# Design and Optimization of Solar-Powered Shared Electric Autonomous Vehicle System for Smart Cities

Pengzhan Zhou, *Student Member, IEEE,* Cong Wang, *Member, IEEE,* and Yuanyuan Yang, *Fellow, IEEE*

**Abstract**—Smart transportation shall address utility waste, traffic congestion, and air pollution problems with least human intervention in future smart cities. To realize the sustainable operation of smart transportation, we leverage solar-harvesting charging stations and rooftops to power electric autonomous vehicles(AVs) solely via design. With a fixed budget, our framework first optimizes the locations of charging stations based on historical spatial-temporal solar energy distribution and usage patterns, achieving $(2+\epsilon)$ factor to the optimal. Then a stochastic algorithm is proposed to update the locations online to adapt to any shift in the distribution. Based on the deployment, a strategy is developed to assign energy requests in order to minimize their traveling distance to stations while not depleting their energy storage. Equipped with extra harvesting capability, we also optimize route planning to achieve a reasonable balance between energy consumed and harvested en-route. As a promising application, utility optimization of shared electric AVs is discussed, and $(2k+1)$-approx algorithm is proposed to manage $k$ vehicles simultaneously. Our extensive simulations demonstrate the algorithm can approach the optimal solution within 10-15% approximation error, improve the operating range of vehicles by up to 2-3 times, and improve the utility by more than 50% compared to other competitive strategies.

**Index Terms**—Optimal scheduling, energy harvesting, vehicle charging, shared autonomous vehicles, approximation algorithm

## 1 Introduction

The future smart cities exemplify how computation and information flow are coordinated between end devices and infrastructure for automation. Transportation is one of the driving impetus for this evolution as most of the metropolises like Los Angles, Beijing and New Delhi suffer from persistent traffic congestion, which remains as one of the major contributors to air pollution. Studies found that traffic congestion is responsible for $56$ billion pounds of carbon dioxide pollution [1] and this number keeps climbing. Electric vehicles have been a green solution and their possession enjoys a rapid growth recently. Meanwhile, the recent advance in artificial intelligence makes it possible to learn from end-to-end for autonomous driving [2], which rises as a promising, or presumably, the ultimate solution to traffic congestion [3]. A marriage of these two powerhouse technologies would reshape the auto industry as major manufacturers like Ford, BMW and Volve have already made their moves to go electrification with autonomous designs.

Unfortunately, the relatively stagnant progress in battery technology fails to catch up with the rising demands in mileage and computation, especially the arrays of sensors are mounted and commanded by power-hungry computing units like GPUs on these battery-powered platforms for real-time processing [2]. The data from sensors like cameras and LiDARs are fused and processed by GPUs, which could generate more than 12 GB of data each minute and consume about 2.5 KWh [4]. With the intensive computations on-board (consumes about 10% energy for the current Tesla Model 3), a fully-charged autonomous vehicle (AV)[1] is expected to last much less than the standard mileage [6]. While energy-efficient neural computation is still at its early stage [5], *charging* is of paramount importance to promote AV as a first-class citizen in the future smart cities. This pushes back on the service provider to offer better coverage of charging stations, operation management, guidance of route planning and deliver them as a whole package to the users. Thus, a solution to the charging problem should entail a holistic approach starting from the charging stations, and assist the AVs with efficient algorithms at the infrastructure backend for optimized decision making.

### 1.1 Retrospect: Solutions with the Main Power Grid

An immediate solution is to build more charging stations to satisfy the emerging energy demands. For instance, Tesla aims to make 99% of the US population within $150$ miles of a charging station. Yet, driving $3$ hours for a charge is obviously not a solution, let alone the mileage for returning. So far, the cost of building and maintaining dedicated charging stations still remains prohibitive for individual and private business owners. Further, inappropriate selection of locations and the dynamics of demand may lead to either low utilization or congestion [7], [8]. These situations cause highly unbalanced distribution of resources and ultimately a loss to the service provider. To this end, existing works

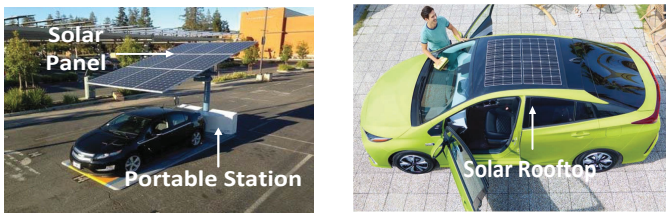
- *P. Zhou and Y. Yang are with Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA. E-mail: {pengzhan.zhou, yuanyuan.yang}@stonybrook.edu*
- *C. Wang is with Department of Cybersecurity, George Mason University, VA 22030, USA. E-mail: cwang51@gmu.edu*

*Manuscript received MONTH 20XX.*

1. We briefly denote "electric AV" as "AV" hereafter for conciseness, since the AV powered by combustion engine is not considered in the paper.

focus on pricing/incentivizing strategies [7], [8], [9], placement of charging stations [10], [11], [12], or energy saving from the user perspective [13], [14]. Users are modeled as self-interested agents to maximize the expected profits of selecting the charging stations [7], [8]. A greedy algorithm is proposed to maximize the charging demand [10]. The mutual interactions among charging stations, drivers, traffic congestion and queuing time are jointly considered in the placement of charging stations [11]. A bi-level optimization model is proposed to arrange the distribution of stations by maximizing the revenue and minimizing the user dissatisfaction [12]. A neural network is used to predict the driving behaviors which extends the mileage by accommodating the controllers [13]. An automated control system is proposed to manage power consumption, improve the battery lifetime and driving range [14].

These strategies focus on the scenarios that the charging stations are connected to the main power grid. Yet, the rapid adoption of AVs would bring excessive load and instability to the grid. With opportunities to generate energy off-the-grid, in this paper, we pursue new directions to power the AVs by *renewable energy*, such as the solar power. Ambient energy is adequate to power self-sustainable wireless sensor networks as shown in [15]. Among the ambient energy, solar is an ideal, green source as they can be harvested for free after a one-time investment. Its success for energy provisioning in distributed systems has been proven in wireless sensor networks [17], [18], [19]. For AVs, it not only alleviates the load to the main power grid [20], but also offers distributed charging opportunities while people are at work. Fig. 1 demonstrates some existing prototypes of this innovation. Charging stations installed with large-size solar panels can serve multiple AVs simultaneously. The platform can be made mobile and relocated to new locations depending on the distributions of ambient energy or emerging demands from users [27]. Similarly, equipped with solar panels on the rooftop, AVs can harvest extra energy anywhere during daytime.




