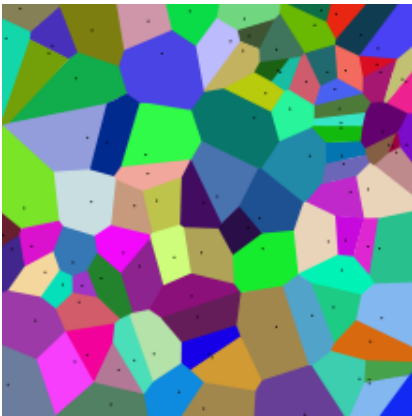


Figure 1. Voronoi Diagram of N random points. Every point in particular region is closest to the shown point of that region than a point in any other region. Figure is borrowed from [1].

A. Key Idea in Using Voronoi Diagrams for VANETs

We describe why Voronoi diagrams provide a promising approach to resolving the RSU placement problem. Recall that

¹Also known as Voronoi Network. We will refer to it as diagram to avoid confusion with vehicular networks.

our goal is to improve the reliability of VANET connectivity while minimizing the number of RSUs needed to cover as large an area of the transportation network as possible. The key insight in our placement algorithm stems from the definition of Voronoi diagrams and the role played by the generating points in the convex polygons. Let us assume that the RSUs are placed at the location of the generating point of the convex polygons of the Voronoi diagram that are formed for a given urban road network. In such a setup, the contours of the convex polygon that surrounds the RSU will be formed based on whatever criteria we choose to form the polygon.

In our case since we are interested in providing reliable connectivity to VANET services that provide assured latencies, one criteria to form the polygons can be the maximum tolerable packet delay. In turn it means how further away can data packets travel from the RSU potentially taking multiple hops between individual vehicles that are traveling on roads away from the RSU before a specified time limit is reached. Thus, the polygon surrounding the RSU may be larger than the RSU's physical signal range since the packet can traverse distance larger than the signal range due to multihop communication. This in turn means that a single RSU can reliably cover a larger area of the transportation network while still meeting the packet delays.

Due to the properties of Voronoi diagram, each vehicle will be now be covered by one RSU only, which also happens to be its nearest RSU from the perspective of delay incurred. This criterion maximizes the area covered by given RSUs while meeting the allowable delay bound for every vehicle since each vehicle based on its location is assigned to nearest RSU and no vehicle is left unassigned. If this criteria were aggregated with maximizing the number of vehicles served by the RSU, *i.e.*, minimizing the packet losses, then by reconciling these two parameters in the formation of Voronoi diagram, we will have a solution for capacity planning for RSU placement problem in VANETs.

B. Illustration via an Example

We explain our idea using a simple example. Consider city planners wanting to deploy n gas stations at optimal locations to offer their service more efficiently. To begin with, they can create a Voronoi diagram by using distance as a parameter and apply it to the city map with gas stations being the head node of the Voronoi diagram. After a Voronoi diagram is created, every other point in the map will be associated with one of the gas stations in such a way that the distance from that other point to a specific gas station is smallest compared to distance to any other gas station. In this way, every point in the city will be associated to one gas station that is nearest to it.

However, the diagram thus formed is based solely on one parameter – distance – but does not account for the population density. As a result, it is likely that gas stations may end up getting uneven number of customers. To offset these load imbalances, we can create another Voronoi diagram using the number of customers as a parameter. After the new Voronoi diagram is formed using customers as a parameter, the area

covered by a specific gas station, say A, will be proportional to its number of customers. Thus, larger the number of customers in an area, larger is the cell size formed by the Voronoi diagram for that gas station. Gas station A will now be surrounded by cells of different sizes in which other gas stations are placed. Consequently, we balance the load among A and its neighbors such that all gas stations will have equal load share.

However, recall that minimizing the driving distance to a gas station was the original criteria. Therefore, it is important to reconcile the Voronoi diagram formed using number of customers as the parameter with the Voronoi diagram formed using distance as a parameter.

