# Quality of Service Optimization for Vehicular Edge Computing with Solar-Powered Road Side Units

Yu-Jen Ku, Po-Han Chiang and Sujit Dey
Mobile Systems Design Lab, Dept. of Electrical and Computer Engineering, University of California, San Diego
Email: {yuku, pochiang, dey} @ucsd.edu

*Abstract*— **This paper shows the viability of Solar-powered Road Side Units (SRSU), consisting of small cell base stations and Mobile Edge Computing (MEC) servers, and powered solely by solar panels with battery, to provide connected vehicles with a low-latency, easy-to-deploy and energy-efficient communication and edge computing infrastructure. However, SRSU may entail a high risk of power deficiency, leading to severe Quality of Service (QoS) loss due to spatial and temporal fluctuation of solar power generation. Meanwhile, the data traffic demand also varies with space and time. The mismatch between solar power generation and SRSU power consumption makes optimal use of solar power challenging. In this paper, we model the above problem with three sub-problems, the SRSU power consumption minimization problem, the temporal energy balancing problem and spatial energy balancing problem. Three algorithms are proposed to solve the above sub-problems, and they together provide a complete joint battery charging and user association control algorithm to minimize the QoS loss under delay constraint of the computing tasks. Results with a simulated urban environment using actual solar irradiance and vehicular traffic data demonstrates that the proposed solution reduces the QoS loss significantly compared to greedy approaches.**

***Keywords— Solar power generation, Road Side Units, User Association, Mobile Edge Computing, Quality of Service.***

## I. INTRODUCTION

Emerging connected vehicles will need to support different levels of assisted and autonomous driving, road safety, infotainment and collaboration services, with increasingly high throughput and low latency computing and communication needs. There is significant work in progress to ensure high throughput wireless connectivity between vehicles (V2V) as well as vehicle-to-infrastructure (V2I) using both traditional cellular licensed spectrum or ITS bands (e.g. ITS 5.9 GHz) [1]. Road-Side Units (RSU) are evolving to play an important role in providing infrastructure support to increase the range of communications as well as help provide various vehicular services. To satisfy the massive growth in communication demands, in particular in urban areas, dense deployment of small cell base stations (SBS) [2] is expected, which can also function as RSUs to satisfy the high throughput requirements of emerging vehicular applications. Furthermore, the small cell-based RSUs can be supplemented with Mobile Edge Computing (MEC), allowing opportunistic use of MEC resources for growing vehicular computing needs while still satisfying low latency requirements. While the use of RSUs consisting of SBS and MEC will make high throughput and low latency emerging vehicular applications viable, it is important to take into consideration the energy consumption and sustainability of the vehicular wireless infrastructure. From [3], it is estimated that the carbon dioxide equivalent ($CO_{2e}$) and the total energy consumption of cellular networks globally will escalate to 235 million tons and 120TWh per year by 2020. Although the power consumption of SBS is 100x to 1000x less than the macro base station (BS) [4], the dense deployment of massive SBSs will still make the accumulated power consumption beyond that consumed by macro BSs. The past few years have seen growing research on cellular networks powered by renewable energy, in particular solar energy [3]. While the power generation rate of solar panels is not sufficient to be the sole source of power for macro BSs, in this paper we show that it is sufficient to power an SBS with a reasonably sized solar panel of a few square meters square. Hence, we propose the use of Solar-powered Road Side Units (SRSU), consisting of SBS, MEC, solar panel and battery. SRSUs will not only achieve reduced power consumption and provide clean wireless and edge computing infrastructure but enable quick and on-demand deployments as needed in urban areas.

A critical challenge of adopting solar energy in a wireless network is their intermittent and fluctuating nature. From the solar irradiance measurement in [5], solar generation varies significantly on location and time. On the other hand, the vehicular data traffic profile also varies with different time and location, which together with the intermittency of solar generation, may lead to mismatches between solar generation and SRSU power demands.

The mismatch between power generation and consumption may lead to severe Quality of Service (QoS) loss, leading to service disruptions for the vehicular applications. We expect each vehicle will continuously upload the information captured by its sensors, including images and video segments recorded by its cameras, to its connected SRSU. The MEC node in each SRSU need to process the received information and form contextual data. The contextual data should be computed and transmitted back to the vehicle with a delay constraint to ensure driving safety. To accomplish the whole process, the SRSU should allocate to each vehicle sufficient uplink, downlink, and computational resources. When the power demand of SRSU cannot be fulfilled by the solar generation or stored energy in the battery, it will need to rearrange its computing and communication resource allocation, so its power consumption can be reduced, consequently adversely affecting the QoS experienced by the served vehicles due to delay constraint violations. In the worst case, when there is no solar power

978-1-5386-5156-8/18/$31.00 ©2018 IEEE

generated and the SRSU battery is fully discharged, the connected vehicles cannot be served at all, leading to service outage.

We model the above challenges as a QoS loss minimization problem by mitigating the temporal and spatial mismatch of the solar power generation profile and SRSU power consumption through battery charging/discharging management and vehicle association. We break down the problem into three sub-problems: 1) Minimizing SRSU Power Consumption problem (MPC) given the data traffic demand, 2) Temporal Energy Balancing problem (TEB) as temporal allocation of solar energy to match the profile of solar generation and power consumption for individual SRSU, and 3) Spatial Energy Balancing (SEB) problem to balance solar energy among multiple SRSUs. Then, we propose the QoS Loss Minimization (QLM) algorithm, a joint solar energy storage and battery charging, user association and SRSU resource allocation mechanism comprising three algorithms solving the above three sub-problems respectively.

### A. Related Work

Various relevant recent works that address the use of renewable energy to minimize grid energy in wireless cellular communications. In [6], the authors optimally adjust the cell size and schedule daily solar energy of BSs to minimize grid-power consumption in a solar-powered wireless networks. In [6], the traffic load is balanced by modifying the transmit power of BSs while our work manages user association without requiring to change cell size. In [7], the authors propose a Lyapunov optimization framework to adapt the BS resource allocation and battery operation to minimize grid power consumption. However, the objective in [7] is to minimize the grid-power consumption while our work is to minimize the QoS loss with RSUs powered solely by solar energy.

The authors in [8] minimize the SBS power outage probability by proposing a power availability oriented user association strategy under transmission rate constraint. In [9], the authors proposed to minimize the overall network latency under limited solar availability by downlink power control and user association management. The above research considers only downlink transmission while we address the problem of both uplink and downlink transmissions and computing resource in order to facilitate vehicles offloading their computing to the SRSU.

In [10], the authors address the problem of minimizing the execution delay and workload failure in a single MEC-enabled BS-user link powered by solar energy. They make online task offloading and transmit power decision on the user side under delay and energy constraints. The authors in [11] focus on minimizing the long-term system cost, including execution delay, downlink transmission delay, battery depreciation and backup diesel power consumption, of a solar-powered MEC-enabled single-BS. They propose a learning-based dynamic workload offloading and MEC server autoscaling strategy to solve the above problem.

The authors in [12] extends the work in [11] with a multi-RSU cellular network. In the network, the workload can be offloaded between MEC servers located in different SBSs. Although the SBSs in the network is connected to grid power, they apply energy budget constraints to each SBS. Regulated by the energy budget, the problem is to minimize the overall system delay due to computation and downlink transmission.

Unlike [10] and [11] which consider only a point-to-point MEC-enabled link, we propose a joint BS resource allocation and user association technique among multiple SBSs. Although [12] considers multiple SBSs, it only applies to the scenario that the workload can be divided into arbitrary portions, and allocated simultaneously to multiple SBSs. In our work, we consider a more practical scenario that the entire workload of the user can only be executed at the associated SBS. Moreover, the long-term energy budget constraint used in [12] cannot efficiently capture the nature of high intermittency of solar power generation.

