Energy Aware Road-Side Unit Placement in Green Vehicular Networks

Naby Nikookaran and Terence D. Todd
Department of Electrical and Computer Engineering
McMaster University
Hamilton, Ontario, CANADA
Email: {nikookn,todd}@mcmaster.ca

George Karakostas
Department of Computing and Software
McMaster University
Hamilton, Ontario, CANADA
Email: karakos@mcmaster.ca

***Abstract*—This paper considers the problem of vehicular road-side unit (RSU) placement so that the sum of capital expenditure (CAPEX) and operating expenditure (OPEX) costs is minimized. The minimum cost solution considers these two cost contributions jointly when making RSU placement decisions. The input to the placement process consists of historical vehicular traffic traces and a set of candidate site locations from which RSU placements are chosen. An integer linear program (ILP) formulation is first given that computes the minimum cost placement based on the input traffic traces and candidate locations. A practical algorithm is then introduced that solves a relaxed version of the ILP and uses a novel rounding procedure to obtain real RSU placements. The algorithm takes into account the energy costs incurred by vehicular requests when the latter are scheduled using a minimum energy online scheduler. Performance results are presented that show that the proposed algorithm performs well compared to the case where placements are done without jointly considering both CAPEX and OPEX cost components.**

***Index Terms*—Road-side unit (RSU) placement, facility location, vehicular networks, service cost, energy efficiency.**

I. Introduction

Equipping vehicles with wireless communication capabilities is expected to be the next step towards Intelligent Transportation Systems (ITS). Vehicular ad-hoc networks (VANETs) will be essential component of this functionality that will help enable future road services [1]. VANETs will eventually support various applications such as road safety, intelligent transportation, location-dependent advertisement, and in-vehicle Internet access [2]. This area has attracted much attention from government, industry, and academia in recent years [1] [3] [4].

Recognizing the importance of vehicular communications, the US Federal Communications Commission (FCC) has allocated a 75 MHz bandwidth for Dedicated Short Range Communication (DSRC) for ITS radio services [5] (a 50 MHz bandwidth was also allocated by the European Conference of Postal and Telecommunications Administrations (CEPT) [1]). DSRC includes the IEEE 802.11p (PHY and MAC layers) and IEEE 1609 (upper layers) standards, which refer to wireless access in vehicular environments (WAVE) [2] [6]. DSRC/WAVE supports an environment in which vehicle speeds can be up to 200 kmh, and the communication range can reach up to 1000m, with a data rate of more than 27 Mbps [6] [7].

VANET operation can include both vehicular onboard units (i.e., OBUs), and fixed roadside unit infrastructure (i.e., RSUs). The latter is typically installed along the road side or at intersections where power grid connectivity is more common [6]. There are two basic modes of communication between RSUs and OBUs: (i) vehicle-to-vehicle (V2V), where an OBU communicates with other OBUs, and, (ii) vehicle-to-infrastructure (V2I), where an OBU communicates directly with an RSU [2] [6] [8]. The main functions of the RSUs include extending the communication range of the ad-hoc network, running safety and non-safety applications, and providing Internet connectivity for OBUs [6]. More specifically, RSUs can provide access to Internet gateways that provide a variety of mobile services as an alternative to cellular-based access technologies [2] [4] [9].

In this paper we consider the problem of RSU placement that minimizes the sum of capital expenditure (CAPEX) and operating expenditure (OPEX) costs. The total RSU cost includes both that of RSU installation, i.e., CAPEX, and long-term energy operating, i.e., OPEX, components. This combination affects both the initial placement costs of the RSUs, and their associated long-term operating costs. A placement that minimizes the total number of deployed RSUs, for example, may result in significantly higher operating costs in the long-term. It is therefore natural to study the placement of RSUs that minimize the combined CAPEX and OPEX costs. To the best of our knowledge, this is the first work that focuses on minimizing this combined cost.

Vehicles are assumed to travel along a given road network and generate requests for service. It is assumed that the requests may be delay tolerant and each has an associated time deadline, which should be satisfied by the deployed network. We are given a set of historical input traffic traces that reflect vehicular behaviour, as well as a set of RSU candidate locations. Each candidate location has an associated (CAPEX) installation cost, and once an RSU is placed and in operation, it incurs long-term operating (OPEX) costs due to the energy use of the RSU. The objective of the design algorithm is to select a subset of the candidate locations such that the total cost is minimized, and all vehicular requests are properly served.

An integer linear program (ILP) is first formulated that computes a minimum total cost placement. The ILP has a prohibitive solution time, even for moderate traffic size instances that make it impractical for real network designs. To address this issue, we develop a novel algorithm, referred to as Minimum Cost Route Clustering (MCRC), based on a rounding procedure applied to the LP relaxation of the ILP, and using the approximation algorithm of [10] for Capacitated Facility Location problems. MCRC is efficient and can be used for large scale problems. A variety of performance results are presented that show that MCRC outperforms RSU placement algorithms which directly solve the ILP, but minimize only CAPEX expenditures. The results demonstrate the inherent inefficiency introduced by considering only CAPEX costs.

The rest of this paper is organized as follows. Section II gives an overview of related work. Section III then describes the system model and a formulation of the RSU placement problem as an integer linear program. In Section IV, the MCRC algorithm is then presented. Performance results are given and discussed in Section V. Finally, we present our conclusions in Section VI.

II. Related Work

Several solutions have been introduced in the literature for the optimal RSU placement problem using a variety of objective functions. For the most part, previous work focuses on minimizing capital costs. There is, however, work that optimizes other criteria, such as road coverage, throughput, packet delivery ratio, and delay [3] [11] [12] [13] [14] [15]. This is typically done under the assumption that there is an upper limit on the number of available RSUs or the maximum permitted CAPEX cost. The minimization of the number of RSUs required under network performance requirements, such as network connectivity and road coverage, has also been studied (cf. References [16] [17] [18] [19] [20] [21] [22] [23] [4] [24]). These references assume that the RSUs are identical, although [25] considers RSUs with different configurations, i.e., power level, antenna type and backhaul connectivity type. Reference [26] seeks to minimize the number of installed RSUs as well as their operating time, by putting them to sleep whenever possible. In our work, no RSU on/off switching functionality is assumed.

III. System Model

Vehicles are assumed to travel along a given road network and generate requests for service that are communicated on an uplink channel to the next RSU that is encountered. The responses to these requests are then scheduled and served by one or more RSUs over time slotted downlink channels. Each request has an associated time deadline. The objective is to place a set of road-side units (RSUs) chosen from a candidate site location set, so that the network can properly schedule vehicular demands, and such that the total of the RSU opening and service costs are minimized.

The RSU placement decisions are made offline, as part of the design of the network to be deployed. This is done using a vehicular traffic input trace that reflects vehicular traffic flow based on historical traffic measurements. The packet scheduling that occurs in this phase is also offline, in that the complete set of inputs over all time is provided to the design algorithm all at once. This permits us to obtain lower bounds on total cost that motivate the performance of the online system.

Once the network is designed and deployed, it is then subject to real online traffic inputs. In online scheduling, the inputs are provided to the system in real time and it must schedule data traffic in a causal fashion, based solely on past and current inputs. To install an RSU at a candidate location we pay an *opening cost*, and to serve a vehicle by an opened RSU, we pay a *service cost*. These are defined as follows:

Opening cost: The opening cost of an RSU is determined by its location and its configuration settings (such as its backhaul connection type, power source, channel capacity, coverage range, antenna type, etc.). A location-based RSU cost analysis was done in [7]. Our model can accommodate non-homogeneous RSUs that are operated with different costs (e.g., operated by the wired electrical power grid or by solar power [7]), in addition to limited but different coverage areas. This limitation on the maximum coverage range is sometimes used to control radio interference levels [5] [6] [7].

Service cost: We assume that the RSUs use power control when communicating with vehicles, i.e., they adapt their transmit power in order to maintain a constant bit rate [8] [9]. The energy cost of this communication depends on the radio link propagation conditions. The total operating cost depends on the *planning time horizon*, i.e., the time period over which the RSU cost is amortized, which may be as long as one or more decades. We assume that the vehicle traffic load input trace is statistically representative of the traffic flow and we can normalize the operating cost to the long-term planning time horizon.

Once an RSU has been deployed, it remains in continuous operation serving vehicular requests. Each vehicle request has a release date, i.e., the time when the request is generated, and a due date, i.e., the deadline of the associated RSU response. A request that is un-served or is served beyond its due date is counted as a *dropped request*. We assume that a vehicle generating a request, communicates its size, release, and due dates to the first RSU it encounters, and, therefore, the system is aware of these parameters for scheduling purposes. We also assume that the route of a vehicle is known, and that each vehicle communicates to the system (through the first RSU encountered) its current location, final destination, and intended route [27] [28]. This is a reasonable assumption, since drivers tend to follow their habits and traffic information in planning their daily route to work, home or other destinations [29]. This assumption is also consistent with the driver-less car functionality that is beginning to appear.

The system model is more formally defined as follows. Let $\mathcal{N} = \{1, \ldots, N\}$ be the set of RSU candidate locations, and $\mathcal{V} = \{1, \ldots, V\}$ be the set of vehicles serviced by the installed RSUs, each with a set of requests $\mathcal{R}_v$, and $|\mathcal{R}_v| = R_v$. Let $\mathcal{R} = \cup_{v \in \mathcal{V}} \mathcal{R}_v = \{1, \ldots, R\}$ be the set of all requests, each with its own size ℓ_r. With a slight abuse of notation, we will refer to an RSU installed at location n as 'RSU n'. We define decision variables Y_n, so that $Y_n = 1$ if RSU n is installed, and $Y_n = 0$ otherwise. The cost of opening an RSU at location n is f_n. Let $\mathcal{T} = \{1, \ldots, T\}$ be the set of time slots; within a time slot, RSU n has the capacity to transmit to at most u_n vehicles, and a vehicle can communicate with at most one RSU. Note that f_n and u_n depend only on the location n, i.e., RSUs installed in different locations are allowed to be of different types with different opening costs.

We define the decision variables, X_{ntr}, such that $X_{ntr} = 1$ if RSU n serves request r of vehicle v during time slot t, and $X_{ntr} = 0$ otherwise. When vehicle v is within the coverage area of RSU n during time-slot t, the energy cost for servicing request r is c_{ntr}, which depends on the RSU-vehicle distance (and other propagation effects) in time slot t. Otherwise, $c_{ntr} = \infty$. In order to enforce the servicing of all requests, if possible, we define the non-serviced portion of request r by variable Z_r, and give it a large cost, D_r. That is, if $Z_r > 0$, then request r is *dropped* and incurs a very large cost $D_r Z_r$. As a result, in the optimization defined below, the scheduler will never drop a request, unless there is a capacity constraint violation. Since this part of the objective function is an artifice to ensure service, it will not be included in the total cost we present in the results obtained.

