Optimal Roadside Units Placement in Urban Areas for Vehicular Networks

Baber Aslam, Faisal Amjad and Cliff C. Zou
University of Central Florida, Orlando, FL, USA
{ababer, czou}@eecs.ucf.edu, faisal@knights.ucf.edu

Abstract— **The most important component of a vehicular ad hoc network (VANET), besides VANET-enabled vehicles, is roadside units (RSUs). The effectiveness of a VANET largely depends on the density and location of these RSUs. During the initial stages of VANET, it will not be possible to deploy a large number of RSUs either due to the low market penetration of VANET-enabled vehicles or due to the deployment cost of RSUs. There is, therefore, a need to optimally place a limited number of RSUs in a given region in order to achieve maximum performance. In this paper, we present two different optimization methods for placement of a limited number of RSUs in an urban region: an analytical Binary Integer Programming (BIP) method and a novel Balloon Expansion Heuristic (BEH) method. BIP method utilizes branch and bound approach to find an optimal analytical solution whereas BEH method uses balloon expansion analogy to find an optimal or near optimal solution. Our evaluations show that both methods perform optimally or near optimally compared with the exhaustive method. Further, BEH method is more versatile and performs better than BIP method in terms of computational cost and scalability.**

Keywords-VANET; roadside unit; initial deployment stage; optimization; placement; urban areas

I. INTRODUCTION

A vehicular ad hoc network (VANET) relies on three types of communication for its setup and provision of services: vehicle to vehicle (V2V) communication, vehicle to infrastructure (V2I) communication and infrastructure to infrastructure (I2I) communication. All VANET applications depend on either one or more of these communication types. V2V communication depends on the number and location of vehicles, V2I communication depends on the number and location of roadside units (RSUs) and I2I communication depends on availability of interconnecting network between RSUs. During the initial deployment stages of VANET, there will be very small number of vehicles and RSUs due to the low market penetration of VANET-enabled vehicles or due to the deployment cost of RSUs. Given a small number of RSUs, there is, therefore a need to optimally place these RSUs in a given region/scenario in order to achieve maximum performance.

Information flow in most VANET applications is either from vehicles to infrastructure or from infrastructure to vehicles. Our focus, in this paper, is on applications that depend on information flow from vehicles to infrastructure (or RSUs), such as collection of information from vehicles about traffic/road conditions, traffic accidents, etc.

We present two different solutions to the RSUs placement problem with objective of maximizing the information flow from vehicles to RSUs in an urban environment: Binary Integer Programming (BIP) method and a novel Balloon Expansion Heuristic (BEH) method. BIP method utilizes branch and bound method to find optimal solution, whereas, BEH method uses balloon expansion analogy to find optimal solution. We have incorporated the vehicle density, vehicle speed, and the occurrence likelihood of an incident/event in our optimization schemes. The optimization aims at minimizing the reporting time for a given number of RSUs; reporting time is defined as the time duration from occurrence of an event till it is reported by a vehicle to an RSU. The RSUs are assumed to be interconnected. Our proposed optimization schemes can easily be extended to applications that depend on information flow from infrastructure to vehicles where the optimization goal can be area covered within some reporting time bounds.

Our contributions in this paper include: 1) study of optimization problem in context of VANET applications that depend on flow of information from vehicles to roadside units in an urban environment, 2) modeling of optimization problem with the objective of minimizing average response time, 3) presentation of two optimization methods (BIP method and BEH method) and formalization of problem into these two optimization methods, and 4) analysis of the two presented optimization methods.

The rest of the paper is organized as follows. Section II discusses the optimization problem modeling, section III presents our proposed optimization schemes, section IV gives the results and discussion, section V discusses the related work and finally section VI presents conclusion and future work.

II. OPTIMIZATION PROBLEM MODELING

A. System Model

The scope of this paper is restricted to urban environment such as the one shown in Fig. 1. Fig. 1(a) shows a partial map of Miami, FL, USA. The map shows a grid of major roads (shown in yellow and red color) and a number of smaller/local streets. The major roads are shared by all users/buses for commuting whereas the smaller streets are used only by users who need to visit a particular home or business on that street. The traffic on smaller streets is therefore very small/negligible as compared to that on major roads and we can safely ignore these for our system model. Fig. 1(a) can be approximated to a grid network of roads as shown in Fig. 1(b) after removing the local/smaller streets.

