Abstract—Vehicular ad-hoc networks (VANETs) has recently attracted a lot of attention due to their great potentials for different applications such as collision avoidance, route finding and autonomous driving. A wide range of coverage and accessibility to the end users in VANETs make them a good target for commercial advertising. This paper addresses the problem of mobile advertising in VANETs. We consider a case where different advertisers compete for the VANET infrastructure. It is assumed that a city is partitioned into a grid of blocks and the central data center manager (CDM) sets the rental price for each block considering the geographical position and the predicted density of vehicles inside the block. The regret-based minimization method is adopted to tackle the problem in response to its dynamic nature. Regret bound of the proposed algorithm and its convergence to the best strategy are shown rigorously. Furthermore, a good potential of the proposed algorithm is revealed through simulations.

Index Terms—Vehicular ad-hoc networks, Mobile advertising, Regret-based minimization method.

I. INTRODUCTION

Vehicular networks have recently emerged as a new area for research and development. These networks have mainly been considered for prevention of possible collisions and collection of traffic information. One of the latest applications of these networks is mobile advertising, the goal of which is to spread the advertisement among vehicles available in a certain area.

There are three modes of communications in a VANET: vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and infrastructure-to-vehicle (I2V) communications, where the infrastructure mainly refers to roadside units (RSUs). In a mobile advertisement scheme, a vehicle can receive the advertisement from the infrastructure or other vehicles which have already received the advertisement.

A. Related Work

The body of research in VANETs is mostly devoted to designing efficient routing protocols [1]. Among few works on mobile advertising in VANETs, the pioneering work [2] is probably the most relevant. The goal in that study is to find the best set of initial seeds among vehicles to maximize the spreading of an advertisement. The authors consider the case where the seeds have to be chosen from public transportation vehicles. They conducted extensive measurement studies based on centrality characteristics of these vehicles in the cities of Shanghai and Shenzhen of China to obtain the temporal correlations for social centralities of vehicles. Based on these observations, an algorithm is designed which can find an initial set of seeds to achieve the goal. However, a more general scenario in which all the vehicles can be the initial seeds remains unaddressed. Also, since the proposed approach is measurement based, its applicability depends on the availability of vehicle measurement data. There is a similar problem in social networks called influence maximization problem [3]–[6]. In these studies, the main focus is on finding the initial set of highly influential nodes which can lead to maximizing the spreading of influence in the network. However, the network is considered to be static. This assumption makes the implementation of the proposed algorithms unrealistic in a dynamic network like VANET. To the best of our knowledge, the problem of VANET advertisement for a general dynamic network has not been studied to date.

B. Contributions

In this work, we introduce a new scheme for mobile advertising in VANETs which no longer involves the influence maximization problem in a direct manner. In the proposed scheme, the given city map is partitioned into small blocks, where the infrastructure in each block can be rented by advertising companies. Considering the positions of blocks in a city, heterogeneous rental prices for blocks are presumed. This scheme aims to find the optimal strategy, which is the best set of blocks to rent, for an advertising company. We adopt the regret-based minimization method mainly applied for the bandit problem in literature [7]–[10], and adapt it for the mobile advertising problem in hand. We derive the bound of regret for the proposed algorithm along with the proof of convergence to the best strategy. Simulation results are provided to corroborate the superiority of the proposed algorithm. To the best of our knowledge, this is one of the first works in this area which proposes a real-time algorithm adaptable to an arbitrary city map while providing analytical support.

Structure of the paper: The system model and the mobile advertising problem in VANETs are introduced in Section II.
The proposed regret-based minimization algorithm and proof of convergence to the best strategy are given in Section III. Simulation results are presented in Section IV. Finally, Section V concludes the paper.

II. System model

As mentioned above, in a VANET there are three possible ways of communications: V2I, I2V, and V2V. The infrastructure contains RSUs where each of them has a limited broadcasting range. All of the distributed RSUs are connected to a central data center where a central data manager (CDM) can control data dissemination among them. A schematic of the proposed model is depicted in Figure 1. For the advertisement spreading scenario, an advertisement can be spread from the infrastructure to vehicles. Moreover, vehicles who have already received the advertisement can spread it to other vehicles.