(a) (b)

Fig. 1: Prototypes (a) mobile solar-powered charging station [27] (b) car with solar rooftop from Toyota.

### 1.2 Contributions of this work

These innovations laid the foundations to power autonomous vehicles with renewable energy in the future smart cities. Its success should not only rely on hardware integration, but also tackle a series of challenges during operation facing the uncertainties in the ambient energy source and user demands. Obviously, the locations of charging stations require careful planning based on the cost profile of rental and daily operation. They should also satisfy the dynamic distribution of charging requests to avoid low utilization or congestion. With external capabilities to harvest solar energy, the AVs can also jointly plan their routes to avoid energy depletion en-route and merge this capability into the entire energy supply chain for system-wide optimization. A recent work [16] has developed a mechanism to plan the route of solar-powered vehicles following the strength of solar radiation. We further refine this approach and integrate it with the optimization framework.

Furthermore, this paper also studies a new framework of shared AVs as an actual application of the proposed autonomous vehicle network solely depending on solar energy. In [22], authors propose novel charging strategies of PEV(Plug-in Electric Vehicle)-based taxis as a game model, which formalizes the problem of pricing the power consumption to realize risk-averse decisions on PEV charging considering the aging of transformers. Without depending on drivers, the shared AVs can operate perpetually through days, weeks, even months only if they have enough energy. By replenishing the energy at the solar-powered charging stations, such a system of shared AVs can operate without human intervention, which is a feasible solution for public transportation for future smart cities. However, due to the non-neglected time required to charge the electric AVs, it is challenging to maximize the utility achieved by each AV while sparing enough time for charging the battery. We intend to address such issues for this novel application in this paper and propose efficient scheduling algorithms for one and multiple AVs with guaranteed performance bound.

In particular, this paper aims to offer a full-stack solution by: 1) optimizing the locations of charging stations by considering extra traveling distance to get charged and spatial-temporal energy distribution of solar power; the historical distributions are analyzed using machine learning algorithms to generate predictions; 2) conducting operation management to establish new stations for emerging demands and pruning the ones with low utilization; 3) assigning the charging requests to different stations based on the real-time energy status in order to, avoid congestion and balance energy income/expenditure; 4) planning routes of the AVs to achieve a balance between energy consumed and harvested en-route with the solar-harvesting rooftops; 5) scheduling the trips served by shared AVs to maximize the overall utility while assuring the energy constraint.

Note that, although there are literatures having studied the charging station placement, energy request assignment, and route planning of vehicles, there are few works having addressed these problems jointly considering the income and expenditure of harvested solar energy to the best of our knowledge. Directly utilizing existing solutions cannot satisfy the design of a holistic electric AV system solely depending on solar energy. Therefore, it is necessary to consider these aspects simultaneously in the new scenario and propose feasible solutions with guaranteed performance bound. The contributions of this paper are summarized in the following.

- We propose a framework to determine the optimal locations of charging stations that maximizes the

energy output while minimizing the driving distance to the charging stations. Based on time series predictions from recurrent neural networks [37], a $(2+\epsilon)$-approximation algorithm is proposed.
- We formulate the charging assignment problem and develop an efficient solution using dynamic programming, based on the predicted solar income and waiting time of users.
- We develop a route planning algorithm achieving a balance between energy consumed and harvested en-route, given the potential capability to harvest solar energy from the vehicle rooftop.
- We propose two approximation algorithms with constant performance bound $2k+1$ to maximize the utility achieved by $k$ AVs operating simultaneously, which considers the time spent for charging AVs in the optimization.
- Based on the real energy traces from [35], our extensive simulations not only demonstrate the algorithm can achieve an average of 30-50% savings compared to the competitive algorithm [36], with 10-15% approximation error, but also improve the operating range of the AV by 2-3 times. For the simulation of shared AVs, our algorithm achieves more than 60% utility improvement compared with the benchmark and achieves about at least 50% of the optimal.

The rest of the paper is organized as follows. Sec. 2 studies related works. Sec. 3 presents the motivation and system overview. Sec. 4 studies the deployment of solar-harvesting charging stations. Sec. 5 studies the assignment of charging requests. Sec. 6 discusses the routing of AVs considering charging and harvesting energy. Sec. 7 addresses scheduling of shared AVs via approximation algorithms. Sec. 8 evaluates the performance of the proposed framework and Sec. 9 concludes the paper.

## 2 Related Works

In [7], authors design a new charging cost mechanism to realize the equilibrium of assigning charging requests in the face of selfish users. The paper minimizes the congestion of electric vehicles at the charging stations and maximizes social efficiency. In [8], authors treat the charging stations and EV users as self-interested agents that aim to maximize their own profit and minimize the impact on their schedule in the face of the congestion of stations. They propose a decentralized mechanism that scales well compared with the centralized one, which realizes the load balance on stations. In [9], an incentivizing mechanism based on multi-armed bandits is proposed to motivate users of shared vehicles to rebalance the vehicles, which considers the heterogeneous levels of difficulties of different tasks. In [10], authors consider the charging station placement problem for the shared EVs. They propose an approximation algorithm intended to maximize the satisfied charging demands for the NP-hard problem, i.e., POI coverage and local charging demand. In [11], authors propose algorithms to solve the charging station placement optimization in distribution systems using genetic algorithms. The problem is formulated as a non-differential combinatorial optimization problem, which minimizes the system loss while satisfying the capacity and system operation constraints. In [12], authors consider the placement of public EV chargers across the city, which maximizes the overall revenue of charging stations while minimizing the driver discomfort. The problem is formulated as a bilevel optimization problem, and an alternating framework is proposed to realize a local minimum. The existing works about charging station placement mainly focus on resolving the congestion, maximizing revenue, and minimizing vehicle traveling. However, to the best of our knowledge, no solution has been proposed to deploy the system consisting of solar-powered charging stations, whose placement tends to achieve the aforementioned objectives while considering the spatial distribution of harvestable energy. Therefore, the existing methods can not be directly utilized to address the newly proposed problem.

In [13], authors propose a context-aware methodology to predict the driving behaviors, which can be used to manage the battery use of EVs better. Based on a nonlinear autoregressive model of artificial neural networks, the behaviors of drivers can be estimated with low loss, and the energy of EVs can be saved. In [14], authors jointly consider the energy use of engines, the heating, ventilation, and air conditioning in the battery management system, which improves the range of EVs via climate control of HVAC devices. In [16], authors propose a route planning method to balance the energy consumption and harvesting of solar-powered EVs. A power-aware optimal routing maximizes the amount of harvested energy choosing from candidate routes based on the bisect k-means clustering method. In [17], authors utilize weather forecasts to choose the time to satisfy the energy demands considering current energy and future energy supply in the energy harvesting system. A model is formulated to transfer the weather forecasts to solar energy harvesting predictions. In [21], authors study the design of a 4-tier system for autonomous electric vehicle sharing in terms of feasibility concerning profit, operations, engineering, and marketing. They conclude that the shared AVs are feasible considering the profits and emissions compared with the ordinary vehicle depending on fuels.