C. Two-Stage Voronoi Diagram Algorithm for VANETs

Our Voronoi diagram approach for VANETs in an urban region to place RSUs follows very much the same approach as in the gas station example. The difference in the VANET case is that in the first pass we use *maximum delay*² encountered by a packet transmitted from one RSU to reach another RSU as a metric to compute the RSU neighborhood map in the Voronoi diagram. Any packet delays exceeding this limit hinders service availability. Recall that an urban region, where these services are to be deployed, comprises a mix of areas with freeways, highways, and residential neighborhoods with smaller roads and lanes. Consequently the movement and speeds of vehicles within the signal range of an RSU will be different depending on where in the urban area that RSU is placed, which in turn will impact the propagation of data packets and the delay they encounter. Our algorithm considers these possibilities. At the end of first phase we will have computed a Voronoi diagram based on delay metric but it has the likelihood of creating load imbalance among RSUs. The second phase of our approach adjusts the RSU placement within this neighborhood map by balancing load to alleviate packet losses. Both these steps together improve the reliability and availability of VANET communications.

1) **Stage I Algorithm:** Algorithm 1 describes the first stage of the RSU placement algorithm for VANETs using Voronoi diagrams. This algorithm makes RSU placement decision assuming one kind of VANET-based service that is to be deployed. The algorithm takes as input the maximum number of RSUs that are allowed by the city planners, a map of an urban area, mobility properties of traffic, and maximum tolerable delay for the service before its quality degrades, and a few other parameters.

The algorithm starts by placing RSUs at some initial position either randomly or by spreading them evenly (Line 1). If they are evenly spread out, the distance between any two RSUs is denoted D_{init} . Since an RSU may be placed in different areas of an urban region, it is highly likely that the density of vehicles in the signal range of that RSU, and the speeds at which they travel are going to be different. To determine the density of vehicles in a region, our algorithm uses population

²An upper limit on the tolerable delay can be defined by the quality of service requirements of the service to be deployed in the VANET.

Algorithm 1: Computing RSU Neighborhood Map

```

Input:
 $N$  = Number of RSUs available // evenly placed to begin with
 $A$  = Area under consideration in square miles // using a map of a region
 $R_s$  = Signal range of an RSU in meters
 $R_{veh}$  = Range of a vehicle's receiver
 $D_{init}$  = Initial distance between two RSUs // in evenly placed case
 $d_{max}$  = Maximum allowable delay suffered by a packet
 $B_{veh}$  = Bandwidth available to each vehicle
 $\alpha$  = Coefficient of effective bandwidth loss due to speed
 $\beta$  = Degree of fine-graininess over traversal direction
 $t_{sim}$  = time period the algorithm is executed (say as a simulation)
Data: // local variables
 $R_{ext}$  = Extended range of an RSU
 $V_h$  = Average number of vehicles per hour in a given region
 $V_m$  = Average number of vehicles per square mile for a given region
 $S_h$  = Average speed of vehicle when on Interstate(highest)
 $S_m$  = Average speed of vehicle when on state highways
 $S_l$  = Average speed of vehicle when on other roads(lowest)
 $B_{loss}$  = Effective Bandwidth loss
 $RouteSet_{ij}$  = set of routes from  $i$  to  $j$ 
Output:
Neighborhood map of RSUs based on packet delay as the parameter
 $N_u$  = Number of useful vehicles /* Computed as a side-effect and used in the
next stage for optimization */
begin
/* Start with an initial placement of RSUs. The value of  $D_{init}$  will depend
on such an initial placement that is chosen. */
1 Initial Condition: Spread  $N$  nodes evenly across  $A$ 
/*  $D_{init}$  is a constant in this case. Also, note that an RSU is referred to as
a Node in the algorithm. */
2 for  $i = 1 \rightarrow N$  do
3   Compute values of  $V_h, V_m, S_h, S_m, S_l$  in the signal range of node  $i$ 
4    $RouteSet_{ij} \leftarrow$  subset of all possible routes from node  $i$  to every other
node  $j$  (other than  $i$ ) that are  $\beta^\circ$  radians apart
5   foreach route  $r$  in  $RouteSet_{ij}$  do
6     Propagate a packet up to distance  $D_{init}$  until time  $d_{max}$  is
reached utilizing all available vehicles moving at speeds  $S_h, S_m,$ 
and  $S_l$  away from node  $i$  along route  $r$  leveraging both V2I and
V2V
7     if  $propagationtime == d_{max}$  then
/* We have reached the maximum allowed propagation
time as prescribed by the quality of service of the
application */
8     Note the location coordinates on the map at this instance when
time  $d_{max}$  is reached
9      $R_{ext_i} \leftarrow R_{ext_i} \cup \{locationcoordinates\}$ 
10    Calculate the bandwidth loss  $B_{loss_i}$  while traveling for each
speed zone,  $S_h, S_m, S_l$ , using the coefficient of effective
bandwidth loss,  $\alpha$ 
11     $N_u = (V_h \times t_{sim}) \times \frac{1 - B_{loss_i}}{B_{veh}}$ 
12    end
13  end
14 endfor
15 foreach pair of nodes  $i, j$  in  $N$  do
16   if  $R_{ext_i} \cap R_{ext_j} \neq \emptyset$  then
17    Nodes  $i$  and  $j$  are neighbors
18   end
19 end
20 Save the neighborhood map for each node with the number of useful vehicles
encountered,  $N_u$  during travel as weight on the links
end