Given the above definitions and for a given input traffic design trace, we formulate the optimum cost as an integer linear program. This provides a lower bound on the total cost and is used in Section IV to obtain a practical RSU placement algorithm using a novel rounding procedure. The optimization is given as follows and discussed below.

$$\min_{X,Y,Z} \sum_{n \in \mathcal{N}} f_n Y_n + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} \sum_{r \in \mathcal{R}} c_{ntr} X_{ntr} + \sum_{r \in \mathcal{R}} D_r Z_r \quad \text{s.t.} \tag{ILP}$$

$$Z_r + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} X_{ntr} = \ell_r \qquad \forall r \in \mathcal{R} \tag{1}$$

$$X_{ntr} \le Y_n \qquad \forall n \in \mathcal{N}, t \in \mathcal{T}, r \in \mathcal{R} \tag{2}$$

$$\sum_{r \in \mathcal{R}} X_{ntr} \le u_n Y_n \qquad \forall n \in \mathcal{N}, t \in \mathcal{T} \tag{3}$$

$$\sum_{n \in \mathcal{N}} \sum_{r \in \mathcal{R}_v} X_{ntr} \le 1 \qquad \forall t \in \mathcal{T}, v \in \mathcal{V} \tag{4}$$

$$Y_n, X_{ntr} \in \{0, 1\} \qquad \forall n \in \mathcal{N}, t \in \mathcal{T}, r \in \mathcal{R} \tag{5}$$

$$Z_r \in \{0, 1, \ldots, \ell_r\} \qquad \forall r \in \mathcal{R} \tag{6}$$

Constraint (1) guarantees that requests are satisfied and Constraint (3) enforces the capacity constraint for the RSUs. Constraint (4) implies that only one request of vehicle v can be serviced during time slot t. Note that (3) and (5) imply Constraint (2), but the latter is crucial for our rounding heuristic, strengthening the LP relaxation presented below.

IV. The Minimum Cost Route Clustering (MCRC) Algorithm

Our proposed heuristic is based on the following primal LP relaxation of (ILP). Unlike (ILP), it can be solved in polynomial time complexity but does not give integral solutions for the decision variables. This issue is addressed by using the rounding procedure discussed below.

$$\min_{X,Y,Z} \sum_{n \in \mathcal{N}} f_n Y_n + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} \sum_{r \in \mathcal{R}} c_{ntr} X_{ntr} + \sum_{r \in \mathcal{R}} D_r Z_r \quad \text{s.t.} \tag{PLPR}$$

$$Z_r + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} X_{ntr} \ge \ell_r \qquad \forall r \in \mathcal{R} \tag{7}$$

$$X_{ntr} \le Y_n \qquad \forall n \in \mathcal{N}, t \in \mathcal{T}, r \in \mathcal{R} \tag{8}$$

$$\sum_{r \in \mathcal{R}} X_{ntr} \le u_n Y_n \qquad \forall n \in \mathcal{N}, t \in \mathcal{T} \tag{9}$$

$$\sum_{n \in \mathcal{N}} \sum_{r \in \mathcal{R}_v} X_{ntr} \le 1 \qquad \forall t \in \mathcal{T}, v \in \mathcal{V} \tag{10}$$

$$Y_n \le 1 \qquad \forall n \in \mathcal{N}, t \in \mathcal{T}, r \in \mathcal{R} \tag{11}$$

$$Z_r \le \ell_r \qquad \forall r \in \mathcal{R} \tag{12}$$

$$Y_n, X_{ntr}, Z_r \ge 0 \qquad \forall n \in \mathcal{N}, t \in \mathcal{T}, r \in \mathcal{R} \tag{13}$$

Rounding the solution of (PLPR) to an integral one is non-trivial, since the integrality gap for this relaxation is infinite [30]. Levi et. al. [10] introduced an LP-based approximation algorithm for the capacitated facility location problem, in which the service has no time constraints, the service cost is time-independent, and there is no capacity associated with clients. It consists of a two-phase clustering procedure, followed by a rounding algorithm, and has a provable approximation factor of 5 when the connection costs are in a metric space. Unfortunately, our model is more complicated than the problem in [10], and, moreover, our operating costs do not come from a metric space; therefore, the known approximation factor guarantees for facility location problems do not necessarily apply in our case. Accordingly, we develop a novel heuristic referred to as the Minimum Cost Route Clustering (MCRC) algorithm, which operates in two steps. In the first, all (partially) opened RSUs from the solution of (PLPR) are partitioned into clusters. In the second step, the rounding algorithm installs all fully opened RSUs in each cluster, and continues installing fractionally open RSUs, until it opens enough RSUs to satisfy all service requirements for that cluster.

The algorithm starts with the fractional solution of (PLPR). This solution consists of (partially) opened RSUs and (fractional) request assignments to the (partially) opened RSUs. In the next step, our algorithm moves the fractional requests of vehicles from one RSU to another, so it can fully open some RSUs and fully close the rest, thus producing an integer solution. It is obvious that the displacement of requests increases the assignment costs and, although we cannot guarantee an upper bound for this increase, as done in [10], our simulation results show that the extra cost of assignment displacements is low.

As argued in [10], moving assignments too much leads to prohibitively expensive results. For this reason, a clustering

Algorithm 1 Minimum Cost Route Clustering

1: Let (X, Y) be the solution to (PLPR) and α the dual variables for constraints (7)
2: Let η be the clustering threshold
3: $\triangleright$ *Step 1: Clustering*
4: $\mathcal{F} = \{n \in \mathcal{N} : Y_n > 0\}$ $\triangleright$ *(partially) opened RSUs*
5: **for all** $v \in \mathcal{V}$ **do**
6: $\quad \mathcal{F}_v = \left\{n \in \mathcal{F} : \sum_{t \in \mathcal{T}} \sum_{r \in \mathcal{R}_v} X_{ntr} > 0\right\}$ $\triangleright$ *RSUs that fractionally serve vehicle* v
7: $\quad \alpha_v = \sum_{r \in \mathcal{R}_v} \alpha_r$
8: **end for**
9: **Initialize:**
10: $\quad \mathcal{C} = \emptyset$ $\triangleright$ *cluster centers*
11: $\quad \mathcal{S} = \mathcal{V}$ $\triangleright$ *cluster center candidates*
12: $\quad \mathcal{N}_v = \emptyset,\ \forall v \in \mathcal{S}$ $\triangleright$ *one potential cluster per vehicle* v
13: **while** $\mathcal{S} \neq \emptyset$ **do**
14: $\quad$ **for all** $v \notin \mathcal{C}$ **do**
15: $\quad\quad \mathcal{B}_v = \{n \in \mathcal{F}_v : n \notin \cup_{k \in \mathcal{C}} \mathcal{N}_k, c_{nv} \leq \min_{k \in \mathcal{C}} c_{nk}\}$
16: $\quad$ **end for**
17: $\quad \mathcal{S} = \{v \notin \mathcal{C} : \max_{n \in \mathcal{B}_v, t \in \mathcal{T}, r \in \mathcal{R}_v} X_{ntr} > \eta\}$
18: $\quad$ Pick $v \in \mathcal{S}$ with the smallest α_v value and if there are more than one, pick the one has the largest $\sum_{n \in \mathcal{B}_v} u_n Y_n$.
19: $\quad$ Set $\mathcal{N}_v = \mathcal{B}_v$, $\mathcal{C} = \mathcal{C} \cup \{v\}$
20: **end while**
21: $\mathcal{U} = \mathcal{F} - \cup_{k \in \mathcal{C}} \mathcal{N}_k$
22: **for all** $n \in \mathcal{U}$ **do**
23: $\quad v = \arg\min_{k \in \mathcal{C}} c_{nk}$
24: $\quad \mathcal{N}_v = \mathcal{N}_v \cup \{n\}$
25: **end for**
26: $\triangleright$ *Step 2: Rounding*
27: **for all** Cluster centers $v \in \mathcal{C}$ **do**
28: $\quad$ Open all of the fully opened RSUs in $\mathcal{N}_v$.
29: $\quad \mathcal{Q}_v = \{n \in \mathcal{N}_v : Y_n < 1\}$
30: $\quad D_{left} = \sum_{n \in \mathcal{Q}_v} u_n Y_n$
31: $\quad$ Sort the RSUs in $\mathcal{Q}_v$ in increasing order of $(f_n/u_n + c_{nv})$
32: $\quad$ **while** $D_{left} > 0$ **do**
33: $\quad\quad$ Let n be the next RSU in the sorted list
34: $\quad\quad$ Open RSU n
35: $\quad\quad D_{left} = D_{left} - \min(D_{left}, u_n)$
36: $\quad\quad \mathcal{Q}_v = \mathcal{Q}_v \setminus \{n\}$
37: $\quad$ **end while**
38: **end for**

step is used before the rounding procedure. It divides the problem into subproblems, and the rounding of their fractional solutions is done separately for each. The clustering step imposes an extra cost, which is due to the aggregated effect of rounding the fractional solutions in each sub-problem.

Algorithm 1 shows the details of our algorithm. Let (X, Y) be the optimal solution to (PLPR) (assuming that all requests are feasible and $Z = 0$), and α_r the optimal dual variables for relaxed constraints (7). The two steps of Algorithm 1 are denoted as *Clustering* and *Rounding*.

In Clustering, we partition the RSUs with $Y_n > 0$ into *clusters*, each of which will be "centered" around a vehicle that we call the *cluster center*. More specifically, for each vehicle v, we define α_v to be the summation of α_r over all its requests. Also, let $\mathcal{F}_v$ be the set of (partially) opened RSUs that (fractionally) serve vehicle v. Let $\mathcal{S}$ be the set of cluster center candidates (initially the set of all vehicles), and $\mathcal{C}$ be the set of current cluster centers (initially empty). We use $\mathcal{N}_v$ to denote a potential cluster centered around vehicle v. Initially, $\mathcal{N}_v$ is empty for all vehicles.

At every iteration of lines 13-20, we define a set $\mathcal{B}_v$ for every vehicle v in $\mathcal{S}$, as long as the latter is non-empty. Of all RSUs in $\mathcal{F}_v$, set $\mathcal{B}_v$ contains only those that are "closer" to v than all cluster centers currently in $\mathcal{C}$, according to a closeness function that is based on the average connection cost between a vehicle v and an RSU n, i.e.,

$$c_{nv} = \frac{\sum_{t \in \mathcal{T}_{nv}} P_{n,v}(t)}{|\mathcal{T}_{nv}|} \tag{14}$$

where $P_{n,v}(t)$ is the communication cost between vehicle v and RSU n at time slot t, and $\mathcal{T}_{nv}$ is the set of time slots during which vehicle v is inside the coverage area of RSU n. This approach prevents the creation of too many clusters, which will eventually reduce the extra costs incurred by opening the fractional RSUs in the clusters. The intuition behind this is as follows: The removal of RSU n from $\mathcal{B}_v$ because it is closer to some other vehicle v', implies that the two vehicles v, v' share part of their routes. Therefore, we can divide the route of v in two parts, the part that is shared with v', and the part that is not; we charge v with only the cost of the latter.

Throughout the Clustering phase, set $\mathcal{S}$ contains all RSUs that are candidates for opening; these are the RSUs that are (partially) open by at least a preset factor η, called the *clustering threshold*. Parameter η can be preset to any value between 0 and 1, but in our case we set $\eta = 0$ to force all partially opened RSUs to be candidates for full opening. In each iteration, we pick the vehicle $v \in \mathcal{S}$ with the smallest α_v value (we break ties by picking the vehicle that has the largest capacity in $\mathcal{B}_v$). We form a cluster centered at v, with $\mathcal{N}_v = \mathcal{B}_v$ and update sets $\mathcal{C}$ and $\mathcal{S}$ accordingly. We continue this procedure while $\mathcal{S}$ is not empty. After that, there can still be RSUs in $\mathcal{F}$ that are not assigned to any cluster. Each of those RSUs is assigned to the cluster whose center is closer to it.

Clustering is followed by Rounding. For each cluster $\mathcal{N}_v$, and after opening all RSUs with $Y_n = 1$, we start opening the rest of the RSUs in this cluster, one-by-one and in increasing order of $(f_n/u_n + c_{nv})$, until all capacity requirements of this cluster are satisfied. After enough RSUs have been opened, we schedule the requests to time-slots and RSUs. Since the capacity of the opened RSUs is sufficient to serve all the requests, this offline scheduling problem is feasible, i.e., the additional request drop ratio is zero (recall that we already have that $Z = 0$).