The rest of the paper is organized as follows. Section II elaborates the system models used, including workload, MEC server, channel, power consumption and battery model.The problem we are addressing is formulated in Section III. Section IV describes the QLM algorithm, our proposed joint solar power-aware SBS energy storage and inter-SBS user association algorithm along with SBS power minimization. We present the simulation results in Section V, and conclude in Section VI.

## II. System Model

### A. Network Model

Consider a set of $N$ SRSUs $\mathcal{B} = \{1,2,\dots,N\}$ along a road $R_1$ and a set of vehicular users (UEs) $I = \{1,2,\dots,\mathrm{I}\}$. Each SRSU $b$ has a SBS and a MEC, each of which we will also refer to as $b^{th}$ SBS and MEC respectively. Each SRSU is powered solely by a solar panel and equipped with a battery. The maximum capacity of the MEC processor in SRSU $b$ is denoted as $U_b$ megabits per second (MIPS), and the maximum bandwidth offered by associated SBS $b$ for downlink and uplink transmission any time are $W_{b,D}$ and $W_{b,U}$, respectively. We divide the duration of time equally into $T$ time slots, each time slot has duration $\tau$.

### B. Workload Model and SBS utilization

At each time slot, we assume that a group of UE will pass through the endpoints of each roads $R_r, r \in \{1,2,\dots,6\}$ with predetermined travel routes and speed, entering the network following a Poisson process with arrival rate $\lambda_r$, while another group of UEs will pass through these endpoints and leave the network. The location and speed of the $i^{th}$ UE at time slot $t$ is denoted by $x_i^t$ and $v_i^t$ respectively. $x_i^t$ and $v_i^t$ of each UE over $T$ time slots are assumed to be known at the start of $1^{st}$. This assumption is valid given that the traffic load difference of a single base station between two consecutive days are limited [13] and by the approach in [14] [15], routine UE movements can be known in advance under negligible prediction error.

Let $a_{bi}^t = \{0,1\}$ be the UE association indicator, where $a_{bi}^t = 1$ if the $i^{th}$ UE is connected to $b^{th}$ SBS and $a_{bi}^t = 0$ otherwise. Without loss of generality, we assume UE will initially connect to the SBS that provides the highest Received Signal Strength Indication (RSSI) measurement.

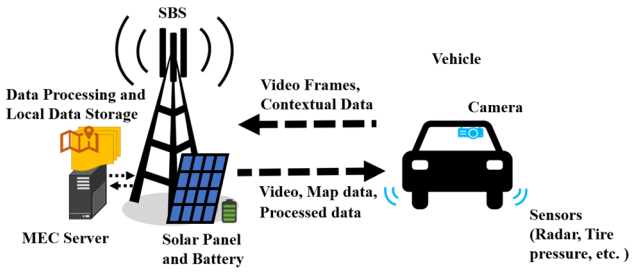


Figure 1. the workload and MEC server model

The workload generated by the $i^{th}$ UE at each time slot $t$ can be represented as $K_i^t = \{ s_i^t, c_i^t, \delta_i^t, \epsilon_i^t, d_i^t \}$, with the notations explained as the following. $s_i^t$ is the size of video and contextual sensor data generated at this time slot, which is uploaded to the MEC server. The data is uploaded once it is generated. we assume the data generation rate is a constant, so the uploading rate is no less than $s_i^t/\tau$. The MEC server will wait for a duration $\upsilon$ before starting the data processing. The duration $\upsilon$ is chosen as large as possible for the MEC server to collect as much data as possible for analysis. The computing resource required to process the data uploaded by the $i^{th}$ UE is denoted as $c_i^t$. The SBS will then transmit the processed information, which has size $\delta_i^t$, back to the $i^{th}$ UE. The delay requirement of the analysis process and downlink transmission is denoted as $d_i^t$. Since the analysis results are critical to driving safety, $d_i^t$ is very small compared to $\tau$ and we refer to this kind of data as *delay sensitive downlink data*. Note that this whole process needs to be finished in one time slot, therefore, the duration $\upsilon$ along with the computation and downlink delay should not be greater than $\tau$.

Furthermore, some of the UEs might concurrently download extra information from the MEC server or the Internet, for example, the map data or video frames captured from other UEs which are stored in the MEC server. The data size and delay constraint of these extra data are set to be $\epsilon_i^t$ and $\tau$ respectively. Since $\tau$ is much longer than $d_i^t$, we refer to these data as *delay tolerant downlink data*. The overall workload and MEC computing model is shown in Fig. 1.

Each $K_i^t$ will utilize a combination of computing and communication resources of the SRSU where it is offloaded. We assume that the MEC sever can concurrently serve multiple workloads from different UE using techniques such as Virtual Machine(VM) [16]. Let $u_{bi}^t$ be the CPU processor's capacity allocated to the $i^{th}$ UE by the $b^{th}$ MEC server; $w_{bi,U}^t$ be the uplink bandwidth allocated to the $i^{th}$ UE by the $b^{th}$ SBS respectively. For downlink transmission, we denote $w_{bi,DS}^t$ and $w_{bi,DT}^t$ as the downlink bandwidth allocated to the $i^{th}$ UE by the $b^{th}$ SBS for delay sensitive and delay tolerant data respectively.

Since the resource of a SBS is limited, we have the following computing and communication resource constraints [17]:

$$\sum_{i \epsilon \zeta^t(b)} u_{bi}^t \leq U_b, \qquad b \in B \tag{1}$$

$$\sum_{i \epsilon \zeta^t(b)} w_{bi,U}^t \leq W_{b,U}, \qquad b \in B \tag{2}$$

$$\sum_{i \epsilon \zeta^t(b)} \left( w_{bi,DS}^t + w_{bi,DT}^t \right) \leq W_{b,D}, \qquad b \in B \tag{3}$$

where $\zeta^t(b) \subset I$ is the set of UEs which are associate to the $b^{th}$ SBS at time $t$.

### C. Channel and Delay Model

We set both of the transmit power density of SBS and UE to be fixed and denoted them as $p_B$ and $p_I$ respectively. The downlink transmission rate from the $b^{th}$ SBS to the $i^{th}$ UE per 1 Hz is

$$r_{bi,D}^t = log_2(1 + \eta_{bi}^t), \tag{4}$$

where $\eta_{bi}^t = (p_B g_{bi}^t / N_0)$ is the signal-to-noise ratio (SNR) with $g_{bi}^t$ denotes the downlink channel gain and $N_0$ is the noise power density. We assume the inter-cell interference (ICI) from other SBSs can be ignored since our approach requires information of UE location and association condition to be shared among SBSs, in the meantime, SBSs can obtain the channel estimation of each UE and eliminate ICI by implementing Coordinated Multipoint (CoMP) [18] Similarly, the uplink transmission rate from $i^{th}$ UE to $b^{th}$ SBS per 1 Hz is given by

$$r_{bi,U}^t = log_2(1 + \frac{p_I g_{bi}^t}{N_0}). \tag{5}$$

For simplicity, we assume that there is no video buffer in UE sides, which means each video frame will be uploaded immediately after it is captured. The uploading transmission rate of the $i^{th}$ UE should be greater than the video size per time slot. Consequently, for each UE, the allocated uplink bandwidth should satisfy the constraint:

$$\sum_{b \in B} a_{bi}^t r_{bi,U}^t w_{bi,U}^t \tau \geq \sum_{b \in B} a_{bi}^t s_i^t, \quad i \epsilon I. \tag{6}$$

while the constraint of the allocated downlink bandwidth for delay tolerant transmission is:

$$\sum_{b \in B} a_{bi}^t r_{bi,D}^t w_{bi,DT}^t \tau \geq \sum_{b \in B} a_{bi}^t \epsilon_i^t, \quad i \epsilon I. \tag{7}$$