```

density data available from census data and computes these parameters (Line 3) for every RSU. The traffic movement generation is explained in section IV-A.

Since our goal is to determine the polygons of the Voronoi diagram based on packet delay as the metric, the algorithm proceeds by determining how much distance can a single packet propagate from one RSU to the other using both V2I and V2V communication until the max delay is reached. We do this by assigning a single packet belonging to the chosen service to all the RSUs in the region and let the packet propagate from a given RSU towards other RSUs (Line 6) by

leveraging the vehicles within its signal range (through V2I channel) as well as by hopping from one vehicle to another (through V2V channel). Since there could exist hundreds of different road routes available to the vehicles to travel from one RSU to another, to prevent unnecessary and exhaustive searching, we choose a subset of such routes by computing a route set for the given RSU (Line 4), and allowing vehicles to travel only on those routes (Line 5).

Every packet will propagate a certain distance within a prescribed maximum duration (d_{max}) and may not necessarily reach the other RSU. The maximum farthest set of locations that a packet can travel given the mix of vehicles and their speeds along the routes they are allowed to take are computed (Line 8) once the packet delay bound is reached (Line 7). A set of all such locations represents the *extended range* of a RSU, which is computed in Line 9. The extended range of RSU determines the contours of the polygon while the position of the RSU becomes the generating point. If the extended range of any two pairwise RSUs overlap (Line 16), then the RSUs in the pair are deemed to be neighbors. Consequently, a neighborhood map of all RSUs for the entire region is computed (Line 20).

Also, each vehicle causes bandwidth loss because of its speed. To take this loss into consideration we compute, N_u which is the portion of total vehicles which respect the delay bound despite bandwidth loss while relaying packet. N_u determines level of overlap between two RSU ranges.

Note that the resulting neighborhood map of RSUs computed by Algorithm 1 will very likely produce areas of overlap of the extended areas of RSUs that are uneven in size as seen in Figure 2. In other words, some overlapping areas may be very large while there may be gaps between the extended range. Large overlapping areas of extended coverage of neighbor RSUs represent wasted resources and lost opportunity to disseminate information over larger areas. Similarly, gaps represent areas where communication is highly unreliable. Thus, if we were to reduce the overlap by moving the RSUs away from each other, there may be an opportunity for an RSU to become a neighbor of someone else that originally was not the case thereby filling some of the gaps. Such a step would in fact improve the overall coverage area over which information can be reliably disseminated.

2) **Stage II Algorithm:** The objectives of the second stage of our algorithm shown in Algorithm 2 are:

- To remove overlapping areas, and
- To remove unattended areas.

Once we achieve above two objectives, resulting diagram becomes a true Voronoi Diagram that is useful for our purpose.