Consider the road network (shown in Fig. 1(b)) as a graph with each intersection as a vertex and each road segment as an edge. $\boldsymbol{V}$ is set of all vertices, let $i \in \boldsymbol{V}$ *(or* $v_i \in$

V). ***E*** is set of all edges, let $j \in \boldsymbol{E}$. Each road segment is further divided into many smaller sub-segments (each of length Δl) ***M*** is the set of all such sub-segments in the complete road network, let $k \in \boldsymbol{M}$ (or $m_k \in \boldsymbol{M}$). For a sub-segment $k \in \boldsymbol{M}$, let d_k be the vehicle density (vehicles/Km), f_k be the event/incident frequency (number of events happened in a given time – frequency of events) and s_k be the vehicle speed (Km/hr). The densities and speed on all sub-segments $k \in \boldsymbol{E_j}$ cannot always be the same because of different surface conditions (bumpy, slippery, etc), different gradient (steep climb, uphill, downhill, etc), different geometry (curving, straight, etc) and proximity to road signals or stop signs etc. Simplified event/incident distributions were considered to ease evaluation. The event/incident distribution functions for the road network of Fig. 1(b) that will be evaluated in this paper are shown in Fig. 2. Fig. 2(a) shows a distribution where the likelihood of an event/incident changes from one road to another but is constant over one particular road. This can be the case when roads have different characteristics such as road widths, speed limits, vehicle densities, and neighborhoods. Fig. 2(b) shows a distribution where the likelihood of an event/incident, in addition to changing from one road to another road, also changes over every road. This corresponds to the more realistic scenario where events/incidents are more likely to happen around intersections than in the road sub-segments that are far away from intersections. The notations used are summarized in Table 1.

TABLE I. NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
$\boldsymbol{V}$	set of all vertices/intersections: $i \in \boldsymbol{V}$
$\boldsymbol{E}$	set of all edges/road-segments: $j \in \boldsymbol{E}$
$\boldsymbol{M}$	set of all sub-segments: $k \in \boldsymbol{M}$
d_k	vehicle density for sub-segment $k \in \boldsymbol{M}$
f_k	event/incident frequency for sub-segment $k \in \boldsymbol{M}$
s_k	vehicle speed for sub-segment $k \in \boldsymbol{M}$
λ_k	vehicles (per unit time) entering sub-segment $k \in \boldsymbol{M}$
$t_{(x,y)}$	reporting time for incident at y to an RSU at x
$T_{(x,y)}$	average reporting time for all the incidents along a path from y to an RSU at x
t_y	time for a vehicle to reach at y
t_{yx}	time for a vehicle to reach x from y
p_{yx}	probability that a vehicle at y will travel to x
D_z	fraction of vehicles travelling from y to x using route z
t_{yxz}	time for a vehicle to reach x from y using route z
$\boldsymbol{N}$	set of sub-segments forming a route from y to x: $n \in \boldsymbol{N}$
C	total number of sub-segments: $n(\boldsymbol{M})=C$
R	total number of intersections: $n(\boldsymbol{V})=R$
r	desired number of RSUs
τ	desired average reporting time
α	desired coverage factor: $m=\alpha C$ gives desired coverage in number of segements
N_k	number of incidents happening on sub-segment $k \in \boldsymbol{M}$
A_{ki}	reporting time for an incident at sub-segment $k \in \boldsymbol{M}$ to an RSU $i \in \boldsymbol{V}$
y_i	= 1 if RSU $i \in \boldsymbol{V}$ exists else y_i = 0
x_{ki}	=1 if sub-segment $k \in \boldsymbol{M}$ is covered by RSU $i \in \boldsymbol{V}$ else x_{ki} =0

If vehicles entering the region follow Poisson distribution then there will be $\lambda_k = s_k\, d_k$ vehicles entering the sub-segment $k \in \boldsymbol{M}$ (or $m_k \in \boldsymbol{M}$) per unit time. If y ($y \in \boldsymbol{M}$) is the location of an incident/event then a vehicle will reach the point of incident with an exponentially distributed time, with an average value of $1/\lambda_y$. If x ($x \in \boldsymbol{V}$) is the location of an RSU, then the reporting time, $t_{(x,y)}$ (time for a vehicle to report an incident happened at location y to an RSU at location x) will be the summation of the time for a vehicle to reach location y (t_y) and the time for the vehicle to reach x from y (t_{yx}), see (1) and Fig. 3.

$$t_{(x,y)} = t_y + t_{yx} = \frac{1}{(s_y d_y p_{yx})} + t_{yx} \qquad (1)$$

where p_{yx} is the probability that a vehicle at $y \in \boldsymbol{M}$ will travel to $x \in \boldsymbol{V}$.