However, recent works mentioned above have only considered the charging stations operating based on power grids. This paper studies the scenario that the AV and charging stations are only powered via solar energy, which needs to jointly consider the charging requests and energy distribution placing the stations. Though some works consider the scheduling of shared AVs, they have not considered maximizing the utility of one or multiple AVs considering the time spent for charging due to the specialty of EVs.

## 3 Preliminary

We motivate this study based on the data analytics of solar data from NREL [29].

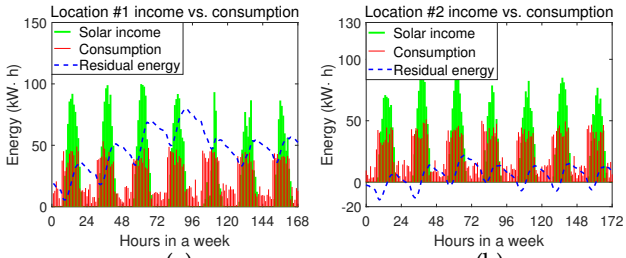
### 3.1 Motivation

From the energy measurements, we utilize the irradiance formulaes in [25] to calculate output power from the solar panel. For a period of one month, the output power of a solar charging station is compared with the energy needed

for satisfying the dynamic charging requests as shown in Fig. 2. The solar data is acquired in Aug. 2018 at El Paso, Texas. The number of charging requests is based on the daily traffic patterns obtained from [28]. We set (assume) a ratio of the electric AVs to the total traffic volume, and apply a moderate-size station equipped with 3-by-5-meter (i.e. 15 $m^2$) solar panel that generates electricity with 2.3 $kW \cdot h$, and stores the energy into a battery with 21.6 $kW \cdot h$ capacity according to [27]. The measurements are based on two representative locations: Location #1 in suburban area and #2 in downtown area. Fig. 2 shows the comparisons of the harvested solar energy vs. the consumed energy with charging demands at two different locations[2]. The residual energy is the net income of solar energy after deducted from the energy used to charge the AVs. Fig. 2 shows that there is sufficient solar energy in location #1 but insufficient in location #2. There are two observations from these preliminary results.

**Observation 1.** It is feasible for a moderate-size station to satisfy the daily charging demands in most of the locations.

**Observation 2.** Energy imbalance still exists at some locations. The additional demands should be re-routed to the neighboring stations.




(a) (b)

Fig. 2: Solar income vs. energy consumed for charging at different locations (a) sufficient energy at loc. #1 (b) insufficient energy at loc. #2.

**Rooftop Charging.** The new solar rooftop can harvest $150W$ according to a recent study from Hyundai [30]. Their study reveals that the harvested energy can extend the mileage by $3\%$ at the speed of $48km/h$ (about 1.3 km additional mileage) for the 2017 Hyundai IONIQ electric car at $11.5kW \cdot h/100km$. This indicates that for two routes with similar distance, choosing the one with more abundant energy can extend the mileage. However, different from [16], we argue that if the additional distance traveled to another route is more than 3%, it is not worthwhile changing the route even if the new one enjoys more solar irradiance. This is because the energy harvested will be offset by the additional distance traveled. Our analysis reveals that the solar rooftops are useful especially when the AVs are stationary during the working hours. The harvested energy can extend the mileage by $10km$, which covers most of the short-distance commutes.

### 3.2 System Model

The system architecture is shown in Fig. 3. It takes historical data of the spatial-temporal distribution of solar energy, usage distribution of AVs and (short-term) energy predictions

2. We consider the worst case scenario that all the AVs in the sub-area are requesting to get charged.

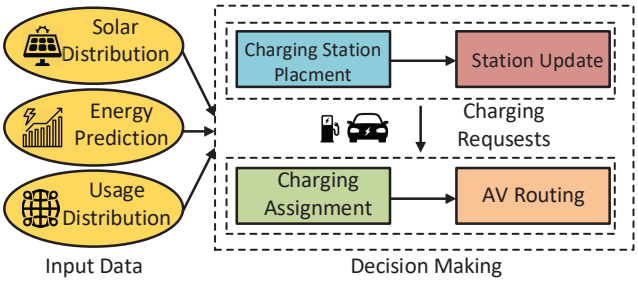



Fig. 3: System architecture.

into the frontend to determine the locations of charging stations. A recurrent neural network [37] is utilized to forecast the short-term energy income and charging requests. The algorithm takes the prediction and optimizes decision making. During operation, the system computes the following: 1) the locations of the charging stations according to the variation of usage patterns; 2) assignment of charging requests to appropriate stations based on energy income and expenditure; 3) route planning of AVs to optimize their energy expenditure.

We assume the system involves two major entities: *service provider* and *users* (passengers). The service provider offers a ride hailing service similar to Uber and Lyft, with the new innovation of self-driving and solar power. To make a trip request, users enter the destination and the system dispatches a dedicated ride. We assume the AVs have Internet connections with the infrastructure to report their real-time status such as location, battery energy, solar energy income (via rooftop sensors) and the infrastructure provides information of charging stations, route planning to optimize the operation. The service provider has an initial budget to allocate $k$ charging stations in a confined region, e.g., a start-up company with only a fixed amount of investment who wants to optimize the operation of the system. For simplicity, we assume the energy consumption of the AVs is proportional to the mileage or traveling time, and do not consider complex situations such as traffic congestions in route planning. These factors can be always introduced as additional weighted factors into the formulation at runtime. Once the AV reports a charging request, or is about to request one, the system collects such request into a pool for charging assignment.

## 4 Deployment of Charging Station

In this section, we first study the deployment of charging stations. The goal is to ensure that the energy demand of AVs can be satisfied at any time with moderate traveling distance to the charging stations. We propose a mechanism with an offline optimization of location selection, and stochastic online adjustment for continuous adaptation to new demand patterns. We summarize the important notations in Table 1.