Algorithm 2 takes as input the number and position of initial RSUs with their neighbors and N_u . It then iterates for each RSU by visiting its neighbors over all possible routes and tries to relocate itself at a place where it will witness maximum N_u . Since every RSU will try to maximize its N_u (at the cost of neighbor), at end of the iteration, N_u will be shared equally among all RSUs. Also this will remove any unattended spaces since in order to maximize its N_u , a RSU will try to

Algorithm 2: Adjusting RSU Placement

```

Input:
   $N$  = Number of RSUs available
   $N_u$  = number of Useful Vehicles
  Neighborhood map of  $N$  RSUs with link weight as  $N_u$ 
Data: // local variables
   $Nei_i$  = Neighbors of  $i$ 
   $Placement_i$  = Placement of RSU  $i$ 
Output:
  Optimal placement positions of  $N$  RSUs
begin
  Initial Condition:  $N_n$ , number of neighborhood maps
  for  $i = 1 \rightarrow N_n$  do
     $Nei_i$  = Neighbors of  $i$ 
    Traverse from  $i$  to  $Nei_i$ 
    foreach Road Segment during traversal do
       $Placement_i$  = Current road segment if  $N_u$  maximizes for
      current RSU
    end
    Move  $i$  to  $Placement_i$ 
  endfor
end

```

incorporate such areas in its extended range. At this point, RSUs will be placed in a place such that they leave no empty spaces and do not encroach on others' space.

Though algorithm 2 is heuristic based, the complexity lies in determining neighborhood map and N_u of each RSU in algorithm 1. The outputs of both algorithms is shown in Figure 2. As can be seen, the RSUs in the center figure is not neighbor of all of its surrounding RSUs. We believe this precise determination of neighbors of each RSU plays an important role for overall extended range improvement for every RSU.

IV. EXPERIMENTAL EVALUATION

This section evaluates the effectiveness of the solutions generated by our Voronoi diagram-based algorithm comparing them with other baseline techniques. We first present the experimental setup followed by the results.

A. Experimental Setup

The evaluations of our algorithm presented in this paper have been performed by using extensive simulations. The simulation includes three different types of experiments. Since no single simulator is readily available to conduct the evaluations, different simulators have been used to simulate different experiments as explained below.

Vehicular Traffic Simulator: To simulate vehicular traffic data, we used the SUMO (Simulation of Urban Mobility) [15]. SUMO is an open source simulator used in the simulation of traffic on large road networks. It focuses on microscopic behavior, *i.e.*, simulation at the level of individual vehicles and behavior of drivers is feasible.

Data for Traffic Simulation: An important consideration for our research was the need to simulate real-world traffic patterns. Though we had access to real-world vehicular traffic information, the kind of data we possessed did not serve our purpose because the information contained data mostly about freeways and highways but not all the roads of an urban area (Nashville, TN, USA in our case). We therefore considered a different approach. We instead collected population census