![Figure 1: Schematic of a city which is divided into 3x3 blocks with multiple RSUs and a CDM.](image)

In our study it is assumed that the city area is partitioned into \( N \) small blocks. Without loss of generality the same size is assumed for all the blocks\(^1\). Note that there may be more than one RSU in each block, and the number of RSUs can vary from one block to another. Also, the number of available vehicles is different in different blocks. Moreover, for any arbitrary block, the average density of vehicles changes over time. For instance, a block could be crowded in the morning and afternoon; however, it becomes uncrowded in the evening.

In the mobile advertising scheme in VANETs, as a block becomes crowded, more vehicles will receive the advertisement and the advertisement will be disseminated among a large number of vehicles with a high probability. However, when a block is uncrowded, the chance for efficient broadcast of advertisement is much lower. Here, we consider a scenario where the CDM is in charge of leasing multiple available RSUs to the advertisers. More precisely, the CDM leases the blocks to advertising companies to let them disseminate their advertisements using the RSUs located inside the blocks. Also, the CDM sets a real-time rental price for each block based on the information about the vehicle density inside the block and the geographical location of the block. For instance, the rental cost for a block located in the downtown of a city should be higher than for a block located in the suburbs. Also, the price should increase corresponding to the increase in the vehicle density.

We partition the time into small time-batches and consider \( T \) time-batches in our observation window. At the beginning of the \( t^{th} \) time-batch, the CDM gathers information and predicts a lower bound on the average density of vehicles inside the \( k^{th} \) block \( \hat{\nu}_k(t) \). Subsequently, CDM sets the constant price \( PR_k(t) \) for this block throughout the time-batch. This pricing strategy implies that the advertising companies (parties) pay a flat price for renting a block in a time-batch. Also, \( PR_k(t) \) can be expressed as:

\[
PR_k(t) = y_k \exp(\hat{\nu}_k(t)) \quad 1 \leq k \leq N, \tag{1}
\]

where \( y_k \) is a constant that reflects the geographical importance of the block.

Assume that there are \( U \) parties willing to disseminate their advertisements through the network. Moreover, each party \( u \) is associated with a limited budget for each block \( k \), \( b_{u,k} \), \( 1 \leq u \leq U, \ 1 \leq k \leq N \). At the beginning of each time-batch \( t \), each party \( u \) chooses a strategy \( A_u(t) \), a subset of the set \( \{1, \ldots, N\} \), which determines the blocks the party will rent from the CDM. For the \( t^{th} \) time-batch the price this party has to pay can be expressed as: \( \sum_{j \in A_u(t)} PR_j(t) \). It is noteworthy that each of the parties can rent multiple blocks in a time-batch.

Suppose there is a feedback channel from the CDM to the parties. Through this channel, the CDM disseminates \( \hat{\nu}_k(t) \), \( 1 \leq k \leq N \), the exact density of vehicles inside the block at the end of each time-batch. Assuming the number of vehicles in each block is upper bounded by \( M \) for every time-batch, the (normalized) utility for the \( u^{th} \) advertiser at the \( t^{th} \) time-batch can be expressed as:

\[
H_u(t)|_{A_u(t)} = \frac{\sum_{k \in A_u(t)} \frac{b_{u,k} - y_k}{\ln(b_{u,k} / y_k)} \hat{\nu}_k(t) - y_k \exp(\hat{\nu}_k(t)) + y_k}{N \left( \frac{B - Y}{\ln(B)} M + Y \right)}, \tag{2}
\]

where

\[
(B, Y) = \arg \max_{(b_{u,k} - y_k)} \left( \frac{b_{u,k} - y_k}{\ln(b_{u,k} / y_k)} M + y_k \right). \tag{3}
\]

In Eq. (3), the payment term does not appear since the utility would reach its global maximum when there is no payment \( \hat{\nu}_k(t) = 0 \). Also, the proposed utility function in Eq. (2) is a concave function of the density of vehicles. This function takes the value of zero when two cases occur:

1) The party has to pay all of his money and the predicted lower bound of the density of vehicles and the exact density

\[^1\]This assumption does not affect the generality of the problem since heterogeneous prices are considered for blocks.
of vehicles are the same. In this case:
\[
v_k(t) = v_k(t), y_k \exp(v_k(t)) = b_{u,k}
\]
\[
\Rightarrow v_k(t) = \ln \left( \frac{b_{u,k}}{y_k} \right)
\]
\[
\Rightarrow b_{u,k} - y_k v_k(t) - y_k \exp(v_k(t)) + y_k = 0. \tag{4}
\]

2) The predicted lower bound of the density of vehicles and the exact density of vehicles are both zero \((v_k(t) = \hat{v}_k(t) = 0)\).