### 4.1 Charging Station Placement (Offline)

The deployment of charging stations is formulated as the *Charging Station Placement* problem (CSP). Similar to gas stations that are built at the road intersections, convenient

TABLE 1: List of Important Notations

| Notation | Definition |
|---|---|
| $\mathcal{H}$ | Locations of charging stations |
| $p_j$ | Center of the $j$-th cluster with energy requests |
| $m_j$ | Number of requests in the $j$-th cluster |
| $f$ | Installation cost of one station |
| $n_i$ | Number of charged AVs |
| $r_i$ | Charging requests $i$ |
| $x_i$ | Location of requests $i$ |
| $d(i,j)$ | Distance between location $i$ and $j$ |
| $E_i$ | Energy demand of request $i$ |
| $s_i$ | Expected solar income at location $i$ |

locations of charging stations can also increase the operating range significantly without worrying about battery depletion. Nowadays, in order to promote emission-free vehicles, charging stations have been built on university campuses, public places like shopping malls [32]. However, few literature has studied stations with extra solar-harvesting capabilities to satisfy the rising energy demand in future smart cities. Existing works have considered energy harvesting in wireless sensor networks through solar-powered sensors [18] and hybrid energy source [19], whereas those solutions cannot be readily applied here due to their limited scale and the new application context like user demands, vehicle mileage and route planning.

Specifically, the formulation of CSP should consider several aspects: 1) the charging stations should be close to the locations of energy requests, in order to reduce the distance traveled to get charged; 2) for most of the energy requests, the charging stations need to make sure at least one of them is reachable to avoid battery depletion; 3) the charging stations should be located in places with abundant solar energy.

**Charging Request.** To facilitate our analysis, the city is divided into grids $\{x_i\}$. The system gathers historical data of energy requests based on their locations and amount. We define the sum of the charging requests $r_i$ within the grid $x_i$ as a parameter associated with the grid location, with its magnitude representing the amount of energy demand in each grid. Neighboring grids can be combined into an aggregated region $R_j$, a location $p_j$ and the amount of energy request $m_j$ are calculated as,

$$p_j = \frac{\sum_{i \in R_j} r_i x_i}{\sum_{i \in R_j} r_i}, \tag{1}$$

$$m_j = \sum_{i \in R_j} r_i \tag{2}$$

Eq. (1) calculates the average position of all the grids weighted by the number of requests, i.e., represented by the density of demands denoted by $p_j$, i.e., the location $p_j$ tends to be closer to grids with more energy requests. This definition helps find an optimal placement of the charging station. Eq. (2) calculates the sum of energy requests in a region, denoted by $m_j$, which determines the number of charging stations near the region. $p_j$ and $m_j$ together form a tuple $C_j = (p_j, m_j)$ of charging requests, and these tuples for all the regions form a set of charging requests $\mathcal{C} = \{C_j\}$. The CSP is defined as follows.

**Charging Station Placement (CSP):** Given the set of charging requests $\mathcal{C}$ from historical data gathered offline, find a location set $\mathcal{S}$, $|\mathcal{S}| = k$, so that locations in $\mathcal{S}$ have sufficient solar energy, and $\max\{p_i \in \mathcal{C}, d(p_i, q) : \forall q \in \mathcal{S}, \}$ is minimized. The problem is defined as the following,

$$\mathbf{P1}: \min\{\max_{p_i \in \mathcal{C}, q \in \mathcal{S}} \{d(p_i, q)\}\}, \tag{3}$$

**Subject to**

$$|\mathcal{S}| = k \tag{4}$$

$$E[GHI(s)] \geq \beta, \forall s \in \mathcal{S}, \tag{5}$$

where *GHI* is *Global Horizontal Irradiance*, and $E[GHI(s)] \geq \beta$ makes sure the harvested solar energy is larger than a lower threshold of $\beta$. CSP finds a group of locations for charging stations, so that for any position in the set of charging request $\mathcal{S}$, there are always enough charging stations near them to satisfy the energy demands and the largest distance from any position to these charging stations is minimized. Next, we prove the problem is NP-hard.

**Theorem 1.** CSP is NP-hard.

*Proof.* The problem can be reduced to the Facility Location Problem (FLP). For given positions $\mathcal{C}$, FLP seeks a location set $\mathcal{S}$, $|\mathcal{S}| = k$ so that $\max_{p_i \in \mathcal{C}}\{\min_{q \in \mathcal{S}}\{d(p_i, q)\}\}$ is minimized. Considering the definitions of both CSP and FLP, if $m_i$, the number of charging stations needed, is set to 1 in CSP, and relax the restrictions of the location set $\mathcal{S}$ with sufficient solar energy, then the CSP is reduced to FLP. Since FLP is NP-hard [33], CSP is also NP-hard. □

### 4.2 $(2+\epsilon)$-factor Offline Algorithm

Since CSP is NP-hard, no algorithm can achieve optimality in polynomial time unless $P = NP$. To this end, we pursue the direction of approximation algorithm. The intuition is to make sure the collection of charging stations covers as many energy requests as possible with minimum traveling distance. The algorithm is described below (Algorithm 1).

1) *Preparation.* We select a number of $m_i$ locations with the largest expected solar energy from the neighborhood of each location $p_i \in \mathcal{C}$. This new set of candidate locations of charging stations is denoted by $\mathcal{A}$.
2) *Selection.* We first pick an arbitrary location in $\mathcal{A}$ to deploy the first charging station, denoted by $H_1$. Then we assign all the remaining locations to the cluster of $B_1$ associated with $H_1$. Next, we find the location in $B_1$ furthest from $H_1$, and designate it as the second charging station $H_2$. For each location $u$ in the original $B_1$, if $d(u, H_2) \leq d(u, H_1)$, then $u$ is re-assigned to $B_2$ associated with $H_2$; otherwise, $u$ stays in its original cluster $B_1$.
3) *Adjustment.* For $j$ clusters $\{B_1, B_2, \ldots, B_j\}$ associated with charging stations $\{H_1, H_2, \ldots, H_j\}$, choose a location $u$ from the charging stations which is the furthest from its assigned charging station, and locate the $(j+1)$-th charging station $H_{j+1}$ at location $u$. For any station $u$ in the original clusters, if $d(u, H_{j+1}) \leq d(u, H_q)$, where $H_q$ is $u$'s charging station assigned originally, then $u$ is re-assigned to the charging station $H_{j+1}$; otherwise, $u$ stays in with $B_q$. This procedure is continued until a total of $k$ charging stations have been established.