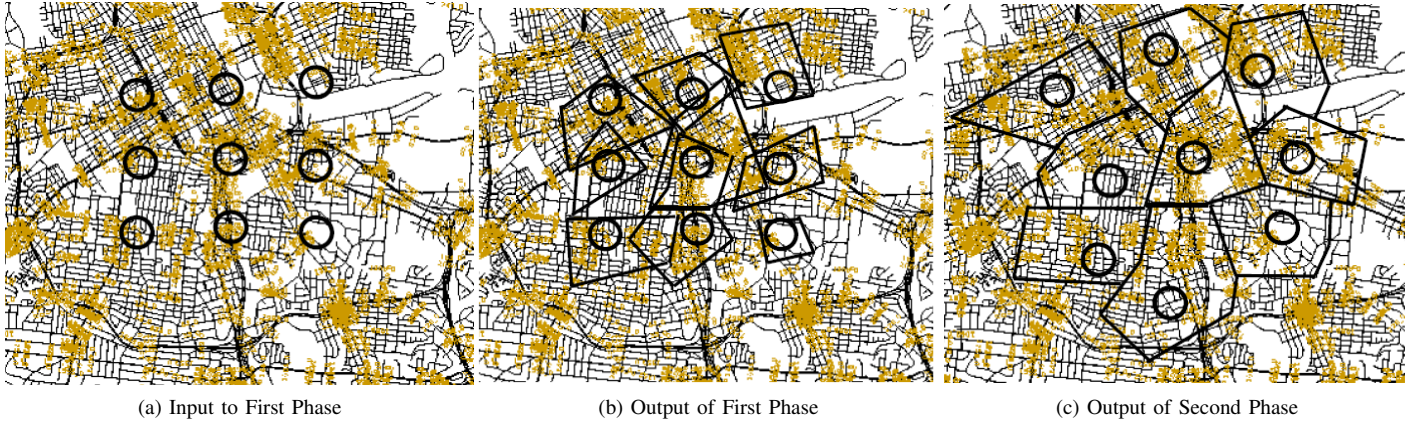


Figure 2. Maximizing RSU Ranges Shown for 9 RSUs (Circle represents physical range; Polygon represents extended range)

data from Tiger maps that are available online for every major US city [7]. This population information is then used within SUMO’s ActivGen API to generate traffic data mimicking the real-world. We considered the urban area for the city of Nashville, TN, USA focusing on an area approximately 20×20 sq. miles. Table I shows the different parameter values we used during this part of the simulation.

TABLE I
SUMO CONFIGURATION PARAMETERS

Parameter	Value
Simulator	SUMO 0.12.3
City	Nashville, TN, USA
Area	20×20 sq. ml
Vehicles	10,000
Vehicle Types	2 (Bus and Car)
Traffic Signals	Yes
Bus Stop	Yes
Bus Stop Time	10 seconds
Route Selection	Shortest Time
Mobility Pattern	Car following model
Simulation Time	Six Different Zones (morning rush, noon lull, evening rush, midnight, early morning, special events/days)
Vehicle Speed	Three zones (high, medium, low)

Network Simulation: Both V2V and V2I communications require a network simulation capability which does not exist within SUMO. Hence we used the widely-used Network Simulator (NS2) version 2.33, where we simulated connectivity among vehicles and also between vehicles and RSUs. The movement of vehicles was obtained through the simulation in SUMO. SUMO was bridged with NS2 using the TracI bridging capability. For our evaluations we focused on just one service to be deployed in the VANET. Thus, every RSU was assumed to contain packets belonging to a single type of service. Packets belonging to the service are assumed to emanate from RSUs, which then spread to the vehicles using both V2I and V2V communications.

TABLE II
NS2 CONFIGURATION PARAMETERS

Parameter	Value
Network Simulator	NS2-2.33
Packet size	variable(16, 32, 64 bytes)
Period	1 min
Queue	Droptail
Queue length	50
Propagation	Two Ray Ground
MAC protocol	802.11p
Range Of RSU	50 meter
Range of vehicle	10 meter
Effective Bandwidth Loss	Velocity model
Maximum Allowed Delay	2 minutes

Baseline and evaluation metrics: For evaluating the effectiveness of our algorithm, we use following performance metrics: Average extended range of RSUs (E_r), packet delay, packet loss and average area covered by RSUs. We also evaluate the reliability of network connectivity using our placement. We compare our algorithm to two baseline approaches. The first baseline performance is obtained by placing RSUs evenly spread across a given area. The second baseline performance was obtained by placing RSUs at the busiest junctions/traffic lights over the chosen urban area. Since it is easier to place RSUs at an existing infrastructure like a traffic light, the transportation agencies usually adopt the second approach. Though this approach appears reasonable and economical, our results demonstrate that it is in fact not vastly better than the naive approach of placing RSUs evenly.