Let us denote the sequence of strategies for the \(i^{th}\) party by \(A_u(t)|_{t=1}^T\). Note that the total utility can be expressed as: \(H_u = \sum_{t=1}^T H_u(t)|_{A_u(t)}\). The optimal solution for the problem in hand can be obtained as:
\[
A_u^*(t)|_{t=1}^T = \arg \max_{A_u(t)|_{t=1}^T} H_u. \tag{5}
\]

In this problem, we face a stochastic environment. Therefore, the sequence of costs for each block is also stochastic and the statistic is unknown to the parties. Hence, finding the best strategy is mathematically intractable. In this work, the regret-based algorithm is used to solve the above optimization problem. This algorithm provides a good match with the inherent dynamism of the model.

III. REGRET-BASED MINIMIZATION

A. Basic idea

Regret-based minimization was originally adopted in the bandit problem, where a gambler must decide which arm of \(K\) non-identical slot machines to play in a sequence of trials in order to maximize his reward [7]–[10]. This method is one of the online learning algorithms which tries to determine the best hand in the current game by observing the history of the game. More precisely, the algorithm tries to determine the best strategy using the history of rewards for strategies. In this section, we explain the basic idea of the regret-based minimization technique and adapt it for the problem in hand. This method is also used in other applications such as finding malicious users in cognitive radio networks [11].

In our mobile advertising scheme, the regret for the \(i^{th}\) party is defined as the difference between the total utility of the best strategy \(H_u^*\) and the total utility of the utilized strategy \(H_u\):
\[
R_u = H_u^* - H_u \geq 0. \tag{6}
\]

The aim is to design an algorithm in which a player’s regret goes to zero sub-linearly with respect to the time. This behavior implies that the strategy of the player converges to the best strategy after a certain period of time.

B. Application in mobile advertising in VANETs

Here, we focus on one of the parties in the considered problem. The set of feasible strategies \(A\), where \(|A| = A\), for this party contains strategies where the party’s utility is positive. Each strategy in \(A\) is associated with a probability of selection based on its utility, where the weights are derived based on the history of observations. In this manner, strategies which have led to more utility are associated with larger weights, and thus higher probability of selection. At the beginning of each time-batch, the party chooses a strategy among a set of mixed strategies. Subsequently, the party measures the utility of the chosen strategy at the end of the time-batch and updates the weights of strategies accordingly. This procedure leads to tracking the smooth changes in the network and adapting the chosen strategy to the current state of the network.

Here, it is assumed that the network changes slowly over time. That means the density of vehicles changes slowly in each block which is an appropriate assumption in VANETs. Therefore, a good strategy for the current time-batch would be a good strategy for the next time-batch with a high probability. The detail of the proposed method is presented in the following.

**Mixed strategy updating rule:** At the \(t^{th}\) time-batch, the probability of selection for the strategy \(a \in A\) is dependent on two main factors: (1) the history of observations, and (2) the probability of selection of strategies which are similar to \(a\). The latter dependency comes from the fact that if multiple strategies target similar blocks, they would have similar utility values. The utilized method should assign high probability of selections to the high weighted strategies along with low probability of selections to the low weighted strategies. Also, the method should consider a minimum probability of selection for all strategies to decrease the probability of trapping into local maxima. Considering all the above, the probability of selection \(p_a(t)\) for a strategy \(a \in A\) for the \(t^{th}\) time-batch is determined by the following rule\(^2\):
\[
p_a(t) = (1 - \gamma - \eta) \frac{w_a(t)}{w(t)} + \gamma + \eta \frac{w_{N_r(a)}(t)}{\sum_{b \in A} w_{N_r(b)}(t)} \tag{7}
\]
where \(\eta, \gamma \in (0, 1]\) are constant parameters chosen such that \(\eta + \gamma \leq 1\), \(w_a(t)\) is the weight of the strategy \(a\) in the \(t^{th}\) time-batch, \(w(t) = \sum_{b \in A} w_b(t)\), and \(w_{N_r(a)}(t) = \sum_{b \in N_r(a)} w_b(t)\). Here, \(N_r(a)\) stands for the strategies which are similar to the strategy \(a\) (neighbor strategies). Mathematically, we define this parameter as follows:
\[
N_r(a) = \{b | |\delta(a, b)| \leq \psi, b \in A\}, \tag{8}
\]
where \(\psi\) is a constant which measures the similarity and \(\delta(a, b) = a \triangle b = a \cup b - a \cap b\).