The overall delay of the delay sensitive downlink data can be expressed as the computation delay of offloaded application task processing plus the downlink transmission delay. To the $i^{th}$ UE, the constraint on the allocated processing speed and downlink bandwidth for delay sensitive data can be expressed as:

$$\sum_{b \in B} a_{bi}^t \left( \frac{c_i^t}{u_{bi}^t} + \frac{\delta_i^t}{r_{bi,D}^t w_{bi,DS}^t} \right) \leq \sum_{b \in B} a_{bi}^t d_i^t, \; i \epsilon I. \tag{8}$$

### D. Power Consumption Model

Since our work focuses on the solar power allocation strategies of SRSU, we will omit the energy consumption of UE in our model. The power consumption of the $b^{th}$ MEC server can be calculated by [16]

$$E_{b,S}^t = E_{S,IDLE} + \left(E_{S,MAX} - E_{S,IDLE}\right)\frac{\sum_{i\epsilon\zeta(b)} u_{bi}^t}{U_b}, \quad (9)$$

where $E_{max}$ is the power consumption when the server is fully-utilized and $E_{S,IDLE}$ as the power consumption when the server is idle.

Transmission power is the major factor for SRSU power consumption. The downlink power consumption of the $b^{th}$ SBS can be modeled as

$$E_{b,D}^t = p_d\left(\sum_{i\epsilon\zeta^t(b)} w_{bi,DS}^t \frac{\delta_i^t}{r_{bi,D}^t w_{bi,DS}^t} + \tau \sum_{i\epsilon\zeta^t(b)} w_{bi,DT}^t\right) \quad (10)$$

$$= p_d\left(\sum_{i\epsilon\zeta^t(b)} \frac{\delta_i^t}{r_{bi,D}^t} + \tau \sum_{i\epsilon\zeta^t(b)} w_{bi,DT}^t\right), \quad (11)$$

where $p_d$ is the overall downlink transmission power density: $p_d = \left(p_B + p_{c,D}\right)$ with $p_{c,D}$ is the circuit power density needed to realize a downlink transmission, for example, it can be the power consumption of modulation as well as channel encoding [19] [20]. Similarly, the power consumed for receiving uplink data from all the associated UEs will be

$$E_{b,U}^t = p_{c,U}\tau \sum_{i\epsilon\zeta^t(b)} w_{bi,U}^t. \quad (12)$$

Denoting $E_{b,IDLE}^t$ as the idle power of the SBS, the total power consumption of SRSU $b$ at time slot $t$ is therefore

$$E_b^t = E_{b,S}^t + E_{b,D}^t + E_{b,U}^t + E_{b,IDLE}. \quad (13)$$

### E. Solar Energy and Battery Model

Each SRSU is accompanied with a solar panel as energy harvesting module and a battery as energy storage module. At each time slot, the solar panel of the $b^{th}$ SRSU will generate $J_b^t$ joules of energy. Let $BAT_b^t$ be the $b^{th}$ SRSU battery level at slot $t$. $BAT_b^t$ is controlled by a battery charging strategy under zero loss on charging, discharging and depletion with the following constraint:

$$0 \le BAT_b^{t'} = \sum_{t=1}^{t'} J_b^t - \sum_{t=1}^{t'} E_b^t \quad (14)$$

where $1 \le t' \le T$, which ensures that the battery cannot be discharged after the stored energy has been exhausted.

Although the solar power generation fluctuates with time and location, our previous work has shown that it can be predicted several hours in advance with very high accuracy [21]. Therefore, we assume advanced knowledge of the solar power generated at SRSUs: $\{J_b^1, J_b^2, \dots, J_b^T\}\ \forall b \in \mathcal{B}$.

## III. Problem Formulation

Note that since the solar power is limited and changes temporally, SRSU might suffer power deficiency in some time slots. For the $b^{th}$ SRSU, the power deficiency happens at time slot $t$ when the energy drained from battery $BAT_b^t - BAT_b^{t-1}$ plus the generated solar energy $J_b^t$ is less than the SRSU power consumption $E_b^t$. Under this condition, the SRSU must reduce $E_b^t$ by either re-associating the UEs in $\zeta^t(b)$ to other SRSUs or dropping their service. If there exists any UE that cannot be re-associated to any SRSU at time slot *t*, the offloaded application of this UE in this time slot will be dropped, leading to a QoS loss since UE cannot offload its data processing application to SRSU. Such QoS loss is denoted by $C_{drop}^t$, which is defined to be equal to the total number of UEs experiencing this type of QoS loss at time slot *t*.

$$C_{drop}^t = \sum_{i=1}^{I}\left(1 - \sum_{b=1}^{B} a_{bi}^t\right). \quad (15)$$

Moreover, SRSU might hand over its UE to other SRSUs after the end of current time slot, making this UE fail to receive the processed information corresponding to the data that is uploaded after duration $\upsilon$. We refer to this information loss as a QoS loss from handover, $C_{handover}^t$, which is defined as the total number of UEs suffering such QoS loss at time slot *t* multiplied by a scaling factor $\kappa = 1 - \upsilon/\tau$. We set $C_{handover}^1 = 0$, and for $t \ge 2$,

$$C_{handover}^t = \kappa \sum_{i=1}^{I} a_{bi}^t\left(1 - \sum_{b=1}^{B} a_{bi}^t a_{bi}^{t-1}\right) \quad (16)$$

Given the solar power generated $J_b^t$ at each time slot over the whole day for each SRSU *b* and the data traffic demand profile $\{K_i^t, x_i^t, v_i^t\}$ for each UE, we focus on minimizing the QoS loss specified by (15) and (16). Our objective is to optimally determine the user association $A_i^t = \{a_{1i}^t, a_{2i}^t, \dots, a_{\mathrm{Bi}}^t\}, i\epsilon I$ and allocate solar energy to each time slot to minimize the overall average QoS loss during timeslots *T,* while satisfying the workload computation and transmission delay, communication resource, computing resource and solar generation constraints. The overall average QoS loss is defined as the overall QoS loss divided by the accumulated total number of UE in the network at each time slots. The problem can be formulated as:

$$\min_{A_i^t, i\epsilon I, 1\le t\le T} \frac{\sum_{t=1}^{T}\left(C_{drop}^t + C_{handover}^t\right)}{\sum_{t=1}^{T}|I|^t}. \quad (17)$$

*s.t.* (1) - (3), (6) - (8), (14).

where $|I|^t$ is the number of UEs in the network at time slot *t*.

Problem (17) is difficult to solve since SRSU can generate different solar power and experience distinct data traffic demand at different time slot. Moreover, different SRSU will have diverse solar power generation and data traffic at the same time slot. Therefore, we propose to heuristically break down (17) into 3 sub-problems: 1) Minimizing SRSU Power Consumption (MPC) problem aims to minimize single SRSU power

consumption given the workload and provides a feasible SRSU computing and communication resource allocation, 2) Temporal Energy Balancing (TEB) problem addresses the problem of mitigating the mismatch between energy generation and power consumption over time for individual SRSU, and 3) Spatial Energy Balancing (SEB) problem is formulated to balance the workload traffic and solar power among all the *N* SRSUs. In the following paragraphs, we will explain these three sub-problems in detail.

*1) MPC Problem:* Subject to the limited power supply of solar panel, SRSU should minimize its power consumption $E_b^t$ given its associated UE set $\zeta^t(b)$ at each time slot $t$ by optimally allocating the communication and computing resources.