Multiple-Choice RSU Placement: In the model described above, there is only one choice for the RSU to be opened at a candidate location. We can easily generalize this, by allowing the RSU to be chosen from a set of different types; however, we still require that at most one RSU is opened at any candidate location. The only change needed is the extension of the Y variables, to have one for each choice at a location (instead of one per location), and the extra constraint that the sum of these variables must be at most 1 at every location. Algorithm 1 does not need to change significantly; we just remove the rest of the choices at a location, once we decide to open an RSU of a specific type.

V. Performance Results

In this section, the performance of the proposed RSU placement algorithm is considered. In order to evaluate the performance of MCRC, two different on-line scheduling algorithms were used to assess the placements that the algorithm generates. The first is the GMCF scheduler introduced in [8]. GMCF schedules requests on a single RSU using a minimum cost flow graph formulation that minimizes total service cost over a finite scheduling window. The second algorithm is the one-objective min-max scheduler presented in [9]. This algorithm schedules requests across multiple RSUs and attempts to minimize the maximum service cost on any of the RSUs. Since our goal is to minimize the total service cost on multiple RSUs, the schedulers are adapted to work in this setting. These two schedulers are referred to as the *Energy Scheduler* and the *Min-Max Scheduler*, respectively, and both are non-preemptive.

Our proposed algorithm is compared with RSU placements that minimize total capital cost, referred to as the Minimum Capital Cost Placement (MCCP) algorithm. This is motivated by the work discussed in Section II, which can be adapted to our problem by adjusting the objective function of the ILP by minimizing the sum of the first and last components of (ILP) (i.e., $\sum_{n\in\mathcal{N}} f_n Y_n + \sum_{r\in\mathcal{R}} D_r Z_r$) subject to constraints (1)-(6). This objective is more general than those that minimize the total number of deployed RSUs, since they do not necessarily minimize total capital cost. The comparisons therefore show the advantage that MCRC has compared to those that focus on CAPEX costs alone.

The performance evaluation is done using 10 vehicular traffic trace inputs. One trace is first used for the offline RSU design and placement, which determines the CAPEX deployment cost. After the design phase, all ten traces are then used as inputs to the online experiments, which determine the OPEX costs. The total cost presented in the simulation results is the sum of the two, and the plotted OPEX cost is obtained by averaging the service costs over each simulation run for the 10 traffic traces. Uniform service request generation is used for all vehicles, i.e., the same arrival rate, size, and time-to-live (TTL). As in [31] [32], we assume that TTL is 40 time slots for each request. The maximum request drop rate is set to 5%.

A Manhattan-like grid road configuration is used, consisting of three horizontal and 5 vertical streets that are all bidirectional. The smallest block has a 1 km square area, which gives a total deployment region of 11.25 km^2. Figure 1 shows the city grid with the candidate RSU site locations used as input. To calculate the RSU candidate locations, we divide each street into segments of length equal to twice the RSU coverage range, and the center of each segment is taken as an RSU candidate location. Note that the beginning and the end of each street are under the coverage of the RSUs at the intersections, and therefore, those two sections are subtracted from the length of the street.

It has been shown that microscopic models are the most appropriate for VANET simulations [33] [34] [35]. Accordingly, we use the Simulation of Urban MObility (SUMO) tool, which is a microscopic mobility generator along with its other capabilities [36]. Vehicles arrive to the city according to a Poisson process. The source and destination of their trip are selected uniformly from the set of intersections [27] [37]. As in other studies that use shortest paths for route planning, Dijkstra's Algorithm [38] is used to determine the paths. The average travel time of each street according to its length, speed limit, and expected traffic density is calculated and is passed to this algorithm [21] [24] [29] [39]. The vehicle traces are 30 minutes in duration.

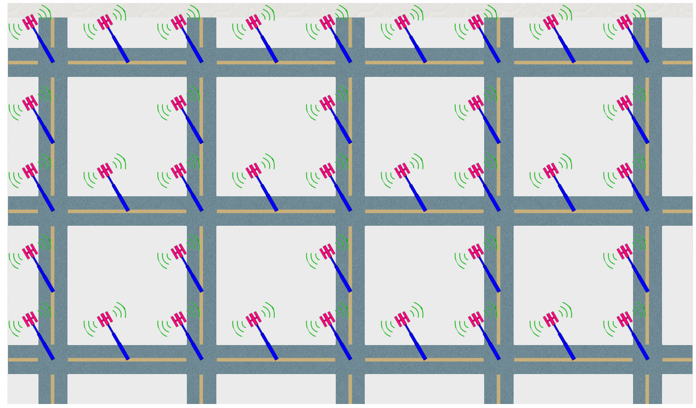

Fig. 1: City Grid with RSU Candidate Site Locations.

TABLE I: Simulation Parameters

Parameter Name	Parameter Value(s)
Planning Time Horizon	20 years
Candidate Site Locations	37 sites
RSU Coverage Range	250 m each side
Data Rate	6 Mbps
Vehicle Arrival Rate	0.5 and 1.0 per sec.
Request Arrival Rate	0.0125 per time slot
Request Size	8 time slots
Request Time-to-Live	40 time slots
Street Speed Limit	50 or 60 km/h
Street Number of Lanes	4 or 5
Traffic Light Control	Yes
Path Loss Exponent	$\alpha = 2.7$
Shadowing Standard Deviation	$\sigma_{\text{dB}} = 4$

A distance dependant exponential path-loss model with log-normal shadowing [40] is used to determine the transmit power needed over a given link. The standard deviation of the shadowing component is taken to be $\sigma_{\text{dB}} = 4$. Table I summarizes these and the other parameters used in our experiments.

The effect of single RSU capital cost, request size, request arrival rate, request time-to-live (deadline), and vehicle arrival rate is studied. Two experiments were performed that show the trade-off between the two components of the total deployment cost, i.e., the opening and service costs. In the first experiment, which is referred to as "single-choice RSU placement", we give one option for the RSU configuration at each candidate site location. Both MCRC and MCCP algorithms decide the locations where RSUs of that type are placed. In the second, referred to as "multiple-choice RSU placement", the output of each algorithm also includes the RSU configuration to be chosen for each selected candidate location.

1) Effect of Per RSU Capital Cost: In this first set of results we evaluate the effect of the per RSU capital cost for two different vehicular traffic load situations, one low, and the other high. RSU placements are compared for both single and multiple choice RSU placement. To properly evaluate the performance of our algorithm, we consider a basic unit cost for each RSU type, and then, we multiply every basic unit cost by the same factor, referred to as the "capital cost factor" during the experiment. The basic unit cost is equal to \$1,000 and \$1,500 for grid-powered and solar-powered RSUs, respectively [7]. The single RSU capital cost at each point is equal to its corresponding cost factor multiplied by the basic unit cost. Figures 2 and 3 show the results of this experiment for single-choice placement, and Figures 4 and 5 show the multiple-choice case.

In Figures 2, 3, 4 and 5, the horizontal axis shows the factor by which the single RSU capital cost is increased. The total cost of RSU deployment and its component opening and service costs are shown in Figures 2, 3, 4 and 5. In each subfigure, the lower bound is shown as a black solid line. The *Energy* and *Min-Max* schedulers for both MCRC and MCCP RSU placement are shown with different line patterns and markers. To follow the same pattern in all subfigures, we draw the opening cost of MCRC and MCCP twice, i.e., once for each scheduling algorithm.

As seen in Figures 2, 3, 4 and 5, the MCCP algorithm is insensitive to the service cost as it opens the minimum number of lower opening cost RSUs that are needed to serve requests, i.e., the MCCP algorithm concentrates the requests in the minimum number of RSUs possible. For this reason, when the per RSU capital cost changes, the MCCP algorithm opens the same set of RSUs. As a result, the opening cost increases linearly with per RSU capital cost and the service cost remains constant for different per RSU capital costs.

On the other hand, the MCRC algorithm tries to create a balance between the opening and service cost components and tends to outperform MCCP. In Figures 2 to 5, there are four regions that can be seen in the service cost subfigures. The first two show the use of load balancing and the second two show the use of load concentration. The first region starts with flat service cost and is followed by a smooth increase in the service cost. In these regions, which correspond to less expensive RSUs, the MCRC algorithm opens more RSUs to bring down the service cost. This approach continues until there is no more decrease in the service cost. This happens either when there are no more RSUs to open, or when opening more RSUs increases the opening cost without improving the service cost.

The third region has a sharper slope compared with the second. As the single RSU capital cost increases, the MCRC algorithm concentrates the requests on a smaller number of RSUs. This approach lowers the ratio by which the opening cost increases. Although the service cost increases by load concentration, the overall cost increases with a lower rate. The reason that the load concentration leads to the higher service

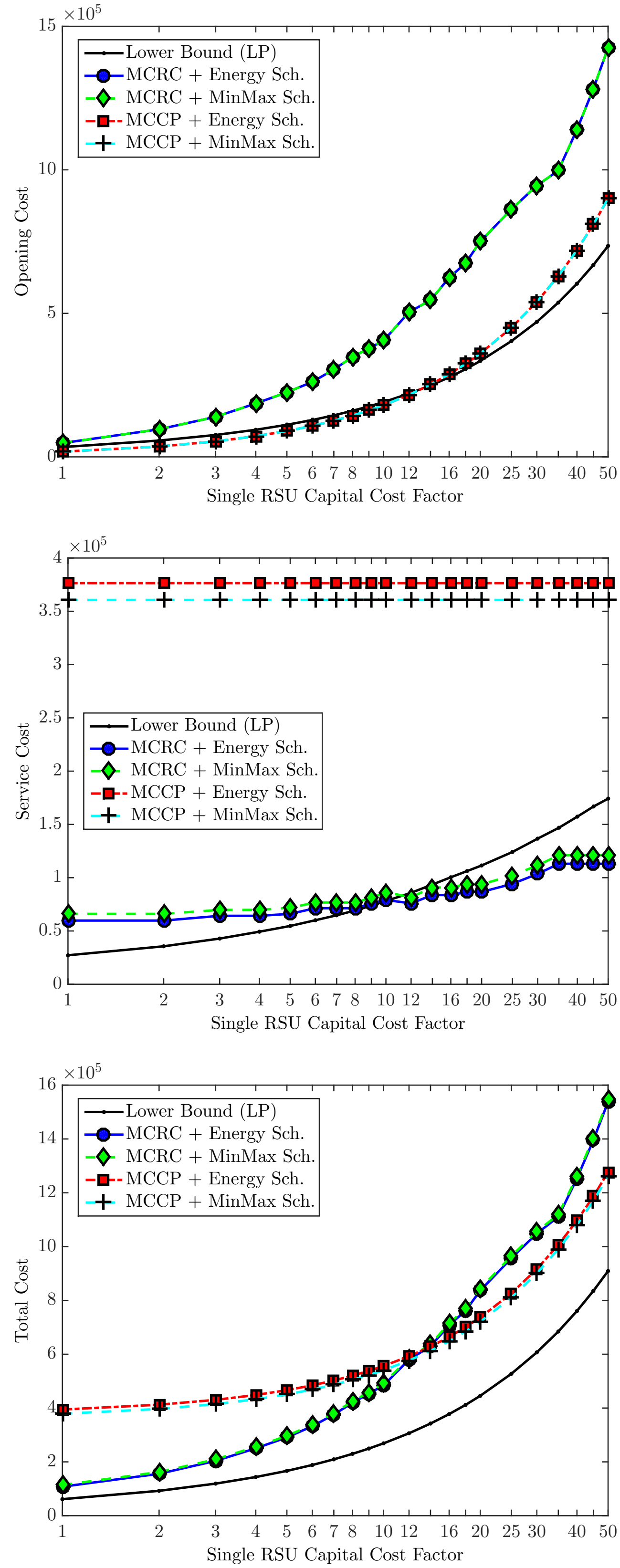

Fig. 2: The Effect of Per RSU Capital Cost on Single-Choice RSU Placement with Low Vehicle Traffic Load.