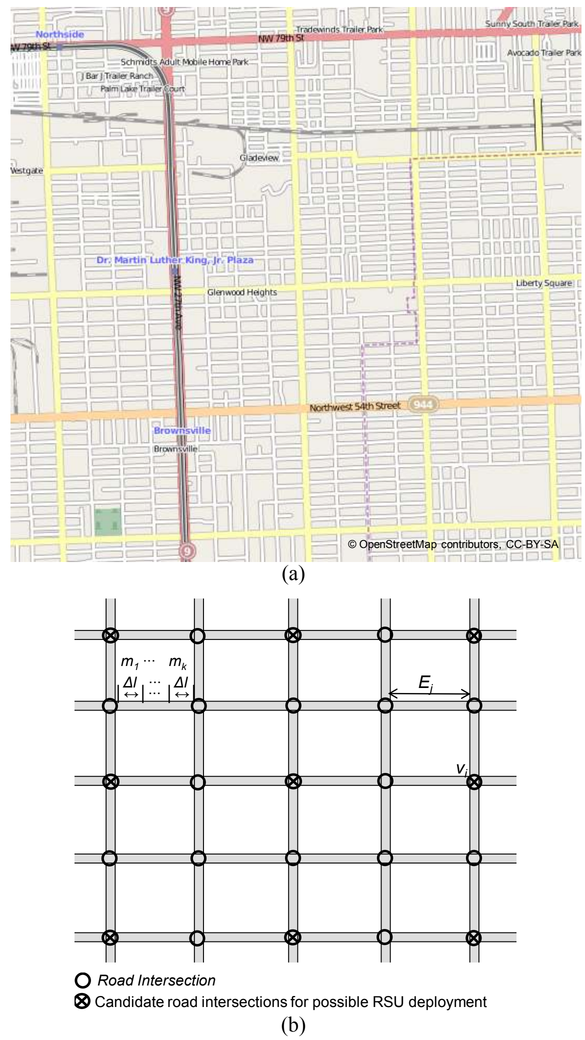
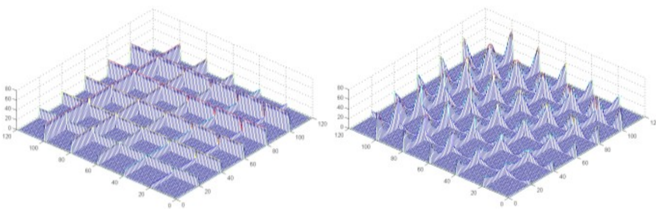


Figure 1. Urban environment. (a) Partial map of Miami, FL, USA. © OpenStreetMap contributors, CC-BY-SA (b) Grid-road network approximation of Fig. 1(a).

If there are more than one paths/routes from y to x, as will be the case in urban environment, then in (1) t_{yx} should represent the average time taken by any vehicle at $y \in M$ to travel to $x \in V$ along all the possible routes. Its value is given by (2).

$$t_{yx} = \sum_z t_{yxz} D_z \quad (2)$$

where z is the number of possible routes from $y \in \boldsymbol{M}$ to $x \in \boldsymbol{V}$, D_z is the fraction of vehicles travelling from $y \in \boldsymbol{M}$ to $x \in \boldsymbol{V}$ that use route z and t_{yxz} *is* the time for the vehicle to reach $x \in \boldsymbol{V}$ from $y \in \boldsymbol{M}$ using route z.



(a) (b)

Figure 2. Event/incident distribution functions: Relative frequency of events (*z axis*) at each segment (*x-y axes*). (a) Stair (b) Wave

Let $\boldsymbol{N}$, $n \in \boldsymbol{N}$, be the set of sub-segments forming a route from $y \in \boldsymbol{M}$ to $x \in \boldsymbol{V}$. The average reporting for any event/incident along this route is given by (3). If a route contains more than one sub-routes then we can use either the average travelling times (as given by (2)) or just the most direct/shortest route.

$$T_{(x,y)} = \frac{\sum_{n \in N} t_{(x,n)} f_n}{\sum_{n \in N} f_n} \quad (3)$$

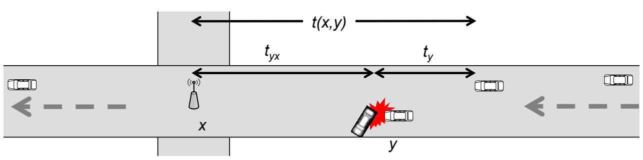


Figure 3. Reporting Time of an incident/event

B. Optimization Problem Modeling

Let C be the total number of sub-segments in a road network, i.e., $n(\boldsymbol{M})=C$ and R be the total number of intersections in a road network, i.e., $n(\boldsymbol{V})=R$ (we use notation $n(\boldsymbol{A})$ for number of elements in set $\boldsymbol{A}$). Each intersection is a candidate location for an RSU. If r is the desired number of RSUs and α is the desired fraction of coverage in the road network ($m=\alpha C$ is the number of covered sub-segments), then the optimization problem can be stated as: Minimize the average reporting time over each route (or an upper bound on the average reporting time over any route) such that at most r RSUs are placed among R candidate locations of set V and at least m out of C sub-segments of set M are covered by these RSUs.