Therefore, we formulate MPC as

$$\min_{\{w_{bi,DS}^t, w_{bi,DT}^t, u_{bi}^t, w_{bi,U}^t\}, i\epsilon\, \zeta^t(b)} E_b^t \tag{18}$$

$$s.t.\ \ (1)\text{ - }(3), (6)\text{ - }(8).$$

The allocation of processing power and channel bandwidth should satisfy constraints (1) - (3) to ensure the allocated SRSU resource will not exceed the resource limitation of the SBS. Meanwhile, the allocation of SRSU communication and computing resources satisfy constraints (6) - (8) to avoid violating the both the data traffic transmission and UE application task computation delay requirements.

*2) TEB Problem:* Let $\{M_b^1, M_b^2, \dots, M_b^T\}$ be the solar energy allocated to the b$^{\text{th}}$ SRSU over the *T* timeslots. Since SRSU allocates solar energy to any time slot through battery charging and dischargin strategies, the allocation strategy should satisfy battery constraint, in other words, $M_b^t = BAT_b^{t-1} - BAT_b^t + J_b^t$. The objective of TEB is to minimize the mismatch between allocated solarenergy $M_b^t$ and power consumption $E_b^t$ for all time slots. Defining the allocate-usage-power-ratio (AUPR) to be $\pi_b^t = M_b^t/E_b^t$ , the problem can be expressed as

$$\max_{\{M_b^1, M_b^2, \dots, M_b^T\}} \min_{1\le t\le T} \pi_b^t \tag{19}$$

$$s.t.\ \ M_b^t \ge E_{s,IDLE} + E_{b,IDLE}, 1 \le t \le T \tag{20}$$
$$(14).$$

The constraint on $M_b^t$ in (20) guarantees that SRSU will never be shut down because of the lack of generated solar power and fully discharged battery.

*3) SEB Problem:* Due to the spatial diversity of data traffic profile, some SRSUs might experience high workload traffic when its allocated solar power is low. Conversely, due to the spatial diversity of solar power generation, some SRSUs may have more solar power generation than its need, resulting in solar power surplus. Note that if SRSU is under power deficiency, namely $\pi_b^t \le 1$, it forces SRSU to reduce its power consumption $E_b^t$ by dropping or re-associating its connected UE to the feasible SRSU satisfyiny constraints (14) (20). When UE is offloaded to a new SRSU by changing the user association, the workload is re-directed to this new host. This provides us the oppurtunity to reduce the QoS loss. Therefore, SEB is formulated as a problem in (21), which is to achieve the minimum QoS loss by maximizing the possible workload balancing across the SRSUs for each time slot.

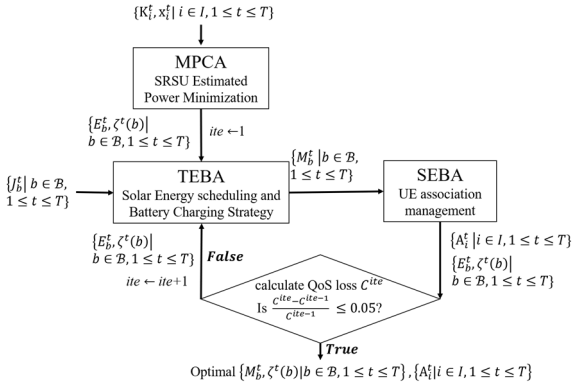

Figure 2. Breakdown of the QLM algorithm

$$\min_{\{A_1^t, A_2^t, \dots, A_I^t\}} C_{drop}^t + \kappa C_{handover}^t \tag{21}$$

$$s.t.\ \pi_b^t \ge 1, b \in \mathcal{B} \tag{22}$$

$$\sum_{b\in\mathcal{B}} a_{bi}^t \eta_{bi}^t \ge \sum_{b\in\mathcal{B}} a_{bi}^t \eta_{th}. \tag{23}$$

$$(1)\sim(3), (6)\sim(8).$$

The first constraint (22) requires every SRSU has its allocated solar power higher than its power consumption. Constraint (23) states that the SNR between $i\hat{}th$ UE and the SRSU it is re-associated to should be greater than a threshold $\eta_{th}$ to satisfy the minimum RSSI requirement of the SBS network. Constraints (1) to (3) ensure that SRSU's communication and computing resource allocation after the change of UE association still satisfies its resource limit. Moreover, for the associated UE, the delay constraints of the workload should never be violated, as stated in constraints (6) to (8).

Since problem (17) is divided into three sub-problems, in the next section, we will first present our proposed solution to these three sub-problems and then introduce an optimal joint solar energy storage and user association technique by adopting these three solutions.

## IV. Algorithm Design

In this section we introduce the QoS Loss Minimization (QLM) algorithm to solve (17). As shown in Fig. 2, QLM leverages the solution of MPC, TEB and SEB problems. At the beginning of every time slot *t*, each UE is associated with the SBS which provides the best SNR $\eta_{bi}^t$. Each SRSU will first minimize its power consumption and find the resulting feasible user set $\zeta^t(b)$. The solution of MPC will optimally allocate the computation and communication resource to the associated UE for each SRSU to minimize the power consumption. Under constraints (1) to (3) SRSU will need to drop some UEs if there is no available communication or computing resources.

After the energy consumption of each SRSU at each time slot *t* has been estimated by MPCA, the Temporal Energy Balancing Algorithm (TEBA) is proposed to solve TEB problem, which decides how to schedule the generated solar over *T* time slots for each SRSU. Based on the result of the TEBA, the

Temporal Energy Balancing Algorithm (TEBA)

**Input:** $\{E_b^t, J_b^t \mid 1 \le t \le T, b \in \mathcal{B}\}$

**Output:** $\{M_b^t \mid 1 \le t \le T, b \in \mathcal{B}\}$

Initialize $M_b^t \leftarrow J_b^t \; \forall t, \forall b$, $E \leftarrow E_{s,IDLE} + E_{b,IDLE}$;
**for** $b = 1$ *to* $N$ **do**
  **for** t $= T - 1$ *to* 1 **do**
    **for** t$' = t + 1$ *to* $T$ **do**
      $\gamma_b^{t'} = \max(\bar{\rho} E_b^{t'} - M_b^{t'}, 0)$;
      $\theta_b^{t'} = \max(E - M_b^{t'}, 0)$;
    **end for**
    calculate $\gamma_{sum} = \sum_{i=t}^{T} \gamma_b^{t'}$, $\theta_{sum} = \sum_{i=t}^{T} \theta_b^{t'}$;
    calculate $\overline{\rho_b^t}$, $\pi_b^t$;
    **If** $\theta_{sum} > M_b^t - E$
      calculate $\theta_{spare} = \max(M_b^t - \mathrm{E}, 0)$;
      calculate $\gamma_{spare} = 0$;
    **else**
      calculate $\theta_{spare} = \theta_{sum}$;
      calculate $\gamma_{spare} = \max(M_b^t - \theta_{spare} - \bar{\rho} E_b^t, 0)$;
    endif
    **for** t$' = t + 1$ *to* $T$ **do**
      **If** $\gamma_{sum} \neq 0$: $M_b^{t'} = M_b^{t'} + \gamma_{spare} \frac{\gamma_b^{t'}}{\gamma_{sum}}$; **endif**
      **If** $\theta_{sum} \neq 0$: $M_b^{t'} = M_b^{t'} + \theta_{spare} \frac{\theta_b^{t'}}{\theta_{sum}}$; **endif**
    **end for**
    $M_b^t = M_b^t - \varepsilon_{spare}$;
  **end for**
**end for**

Figure 4. TEBA algorithm

Spatial Energy Balancing Algorithm (SEBA) will balance the offloaded UE application among all SRSUs considering the ratio of allocated green energy and power consumption by UE offloading technique to minimize number of UEs experiencing offloaded application QoS loss by dropping or hand over. Since the result of SEBA will change SRSU's estimated power consumption, and hence affects the result of TEBA, the TEBA and SEBA will be iteratively executed until the optimum is achieved.