**Updating rule for strategies weights:** As can be seen from Eq. (7), the weights for strategies should be calculated at the beginning of each time-batch. We assume the initial weights are set as \(w_i(1) = 1, \forall i \in A\).

In our study, the weights of strategies for the next time-batch \(w_j(t + 1)\), \(\forall j \in A\) are updated as follows:
\[
w_j(t + 1) = w_j(t) \exp \left( \gamma Q'_j(t)/A \right), \tag{9}
\]
\(^2\)We use \(a\) instead of \(A(t)\) for simplicity.
where $Q'_j(t)$ is the virtual strategy reward and it is the ratio between the utility of the corresponding strategy and the probability of choosing the strategies which target at least a common block to the strategy of interest. Mathematically:

$$Q'_j(t) = \frac{H(t)|j}{\sum_i\sum_{s,t} p_s(t)}.$$  

(10)

It is worth mentioning that strategies weights vary over time since the network changes over time.

A dynamic network which varies smoothly over time can be considered as a sequence of static networks, where in each time slot the network preserves its status. Hence, in our scenario, if an algorithm is able to converge to the best strategy in the static network setting, it will be able to adapt itself to the network evolution if the network status varies smoothly over time. Since in a VANET the network changes slowly over time, there is a marginal difference in the network status between consecutive time slots. Also, vehicles located inside a block will either stay in the block or move into the neighboring blocks in the following time-batch. Therefore, the best strategy for the next time-batch is similar to the best strategy of the current time-batch. Using this fact and the updating rule defined in (7) which takes the strategies of neighboring blocks into account, our algorithm tracks the smooth changes in the network and adapts itself to the current state of the network. In the following, an upper bound of regret for the proposed algorithm is derived. Using this upper bound, it is proved that the proposed algorithm converges to a globally optimal strategy in a static network setting with unknown statistics about the estimated density of vehicles inside the blocks.

**Theorem 1.** In the proposed regret-based algorithm, for any $\gamma, \eta \in (0, 1]$ we have:

$$E[R_a] = E[H^* - H_a] \leq (e - 1)\gamma H^* + A(1 - e^{(1 - \gamma)\eta}) \ln(A) + \eta H^*,$$

(11)

where $a = \{a(t)\}_{t=1}^T$ denotes the utilized strategy of the proposed algorithm.

**Proof.** The proof is inspired by the proof of Theorem 3.1 in [7] with substantial modifications. For $\gamma = 1$ there is nothing to prove since the inequality automatically holds. Let us consider the case where $\gamma \in (0, 1)$. Let $W_t = \sum_{i \in A} w_i(t)$. We construct the fraction $\frac{W_{t+1}}{W_t}$ and find an upper bound and a lower bound for it in the designed algorithm as follows:

**Upper bound derivation:**

$$\frac{W_{t+1}}{W_t} = \sum_{i \in A} w_i(t + 1)$$

$$= \sum_{i \in A} w_i(t) \exp \left( \left( \frac{1}{\gamma} \right) Q'_i(t) \right)$$

$$= \sum_{i \in A} \frac{p_i(t) - \gamma - \eta \sum_{i \in A} W_{N,r}(i)(j)}{(1 - \gamma - \eta) W_t} \frac{W_t}{W_t}$$

$$\exp \left( \left( \frac{\gamma}{\gamma} \right) Q'_i(t) \right).$$

Using the fact that $\exp(x) \leq 1 + x + (e - 2)x^2$ for $x \leq 1$ gives rise to:

$$\frac{W_{t+1}}{W_t} \leq \sum_{i \in A} \left[ p_i(t) - \frac{\gamma - \eta \sum_{i \in A} W_{N,r}(i)(j)}{(1 - \gamma - \eta)} \right]$$

$$\left[ 1 + \frac{\gamma}{A} Q'_i(t) + (e - 2) \left( \frac{\gamma}{A} Q'_i(t) \right)^2 \right].$$