C. Problem Complexity

One possible optimization option is to exhaustively check all possible combinations to find an optimal solution. The number of possible combinations for optimization problem is given by (4). The solutions that check all possible combinations to find an optimal solution increasingly become inefficient with the increase in size of area/region. For a 10Km x 10Km urban area with a grid-road topology, we may have a total road length of 100Km and 25 intersections. For a sub-segment size of 250m, we will have a total of 400 sub-segments. If we want to minimize the average reporting time for a total of r=5 RSUs and α=0.8 (80%) coverage, the number of possible combinations will be 1.15 x 10^{313}.

$$\binom{R}{r}\binom{C}{m} r! \left\{ {m \atop r} \right\} \quad (4)$$

Where, $\binom{C}{m}$ is *combinations* that gives the number of subsets with size m when picking from a larger set of size C, and $\left\{ {m \atop r} \right\}$ is the *Stirling numbers of the second kind* that gives the number of ways to partition a set of m elements into r nonempty (and non-distinct) subsets [11]. It is given by (5).

$$\left\{ {m \atop r} \right\} = \frac{1}{r!} \sum_{p=0}^{r} (-1)^p \binom{r}{p} (r-p)^m \quad (5)$$

III. Optimization Schemes

A. Binary Integer Programming (BIP) Optimization

The linear programming formalizations are not aware of the road topology so we need to relax the condition of the average reporting time that is defined over a single path/route to the average reporting time defined over entire region. The two performance metrics are not the same but the relaxation helps us to solve the optimization problem analytically using linear programming. It is important to note that averaging over entire region is more relaxed; it may include some routes whose average reporting time will be greater than the average reporting time over the entire region.

$\boldsymbol{M}$ is the set of all sub-segments in the road network, let ($k \in \boldsymbol{M}$); and $\boldsymbol{V}$ is the set of all intersections (candidate RSU locations), let ($i \in \boldsymbol{V}$). Let N_k be the number of incidents happening on any sub-segment $k \in \boldsymbol{M}$ and A_{ki} be the reporting time of an incident at sub-segment $k \in \boldsymbol{M}$ to an RSU $i \in \boldsymbol{V}$. Let y_i and x_{ki} be two binary decision variable; such that, y_i equals to 1 if RSU $i \in \boldsymbol{V}$ exists and 0 otherwise and x_{ki} equals to 1 if sub-segment $k \in \boldsymbol{M}$ is covered by RSU $i \in \boldsymbol{V}$ and 0 otherwise.

The optimization goal is to minimize the average reporting time for a given number of RSUs and area coverage. As discussed earlier, we have relaxed the minimization of the average reporting time over each route constraint and replaced it with the average reporting time over the entire region. Specifically, here we minimize the total reporting time over the entire region. The binary integer programming formalization of this optimization problem is given in Fig. 4.

Constraint (1) requires that the number of RSUs should be less than or equal to the desired value (r). Constraint (2) requires that each sub-segment is assigned to one or more than one RSUs, this ensures 100% coverage. Constraint (3) ensures that sub-segments are assigned to only those RSUs that are included in the solution. Constraints (2b and 6) replace constraint (2) if the required coverage is less than 100% coverage but equal to or greater than some coverage threshold (given by αC).

Binary Integer Programming implements a branch and bound algorithm to solve the problems; the *branch and bound* algorithm uses binary search tree whose size grows tremendously with the size increase of a problem. In worst cases, the branch and bound algorithm searches all possible combinations to find the best solution [13] and we have already shown in Eq. (4) that the number of possible combinations (defined by C, R, m, and r) for this problem is very large. Further, the memory requirements of *BIP* are proportional to C^2R.

$$f(r) = min \quad \sum_{i\in V}\sum_{k\in M} N_k A_{ki} x_{ki}$$

$s.t.$

- $For \quad \alpha = 1 \qquad (100\% \; coverage)$

$$(1) \quad \sum_{i\in V} y_i \le r$$

$$(2) \quad \sum_{i\in V} x_{ki} \ge 1 \qquad \forall\, k, k \in M$$

$$(3) \quad \sum_{k\in M} x_{ki} \le C y_i \quad \forall\, i, i \in V$$

$$(4) \quad y_i \in \{0,1\} \qquad i \in V$$

$$(5) \quad x_{ki} \in \{0,1\} \qquad k \in M, i \in V$$

- $For \; \alpha < 1 \qquad (100\alpha \; percent \; coverage)$

$$(2b) \quad \sum_{i\in V} x_{ki} \le 1 \qquad \forall\, k, k \in M$$

$$(6) \quad \sum_{i\in V}\sum_{k\in M} x_{ki} \;\ge\; \alpha C$$

Figure 4. Formulization BIP: Minimizing total reporting time $f(r)$ for given r number of RSUs and α area coverage. For 100% coverage, the constraints are (1) to (5); for 100α percentage of coverage, the constraints are (1), (2b), (3), (4), (5) and (6).

B. Balloon Expansion Heuristic (BEH) Optimization

In this optimization, each RSU and its coverage area is considered as a balloon that dynamically expands in a 2 dimensional space. A balloon's boundary represents the area covered by an RSU within a given average reporting time. The balloons are dynamically expanded as we gradually relax the average reporting time constraint till the desired percentage/fraction of area is covered by them.

The roads inside a balloon's boundary at any time include all the segments that can be covered by the RSU within some average reporting time via some route/path. The balloon expansion follows road network and the expansion is independent on each side, that is, if the RSU is located at an intersection then the balloon boundary on each of the four sides will expand independent of other three sides. The expansion depends on vehicle speed, vehicle density, event/incident distribution and probability of vehicles following a route. The segments, along a route, with high frequency of events/incidents will have more impact on computing the average reporting time than those with low frequency of events.