**Algorithm 1:** $(2+\epsilon)$-Charging Station Placement Algorithm (Offline)

**Input**: Set of charging requests $\mathcal{C} = \{C_i : C_i = (p_i, m_i)\}$, distribution of solar energy, neighborhood radius $r$, set of candidates locations $\mathcal{A} \leftarrow \phi$, # of stations to establish $k$.
**Output**: locations of charging stations $\mathcal{H}$.
**for** $i = 1, 2, \ldots, |\mathcal{C}|$ **do**
  $\mathcal{A} \leftarrow \mathcal{A} \bigcup \{m_i$ largest solar-rich locations in $r$ range of $p_i\}$
Pick any point $v \in \mathcal{A}$; $H_1 \leftarrow v$, $B_1 \leftarrow \mathcal{A} \setminus \{v\}$; $j = 1$.
**while** $j \leq k-1$ **do**
  $H_{j+1} \leftarrow \underset{\forall 1 \leq i \leq j, \forall u \in B_i}{\arg\max} d(u, H_i)$, $B_{j+1} \leftarrow \phi$.
  $\forall 1 \leq i \leq j, \forall u \in B_i$
  **if** $d(u, H_{j+1}) \leq d(u, H_i)$ **then**
    $B_{j+1} \leftarrow B_{j+1} \bigcup \{u\}$, $B_i \leftarrow B_i \setminus \{u\}$, $j \leftarrow j+1$.
  $j \leftarrow j+1$

The time complexity is analyzed below. For the first step of finding candidate locations, the time requirement for $m$ candidate locations is $O(m)$. For determining cluster head, each one has complexity of $O(m)$ and there are $k$ in total. Therefore, the total complexity for our algorithm is $O(km)$.

**Theorem 2 (Approximation Bound).** Algorithm 1 has $(2+\epsilon)$ approximation ratio to the optimal solution.

*Proof.* Denote the maximal distance of this solution as $SOL$. We first prove that, for this case, the solution found by the algorithm is no greater than $2h$, where $h$ is defined as the largest distance from any node $u$, to its assigned cluster head $H_j$ among all $k$ clusters. $SOL$ is the distance between two nodes, e.g., $A$ and $B$ in a cluster, $SOL = d(A, B)$. For the cluster head $H_i$ associated with $A$ and $B$, $d(H_i, A) \leq h$ and $d(H_i, B) \leq h$. Therefore, according to the triangle inequality, $d(A, B) \leq d(H_i, A) + d(H_i, B)$, and $SOL \leq 2h$.

Next, we prove that any two nodes in $\{H_1, H_2, \ldots, H_k, u\}$ have distance greater than or equal to $h$, i.e., $u$ is not assigned to another cluster head yet. It implies that $d(u, H_i) \geq h$. Since $u$ is not picked while looking for $H_2$, $d(u, H_1) \leq d(H_1, H_2)$. Since $h \leq d(u, H_1)$, $h \leq d(H_1, H_2)$. By applying similar procedure in the process of determining $H_3$, it is proved that $h \leq d(H_1, H_3)$ and $h \leq d(H_2, H_3)$. By continuing this process for all $k$ cluster heads, we conclude that the distance between any two heads in $\{H_1, H_2, H_3, \ldots, H_k\}$ is at least $h$.

Suppose the optimal solution has the largest distance denoted by $OPT$. Since $\{H_1, \ldots, H_k, u\}$ exists, according to the *pigeonhole principle*, there is at least one cluster in the optimal solution containing two or more nodes in the above set. Since it has been proved that the distance between any two nodes in the set is at least $h$, $OPT \geq h$. With $SOL \leq 2h$, $SOL \leq 2 \cdot OPT$.

For candidate locations in the neighborhood within range of $r > 0$, the distance between any two locations $A$ and $B$ is no greater than $d(A, B) + 2r$. The new solution $SOL^*$ is no greater than $SOL + 2r$, $SOL^* \leq 2(OPT + r)$. Since $r$ is much smaller than the $OPT$, the approximation bound is $SOL^* \leq (2+\epsilon)OPT$. □

*Example.* An example of CSP is demonstrated in Fig. 4 when $k = 3$. $(x_i, 5)$ stands for a charging request located at $x_i$ with a number of 5 times. $(p_j, 100)$ is the 100 aggregated charging requests in a region, represented by the center $p_j$ as a weighted average of energy requests. As shown by Fig. 4, 3 charging stations are picked in the order of the arrows from some candidate locations with sufficient solar energy. Then the requests are assigned to their closest charging stations.

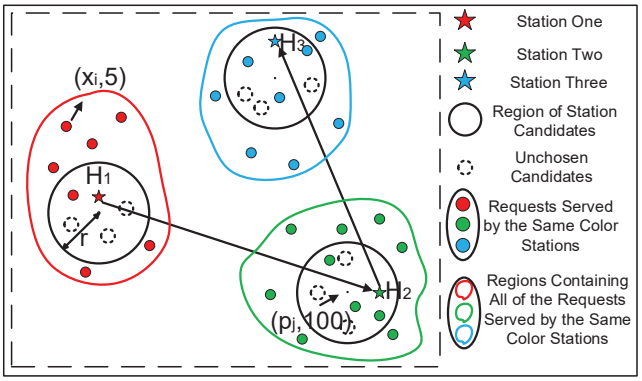



Fig. 4: Example of solving CSP when $k = 3$.

### 4.3 Update Charging Stations (Online)

Near-optimal solutions can be achieved offline if the successive occurrences of charging requests are known. A method is to use machine learning for prediction. Then Alg. 1 is applied to derive the initial deployment of the charging stations that can "best" accommodate the future energy income and demands.

Nevertheless, the future occurrence could exhibit significant deviation from the historical data. For example, the spatial-temporal variation of energy is subject to the seasonal change of sun's angle towards earth surface. The building obstructions and natural surroundings may have different impact on the energy captured based on the actual locations and the time of the year. Similarly, the patterns of utilization may undergo substantial variation because of traffic, construction and planning. As a result, to adapt the varying nature of these factors, new charging stations should be added while the ones with low utilization should be removed or relocated. Based on the offline solutions on the historical data, we propose a stochastic online algorithm to adapt to these changes.

**Adding**. Whenever a charging request $r_i$ is received, its position $x_i$ is recorded. There are two criteria of determining whether a new charging station should be established at location $x_i$ or not: 1) There is no charging station within distance $d$ from $x_i$, where $d = E[|h - h'|]$, $\forall h, h' \in \mathcal{H}$ (the average distance between any two stations). 2) With probability $1/f$ (or equal to one if $f < 1$), the location $x_i$ is chosen as the new location, where $f$ is the cost of establishing one charging station. When these two criteria are satisfied, a new charging station is introduced at location $x_i$.