B. Experimental Results

We present results evaluating the effectiveness of our Voronoi diagram-based approach using following metrics. **(1) Average Extended Range of RSUs (E_r):** Because vehicles are assigned to each RSU in near optimal fashion in our approach, the overall V2V communication improves and results in an extended range of RSU. This occurs due to vehicles becoming carriers of RSU-generated packets to an area outside the real signal range of RSU and hence effectively increases the logical range of RSU, which is called as extended

range, E_r , in this paper. Figure 3 demonstrates that RSUs get maximum E_r when placed according to Voronoi-based network. With our approach, the improvement in the extended range is remarkable and also increases more rapidly with additional RSUs. The same trend is not seen with the other two approaches as we can see that both other approaches (busiest junction placement and even placement) not only perform worse but also do not improve performance with added RSUs. It can be assumed in these two approaches that the range is not extensible any more. The range remains static in fact irrespective of the number of RSUs. Contrary to these approaches, in the Voronoi-based approach, RSUs will have increased range with more number of RSUs deployed. This decreases the cost of RSUs for large metropolitan areas because with addition of just one RSU, we are able to derive more range per existing RSUs.

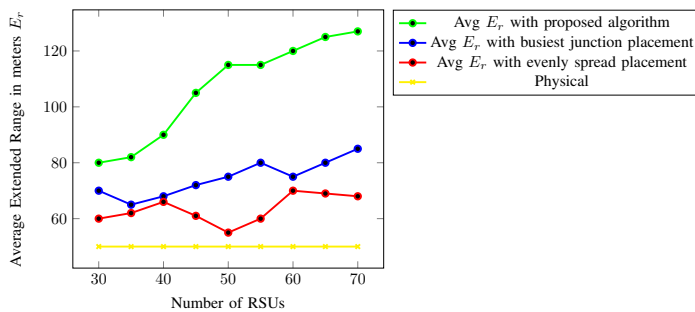


Figure 3. Evaluation of Average Extended Range, E_r .

(2) Packet loss: The second metric is the number of packets lost by vehicles. This metric is dependent on the maximum allowable delay. Any packet received after this limit is considered as lost. We set the maximum allowable delay to 2 minutes for this evaluation. As noted earlier, each vehicle is supposed to receive one packet of x bytes per minute either from one of the RSUs or from any other vehicle. We have repeated same experiment for different sizes of packets like 16, 32, 64 and so on. It validates the suitability of our approach for different types of applications like traffic safety and multimedia applications. For this metric, the results are shown in Figure 4. As can be seen, our placement causes less packet loss compared to the other two approaches. Also this performance is consistent for different number of RSUs.

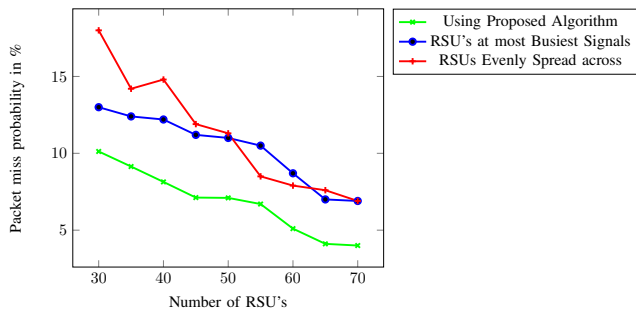


Figure 4. Evaluation with Packet Loss metric

(3) Packet Delay: Not just receiving a packet is important, but receiving it within the maximum allowable delay is desirable in VANETs for assuring the quality of service for the VANET service. This is important because most of the information, such as traffic information, weather or accident information are temporally sensitive. So we have compared our approach against other two approaches with the packet delay metric. Here, we consider packet delay for only those packets which are received before the maximum packet delay limit. Any packet received after this time limit is simply discarded as we consider such packet of no use or of very little utility. The results are shown in Figure 5. As seen, the placement of RSUs using our algorithm outperforms other cases by causing less packet delay than other two placements irrespective of number of initial RSUs used.