Some algebra yields the following:

$$\frac{W_{t+1}}{W_t} \leq \sum_{i \in A} \left[ p_i(t) - \frac{\gamma - \eta \sum_{i \in A} W_{N,r}(i)(j)}{(1 - \gamma - \eta)} \right]$$

$$+ \frac{\gamma}{A(1 - \gamma - \eta)} \sum_{i \in A} p_i(t) Q'_i(t)$$

$$- \frac{\gamma}{A(1 - \gamma - \eta)} \sum_{i \in A} \frac{W_{N,r}(i)(j)}{1 - \gamma - \eta} Q'_i(t)$$

$$+ \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{i \in A} p_i(t)(Q'_i(t))^2$$

$$- \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{i \in A} \left( \gamma \right)$$

$$+ \frac{\eta}{A} \sum_{i \in A} W_{N,r}(i)(j) \frac{1}{(Q'_i(t))^2}.$$ (14)

By neglecting the terms with negative signs above and further expanding the first term, we can obtain:

$$\frac{W_{t+1}}{W_t} \leq \frac{1 - \gamma - \eta}{1 - \gamma - \eta}$$

$$+ \frac{\gamma}{A(1 - \gamma - \eta)} \sum_{i \in A} p_i(t) Q'_i(t)$$

$$+ \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{i \in A} p_i(t)(Q'_i(t))^2.$$ (15)

Consider the sequence of strategies taken by the proposed algorithm to be $\{a(t)\}_{t=1}^T$. Using Eq. (10) and Eq. (2), it is straightforward to validate that Eq. (15) can be written as:

$$\frac{W_{t+1}}{W_t} \leq 1 + \frac{\gamma}{A(1 - \gamma - \eta)} H(t) a(t) +$$

$$\frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} Q'_a(t).$$ (16)

Using the inequality $1 + y \leq \exp(y)$ on the right hand side along with taking the natural logarithm of both sides gives rise to:

$$\ln \frac{W_{t+1}}{W_t} \leq \frac{\gamma}{A(1 - \gamma - \eta)} H(t) a(t) +$$

$$\frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} Q'_a(t).$$ (17)
Now, by taking the summation over time, we get:

\[
\ln \frac{W_{T+1}}{W_1} \leq \frac{\gamma}{A(1 - \gamma - \eta)} H_a + \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{t=1}^{T} \sum_{i \in A} Q_i'(t). \tag{18}
\]

**Lower bound derivation:** By expanding the fraction of interest \( \ln \frac{W_{T+1}}{W_1} \) we get:

\[
\ln \frac{W_{T+1}}{W_1} = \ln \frac{\sum_{i \in A} w_i(T + 1)}{W_1} = \ln \frac{\sum_{i \in A} \prod_{t=1}^{T} \exp \left( \frac{\gamma Q_i'(t)}{A} \right)}{W_1} \geq \ln \frac{\prod_{t=1}^{T} \exp \left( \frac{\gamma Q_i'(t)}{A} \right)}{W_1} \forall i \in A.
\]

Note that the best strategy is among the possible strategy set. Let us denote it by \( b \). This results in:

\[
\ln \frac{W_{T+1}}{W_1} \geq \frac{\gamma}{A} \sum_{t=1}^{T} Q_b'(t) - \ln W_1 = \frac{\gamma}{A} \sum_{t=1}^{T} Q_b(t) - \ln A.
\]

By comparing the upper and the lower bounds we get:

\[
\frac{\gamma}{A} \sum_{t=1}^{T} Q_b'(t) - \ln A \leq H_a + \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{t=1}^{T} \sum_{i \in A} Q_i'(t).
\]

Now, by taking the expectation of both sides with respect to the distribution of the previous actions, and considering the fact that \( E[Q_{i(a)}'(t)] = E[H(t)\mid a(t)] \) we get:

\[
\frac{\gamma}{A} H^* - \ln A \leq \frac{\gamma}{A(1 - \gamma - \eta)} E[H_a] + \frac{(e - 2)\gamma^2}{A^2(1 - \gamma - \eta)} \sum_{t=1}^{T} E[H(t)\mid a(t)].
\]