**Pruning**. A station contains multiple charging piles and each one maintains a counter of $n_i$ for the number of AVs served per day. The expected number of vehicles served is $e$, and the cost of removing one charging pile is $f$ (originated from removing the previous installation). We mandate a charging pile to be removed stochastically with probability $\max(0, 1/f - n_i/ef)$. If there is no AV served at the charging pile during that day, it is removed with probability $1/f$; if

**Algorithm 2:** Station Adjustment Algorithm (Online)

**Input**: Set of charging requests $\mathcal{R}$, set of charging stations $\mathcal{H}$, the installation cost $f$ of one station, number of served requests $n_i$ at $i$-th station during a day, expected number of served requests $e$.
**Output**: New set of charging stations $\mathcal{H}$.
**for** $\forall r_i \in \mathcal{R}$ **do**
  **if** $d(r_i, h) > E[\|h - h'\|],\ \forall h, h' \in \mathcal{H}$ **then**
    $z \leftarrow$ random number in $[0, 1]$
    **if** $z \leq 1/f$ **then**
      $\mathcal{H} \leftarrow \mathcal{H} \bigcup \{x_i\}$
**for** $H_j \in \mathcal{H}$ **do**
  $z \leftarrow$ random number in $[0, 1]$
  **if** $z \leq \max(0, 1/f - n_i/ef)$ **then**
    $\mathcal{H} \leftarrow \mathcal{H} \setminus \{H_j\}$

$n_i \geq e$ (i.e. the served requests are more than expectation), then the removal probability is 0; if $0 < n_i < e$, the probability is between 0 to $1/f$. The online algorithm is summarized in Algorithm 2.

## 5 Assignment of Charging Requests

After the charging stations have been established, the runtime performance of the system necessitates coordination between the charging infrastructure and the AVs; otherwise, the AVs would simply swarm into the closest stations that ultimately causes energy depletion at some popular locations. To tackle this problem, we design an assignment algorithm to re-route some of the requests and achieve the following objectives: 1) the traveling cost of AVs to the assigned charging stations is minimized; 2) no station depletes its energy storage.

For optimal scheduling, the system does not make online decisions that take streaming requests for immediate response. Instead, it plans ahead to forecast the number of $n$ charging requests that would be sent from the AVs depending on their energy status. For example, if the destination is still far but the battery is running low, the system could provide an estimate of when the charging request would be sent and form a number of pending requests. This way, the charging assignment can be conducted more effectively for a better solution. Thus, we consider an offline setting of the assignment problem.

**Charging Assignment Problem (CAP):** The goal is to assign $n$ charging requests (of energy demands $E_i$) to $m$ stations. Compared with traditional AVs charged by the stationed connected to the power grid, the solar-powered AVs are constrainted by the output of harvested solar energy. Therefore, the energy constraint must be considered in the formulation of the Charging Assignment Problem here. With the predicted energy income $s_j$ (from which the charging consumptions have been subtracted), assigning which request to which station is governed by the following aspects. First, from the perspective of the passengers who receive the charging service, the extra mileage traveled from $x_i$ to the designated station $h_j$ is a dominant factor of user satisfaction. It is proportional to the distance measure of $L - d(i, j)$, in which $d(i, j)$ is the traveling distance between location $i$ and $j$, and $L$ is a large number to make $L - d(i, j)$ positive. In other words, a larger $d(i, j)$ results a lower satisfaction and vice versa. The objective is to maximize user satisfaction of all $n$ requests. Meanwhile, we should guarantee that the total energy requests assigned to a station do not exceed the expected energy income. The problem is formalized below.

$$\mathbf{P2}: \quad \max \sum_{j=1}^{m} \sum_{i=1}^{n} d_{ij} y_{ij} \tag{6}$$

**Subject to**

$$\sum_{i=1}^{n} E_i y_{ij} \leq s_j, \tag{7}$$

$$\sum_{j=1}^{m} y_{ij} = 1, \tag{8}$$

where $d_{ij} = L - d(i, j)$ is the user satisfaction, $y_{ij}$ is a 0-1 decision variable of whether a request is assigned to station $y_j$. Eq. (6) maximizes user satisfaction in terms of distance. Eq. (7) ensures the energy demand assigned to each station is bounded by the harvested energy. Eq. (8) states that each charging request is assigned to one station.

### 5.1 Solution by Dynamic Programming

CAP can be solved in polynomial time by converting it to the Knapsack problem [31], which finds the most valuable items to fit into a fixed-size knapsack. Here, the problem has a difference since the user satisfaction for each request would change during assignment. If a request has not been assigned to any station yet, the satisfaction is $d_{ij}$ for being assigned to station $j$; if the request has been assigned to another station $l$, the satisfaction becomes $d_{ij} - d_{il}$. Note that the former assignment of some requests may be altered by the process of another station. Since the satisfaction is updated to $d_{ij} - d_{il}$, the request is less likely to be assigned to $j$, and if this happens, it means the new assignment can always increase the total satisfaction.

We leverage dynamic programming to solve the problem, which efficiently trades computational time with memory space. The key step is to comp up with the transition from step $i$ to $i + 1$ assuming we know the optimal solution at step $i$. The optimal assignment with the maximum satisfaction for the first $i$ charging requests with $s$ harvested energy is denoted by $F(i, s)$. $F(0, s) = 0$ $(0 \leq s \leq s_j)$ is set as the initial value. $F(i, s)$ is updated of $i$ and $s$ towards the number of requests $n$ and maximum harvestable energy $S$,

$$F(i+1,s) = \begin{cases} F(i,s), & E_{i+1} > s \\ \max(F(i,s), F(i,s-E_{i+1}) + d_{i+1,l}), & E_{i+1} \leq s. \end{cases}$$

The dynamic programming method runs in $\mathcal{O}(s_i n)$ time with $\mathcal{O}(s_i n)$ space that finds the optimal solution of each assignment. During the process, $\mathcal{O}(m)$ is needed for updating the assignment for each station, with a total of $m$ times. Therefore, the time complexity is $\mathcal{O}(snm) + \mathcal{O}(m^2)$. Since the Knapsack solution provides an $(\alpha + 1)$-approximation [23] to the assignment problem, with dynamic programming $(\alpha = 1)$, the approximation ratio is 2 compared to optimal assignments.

## 6 Energy-aware Routing of AV

With the additional capabilities to harvest solar power, we also integrate energy-aware route planning for AVs into the optimization framework. Recall that the AVs have two ways to replenish their battery energy either through the charging station or from the solar rooftop. The previous charging station placement and request assignment guarantee the regular energy replenishment is satisfied. In this section, with additional capabilities to harvest solar energy anywhere from the rooftop, we consider energy-aware route planning for AVs.