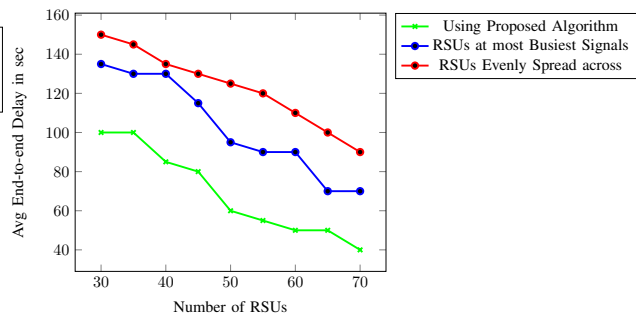


Figure 5. Evaluation with Packet Delay metric

(4) Coverage of Area: A direct consequence of increased extended range of RSU is that our algorithm covers much more area for a given number of RSUs (and for fixed packet delay and loss) compared to other approaches. As can be seen in Figure 6, the area covered by Voronoi Algorithm placement for 30 RSUs is more than double that of placing RSUs at busiest junction. Also this improvement persists for higher number of RSUs as high as 70. This clearly shows that the proposed placement algorithm improves the coverage area of RSUs and hence requires less number of RSUs for a given area, which makes the solution more economically viable compared to other approaches.

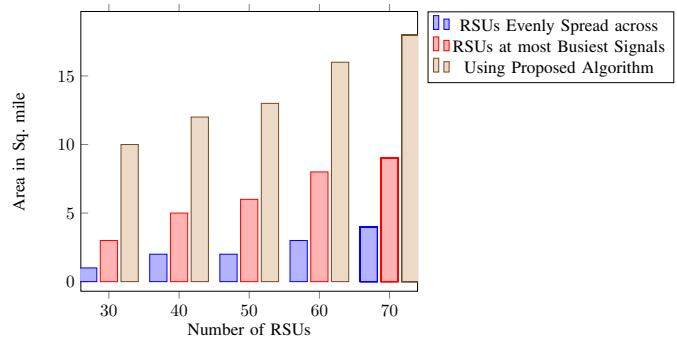


Figure 6. Evaluation of Area Covered

V. CONCLUSIONS AND FUTURE WORK

The societal and environmental benefits stemming from the VANET-based services can be adjudged only when there exists an effective dynamic resource management over the highly dynamic, wireless and mobile vehicular networks. To achieve these end results, however, requires resolving a wide range of challenges in the realm of VANETs. We focused on one of the many challenges in this space, namely, capacity planning and deployment of RSUs, which serve as agents that enable collaborative dynamic resource management. By no means does the resolution of just the capacity planning and deployment challenge solve all problems in this space. Nonetheless, it is a necessary dimension of the problem space that requires new solutions, and we present our solution based on Voronoi diagrams.

Our algorithm adjusts the placement of RSU in accordance to the traffic data populated from census data of Tiger maps of US census bureau. In doing so, it benefits from the traffic patterns and maximizes the normal range of RSU to an extended range. This in turn minimizes the number of RSUs required to achieve a specific quality of service of connectivity. Our approach performs better than other baseline approaches that use evenly spaced RSUs or RSUs placed at junctions in terms of lesser delay and lesser packet loss. We have validated our claims with extensive simulations using NS2 for network, SUMO for traffic mobility, and Tiger Maps for populating traffic patterns for the city of Nashville, TN, USA.

The following limitations exist in our approach, and our future work is investigating solutions to overcome these limitations.

- The placement decisions made by our algorithm may not always be implementable because the locations where they are to be deployed may involve private land, nor does our work take into consideration obstructions, such as hills and buildings. Addressing these additional constraints in the context of our algorithm will be part of our future work.
- For faster evaluation and scalability of the algorithm, we intend to integrate our algorithm in a programmatic way into an existing OMNET simulator using TraCI interface. TraCI is the interface for connecting OMNET with other tools such as SUMO. This would help in faster evaluation of our algorithm using different city areas and different traffic patterns or real world traffic data.
- Our future work will seek how dynamic resource management solutions using our RSU placements improve application QoS, and scale.