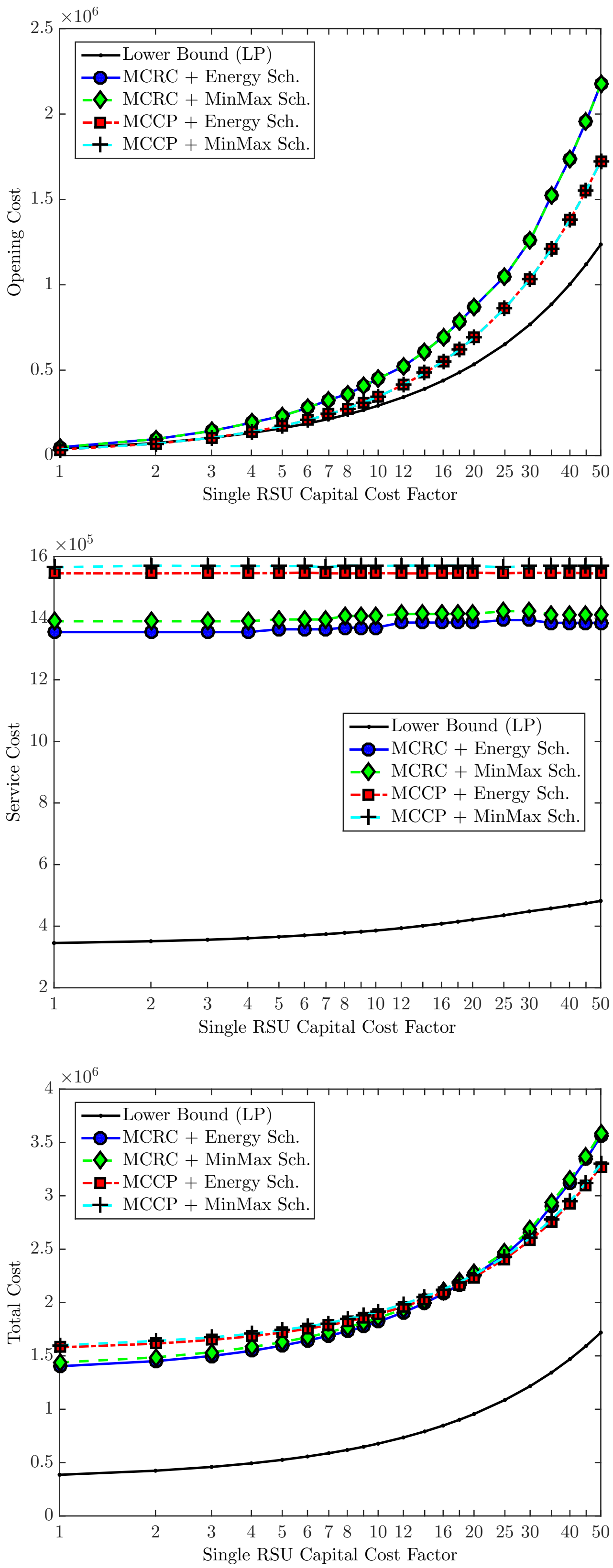

Fig. 3: The Effect of Per RSU Capital Cost on Single-Choice RSU Placement with High Vehicle Traffic Load.

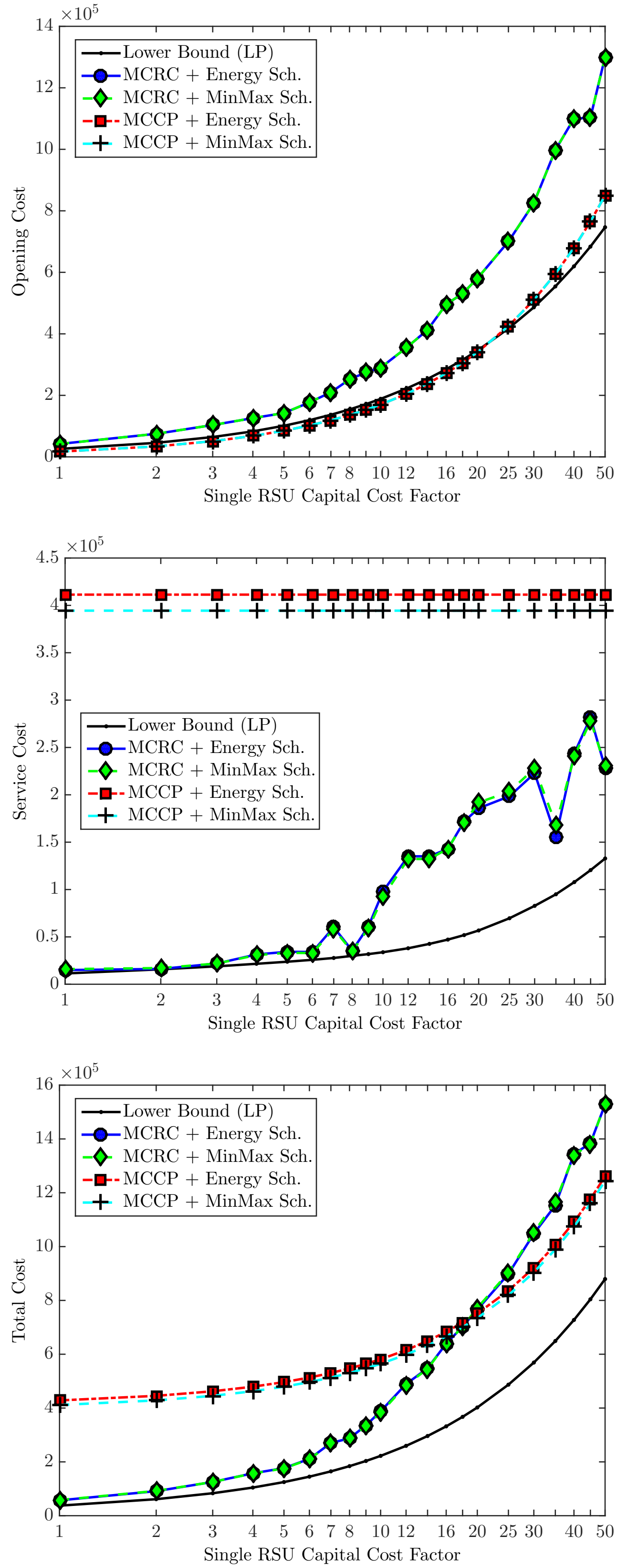

Fig. 4: The Effect of Per RSU Capital Cost on Multiple-Choice RSU Placement with Low Vehicle Traffic Load.

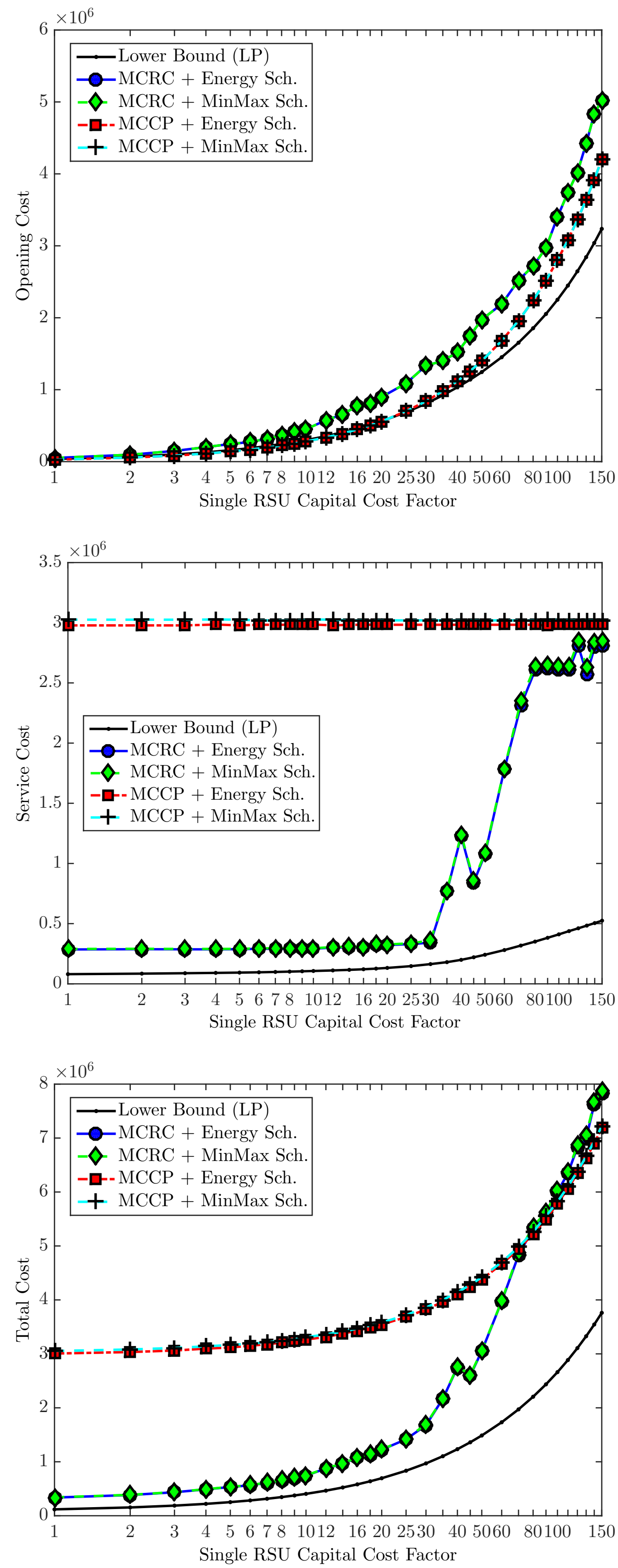

Fig. 5: The Effect of Per RSU Capital Cost on Multiple-Choice RSU Placement with High Vehicle Traffic Load.

cost is because of the RSU capacity limitation, i.e., it is not possible to serve all of the transferred requests at their most favourable times. Therefore, some requests will be served with higher costs, when their issuing vehicles are more distant from the RSUs. After a certain point, there is no way to decrease the number of opened RSUs. This corresponds to the fourth region and has the lower rate by which the service cost increases. There are two reasons for this. Either request deadlines prevent the MCRC algorithm from transfering them from one RSU to another, or, one or more RSUs reach their capacity limit, so that they cannot accept more requests from other RSUs. If both of these happen, the RSU inevitably drops requests.

Note that at the end of the third region and during the fourth, where the MCCP algorithm shows better performance, the opening cost becomes the dominant component of the objective function. The MCRC algorithm opens a smaller fraction of RSUs to bring down the opening cost, but this increases the integrality gap between the LP optimum value and the approximated value.

The four regions discussed above happen at different per RSU capital cost factors and depend on the vehicle traffic load, the data traffic load, and the RSU placement model, i.e., single-choice or multiple-choice. For example, when vehicle traffic load increases, as in Figure 3 compared to Figure 2 or Figure 5 compared to Figure 4, the data traffic load increases, which causes an increase in the service cost. But when there is insufficient capacity, opening more RSUs also leads to higher opening costs. When there are no more RSUs to open or when request deadlines do not allow additional load concentration, increasing the vehicle arrival rate increases only the service cost. At this point, the MCRC algorithm shows its advantage by moving the crossing point of the total cost from a capital cost factor of 12 in Figure 2 to a capital cost factor of 18 in Figure 3. The effect of vehicle arrival rate is discussed in more detail in Section V-5.