### *A. The MPCA algorithm*

Note that $E_b^t$ in (13) is a linear combination of resource allocation indicators $w_{bi,DS}^t, w_{bi,DT}^t, u_{bi}^t, w_{bi,U}^t, i\epsilon\, \zeta^t(b)$. Since the constraints on $w_{bi,DT}^t$ and $w_{bi,U}^t$ are independent of other variables and $E_b^t$ is strictly increasing with $w_{bi,DT}^t$ and $w_{bi,U}^t$, by (6) and (7), the optimal value of $w_{bi,DT}^t$ and $w_{bi,U}^t$ should be

$$w_{bi,U}^t = \frac{s_i^t}{r_{bi,U}^t \tau}, w_{bi,DT}^t = \frac{\epsilon_i^t}{r_{bi,D}^t \tau}, i\epsilon\, \zeta^t(b). \quad (24)$$

On the other hand, since $E_b^t$ is independent of the value of $w_{bi,DS}^t$, the resource allocation problem in (18) is equivalent to minimizing the MEC server power consumption (9) subject to the processor speed constraint (1), downlink bandwidth limitation (3) and workload delay requirement (8). Note that since (1), (3), (8), (9) are convex functions when $u_{bi}^t > 0$ and $w_{bi,DL}^t > 0$, the optimal solution can be achieved by analyzing its Karush–Kuhn–Tucker (KKT) conditions. The proof is similar to Boyd in [22], which is omitted for brevity.

To show the optimal result in (26), we define the following terms for all i$^{\text{th}}$ UE in $\zeta^t(b)$,

$$\gamma_i^t = \frac{\delta_i^t}{r_{bi,D}^t d_i^t}, \;\; \sigma_i^t = \frac{\gamma_i^t c_i^t}{d_i^t}, \;\; \varpi_b^t = \sum_{i\epsilon\zeta^t(b)} w_{bi,DT}^t,$$

$$Q_b^t = \frac{\sum_{i\epsilon\zeta^t(b)} \sqrt{\sigma_i^t}}{W_{b,D} - \varpi_i^t - \sum_{i\epsilon\zeta^t(b)} \gamma_i^t}, \;\; y_i^t = \frac{\sqrt{c_i^t}}{d_i^t} + \frac{\sqrt{\gamma_i^t d_i^t}}{d_i^t} Q_b^t. \quad (25)$$

Then the optimal resource allocation for $u_{bi}^t$ and $w_{bi,DL}^t$ is

$$u_{bi}^t = y_i^t \sqrt{c_i^t}, \;\; w_{bi,DS}^t = \frac{y_i^t \sqrt{\gamma_i^t d_i^t}}{Q_b^t}, i\epsilon\, \zeta^t(b). \quad (26)$$

However, the above analysis is under the condition that $\zeta^t(b)$ is a feasible set for the $b^{th}$ SRSU. When $\zeta^t(b)$ is not feasible, MPCA will drop the UE which consumes the most communication and computing resource until constraints of (18) are satisfied.

### *B. The TEBA algorithm*

Given the minimized power consumption of each SRSU at each time slot *t* from MPCA, solar energy generation and power consumption of a single SRSU are matched by solving the TEB problem. The solution is provided by TEBA, as shown in Fig. 4. To make the smallest AUPR $\pi_b^t$ among all time slots as large as possible, the idea of TEBA is to allocate the current generated solar energy $J_b^t$ to future time slots $t+1$ to $T$. In the beginning of TEBA, the allocated solar energy $M_b^t$ for each time slot is initialized to be equal to $J_b^t$. To satisfy constraint (14), TEBA will start the allocation process from the last slot to the first slot. Since all SRSU has to remain on at all time, the algorithm will first check if there exists any time slot $t'$ between $t+1$ to $T$ which has $M_b^{t'} < E_{s,IDLE} + E_{b,IDLE}$. If yes, TEBA will allocate part of $M_b^t$ to $t'$ until no such $t'$ exists or $M_b^t$ itself reaches the lower limit, namely $E_{s,IDLE} + E_{b,IDLE}$. $\pi_b^t$ is recalculated by the residual $M_b^t$, if the updated $\pi_b^t$ is greater than the average AUPR $\overline{\rho_b^t}$ after time slots *t*, where $\overline{\rho_b^t}$ is calculated as:

$$\overline{\rho_b^t} = \frac{\sum_{i=t}^{T} M_b^i}{\sum_{i=t}^{T} E_b^i}, \quad (27)$$

TEBA will let $\pi_b^t$ equal $\overline{\rho_b^t}$ by reducing $M_b^t$. The redundant $M_b^t$ will be allocated to time slots $t'$, $t' > t$, with $\pi_b^{t'} < \overline{\rho_b^t}$ by a value proportional to the solar energy required to make $\pi_b^{t'} = \overline{\rho_b^t}$. TEBA will repeat the above process until all of the time slots are processed.

### *C. The SEBA algorithm*

Based on the result of TEBA, we propose the algorithm SEBA shown in Fig. 5 to balance the solar energy among different SRSUs. SEBA has three steps: 1) choose the SRSU which has the smallest AUPR with its value less than 1, namely the allocated solar power is less than its estimated power consumption; 2) drop or offload associated UEs of the above SBS until its AUPR is at least 1. 3) iteratively apply above steps to other SRSUs until all SRSUs have AUPR greater than 1.

At each time slot *t*, SEBA will first sort all SRSUs by their AUPR in an ascending order. Following the sorted list, SEBA

Spatial Energy Balancing Algorithm (SEBA)

**Input:** $Time\ slot\ t, \{E_b^t, M_b^t, \zeta^t(b) \mid b \in \mathcal{B}\}$, $\{K_i^t, A_i^t | i \in I\}, \{u_{bi}^t, w_{bi,U}^t, w_{bi,DS}^t, w_{bi,DT}^t \mid i \in I, b \in \mathcal{B}\}$, $U_b, E_{S,MAX}, \alpha, p_d, p_{c,U}, \tau$

**Output:** $\{A_i^t \mid i \in I\}, \{E_b^t, \zeta^t(b) \mid b \in \mathcal{B}\}$, $\{u_{bi}^t, w_{bi,U}^t, w_{bi,DS}^t, w_{bi,DT}^t \mid i \in I, b \in \mathcal{B}\}$.