Finally, we obtain the following bound:

\[
H^* - E[H_a] \leq (e - 1)\gamma H^* + \frac{A(1 - \gamma - \eta)}{\gamma^2} \ln(A) + \eta H^*. \tag{23}
\]

**Proof.** Note that:

\[
E[R_a] = E[H^* - H_a] \leq [(e - 1)\gamma H^* + \frac{A(1 - \gamma - \eta)}{\gamma} \ln(A) + \eta H^*] \leq (e - 1)\gamma T + \frac{A}{\gamma} \ln(A) + \eta T \Rightarrow E[R_a/T] \leq (e - 1)\gamma + \frac{A}{\gamma T} \ln(A) + \eta. \tag{25}
\]

Now, it is obvious that choosing \( \gamma = \Omega_1 T^{-\Gamma_1} \) and \( \eta = \Omega_2 T^{-\Gamma_2} \), where \( \Omega_1, \Omega_2 \) are positive real constants and \( 0 < \Gamma_1, \Gamma_2 < 1 \) implies the theorem. For instance, let us choose \( \gamma = \Omega T^{-1/2} \in (0, 1) \) and \( \eta = T^{-1/2} \) such that \( \Omega T^{-1/2} + T^{-1/2} < 1 \). This choice of parameters leads to the following:

\[
E[R_a/T] \leq (e - 1)\Omega T^{-1/2} + \frac{A}{\Omega T} \ln(A)T^{-1/2} + T^{-1/2}, \tag{26}
\]

which leads to the result when the limit is taken. \( \square \)

**IV. SIMULATION RESULTS**

In the simulation, we compare the performance of three algorithms including random, exhaustive search and regret-based minimization. We consider a city map which is divided into 400 blocks (20 \( \times \) 20 grid). In this map, the number of vehicles in each block are distributed following a uniform distribution in the range of [10, 2000]. The network varies over time according to a Gaussian random walk process, where the number of vehicles in each block changes with standard deviation of 100. It is assumed that changes in the block densities occur in every 100 time instances. Also, it is assumed that \( \eta = \gamma = 0.1 \) for the performed simulations.

Figure 2 depicts the convergence of the algorithm to the best strategy with respect to time. Note that in this case, \( \gamma = 0.1 \); hence, the maximum possible probability of selection would be limited to 90\% for all possible strategies. The convergence to the best strategy along with the adaptation to the new situations can be seen in every change in the blocks densities.

Figure 3 compares the non-normalized utilities of the regret-based minimization algorithm and random strategy selection. It can be seen that both algorithms start from the same utility value; however, the regret-based minimization algorithm dominates after a settling time. Also, the convergence to the maximum attainable utility can be observed for the regret-based minimization algorithm.

Figure 4 shows the instantaneous and average regret for the regret-based minimization algorithm and random strategy selection. Note that the average regret at each time-step refers to the cumulative regret up to that time-step divided by the time-step. Hence, it is decreasing slower than the instantaneous regret. Also, the decay in regret is evident for the proposed regret-based minimization algorithm.

Performed simulations imply the great potential of the regret-based minimization algorithm. It can be seen that the algorithm is able to converge to the best strategy for the mobile advertising scheme in a VANET.
In this work, we considered the problem in which an advertising company competes for the VANETs’ roadside units to spread its advertisements among vehicles. The city map is partitioned into a grid of blocks, and heterogeneous prices for blocks corresponding to their vehicle densities and geographical positions are considered. The goal is to find the best set of blocks which maximizes the utility of advertising companies. For this purpose, we adapted the regret-based minimization method to the dynamism of the network in hand. Rigorous proofs are provided for the bound of regret and convergence to the best strategy. The simulations revealed the great potential of using the proposed algorithm in VANETs.

V. CONCLUSION

In this work, we considered the problem in which an advertising company competes for the VANETs’ roadside units to spread its advertisements among vehicles. The city map is partitioned into a grid of blocks, and heterogeneous prices for blocks corresponding to their vehicle densities and geographical positions are considered. The goal is to find the best set of blocks which maximizes the utility of advertising companies. For this purpose, we adapted the regret-based minimization method to the dynamism of the network in hand. Rigorous proofs are provided for the bound of regret and convergence to the best strategy. The simulations revealed the great potential of using the proposed algorithm in VANETs.

REFERENCES