A similar behavior can be seen by comparing Figures 4 and 5, but for different reasons. There are two options available at each candidate site location, i.e., grid and solar-powered RSUs. The latter are more expensive, but their service cost is cheaper. This gives the MCRC algorithm more flexibility to balance these cost components. This can be done using fewer RSUs compared to the similar scenario in single-choice RSU placement. For example, in Figures 2 and 3, at the capital cost factor of 1, the service cost cannot be reduced, since there are no more RSUs to open. But at the same capital cost factor, the service cost in Figure 4 is almost one third of the service cost in Figure 2, and the service cost in Figure 5 is almost one fourth of the service cost in Figure 3. This is true even though there are more RSUs to be opened and happens during the first and second regions by opening more solar-powered RSUs instead of grid-powered RSUs. But during the third and the fourth regions, when the single RSU capital cost becomes higher than the service cost, the MCRC algorithm not only concentrates the requests to a smaller number of RSUs, but also prefers grid-powered RSUs. This causes the sharper slope in the service cost during the third region.

By comparison, the opening cost remains almost the same in Figure 4 as compared to Figure 2, and in Figure 5 compared to Figure 3. This is because even though the more expensive RSUs are used, a fewer number are opened. As a result, the total cost from the MCRC algorithm in the multiple-choice RSU placement shows improvement compared to the single-choice RSU placement. The crossing point of the two algorithms moved from a capital cost factor of 12 in Figure 2 to 20 in Figure 4, and from 18 in Figure 3 80 in Figure 5. On the other hand, the MCCP algorithm ignores the service cost, and only opens grid-powered RSUs. As a result, the service cost of the MCCP algorithm in Figure 5 is almost double that in Figure 3, which degrades its overall performance.

As discussed earlier, if the traffic input surpasses the network capacity, some of the requests will be dropped. In this case, the LP solutions show the regions in which the network is saturated and this can be detected at early stages of the network design. The request drop ratio of the offline LP, lower bound, in Figures 3 and 5 is equal to 0.1%.

Another reason that requests are dropped is due to imperfections in online scheduling. The online scheduler, because of its causal nature, may inadvertently create RSU congestion, leading to an increase in the request drop ratio. In Figures 2 to 5, it can be seen that the min-max scheduler has a slightly higher service cost than the energy scheduler. This gap is higher when these two schedulers are used for the MCCP algorithm output. However, the results from these figures show that the min-max scheduler has a better performance in terms of the request drop ratio. In these results, the min-max scheduler has a smaller request drop ratio than the energy scheduler. This is because it balances the load between RSUs, and generally causes less congestion than the energy scheduler.

The comparison between the MCRC and MCCP algorithms in terms of the request drop ratio shows that the former has better performance. In the single-choice RSU placement, i.e., Figures 2 and 3, the request drop ratio of the MCRC algorithm, regardless of the scheduling algorithm is about 0.02%, while the MCCP algorithm has the request drop ratio of 0.6% and 0.5% for the energy scheduler and the min-max scheduler, respectively. As the vehicle arrival rate increases, the competition between vehicles increases. Therefore, more requests are expected to be dropped. In the multiple-choice RSU placement, i.e., Figures 4 and 5, the request drop ratio of the MCRC algorithm is equal to 2.2% and 1.6% for using the energy scheduler and the min-max scheduler, respectively. The request drop ratio of the MCCP algorithm is equal to 3.0% and 2.4% for the energy scheduler and the min-max scheduler, respectively. Similarly, in multiple-choice RSU placement, shown in Figures 4, the request drop ratio of the MCRC algorithm is equal to 0.91% and 0.84% for the energy and the min-max schedulers, respectively. The request drop ratio of the MCCP algorithm is equal to 0.67% and 0.51% for the energy scheduler and the min-max scheduler, respectively. In Figure 5, the request drop ratio of the MCRC algorithm is equal to 2.8% and 2.2% for the energy and the min-max schedulers. The request drop ratio of the MCCP algorithm is equal to 3.8% and 3.0% for the energy scheduler and the min-max scheduler.

2) The Effect of Request Size: To evaluate the effect of request size on algorithm performance, we set the capital cost factor to 10 from Section V-1. This means that the capital cost of each grid-powered RSU and each solar-powered RSU are equal to $10,000 and $15,000, respectively. Because of space limitations, we only present the results for high vehicle traffic load. This means that the vehicle arrival rate is equal to 1 vehicle per second, i.e., 2 vehicles per time slot. Figures 6 and 7 show these results for the single-choice RSU placement and the multiple-choice RSU placement, respectively.

In these results, we increase the size of each individual vehicle request from 1 to 10. The MCRC algorithm shows a slight advantage over the MCCP algorithm in Figure 6. However, in Figure 7, the MCRC algorithm significantly outperforms MCCP. As in the previous section, there are four regions. The first two are load concentration, which correspond to the lower data traffic load. When the request size is small, the service cost is low. Therefore, the MCRC algorithm reduces the number of opened RSUs and transfers requests to the opened RSUs. After this, the only way to bring down the opening cost is to open a smaller fraction of RSUs. This introduces an extra opening cost in the rounding step of the MCRC algorithm.

The second two regions correspond to load balancing. As the request size increases, so does the service cost. However, it can be seen in Figures 6 and 7 that this increase is not linear. This comes from the fact that vehicles have a capacity limitation since they are equipped with single-radio transceivers and cannot receive more than one unit of data at a time. Therefore, requests with the size of more than one unit of data must be received in different time slots. If some parts of the request cannot be transferred to the next RSU, it is more likely that they will be served when the vehicle is farther from the RSU. Since the communication cost is nonlinear, the service cost increases at a nonlinear rate. Also, the increase in the request size requires more network capacity and as a result, both algorithms open more RSUs. The MCRC algorithm opens more RSUs to moderate the service cost increase. In Figure 7, the MCRC algorithm also switches to solar-powered RSUs to take advantage of their low service cost.

In terms of request drop ratio, the MCRC algorithm shows better performance as before. In both algorithms, the request drop ratio increases rapidly with the request size. Even after a request size of 8, the lower bound and offline LP drops requests.

3) The Effect of the Request Arrival Rate: In this section, the effect of the request arrival rate on the MCRC algorithm performance is investigated. For the per RSU capital cost and the vehicle arrival rate, the same values are selected as in section V-2. Figures 8 and 9 show the results of the single-choice and multiple-choice RSU placements, respectively.

To evaluate the MCRC algorithm, we increase the rate by which vehicles generate their requests from 0.0025 requests per time slot to 0.015. All vehicles have the same request arrival rate and each request has a size of 8. The results are similar to the results of Section V-2 and the same arguments apply here. There is only one significant difference, i.e., the rate that the service cost increases, as shown in Figures 8 and 9. The service cost increases almost linearly with the request arrival rate, since there is more flexibility for load balancing. In Section V-2, the request arrival rate was fixed

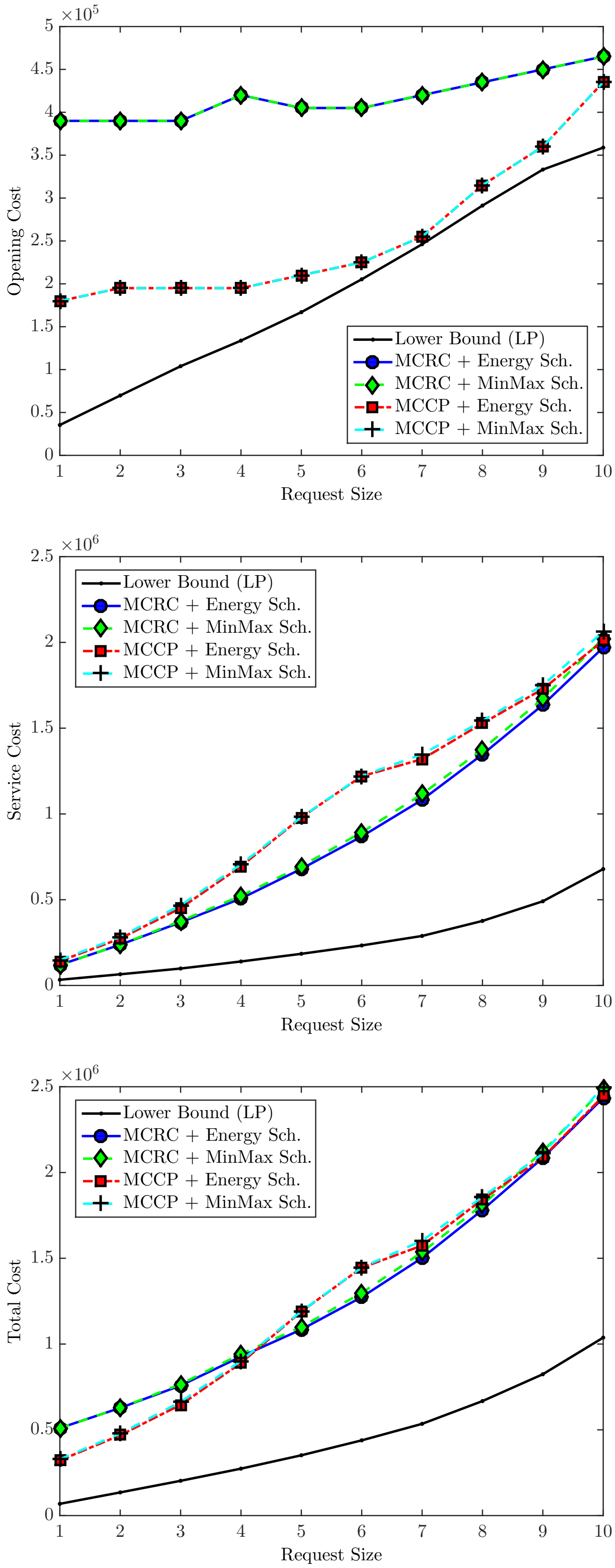

Fig. 6: The Effect of Request Size on Single-Choice RSU Placement with High Vehicle Traffic Load.