**Initialize** $\Omega_{bi}^t \leftarrow \{u_{bi}^t, w_{bi,U}^t, w_{bi,DS}^t, w_{bi,DT}^t\}$;
**Sort** SRSU by $\pi_b^t$ in an ascending order, store the index in D.
**while** D is not empty **do**
  $b' \leftarrow \mathrm{D}(1)$.
  $\Lambda_{b'} \leftarrow \{\max(b'-1,1), \min(b'+1, N)\}$;
  **Step 1**: Divide $\zeta^t(b')$ into subsets $S_1, S_2, S_3$
  **Sort** UE in each group $S_i$ by the corresponding power consumption in a descending order, let the sorted index be $H_1, H_2, H_3$ respectively.
  $H \leftarrow \{H_1, H_2, H_3\}$;
  **Step 2**: change association status of UEs
  **while** $\pi_{b'}^t \geq 1$ **do**
    $i_M \leftarrow H(1)$.
    **sort** all SRSUs $b'' \in \Lambda_{b'} \backslash \{b'\}$ by $\eta_{b'' i_M}^t$ in descend order, let the sorted index be $Z$;
    $f_{drop} \leftarrow true$;
    **for** $b'' = 1\ to\ |\Lambda_{b'} \backslash \{b'\}|$ **do**
      $b_m \leftarrow Z(b'')$, $\zeta^t(b_m)_v \leftarrow \zeta^t(b_m) \cup \{i_M\}$
      calculate $\left(\Omega_{b_m i_M}^t\right)_v$ by (23), (25) use $\zeta^t(b_m)_v$
      **if** $\zeta^t(b_m)_v$ and $\left(\Omega_{b_m i_M}^t\right)_v$ satisfy (1) - (3), (6) - (8), and $\left(E_{b_m}^t\right)_v \leq M_{b_m}^t$ **do**
        $\zeta^t(b_m) \leftarrow \zeta^t(b_m)_v$, $\zeta^t(b') \leftarrow \zeta^t(b') \backslash \{i_M\}$
        $a_{b' i_M}^t \leftarrow 0, a_{b_m i_M}^t \leftarrow 1, \Omega_{b_m i_M}^t \leftarrow \left(\Omega_{b_m i_M}^t\right)_v$
        update $\Omega_{b' i}^t, i \in \zeta^t(b')$ by (24), (26) use $\zeta^t(b')$
        update $E_{b_m}^t, E_{b'}^t$ by (13), update $\pi_{b_m}^t, \pi_{b'}^t$;
        $H \leftarrow H \backslash \{H(1)\}$; $f_{drop} \leftarrow false$;
        **break**;
      **end if**
    **end for**
    **if** $f_{drop} == true$ **do**
      $\zeta^t(b') \leftarrow \zeta^t(b') \backslash \{i_M\}, a_{b' i_M}^t \leftarrow 0,$
      update $\Omega_{b' i}^t, i \in \zeta^t(b')$ by (24), (26) use $\zeta^t(b')$
      update $E_{b'}^t$, update $\pi_{b'}^t$;
      $H \leftarrow H \backslash \{H(1)\}$;
    **end if**
  **end while**
D $\leftarrow D \backslash \{\mathrm{D}(1)\}$;
**end while**

Figure 5. SEBA algorithm

will check if the every SRSU's AUPR is less than 1. For such SRSU, SEBA separates UEs in $\zeta^t(b)$ into three subsets: $S_1, S_2$, and $S_3$. $S_1$ includes the UEs that are handed over to the $b^{th}$ SRSU from other SRSUs at current time slot; $S_2$ includes the UEs that previously have no connection to any SRSUs and the UEs that are previously associated to the $b^{th}$ SRSU is included in $S_3$. The rationale of the grouping is that changing the association of UEs in $S_1$ will not increase the value of $C_{handover}^t$ since these UEs already suffer the QoS loss from handover. On the other hand, although dropping the UEs in $S_2$ and $S_3$ will both

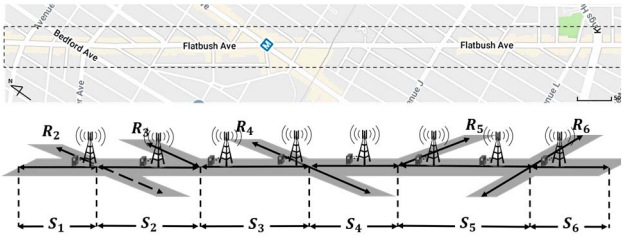


Figure 6. (a) above, a neighborhood around Flatbush Avenue in Brooklyn. (b) below, network topology showing deployment of roads and SBSs.

increase $C_{drop}^t$, re-associating the UEs in $S_2$ will not increase $C_{handover}^t$. Therefore, SEBA will start to change the association status of UEs in $S_1$, followed by UEs in $S_2$ and $S_3$. In each subset, the UE that accounts for the highest power consumption, which is calculated by the computing and communication resource allocated to UE will be first chosen by SEBA.

Whenever SEBA is going to change the association status of a UE, it will check if the two neighbor SRSUs can accommodate this UE without violating constraints (1) - (3) and keep their own $\pi_b^t < 1$. If not, SEBA will drop this UE.

Since $E_b^t$ for each SRSU will be modified after the UE offloading process, TEBA needs to be applied to find the new optimal solar energy allocation. The system will iteratively run TEBA and SEBA until the cost cannot be further decreased. We use a threshold shown in Fig. 2 to terminate the iterations.

## V. Experimental Result

In this section, we first introduce our MATLAB-based simulation framework and simulation parameters. We further present the effects of MCPA, TEBA and SEBA algorithms on the final power consumption and UE association result to SRSUs. In the end, we compare the performance of our overall QLM algorithm with two Greedy SRSU Energy Management (GSEM) strategies.

### *A. Simulation Framework*

The objective of our simulation framework is to observe the effects of different solar energy management and UE association strategies on QoS loss of the offloaded application with real-world environment. Therefore, our simulation environment consists of the solar photovoltaic (PV) model, the SRSU power consumption model, the channel model, the user location and traffic model, the vehicular applications workload model and the MEC server model. The system parameters are summarized in Table I.

We choose to study the streets in a neighborhood in Brooklyn, New York City, shown in Fig. 6(a), so that we can utilize historical traffic data for the streets collected by New York State Department of Transportation and available in [23]. In our simulation environment, we implement the topology of the streets with the placement of SRSUs as shown in Fig. 6(b). The topology comprises of a bidirectional long road $R_1$, crossed by 5 short bidirectional streets $\{R_2, R_3, \dots, R_5\}$, dividing R1 into 6 bidirectional segments $\{S_1, S_2, \dots, S_6\}$.

The SRSUs are set along $R_1$, each separated by 400 meters. The total bandwidth of each SBS of each direction in our

Table I. Simulation Parameters

| Parameter | Description | Value |
|---|---|---|
| N | Number of SRSUs | 8 |
| $\tau$ | Duration of a time slot | 1 (s) |
| $U_b$ | Computing resource limit | 4744 (MIPS) |
| $W_{b,U}$ | UL bandwidth limit | 129.6 (MHz) |
| $W_{b,D}$ | DL bandwidth limit | 129.6 (MHz) |
| $p_B$ | SBS antenna power density | 20 (dBm/MHz) |
| $p_I$ | UE antenna power density | 17 (dBm/MHz) |
| $p_{c,U}$ | UL[a] SBS circuit power density | 15.6 (mW/MHz) |
| $p_{c,D}$ | DL[b] SBS circuit power density | 15.6 (mW/MHz) |
| $N_0$ | Noise power density | -184 (dBm/MHz) |
| $E_{b,IDLE}$ | SBS power consumption in idle | 6 (W) |
| Channel model | | |
| $g_{bi}^t$ | Path loss | Shadowing |
| | 128.1 + 37.6log10(R), R in kilometers, R is the distance between SBS and UE | 8 (dB) |

[a]UL: Uplink, [b]DL: Downlink

simulation is 5MHz, which is further divided into 25 resource blocks. Note that the minimum bandwidth available to a UE is equal to the bandwidth of a resource block, namely 0.18 MHz. We model the power consumption and computing resource profile of the MEC server as Raspberry Pi 2 Model B, which has been shown to be a feasible solution for low-power cloud servers [24]. The power consumed for the MEC server at idle and fully utilized mode is 4.8W and 6.25W respectively. For solar generation profile, we use the solar irradiance data in [5], which provides temporal variation at each SRSU as well as spatial variation across the SRSUs. We choose to simulate for 24 hours from 9AM to 9AM, so that SRSUs can leverage the solar energy generated during the day time to power itself at night, when there is no solar energy generation.

As mentioned in Section II, the number of UE entering the network at each time slot follows a Poisson distribution with rate parameter $\lambda_r$. Each UE is entering the network with predetermined travel routes and speed. The travel routes decision, speed and $\lambda_r$ are set in a manner that the average traffic volume of each road segments and cross streets $S \cup R$ will satisfy the historical data in [23]. Furthermore, the transmission channel model is specified by the Vehicular to Evolved Node B (eNB) type RSU channel model in [25] with path loss model listed in Table 1 and the transmitted power of SBS and UE are specified as a typical small cell network [26].