Fig. 5 shows a road intersection where an RSU is located; A, B, C, and D is the balloon boundary for some average reporting time (τ). Initially, $|\boldsymbol{XA}|=|\boldsymbol{XB}|=|\boldsymbol{XC}|=|\boldsymbol{XD}|=0$, i.e., points {A, B, C, D} are located at X and $T_{(x,a)}=T_{(x,b)}=T_{(x,c)}=T_{(x,d)}=0$, where $T_{(x,y)}$ is the average reporting time along path $\boldsymbol{XY}$ including point Y. The balloon is then expanded independently on all four routes for some average reporting time (τ). The size of expansion on each route will vary depending on vehicle speed, vehicle density, incident/event distribution and probability of vehicles following a route. Fig. 5 shows a balloon expansion where $|\boldsymbol{XA}|=|\boldsymbol{XD}|\le|\boldsymbol{XC}|\le|\boldsymbol{XB}|$.

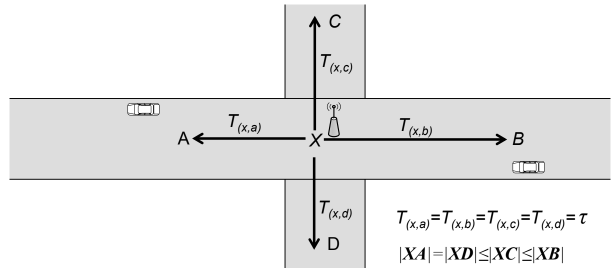
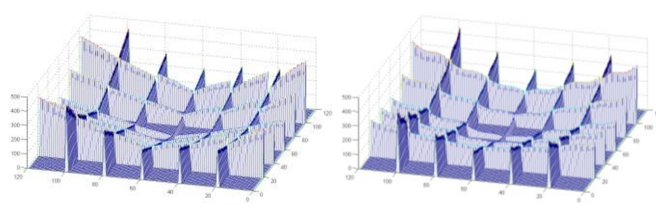


Figure 5. Balloon expansion: The expansion is independent along each direction and depends on vehicle speed, vehicle density, event/incident distribution and probability of vehicles following a route. $|XA|$, $|XD|$, $|XC|$, and $|XB|$ gives the size of expansion towards A, B, C and D respectively for τ average reporting time over each route.

Unrestricted expansions may form loops especially in urban environment. In order to avoid loops, we assume that the boundary expansion of an RSU is towards the direction away from the RSU; if the expansion encounters an intersection then it only continues in directions that are away from the RSU. Fig. 6 gives the average reporting times for an RSU located at the center of an urban environment (Fig. 1(b)) for event/incident distributions given at Fig. 2.



(a) (b)

Figure 6. Average reporting times, for different event/incident distributions, of urban environment given at Fig. 1(b). (a) Stair (b) Wave

BEH optimization method, in general, starts with placing an RSU at each candidate location. The coverage of each RSU is then expanded on each side (along each route) for some value of average response time. The expansion continues till a sufficient number of sub-segments are covered by more than one neighboring RSUs (or the average reporting time threshold has reached). At this moment, the RSU with the least "impact factor" is removed (similar to the bursting of a balloon due to the too tight compression from neighboring balloons). The process repeats until the optimization objective is achieved. The *impact factor* of an RSU is the number of sub-segments that will not be covered if the RSU is removed; it is computed by subtracting the number of overlapped-sub-segments (sub-segments that are covered by this RSU and also by some other RSUs) from the number of sub-segments covered by this RSU.

The BEH algorithm for this optimization problem is given in Fig. 7. The optimization objective is to minimize the average reporting time over each route (or the upper bound on average reporting time over any route) for given number of RSUs (r) and area coverage (αC). The method starts with placing an RSU at each candidate location (line 2), the average reporting time (of each route) is then iteratively incremented by a small value (line 5-8) until area coverage constraint is met (line 9-10). The impact factor of each RSU is calculated (line 11-13) and the one with the least impact factor (line 14-15) is removed provided the removal does not affect area coverage constraint (line 16-19); otherwise the average reporting time (of each route) will be further incremented (line 5-8). The process continues until the number of RSU constraint is met (line 4).

1. **begin**
2. $\boldsymbol{B} \leftarrow \boldsymbol{V}$
3. $\tau' \leftarrow 0$
4. **while** $n(\boldsymbol{B}) > r$ **do**
5. Increment $\tau'$by a small value
6. **for** $each\ b \in \boldsymbol{B}$ **do**
7. Expand area coverage of b to $y\ s.t.\ T_{(b,y)} = \tau'$
8. **end**
9. Compute Q:
   combined area coverage by all RSUs in set $\boldsymbol{B}$
10. **if** $Q \geq \alpha C$ **then**
11. **for** $each\ b \in \boldsymbol{B}$ **do**
12. Compute Impact Factor $IF(b)$
13. **end**
14. Find $w \in \boldsymbol{B}\ \ s.t. min(IF) = IF(w)$
15. $\boldsymbol{B}' \leftarrow \{w\}$
16. Compute Q:
    combined area coverage by all RSUs in
    set $\{\boldsymbol{B} - \boldsymbol{B}'\}$
17. **if** $Q \geq \alpha C$ **then**
18. $\boldsymbol{B} \leftarrow \boldsymbol{B} - \boldsymbol{B}'$
19. **end**
20. **end**
21. **end**
22. **Return** $(\boldsymbol{B})$
23. **end**

Figure 7. Algorithm BEH: Minimizing average reporting time over each route (i.e., the upper bound on average reporting time over any route) for given number of RSUs and area coverage. After the algorithm finishes, τ' gives the upper bound on the average reporting time over any route.