ACKNOWLEDGMENTS

This work was supported in part by NSF CAREER CNS 0845789, SHF/CNS Award CNS 0915976, and Vanderbilt IDEAS grant. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the

author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] Franz Aurenhammer. Voronoi diagrams a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23:345–405, September 1991.
- [2] Mark Brackstone and Mike McDonald. Car-following: a historical review. *Transportation Research Part F: Traffic Psychology and Behaviour*, 2(4):181 – 196, 1999.
- [3] Elmar Brockfeld, Robert Barlovic, Andreas Schadschneider, and Michael Schreckenberg. Optimizing traffic lights in a cellular automaton model for city traffic. *Phys. Rev. E*, 64:056132, Oct 2001.
- [4] Debashish Chowdhury, Ludger Santen, and Andreas Schadschneider. Statistical physics of vehicular traffic and some related systems. *Physics Reports*, 329(4-6):199 – 329, 2000.
- [5] S.M. Das, H. Pucha, and Y.C. Hu. Performance comparison of scalable location services for geographic ad hoc routing. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 2, pages 1228 – 1239 vol. 2, march 2005.
- [6] G. Ferrari, S. Busanelli, N. Iotti, and Y. Kaplan. Cross-network information dissemination in vanets. In *ITS Telecommunications (ITST), 2011 11th International Conference on*, pages 351 –356, aug. 2011.
- [7] US government website. <http://www.census.gov/geo/www/tiger/>, November 2008.
- [8] Dirk Helbing. Improved fluid-dynamic model for vehicular traffic. *Phys. Rev. E*, 51:3164–3169, Apr 1995.
- [9] Dirk Helbing. Traffic and related self-driven many-particle systems. *Rev. Mod. Phys.*, 73:1067–1141, Dec 2001.
- [10] Kai Nagel and Michael Schreckenberg. A cellular automaton model for freeway traffic. *J. Phys. I France*, 2(12):2221–2229, 1992.
- [11] G. Korkmaz, E. Ekici, and F. Ozguner. A cross-layer multihop data delivery protocol with fairness guarantees for vehicular networks. *Vehicular Technology, IEEE Transactions on*, 55(3):865 –875, may 2006.
- [12] H. K. Lee, H.-W. Lee, and D. Kim. Macroscopic traffic models from microscopic car-following models. *Phys. Rev. E*, 64:056126, Oct 2001.
- [13] Junghoon Lee and Cheol Kim. A roadside unit placement scheme for vehicular telematics networks. *Physics Reports*, 329(4-6):199 – 329, 2000.
- [14] T. Mangel and H. Hartenstein. An analysis of data traffic in cellular networks caused by inter-vehicle communication at intersections. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 473 –478, june 2011.
- [15] Simulation of Urban Simulation. <http://sumo.sourceforge.net/>, November 2008.
- [16] P.G. and Gipps. A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological*, 15(2):105 – 111, 1981.
- [17] W. Saad, Zhu Han, A. Hjørungnes, D. Niyato, and E. Hossain. Coalition formation games for distributed cooperation among roadside units in vehicular networks. *Selected Areas in Communications, IEEE Journal on*, 29(1):48 –60, january 2011.
- [18] B. Shrestha, D. Niyato, Zhu Han, and E. Hossain. Wireless access in vehicular environments using bittorrent and bargaining. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1 –5, 30 2008-dec. 4 2008.
- [19] Sok-Lan Sou. A power-saving model for roadside unit deployment in vehicular networks. *Communications Letters, IEEE*, 14(7):623 –625, july 2010.
- [20] Elad Tomer, Leonid Safonov, and Shlomo Havlin. Presence of many stable nonhomogeneous states in an inertial car-following model. *Phys. Rev. Lett.*, 84:382–385, Jan 2000.
- [21] Pu Wang, Marta C. González, César A. Hidalgo, and Albert-László Barabási. Understanding the spreading patterns of mobile phone viruses. *Science*, 324(5930):1071–1076, 2009.
- [22] Yang Zhang, Jing Zhao, and Guohong Cao. On scheduling vehicle-roadside data access. In *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks, VANET '07*, pages 9–18, New York, NY, USA, 2007. ACM.