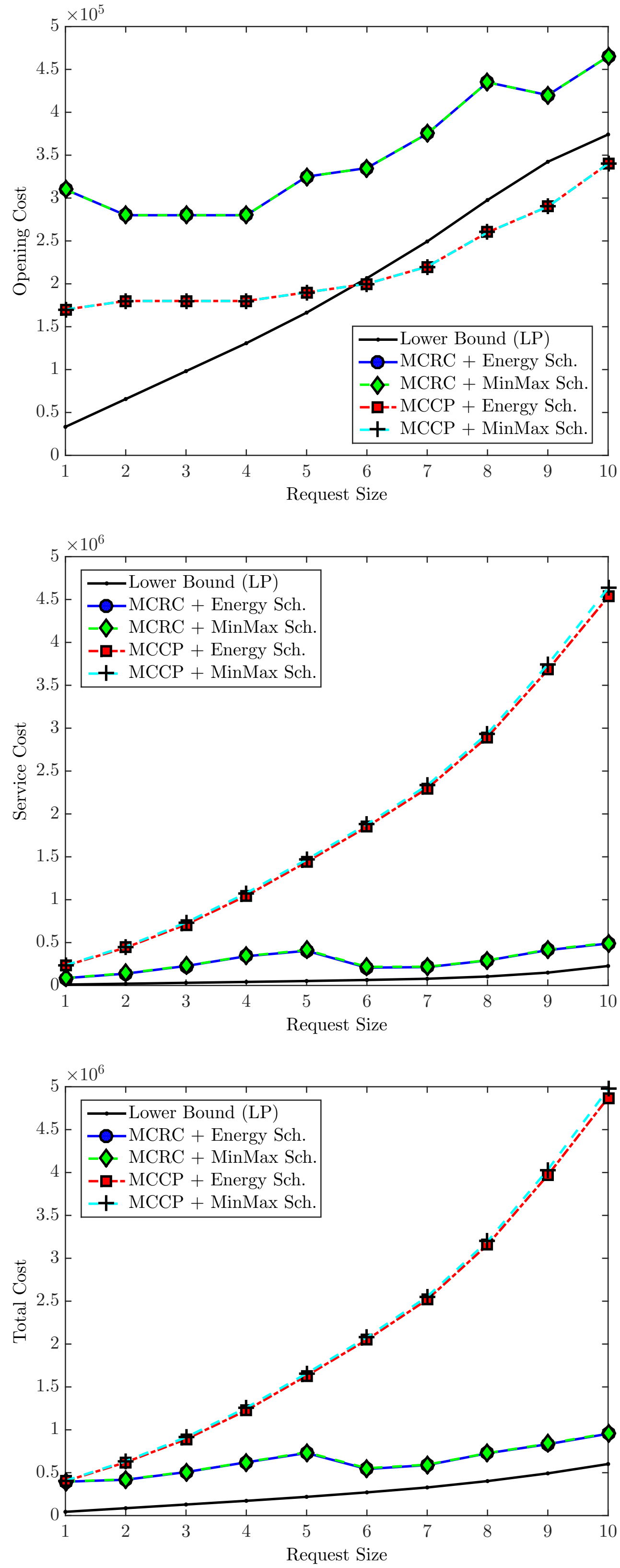

Fig. 7: The Effect of Request Size on Multiple-Choice RSU Placement with High Vehicle Traffic Load.

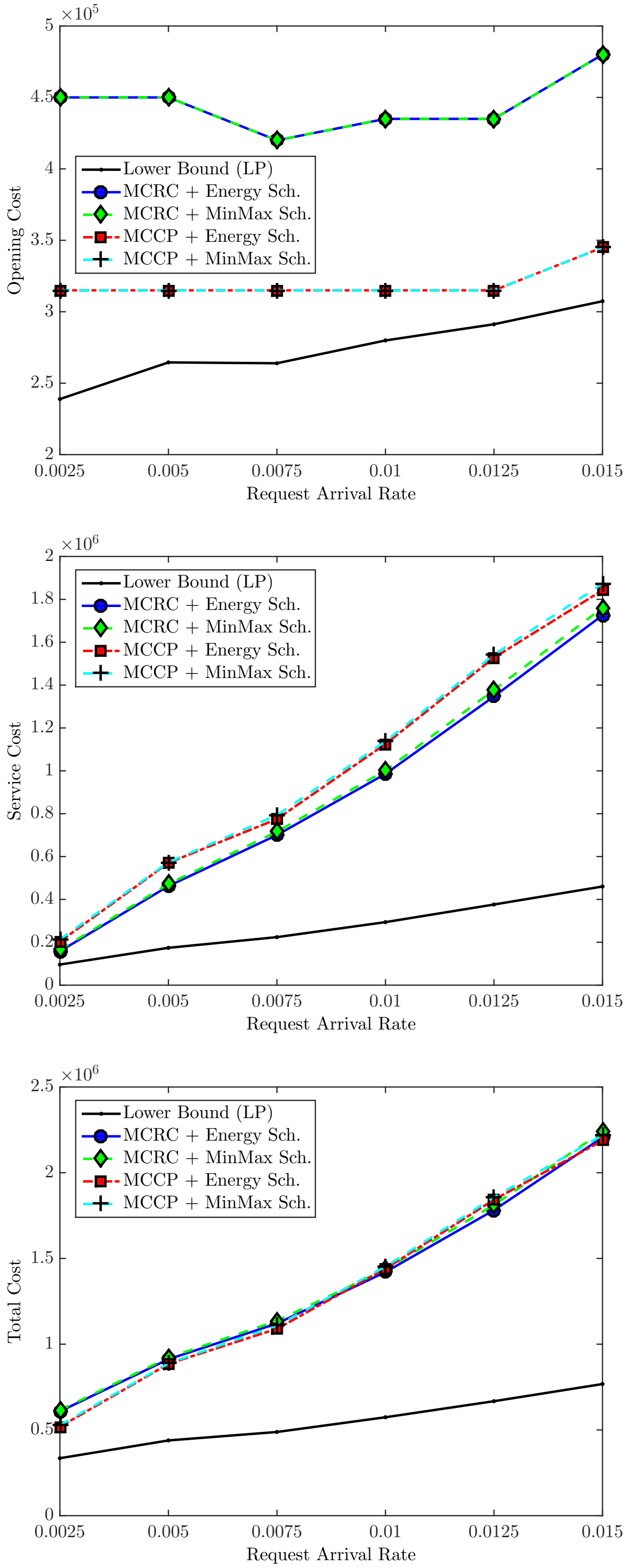

Fig. 8: The Effect of Request Arrival Rate on Single-Choice RSU Placement with High Vehicle Traffic Load.

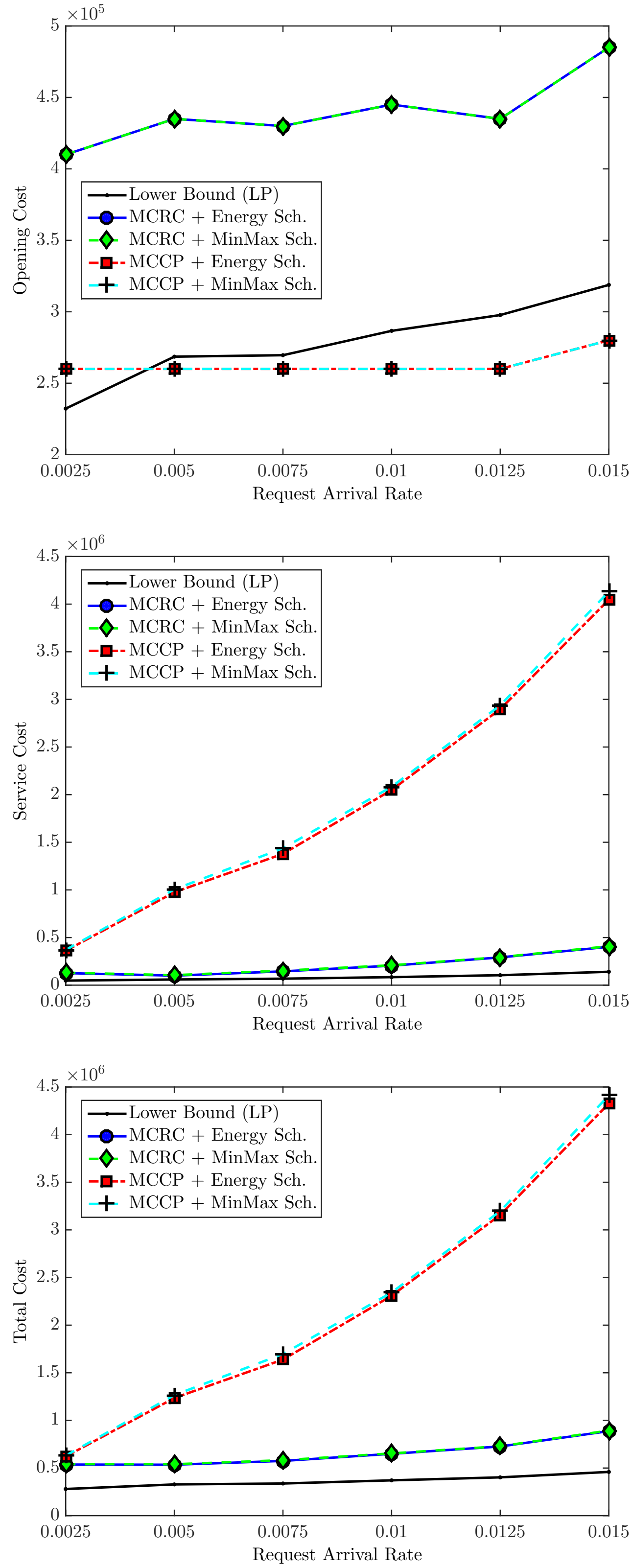

Fig. 9: The Effect of Request Arrival Rate on Multiple-Choice RSU Placement with High Vehicle Traffic Load.

and we increased the size of the requests. However, in this case we fix the request size and increase the request arrival rate.

4) The Effect of the Request Time-to-Live (Deadline): The effect of the request time-to-live (TTL) on the MCRC algorithm is evaluated in this section. These results are presented in Figures 10 and 11. The single RSU capital cost factor is 10, the vehicle arrival rate is 2 per time slot, the request arrival rate is 0.0125 requests per time slot, and the request size is 8.

The request TTL is the time duration that a request is valid and can be served. In this section, we change the request TTL from 20 time slots to 160. It can be seen from Figures 10 and 11 that as the request TTL increases, the RSU deployment cost, mainly because of the service cost, decreases. In the lowest TTL value, both algorithms in both the single-choice and multiple-choice RSU placement case, open RSUs at all candidate site locations. This is because the short TTL does not allow any request transfer between RSUs for the purpose of load concentration. In this case, the request TTL is shorter than the travelling time of a vehicle inside the coverage area of an RSU. However, as requests become more delay tolerant, the MCRC algorithm transfers more requests between RSUs, so that it can both reduce the number of opened RSUs and also serve more requests at their energy favourable positions. As a result, the opening cost and the service cost, and consequently, the total cost of RSU deployment decreases. Similar to the previous sections, the MCRC algorithm shows its advantage in the multiple-choice RSU placement by selecting more solar-powered RSUs over grid-powered RSUs.

5) The Effect of the Vehicle Arrival Rate: The effect of the vehicle arrival rate on the RSU placement algorithms was briefly discussed in V-1. In this section, we further evaluate this effect. The single RSU capital cost factor is 10, the vehicle arrival rate is 2 per time slot, the request arrival rate is 0.0125 requests per time slot, the request size is 8, and the request TTL is 40 time slots. Figures 12 and 13 show the results for the single-choice and multiple-choice RSU placements, respectively. The vehicle arrival rate is changed from 0.5 vehicles per time slot to 2.5 vehicles per time slot.

It can be seen in Figures 12 and 13 that the MCRC algorithm is as good as MCCP in single-choice RSU placement, but not in the multiple-choice case. Similar to the previous results, when the data traffic load is low, the integrality gap between the fractional solution and the rounded solution pushes the opening cost of the MCRC algorithm to higher values. However, as the vehicle arrival rate increases, it balances the load between more RSUs. Specially, in Figure 13, as the service cost increases, the MCRC algorithm switches to more solar-powered RSUs, which brings down the service cost and consequently, improves the total cost of the deployment. The results also show that the request drop ratio increases rapidly after the vehicle arrival rate of 1.5 vehicles per time slot. Also, after 2 vehicles per time slot, the offline LP starts to drop more requests. In these results, the MCRC algorithm has a smaller request drop ratio than the MCCP algorithm.