The number of iterations in *BEH* optimization method depends on the increment size of the average-response-time in each iteration, not on the problem size (defined by *C, R, m,* and *r*). Therefore, BEH is scalable and suitable for solving large-size problems. The memory requirements of *BIP* are proportional to *CR.*

IV. Simulation Results and Discussion

A. Simulation Setup

The simulation is based on an urban region with five vertical and five horizontal roads, as shown in Fig. 1(b). The problem size (as earlier discussed in section II.D) is defined by *C, R, m,* and *r*. The urban topology (grid/Manhattan topology) selected for simulation, results in a higher number of intersections (hence large *R*) and a larger total road length (hence large *C*) in a given area as compared to other topologies. A solution that produces desired results for this topology can be assumed to work well with other topologies.

The region is 3 Km x 3 Km, with a total road length of 30 Km. The sub-segment size is 250m, resulting in a total of 120 sub-segments. There are a total of 25 intersections; in order to reduce problem complexity for BIP methods (explained earlier in section II.D), only 9 out of the 25 intersections are considered as candidate locations for RSUs (refer Fig. 1(b)). The transmission range of both the RSU and vehicles was taken as 250m. Two different incident/event distributions are defined, as shown in Fig. 2. Fig. 2(a) shows a distribution where the likelihood of an event/incident changes from one road to another but is constant over one particular road. This can be the case when roads have different characteristics such as road widths, speed limits, vehicle densities, and neighborhoods. Fig. 2(b) shows a distribution where the likelihood of an event/incident, in addition to changing from one road to another road, also changes over every road. This corresponds to the more realistic scenario where events/incidents are more likely to happen around intersections than in the road sub-segments that are far away from intersections. Vehicles entering the region follow Poisson distribution, with $\lambda = SD$ vehicles entering any sub-segment per unit time. A constant vehicle density of D = 4 vehicle/Km and a constant speed of S = 50 Km/hr is assumed for this simulation. The probability that a vehicle at a point of event/incident will travel to a particular RSU is considered to be inversely proportional to the number of intersections (or routes) between the vehicle and the RSU. The most direct and shortest path is used to calculate the reporting time of an event/incident to a particular RSU; only the vehicles following that route are considered in computing the average reporting time and the contribution by the rest of vehicles for reporting the event/incident is ignored (which, if considered, may improve the event/incident reporting likelihood). It is important to note that in real scenarios/applications vehicle traces can be used to generate all these statistics; the statistics based on vehicle traces are generally reliable as daily traffic patterns are often repeated.

In order to study how well our proposed optimization methods could achieve, enumeration method was used to exhaustively search for the true optimal solution. As discussed earlier (in section II.D), enumeration method increasingly becomes inefficient with the increase in size of area/region. In order to reduce the number of combinations to be checked to find an optimal solution, so that we can actually finish the enumerating operation using personal computers, the average reporting time over each route constraint was relaxed and was replaced with the average reporting time over the entire region. With this relaxation, we can simply consider a segment to be covered by one RSU (out of all the currently considered RSUs) that has minimum reporting time from that segment instead of considering all of RSUs.

More than one thousand simulation runs were carried out and the reported results are based on average of these simulation runs. The specifications of system used for simulation are: *Processor* - Intel® Core^{TM} 2 Quad CPU Q6700 @ 2.66 GHz, *RAM* - 4 GB, *Hard Disk* - 232 GB (200 GB free), and *OS* - Windows 7 Enterprise 64-bit.

B. Results

1) Enumeration/Exhaustive Search: The minimum average reporting time (over the entire region) for different number of RSUs, using enumeration/exhaustive search for different event/incident distributions of urban environment given at Fig. 1(b), are shown in Fig. 8.