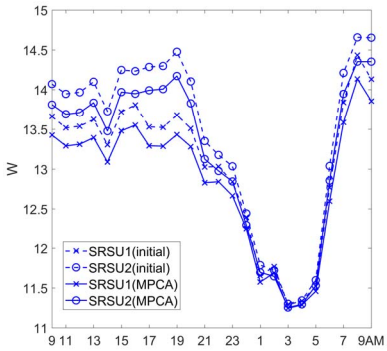


Figure 7. Initial power consumption for two SRSUs and after MPCA

To model the offloading of the vehicle tasks and the uplink/downlink data, we assume that each UE will upload a 1080p 30fps H.264 encoded video file to its associated SRSU at each time slot, which requires approximately 10 Million instructions per second (MIPS) for video processing including decoding and object detection by the MEC [27] [28]. We set the uploaded data size to be uniformly distributed between 11 to 13.5 MB. Since the delay sensitive downlink data size depends on the information in the uplink data, we assume the data size is uniformly distributed between 0.1 to 0.3 MB. In the meantime, half of the vehicles will request to download a 720p or 1080p 30fps H.264 encoded video with duration $\tau$, each with 0.5 probability and delay constraint equals $\tau$.

In the next subsection, we first present the effects of MPCA, TEBA and SEBA by showing the change of individual SRSU's power consumption and user association. Then, we compare the performance of QLM in terms of its weighted QoS loss of the offloaded task processing by using two Greedy SRSU Energy Management (GEM) strategies for temporal solar power allocation. The first GEM, the Ordinary GEM (OGEM), allows each SRSU to utilize any available solar and battery energy to satisfy its power demand at time slot $t$ and store the remaining solar power to the battery. Consequently, OGEM will make each SRSU's AUPR equal to 1 at every time slot, which restraints UE from being re-associated between different SRSUs since SRSUs have no redundant energy to serve extra workload. The second GEM strategy sets the allocated solar energy at each time slot to the value that is 1.2 times higher than its estimated power consumption when the battery level is higher than a threshold 50 Whr. We denote this strategy as the Reserved GEM

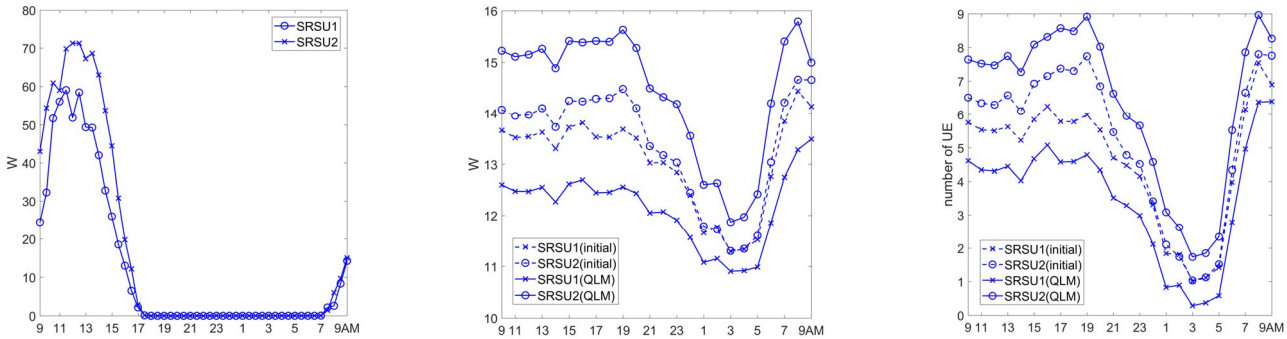


Figure 8. Solar generation, power consumption and vehicle associations for two SRSUs. (a) left, solar generation profile; (b) middle, initial power consumption and after QLM algorithm; (c) right, initial vehicle association and after QLM.

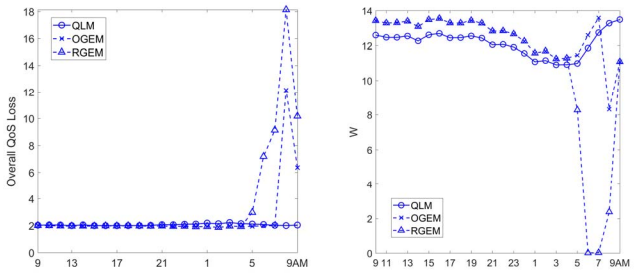


Figure 9. (a) left, overall QoS loss of three algorithms for each time slot; (b) right, power consumption of a SRSU for three algorithms: QLM, OGEM and RGEM.

(RGEM). If the battery level is less than 50 Whr, RGEM operates in the same way as OGEM. We assume RGEM performs the same UE association strategy as SEBA since RGEM allows SRSU's AUPR be greater than 1.

### *B. Simulation Result*

We first observe the effect of MPCA by comparing its result with SRSU's power consumption without MPCA. We assume that without MPCA, SRSU communication and computing resources are allocated by the following rules: $w_{bi,U}^t$ and $w_{bi,DT}^t$ are allocated by (24); $u_{bi}^t$ and $w_{bi,DS}^t$ are calculated so that both of the computation delay and downlink transmission delay are equal to $0.5\,d_i^t$. The results in Fig. 7 shows that the power consumption of SRSU1 and SRSU2 decreases after MCPA under the same offloaded task profiles.

Fig, 8 shows the effects of TEBA and SEBA. Fig 8 (a) depicts the solar power generation profiles of two SRSUs during the 24-hour simulation period. Fig. 8 (b) shows the initial power consumption and final power consumption of SRSU 1 and SRSU 2 after applying QLM algorithm. As mentioned in Section III, at time slot *t*, UEs in the network will initially connect to SRSU which provides the highest RSSI. Based on the communication and computing constraints, SRSU then finds its feasible association set $\zeta^t(b)$, which leads to the initial estimation of SRSU power consumption at time slot *t*. The effect of TEBA can be observed from final SRSU power consumption result shown in Fig. 8(b), as the solar generation profiles in Fig. 8(a) are arranged by TEMA to match the initial power consumption estimation. Since solar generated by SRSU 1 is not sufficient to serve all the workload from UEs in the initial feasible association set $\zeta^t(1)$, during SEBA, SRSU 1 will associate some of the UEs in $\zeta^t(1)$ to SRSU 2, reducing its power consumption and preventing these UEs from experiencing QoS loss. Consequently, the number of associated UEs decreases in SRSU 1 while SRSU 2 shows the opposite trend, as shown in Fig. 8(c).

Table II shows the performance comparison of OGEM, RGEM and QLM, where *Drop* and *Handover* in Table II represents the QoS loss of offloaded UE application from UE drop and UE handover introduced by (15) (16) divided by the number of UEs in the network respectively. On the other hands, *Overall* in Table II represents the overall average QoS loss introduced in (17). QLM can reduce the weighted QoS loss by 31% compared to OGEM and 52% compared to RGEM. As mentioned in previous subsection, OGEM cannot balance the offloaded task profile among different SRSUs. If a SRSU is running out of solar energy and its battery is fully discharged, it will drop the served UEs instead of trying to offload them, leading to higher drop rate compared to QLM. In comparison, although RGEM allows UEs to be re-associated between SRSUs, its power consumption unaware battery charging strategy makes SRSUs either have surplus allocated solar power or suffer power deficiency simultaneously most of the time.

Table II. QoS Loss Performance

| QoS Loss[a] | Algorithms | | |
|---|---|---|---|
| | ***QLM*** | ***OGEM*** | ***RGEM*** |
| **Drop** | 0.041 | 0.982 | 2.284 |
| **Handover** | 19.45 | 19.29 | 18.60 |
| **Overall** | 1.985 | 2.911 | 4.144 |

[a]Unit: %

Therefore, RGEM fails to take the advantage of UE reassociation to reduce the QoS loss.