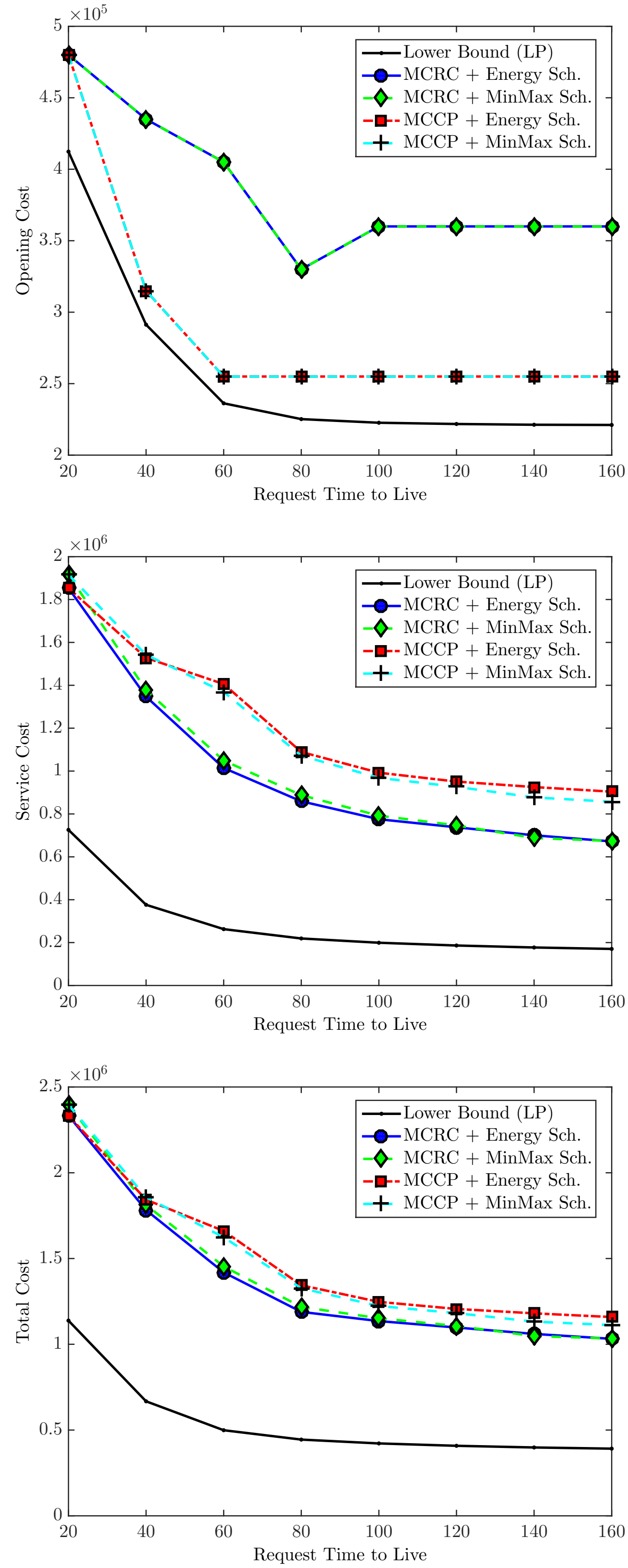

Fig. 10: The Effect of Request Time-to-Live on Single-Choice RSU Placement with High Vehicle Traffic Load.

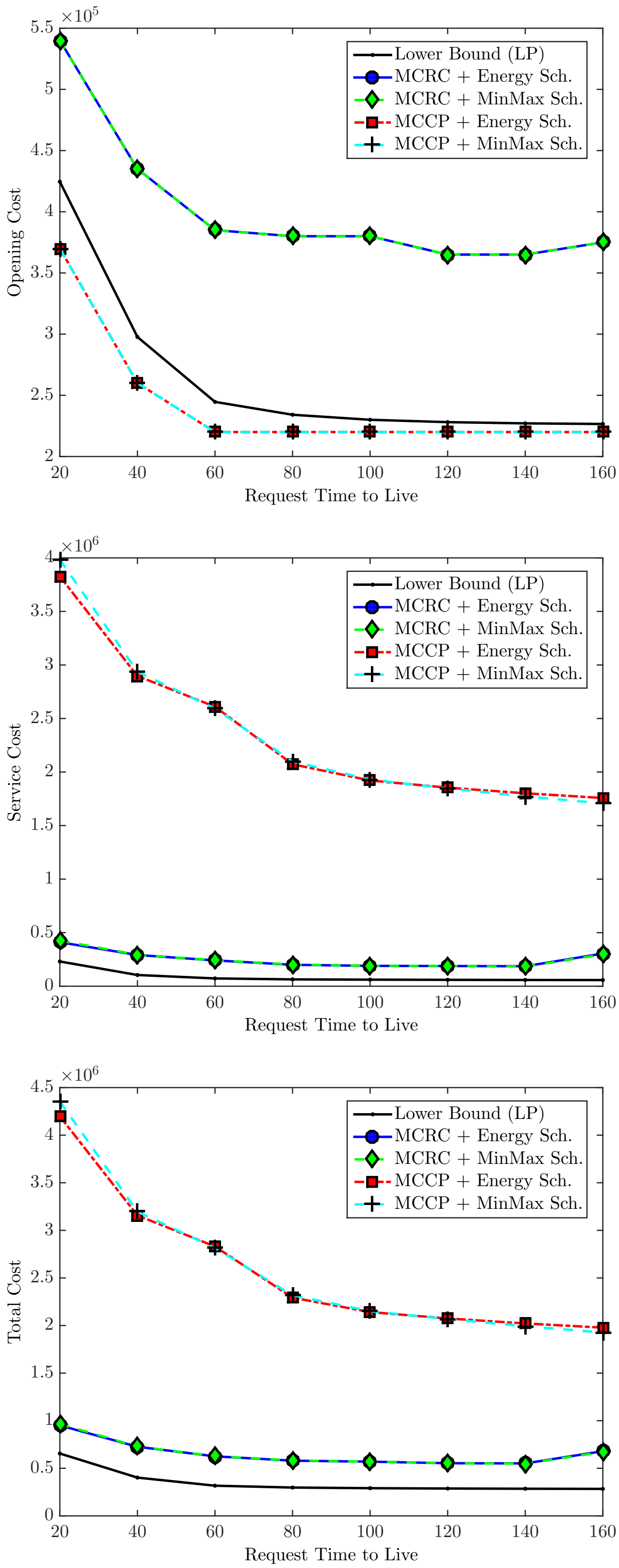

Fig. 11: The Effect of Request Time-to-Live on Multiple-Choice RSU Placement with High Vehicle Traffic Load.

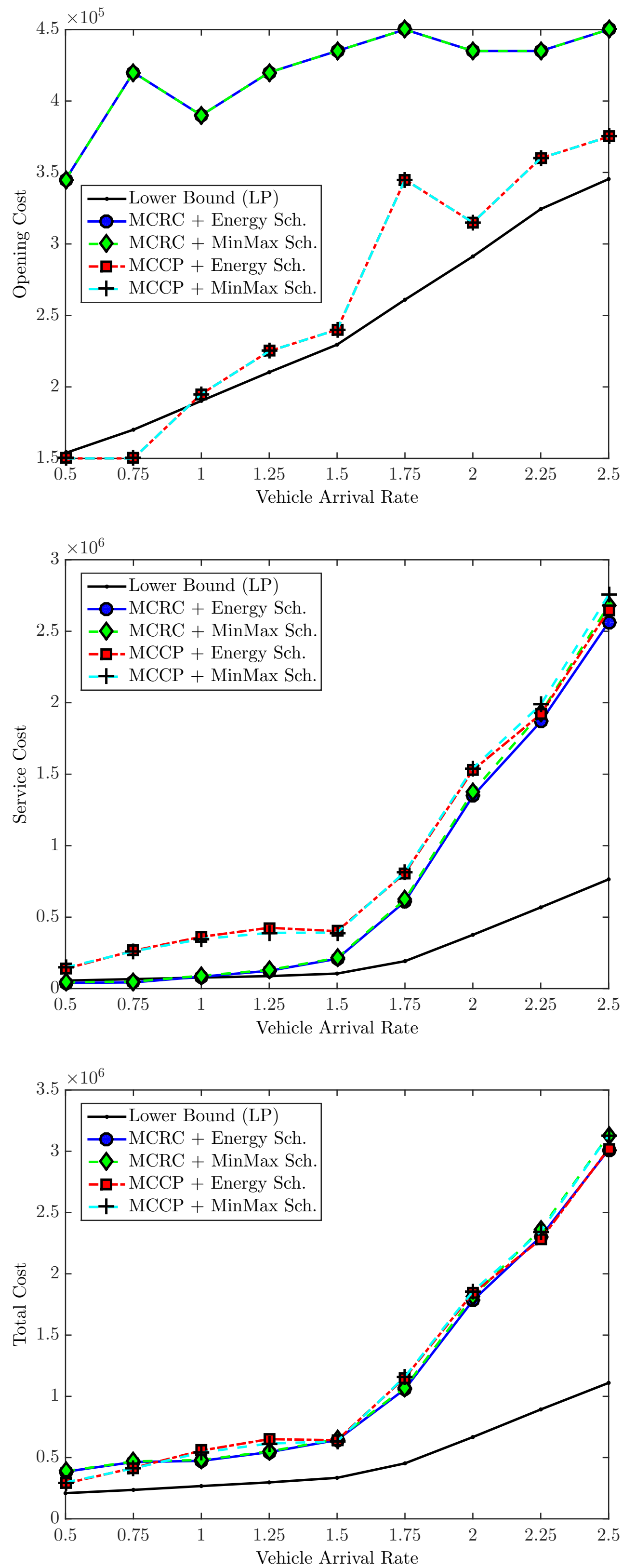

Fig. 12: The Effect of Vehicle Arrival Rate on Single-Choice RSU Placement.

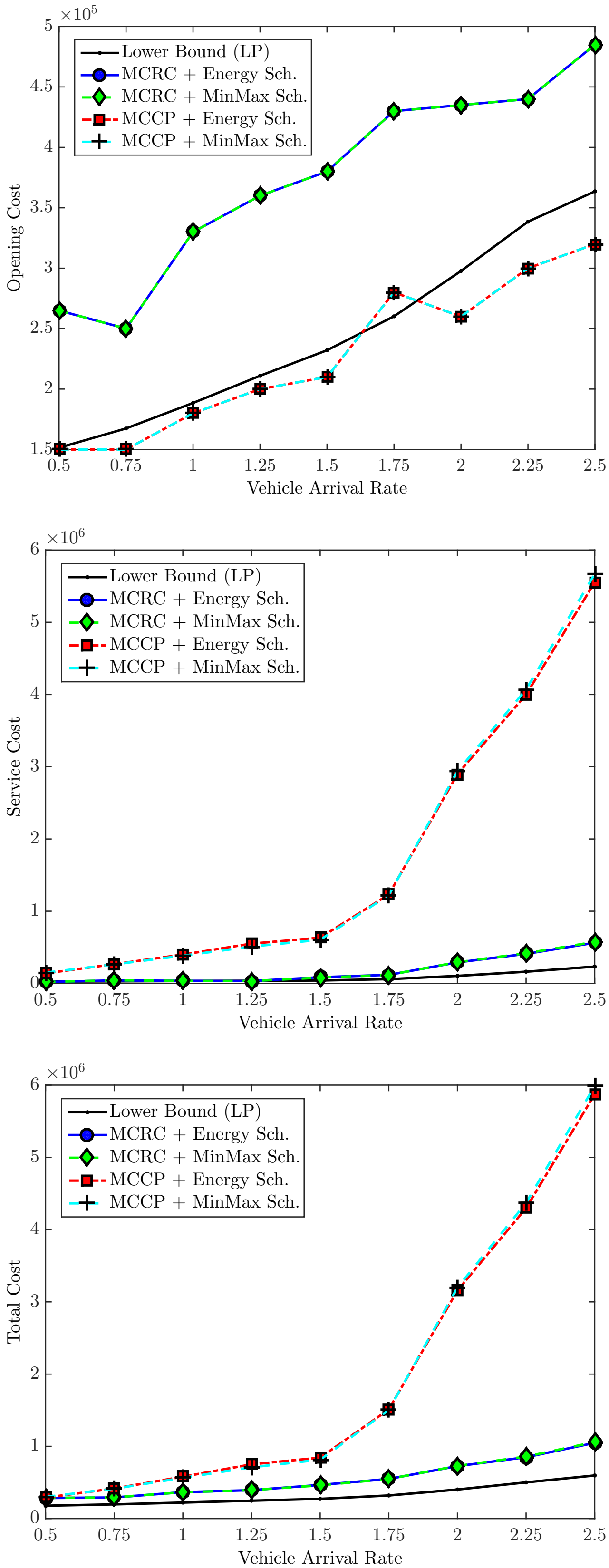

Fig. 13: The Effect of Vehicle Arrival Rate on Multiple-Choice RSU Placement.