**Energy-aware Route Planning**. The AVs can select a path with more solar exposure if the candidate paths are identical in traveling time (energy consumption). Hence, route planning should consider factors of: 1) energy harvested and consumed from a chosen path, 2) residual energy of the AV. Here, we consider the general case that energy consumption is proportional to the traveling time. The AV travels from the source to destination in a grid-based coordinate system following different paths. Our objective is to maximize the residual energy of the AV by selecting a path $P$, while making sure the residual energy is above a lower threshold.

The problem is analogous to the Traveling Salesman Problem with Profits, a variant of the classic Traveling Salesman Problem (TSP). TSP aims to find the shortest path traversing a set of locations exactly once [24]. In addition to the traveling cost, a reward $p_i$ is associated with each vertex. The problem finds the shortest path with the maximum profits. These two objectives are indeed conflicting, since the first objective urges the salesman to travel as less as possible while the second one encourages him to traverse as many vertices as possible to maximize the profits collected. In close analogy, the profits here are the solar energy enroute to be harvested and the objective is to minimize the energy consumed (assuming it is proportional to traveling distance) while maximizing the solar energy collected. The problem is formulated in the following,

$$\mathbf{P3}: \quad \min \sum_{i,j\in\mathcal{V}} d(i,j)a_{ij} - \sum_{i\in\mathcal{V}} p_i y_i \tag{9}$$

**Subject to**

$$\sum_{j\in\mathcal{V}\setminus\{i\}} a_{ij} = y_i, \tag{10}$$

$$\sum_{i\in\mathcal{V}\setminus\{j\}} a_{ij} = y_j, \tag{11}$$

$$\overline{P}\cdot \sum_{i,j\in\mathcal{V}} d(i,j)a_{ij} < E_{AV}, \tag{12}$$

$$y_1 = 1, \tag{13}$$

$$a_{ij} \in \{0,1\}, \tag{14}$$

$$y_j \in \{0,1\}, \tag{15}$$

where $i$, $j$ represents the index of locations from a vertex set $\mathcal{V}$, $a_{ij}$ is an indicator denoting whether the arc between $i$ and $j$ is chosen or not (1 for chosen, 0 for not), $y_i$ is the indicator denoting whether location $i$ is chosen or not, $\overline{P}$ is the average power consumed for unit distance. Eq.(10) and Eq.(11) are the assignment constraint, Eq.(12) ensures the chosen path does not deplete the residual energy of AV, Eq.(13) ensures the chosen of starting location, and Eq.(14) and Eq.(15) represent the indicators.

**Greedy Algorithm.** We propose a greedy algorithm. Consider the map with grids $\{x_{ij}\}$ and the mileage measured in Manhattan distance. According to the forecast, each grid $x_{ij}$ has a potential solar energy income $s_{ij}^t$ during time period $t$. At $x_{ij}$, the AV can move to four adjacent grids $\{x_{i-1,j}, x_{i+1,j}, x_{i,j-1}, x_{i,j+1}\}$ in the next step, that either increases or decreases the distance by one unit towards the destination. Based on the solar energy at these locations, profits can be calculated. Take $x_{i-1,j}$ for example, assume that moving to this location would decrease the distance to the destination by 1, then the profit is $p_{i-1,j} = s_{i-1,j}^t - 1$. In each step, the AV moves to the grid with the largest profit if the energy constraint is not violated. Before taking this move, the AV also makes sure that there is at least one charging station within the operating range of the AV before energy depletion. If a location has the largest profit but fails to satisfy the previous condition, it is not chosen. The above process is repeated until the AV reaches the destination.

**Time Complexity.** At each step, it takes constant time to find the next destination with the largest profit without violating the energy constraint. While the maximum number of steps is in the order of the number $n$ of grids, the time complexity of the proposed greedy algorithm is $\mathcal{O}\{n\}$.

## 7 Scheduling of Shared Autonomous Electric Vehicles

We have constructed a vehicular network of autonomous vehicles that can be operated solely based on solar energy in previous sections. Such a framework makes the system consisting of shared AVs that can serve the citizens being possible. This section will discuss this promising application of the proposed framework and provide efficient algorithms with theoretically guaranteed performance. Note that this section focuses on the task assignment problem of shared AVs. As for when, where, and how the shared AVs may replenish their energy needs to be addressed using the proposed methods in Sec.5, Sec.6, and Sec.7 to form a holistic system.

### 7.1 Scheduling Trips for One AV

Benefiting from autonomous driving, the operation and scheduling of taxis in future smart cities must also be highly intelligent. By getting rid of taxi drivers, the autonomous electric taxis can operate 24 hours a day and seven days a week if possible, significantly improving the usage and efficiency of an electric vehicle. Using such a system depending on AVs, users just need to post their position, beginning time $b_i$, and finishing time $f_i$, and then a taxi will be assigned by the system to pick them up and finish the trip. In this way, taxis in the future will be more like shared vehicles among people, which motivates us to call such scenarios the *Shared AVs*.

From the perspective of the taxi, it aims to maximize the number of trips it fulfills to consider the profits and social efficiency jointly. Note that the shared AVs are treated like public facilities, which naturally motivates us to use the number of trips served as the optimization goals instead of other matrices such as the total money earned in a certain period. However, besides the ability of perpetual operation,

another significant difference between the shared AVs and the ordinary taxis is the time spent for charging: the ordinary cars using fuels can refill their tank immediately, while the time spent for charging the shared AVs should not be neglectable due to the limitation of the charging speed. Therefore, when the system intends to schedule the routing of AVs, it should also spare some time to charge the AVs to avoid the depletion of energy. The taxi is restricted to serving the requests in a region to avoid traveling too far to catch the following passengers. Due to this restriction, the average time spent to travel to the following passengers can be derived according to the analysis of the collected data or a prior survey, which is added to the starting time of each trip. Therefore, the reported starting time of each trip is earlier than the actual starting time, which allows efficient time for the AV to pick up the passenger. The reported finishing time is reasonable and precise by analyzing the usual time spent on the trip in corresponding traffic volume. A more precise solution can be derived based on the exact locations of the origin and destination of each trip and the exact time requirement. However, considering these makes the problem intractable. The Maximizing Utility of Shared Autonomous Vehicle (MUSAV) problem is formulated as following,

$$\mathbf{P4}: \quad \max_{z_i} |\mathcal{U}| \tag{16}$$

**Subject to**

$$z_i = 0 \text{ or } 1, \forall i, \tag{17}$$

$$t_{ch} \cdot P_{ch} + E_{AV} \geq \sum_{i \in \mathcal{U}} (f_i - b_i) \cdot \overline{P}_{tr} + E_{cs}, \tag{18}$$

$$t_{ch} + \sum_{i \in \mathcal{U}} (f_i - b_i) + t_{cs} \leq T. \tag{19}$$

$$b_i + \overline{w} \leq f_i, \forall i, \tag{20}$$

$$\mathcal{U} = \{i | z_i = 1, \forall i\}, \tag{21}$$

$$f_i \leq b_j, \text{ or } f_j \leq b_i, \forall i, j \in \mathcal{U}, \tag{22}$$