Fig. 9(a) elaborates the QoS loss performance of OGEM, RGEM and QLM at each time slots. Before 5 AM, all of the algorithms have similar performance. This is because the energy stored in the battery is enough to satisfy the SRSU's initial power consumption estimation during this period. After 5 PM, the overall average QoS loss of OGEM and RGEM escalates due to insufficient solar generation and fully discharged battery. The same phenomenon can be observed in Fig. 9(b), where the SRSU is forced to shut down during 5 to 9 AM by RGEM and 7 to 9 AM by OGEM under the absence of solar power and an empty battery. On the contrary, QLM keeps the QoS loss average during the whole simulation period low by optimally allocating the solar energy to each time slots through scheduled charging and discharging of the battery and balance the offloaded task among SRSUs.

The result indicates that our algorithm not only can balance the mismatch of SRSU solar energy generation and power consumption over different time slots but can also optimally compensate the power deficiency of a single SRSU by utilizing resources from its neighboring SRSUs.

## VI. Conclusion

In this paper, we show the feasibility of a green road infrastructure of solar-powered RSUs, consisting of small cell bases station and edge computing, to support the computing and communications needs of vehicles. We propose the QLM algorithm, which is a joint solar power conservation and user association algorithm which minimizes the average QoS loss due to service outage and handover loss possible when a SRSU runs out of solar or battery power. We break down the problem into three sub-problems and propose algorithms for each sub-problem including combinations of SRSU's communication and computing resource allocation, solar power conservation and UE association techniques. Our simulation results shows that QLM significantly reduces the average QoS loss caused by power deficiency compared to greedy algorithms.

## ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant No. CNS-1619184.

## REFERENCES

[1] J. Choi, V. Va, N. Gonzalez-Prelcic, R. Daniels, C. R. Bhat and R. W. Heath, "Millimeter-Wave Vehicular Communication to Support Massive Automotive Sensing," in *IEEE Communications Magazine*, vol. 54, no. 12, pp. 160-167, December 2016.

[2] X. Ge, S. Tu, G. Mao, C. X. Wang and T. Han, "5G Ultra-Dense Cellular Networks," in *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72-79, February 2016.

[3] A. M. Aris and B. Shabani, "Sustainable Power Supply Solutions for Off-Grid Base Stations," *Energies*, vol. 8, no 10, pp. 10904-10941, September 2015.

[4] A.S.Y. Poon, "An Energy-Efficient Reconfigurable Baseband Processor for Wireless Communications," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.15, no.3, pp.319-327, March 2007.

[5] P. H. Chiang, R. Guruprasad and S. Dey, "Renewable energy-aware video download in cellular networks," *2015 IEEE 26th PIMRC,* Hong Kong, 2015, pp. 1622-1627.

[6] T. Han and N. Ansari, "On Optimizing Green Energy Utilization for Cellular Networks with Hybrid Energy Supplies," *IEEE Transactions on Wireless Communications*, vol. 12, no. 8, pp. 3872-3882, August 2013.

[7] P. Chiang, R. Guruprasad and S. Dey, "Optimal Use of Harvested Solar, Hybrid Storage and Base Station Resources for Green Cellular Networks," to appear in *IEEE Transactions on Green Communications* and Networking

[8] F. Parzysz, M. Di Renzo and C. Verikoukis, "Power-Availability-Aware Cell Association for Energy-Harvesting Small-Cell Base Stations," *IEEE Transactions on Wireless Communications*, vol. 16, no. 4, pp. 2409-2422, April 2017.

[9] V. Chamola, B. Krishnamachari and B. Sikdar, "Green Energy and Delay Aware Downlink Power Control and User Association for Off-Grid Solar-Powered Base Stations," *IEEE Systems Journal*. January 2017.

[10] Y. Mao, J. Zhang and K. B. Letaief, "Dynamic Computation Offloading for Mobile-Edge Computing With Energy Harvesting Devices," in *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590-3605, Dec. 2016.

[11] J. Xu and S. Ren, "Online Learning for Offloading and Autoscaling in Renewable-Powered Mobile Edge Computing," *2016 IEEE GLOBECOM*, Washington, DC, 2016, pp. 1-6.

[12] L. Chen, J. Xu and S. Zhou, "Computation Peer Offloading in Mobile Edge Computing with Energy Budgets," *2017 IEEE Global Communications Conference (GLOBECOM),*, Singapore, 2017, pp. 1-6.

[13] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, "Traffic-driven Power Saving in Operational 3G Cellular Networks," *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, pp. 121–132, September 2011.

[14] Y. Sun, Y. Chang, M. Hu and T. Zeng, "A Universal Predictive Mobility Management Scheme for Urban Ultra-Dense Networks with Control/Data Plane Separation," *IEEE Access*, vol. 5, pp. 6015-6026, 2017.

[15] J. McInerney, S. Stein, A. Rogers, and N. R. Jennings, ''Breaking the habit: Measuring and predicting departures from routine in individual human mobility,'' *Pervasive Mobile Comput.*, vol. 9, no. 6, pp. 808–822, 2013.

[16] Y. Mao, C. You, J. Zhang, K. Huang and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322-2358, Fourthquarter 2017.

[17] S. Barbarossa, S. Sardellitti and P. Di Lorenzo, "Joint allocation of computation and communication resources in multiuser mobile cloud computing," *2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Darmstadt, 2013, pp. 26-30.

[18] R. Irmer et al., "Coordinated multipoint: Concepts, performance, and field trial results," *IEEE Communications Magazine*, vol. 49, no. 2, pp. 102-111, February 2011.

[19] K. Zhang et al., "Energy-Efficient Offloading for Mobile Edge Computing in 5G Heterogeneous Networks," *IEEE Access*, vol. 4, pp. 5896-5907, 2016.

[20] N. T. Ti and L. B. Le, "Computation offloading leveraging computing resources from edge cloud and mobile peers," 2017 *IEEE International Conference on Communications*, Paris, 2017, pp. 1-6.

[21] P. Chiang, S. Chiluvuri, S. Dey and T. Nguyen, "Forecasting of Solar Photovoltaic System Power Generation using Wavelet Decomposition and Bias-compensated Random Forest," *Proc. of the 9th Annual IEEE Green Technologies Conference*, Denver, CO, Mar. 2017, pp. 260-6.

[22] S. Boyd and L. Vandenberghe, Convex Optimization. New York, NY, USA: Cambridge University Press, 2004.

[23] "NYS Traffic Data Viewer" [Online]. Available: https://gis3.dot.ny.gov/html5viewer/?viewer=tdv [Accessed: Mar. 6, 2018].

[24] F. P. Tso, D. R. White, S. Jouet, J. Singer and D. P. Pezaros, "The Glasgow Raspberry Pi Cloud: A Scale Model for Cloud Computing Infrastructures," *2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops*, Philadelphia, PA, 2013, pp. 108-112.

[25] The 3rd Generation Partnership Project, "Study on LTE-based V2X services," 3GPP-REF-36885, rel. 14, 2016. [Online]. Available: http://www.3gpp.org/. [Accessed: Jan. 19, 2018].

[26] The 3rd Generation Partnership Project, "Small cell enhancements for E-UTRA and E-UTRAN - Physical layer aspects," 3GPP-REF-36.872, rel. 12, 2013. [Online]. Available: http://www.3gpp.org/. [Accessed: Jan. 7, 2018].

[27] C. Cheng, et al. "An in-place architecture for the deblocking filter in H.264/AVC." *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol.53, pp.530 - 534, July, 2006

[28] T. Endeshaw, J. Garcia and A. Jakobsson, "Classification of indecent videos by low complexity repetitive motion detection," *37th IEEE Applied Imagery Pattern Recognition Workshop*, Washington DC, pp. 1-7, 2008