VI. Conclusions

In this paper we have considered the problem of road-side unit (RSU) placement so that the sum of installation and operating costs is minimized. In this type of system, the total cost is a function of both capital expenditure (CAPEX) installation/opening costs, and long-term energy operating (OPEX) costs. An integer linear program (ILP) was first formulated that gives the minimum cost placement using a given set of inputs. This was used as a lower bound on total cost and is attainable for small problem sizes where the solution complexity is reasonable. To address larger problems, an algorithm was then proposed that solves a relaxed version of the ILP and uses a novel rounding procedure to obtain RSU placements, referred to as Minimum Cost Route Clustering (MCRC). The placement decisions take into account the service costs associated with the energy used to operate the RSUs, which is done using a minimum energy online scheduler.

The performance of MCRC algorithm was investigated in different scenarios, where per RSU capital cost, request parameters (such as arrival rate, size, and time-to-live), and vehicle arrival rate were changed. MCRC was compared the Minimum Capital Cost Placement (MCCP) algorithm that generates RSU placements that only minimize capital costs. As was discussed in Section V, MCRC outperforms MCCP in RSU deployment cost and it has less request drop ratio. In contrast to the MCCP algorithm, which is insensitive to the service cost, the MCRC algorithm creates a balance between the opening and service cost components. As a result, for different per RSU capital costs, MCRC outperforms MCCP through load concentration and load balancing, whichever is more appropriate. Since MCRC is an LP-based approximation algorithm, for very expensive per RSU capital costs, it looses its advantages to MCCP because of the high integrality gap. As we increase the size of the vehicle requests, the MCRC algorithm shows a slight advantage over the MCCP algorithm. However, in multiple-choice RSU placement, the MCRC algorithm significantly outperforms MCCP. Similar results were found as the request arrival rate and vehicle arrival rate were increased. By increasing the request time-to-live (i.e., extending the request deadline), the RSU deployment cost decreases through better load concentration. The MCRC algorithm also shows its advantage in the multiple-choice RSU placement case by selecting more solar-powered RSUs over grid-powered RSUs, where appropriate.

References

[1] C. Campolo and A. Molinaro, “Multichannel Communications in Vehicular Ad Hoc Networks: A Survey,” *Communications Magazine, IEEE*, vol. 51, no. 5, pp. 158–169, May 2013.

[2] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. Mark, “Connected Vehicles: Solutions and Challenges,” *Internet of Things Journal, IEEE*, vol. 1, no. 4, pp. 289–299, Aug 2014.

[3] T.-J. Wu, W. Liao, and C.-J. Chang, “A Cost-Effective Strategy for Road-Side Unit Placement in Vehicular Networks,” *Communications, IEEE Transactions on*, vol. 60, no. 8, pp. 2295–2303, August 2012.

[4] A. Abdrabou and W. Zhuang, “Probabilistic Delay Control and Road Side Unit Placement for Vehicular Ad Hoc Networks with Disrupted Connectivity,” *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 1, pp. 129–139, January 2011.

[5] U.S. Federal Communications Commission, "Report and Order FCC 03-324," Washington, D.C. 20554, February 2004.

[6] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A Comprehensive Survey on Vehicular Ad Hoc Network," *Journal of Network and Computer Applications*, vol. 37, pp. 380–392, January 2014.

[7] J. A. Volpe National Transportation Systems Center, "Vehicle-Infrastructure Integration (VII) Initiative Benefit-Cost Analysis, Version 2.3," Intelligent Transportation Systems Joint Program Office, United States Department of Transportation, Washington, DC, Tech. Rep., May 2008.

[8] A. Hammad, T. Todd, G. Karakostas, and D. Zhao, "Downlink traffic scheduling in green vehicular roadside infrastructure," *Vehicular Technology, IEEE Transactions on*, vol. 62, no. 3, pp. 1289–1302, March 2013.

[9] A. Khezrian, T. Todd, G. Karakostas, and M. Azimifar, "Energy-Efficient Scheduling in Green Vehicular Infrastructure With Multiple Roadside Units," *Vehicular Technology, IEEE Transactions on*, vol. 64, no. 5, pp. 1942–1957, May 2015.

[10] R. Levi, D. B. Shmoys, and C. Swamy, "LP-based Approximation Algorithms for Capacitated Facility Location," *Mathematical Programming*, vol. 131, no. 1-2, pp. 365–379, 2012.

[11] M. Patra, S. Mishra, and C. Murthy, "An Analytic Hierarchy Process Based Approach for Optimal Road Side Unit Placement in Vehicular Ad Hoc Networks," in *Vehicular Technology Conference (VTC Spring), 2014 IEEE 79th*, May 2014, pp. 1–5.

[12] X. Liya, H. Chuanhe, L. Peng, and Z. Junyu, "A Randomized Algorithm for Roadside Units Placement in Vehicular Ad Hoc Network," in *Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on*, Dec 2013, pp. 193–197.

[13] B. Aslam, F. Amjad, and C. Zou, "Optimal Roadside Units Placement in Urban Areas for Vehicular Networks," in *Computers and Communications (ISCC), 2012 IEEE Symposium on*, July 2012, pp. 423–429.

[14] B. Aslam and C. Zou, "Optimal Roadside Units Placement along Highways," in *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, Jan 2011, pp. 814–815.

[15] J. Lee, "Design of a network coverage analyzer for roadside-to-vehicle telematics networks," in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD '08. Ninth ACIS International Conference on*, August 2008, pp. 201–205.

[16] C.-C. Lin and D.-J. Deng, "Optimal Two-Lane Placement for Hybrid VANET-Sensor Networks," *Industrial Electronics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, April 2015.

[17] R. Rizk, R. Daher, and A. Makkawi, "RSUs Placement Using Overlap Based Greedy Method for Urban and Rural Roads," in *Communication Technologies for Vehicles (Nets4Cars-Fall), 2014 7th International Workshop on*, Oct 2014, pp. 12–18.

[18] T. Kumrai, K. Ota, M. Dong, and P. Champrasert, "RSU Placement Optimization in Vehicular Participatory Sensing Networks," in *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on*, April 2014, pp. 207–208.

[19] A. Abdrabou, B. Liang, and W. Zhuang, "Delay Analysis for Sparse Vehicular Sensor Networks with Reliability Considerations," *Wireless Communications, IEEE Transactions on*, vol. 12, no. 9, pp. 4402–4413, September 2013.

[20] J. Chi, Y. Jo, H. Park, and S. Park, "Intersection-Priority based Optimal RSU Allocation for VANET," in *Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on*, July 2013, pp. 350–355.

[21] P. Patil and A. Gokhale, "Voronoi-based Placement of Road-side Units to Improve Dynamic Resource Management in Vehicular Ad Hoc Networks," in *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, May 2013, pp. 389–396.

[22] M. Rebai, L. Khoukhi, H. Snoussi, and F. Hnaien, "Optimal Placement in Hybrid VANETs-Sensors Networks," in *Wireless Advanced (WiAd), 2012*, June 2012, pp. 54–57.

[23] M. Rashidi, I. Batros, T. Madsen, M. Riaz, and T. Paulin, "Placement of Road Side Units for Floating Car Data Collection in Highway Scenario," in *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2012 4th International Congress on*, Oct 2012, pp. 114–118.

[24] S.-W. Wang and M.-Y. Chang, "Roadside Units Allocation Algorithms for Certificate Update in VANET Environments," in *Communications (APCC), 2011 17th Asia-Pacific Conference on*, Oct 2011, pp. 472–477.

[25] Y. Liang, H. Liu, and D. Rajan, "Optimal Placement and Configuration of Roadside Units in Vehicular Networks," in *Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th*, May 2012, pp. 1–6.

[26] D. Vageesh, M. Patra, and C. Murthy, "Joint Placement and Sleep Scheduling of Grid-Connected Solar Powered Road Side Units in Vehicular Networks," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on*, May 2014, pp. 534–540.

[27] F. Zou, J. Zhong, W. Wu, D.-Z. Du, and J. Lee, "Energy-Efficient Roadside Unit Scheduling for Maintaining Connectivity in Vehicle Ad-hoc Network," in *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication*, ser. ICUIMC '11. New York, NY, USA: ACM, 2011, pp. 64:1–64:8.

[28] G. Md Nawaz Ali, M. Salim Mollah, S. Samantha, and S. Mahmud, "An Efficient Cooperative Load Balancing Approach in RSU-based Vehicular Ad Hoc Networks (VANETs)," in *Control System, Computing and Engineering (ICCSCE), 2014 IEEE International Conference on*, Nov 2014, pp. 52–57.

[29] E. Cascetta, *Transportation Systems Analysis: Models and Applications*, 2nd ed., ser. Springer Optimization and Its Applications. Springer US, 2009, vol. 29.

[30] S. G. Kolliopoulos and Y. Moysoglou, "Integrality Gaps for Strengthened Linear Relaxations of Capacitated Facility Location," *Mathematical Programming*, pp. 1–43, 2015.

[31] G. Maia, L. Villas, A. Boukerche, A. Viana, A. Aquino, and A. Loureiro, "Data Dissemination in Urban Vehicular Ad hoc Networks with Diverse Traffic Conditions," in *Computers and Communications (ISCC), 2013 IEEE Symposium on*, July 2013, pp. 000 459–000 464.

[32] C. Dai, C.-Y. Chow, and J. Zhang, "Utilizing Road-side Infrastructure for Location-based Services in Vehicular Ad-hoc Networks," in *Communications and Networking in China (CHINACOM), 2013 8th International ICST Conference on*, Aug 2013, pp. 546–551.

[33] F. J. Martinez, C. K. Toh, J.-C. Cano, C. T. Calafate, and P. Manzoni, "A Survey and Comparative Study of Simulators for Vehicular Ad Hoc Networks (VANETs)," *Wirel. Commun. Mob. Comput.*, vol. 11, no. 7, pp. 813–828, Jul. 2011.

[34] J. Harri, F. Filali, and C. Bonnet, "Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy," *Communications Surveys Tutorials, IEEE*, vol. 11, no. 4, pp. 19–41, Fourth 2009.

[35] E. Spaho, L. Barolli, G. Mino, F. Xhafa, and V. Kolici, "VANET Simulators: A Survey on Mobility and Routing Protocols," in *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2011 International Conference on*, Oct 2011, pp. 1–10.

[36] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent Development and Applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.

[37] Z. Li, Y. Zhu, and Z. Li, "Practical Location-based Routing in Vehicular Ad Hoc Networks," in *Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on*, Oct 2009, pp. 900–905.

[38] E. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[39] J. Song, Y. Wu, Z. Xu, and X. Lin, "Research on Car-Following Model Based on SUMO," in *Advanced Infocomm Technology (ICAIT), 2014 IEEE 7th International Conference on*, Nov 2014, pp. 47–55.

[40] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, 2001.