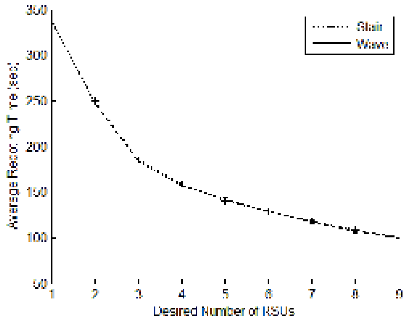
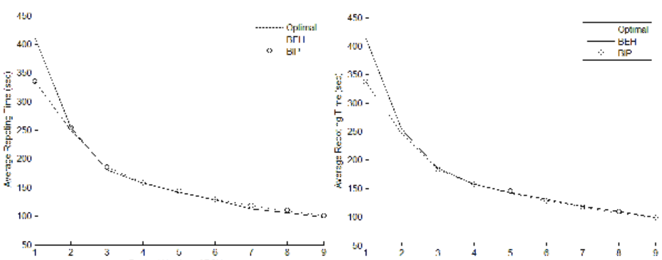


Figure 8. Minimum average reporting time (over the entire region) for different number of RSUs using enumeration/ exhaustive search for different event/incident distributions of urban environment given at Fig. 1(b).

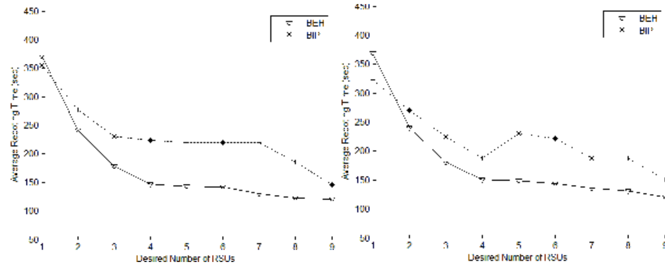
2) Binary Integer Programming (BIP) Optimization: The minimum average reporting time (over the entire region) for different number of RSUs, using BIP for different event/incident distributions of urban environment given at Fig. 1, is shown in Fig. 9. The minimum average reporting time over the entire region of BIP is the same as that of *enumeration/exhaustive* search solution.



(a) (b)

Figure 9. Minimum average reporting time (over the entire region) for different number of RSUs and different event/incident distributions of urban environment given at Fig. 1(b). (a) Stair (b) Wave

The minimum average reporting time over each route (or an upper bound on the average reporting time over any route) for different number of RSUs, corresponding to optimal solutions of *BIP* for different event/incident distributions of urban environment given at Fig. 1, are shown in Fig. 10. The minimum average reporting time over each path of *BIP* is higher than that of *BEH*. The execution times for *BIP* is given in Fig. 11.



(a) (b)

Figure 10. Minimum average reporting time over each route (or an upper bound on the average reporting time over any route) for different event/incident distributions of urban environment given at Fig. 1(b). (a) Stair (b) Wave

3) Balloon Expansion Heuristic (BEH) Optimization: The minimum average reporting time over each route (or an upper bound on the average reporting time over any route) for different number of RSUs, using BEH for different event/incident distributions of urban environment given at Fig. 1, are shown in Fig. 10. The minimum average reporting time over each path achieved by BEH is better than that of BIP.

The minimum average reporting time (over the entire region) for different number of RSUs, corresponding to optimal solutions of *BEH* for different event/incident distributions of urban environment given at Fig. 1, are shown in Fig. 9. The minimum average reporting time over the entire region achieved by *BEH* closely follows that of enumeration/exhaustive search. The execution times for the *BEH* algorithm is given in Fig. 11.

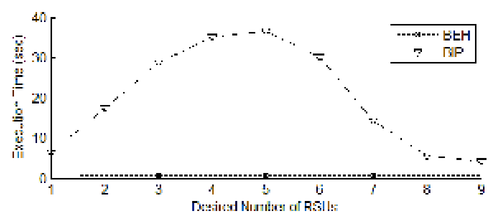


Figure 11. Execution times for BEH and BIP for different number of desired RSUs

C. Discussion

BIP successfully produced optimal solutions. The minimum average reporting time over the entire region is the same as that of enumeration/exhaustive search (Fig. 9). However, the minimum average reporting time over each path is higher than that of BEH (Fig. 10).

BEH incorporate the knowledge of road topology to find the optimal solution. The *BEH* successfully produced optimal solution. The execution times for the *BEH* method are much less than that of *BIP* method (refer Fig. 11).

In addition, the minimum average reporting time over each path achieved by BEH is better than that of BIP (Fig. 10). The minimum average reporting time over the entire region achieved by BEH closely follows that of enumeration/exhaustive search (Fig. 9).

BEH use average reporting time over a path as a timing constraint. Average reporting time over a path is more useful metric then average reporting time over entire region; it guarantees that on average an event/incident will be reported within the timing constraint whereas the average reporting time over entire region cannot guarantee this.

V. Related Work

Earlier works in optimal placement in VANET include [1-9]. Lee et al. [1] seek optimal placement of RSUs to improve connectivity. Each intersection is considered as a potential RSU location. These potential locations are then ordered based on number of vehicle-reports received within communication range of each RSU. The placement scheme only considers taxi location reports and does not consider speed or density of all vehicles.

Li et al. [2] consider the optimal placement of gateways, which connect RSUs (access points - AP) to the Internet, while minimizing the average number of hops from APs to gateways. They consider pervasive APs such that every vehicle is connected to an AP. They do not consider vehicle speed, density or movement patterns.

Zhao et al. [3] optimize placement of Thowboxes, standalone units that act as relays, to improve contact and data-rate/throughput within context of a delay tolerant network. They aim at improving V2V communication and not the V2I communication. Lochert et al. [4] use genetic algorithm for optimal placement of RSUs for a VANET traffic information system. The optimal placement is to minimize travel for some fixed landmarks and may not be useful for travel between any two points in an area.

Sun et al. [5] optimize the location of RSUs such that vehicle can reach an RSU within some timing constraint, given by sum of driving time and an overhead time (for adjusting the route), to update short term certificates. The optimization scheme may require vehicles to change their route which may have effects on local traffic condition. We do not have any route changing condition; we optimally place the RSUs considering the vehicles current routes only

Fiore et al. [6] optimally place RSUs (Access Points - AP) in an urban environment to improve cooperative download of data among vehicles. They aim at placing the APs at point where maximum vehicles cross each other, this helps in relaying the data from AP to a downloading vehicle via other vehicles. Trullols et al. [7] optimally deploy RSUs (Dissemination Points – DPs) in an urban area to maximize the number of vehicles that contact the DPs. Malandrino et al. [8] optimally deploy the RSUs (APs) to maximize the system throughput. They consider both the V2I (or I2V) and V2V communications for optimal placement of APs. Vehicle trajectory information (time and location) forms basis of this optimization which may not be available in many cases. Zheng et al. [9] optimally deploy APs to improve contact opportunity; defined in terms of time for which a user remains in contact with an AP. These optimizations aims at transfer of data from RSUs to vehicles whereas, our optimization aims at transfer of data from vehicles to RSUs with an area coverage constraint. Also, we do not consider V2V communication in our optimization problem.

Our work is also related to the problem of facility location, where one or more facilities are optimally located in a region to reduce the overall costs (to consumer and facility) [10, 11]. We do not aim at minimizing the overall costs (reporting time of events) rather we aim at minimizing the average reporting time on each path/route basis; this need awareness to road topology. Further, we also incorporate vehicle speed, vehicle density, probability of a vehicle to follow a particular route and event distribution.

VI. Conclusion and Future work

We have presented two optimization methods: Binary Integer Programming (BIP) method and Balloon Expansion Heuristic (BEH) method. Both optimization methods were used to solve the optimization problem of minimizing the average reporting time. We have shown that the novel BEH method is more versatile and can be used to solve the optimization problem without any further relaxations. In future work, we intend to use more complex road topology with statistics generated from realistic traffic traces to further ascertain the effectiveness of our proposed optimization methods.

References

[1] J. Lee and C. Kim, "A roadside unit placement scheme for Vehicular Telematics networks", in AST'10.

[2] P. Li, X. Huang, Y. Fang and P. Lin, " Optimal placement of gateways in Vehicular Networks", in IEEE Transactions on Vehicular Technology 2007, Vol. 56/ 6, pt 1, pp. 3421-3430, 2007.

[3] W. Zhao, Y. Chen, M. Ammar, M. Corner, B. Levine and E. Zegura, "Capacity enhancement using Throwboxes in DTNs". In MASS'06.

[4] C. Lochert, B. Scheuermann, C. Wewetzer, A. Luebke and M. Mauve, "Data aggregation and roadside unit placement for a VANET traffic information system", In VANET'08

[5] Y. Sun, X. Lin, R. Lu, X. Shen, J. Su, "Roadside Units Deployment for Efficient Short-time Certificate Updating in VANETs", In IEEE ICC'10, 2010.

[6] M. Fiore, J. Barcelo-Ordinas,"Cooperative download in urban vehicular networks", in MASS '09.

[7] O. Trullols, M. Fiore, C. Casetti, C.F. Chiasserini, J.M. Barcelo Ordinas, "Planning roadside infrastructure for information dissemination in intelligent transportation systems", in Computer Communications, Vol. 33/ 4, pp 432-442, March 2010.

[8] F. Malandrino, C. Casetti, C. Chiasserini, M. Fiore , "Content downloading in vehicular networks: What really matters," in INFOCOM'11.

[9] Z. Zheng, Z. Lu, P. Sinha, S. Kumar, "Maximizing the Contact Opportunity for Vehicular Internet Access," in INFOCOM'10.

[10] M. L. Brandeau and S. S. Chiu, "An overview of representative problems in location research", Management Science, 35(6):645–674, 1989.

[11] R. Francis and J. White, "Facility Layout and Location: An Analytical Approach", Englewood Cliffs, NJ: Prentice-Hall, 1974.

[12] M. Abramowitz, and I.A. Stegun, (Eds.). "Stirling Numbers of the Second Kind." §24.1.4 in Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing. New York: Dover, pp. 824-825, 1972.

[13] J. Clausen, "Branch and Bound Algorithms—Principles and Examples," Dept. Comput. Sci., Univ. Copenhagen, [Online]. Available: http://www.diku.dk/OLD/undervisning/ 2003e/datV-optimer/ JensClausenNoter.pdf.