For a given period $T$, MUSAV aims to maximize the number of trips that one AV fulfills by choosing the trips it accepts carefully. $z_i$ is the indicator showing whether the trip $i$ is chosen or not. $t_{ch}$ is the time spent for charging, $P_{ch}$ and $\overline{P}_{tr}$ are the average charging power and average power spent for traveling, $E_{AV}$ is the current energy of AV, and $E_{cs}$ and $t_{cs}$ are the energy and time spent to reach the charging station. Eq. (18) ensures the energy that AV receives is larger than the energy spent, while Eq. (19) ensures that there is enough time left for charging the AV. Eq. (20) is the natural requirement that the beginning time of a trip plus the average waiting time $\overline{w}$ is no later than its finishing. $\mathcal{U}$ is the set of trips accepted by the AV, and Eq. (22) ensures any two trips accepted by the AV do not have any overlapping since the AV is not allowed to serve two requests simultaneously.

According to P4, the AV can choose any set of trips that do not coincide with each other if only the constraints about the energy are satisfied. Since $t_{ch}$ appears both in Eq. (18) and Eq. (19), combining these two constraints, the condition which needs to be satisfied to guarantee the energy being consistently enough for AV is as following,

$$\sum_{i \in \mathcal{U}} (f_i - b_i) \leq \frac{T \cdot P_{ch} + E_{AV} - E_{cs}}{\overline{P}_{tr} + P_{ch}}. \tag{23}$$

**Algorithm 3:** Earliest Starting Time Algorithm

```
Input: n pending trips Z = {z_i} in a certain period T, z_i = (b_i, f_i).
Output: The set of chosen trips U for one AV.
U ← φ
while Σ_{i∈U}(f_i − b_i) ≤ (T·P_ch + E_AV − E_cs)/(P̄_tr + P_ch) do
    if Ẑ ≠ φ then
        z_h ← arg min_{z_i∈Ẑ}{b_i}, U ← U ⋃ {h}
        Ẑ ← Ẑ \ {z_j : z_j ∩ z_h ≠ φ, ∀j}
AV fulfills the trips according to the time sequence of U
```

Since the right-hand side only contains constants, the sum of the duration of trips only needs to be smaller than a constant.

### 7.2 Earliest Starting Time Algorithm

The MUSAV problem is apparently NP-hard. First, a heuristic greedy algorithm named Earliest Starting Time Algorithm is proposed. This algorithm will find the trip with the earliest starting time in the set of pending trips and check whether adding this new trip will still satisfy the constraint Eq. (23) or not. If the equation is not satisfied anymore, then the algorithm ends, and the AV will serve the current set of chosen trips; If the equation is still satisfied, it will remove all the pending trips that have any overlapping with the chosen trip from the set, since one AV can only execute one trip at the same time. The whole processes are conducted iteratively until the algorithm ends, or there are no more pending trips in the set. This algorithm is summarized in Algorithm 3.

However, this algorithm does not have any performance guarantee, i.e., the achieved utility by the algorithm is unbounded compared with the optimal solution. One example is constructed to show such a case. The time starts at 0. The starting and finishing time of the set of trips are as following,

$$\begin{cases} b_1 = 0, f_1 = \dfrac{T \cdot P_{ch} + E_{AV} - E_{cs}}{\overline{P}_{tr} + P_{ch}}; \\ b_i = (i-1) \cdot \delta + (i-1) \cdot \overline{w}, f_i = (i-1) \cdot (\delta + \overline{w}), \\ 2 \leq i \leq \left\lfloor \dfrac{T \cdot P_{ch} + E_{AV} - E_{cs}}{(\overline{P}_{tr} + P_{ch}) \cdot (\delta + \overline{w})} + 1 \right\rfloor, \delta \approx 0. \end{cases}$$

The AV will choose the first trip $(b_1, f_1)$ using the proposed greedy algorithm. Since this trip is the longest trip that can be accepted, no more trips can be added, which achieves the utility of 1. However, if the AV first serves $(b_2, f_2)$, which starts a very short time $\delta$ after $b_1$, then the utility can be as much as $\lfloor (T \cdot P_{ch} + E_{AV} - E_{cs})/(\overline{P}_{tr} + P_{ch}) \cdot (\delta + \overline{w}) + 1 \rfloor$. Since $T$ is the only changing variable in the utility achieved by the second method, the ratio of the utility achieved by the non-greedy and greedy method is unbounded as $T$ approaches to $\infty$. Therefore, the proposed greedy algorithm does not have any performance bound.

*Time complexity.* If the total number of trips in the set is $n$ at the very beginning, finding the trip with the earliest starting time takes $\mathcal{O}(n)$ time, and removing the trips that

**Algorithm 4:** Shortest Trip Algorithm

**Input**: $n$ pending trips $\mathcal{Z} = \{z_i\}$ in a certain period $T$, $z_i = (b_i, f_i)$.
**Output**: The set of chosen trips $\mathcal{U}$ for one AV.
$\mathcal{U} \leftarrow \phi$
**while** $\sum_{i \in \mathcal{U}} (f_i - b_i) \leq \frac{T \cdot P_{ch} + E_{AV} - E_{cs}}{\overline{P}_{tr} + P_{ch}}$ **do**
  **if** $\widehat{\mathcal{Z}} \neq \phi$ **then**
    $z_h \leftarrow \arg\min_{z_i \in \widehat{\mathcal{Z}}} |f_i - b_i|, \mathcal{U} \leftarrow \mathcal{U} \bigcup \{h\}$
    $\widehat{\mathcal{Z}} \leftarrow \widehat{\mathcal{Z}} \setminus \{z_j : z_j \cap z_h \neq \phi, \forall j\}$
AV fulfills the trips according to the time sequence of $\mathcal{U}$

overlap with the chosen trip takes $\mathcal{O}(n)$ time. The number of iterations is in the order of $\mathcal{O}(T)$, so the total time complexity is in the order of $\mathcal{O}(Tn)$

### 7.3 Approximation Algorithm

The previous greedy algorithm does not have any guaranteed performance bound, making it difficult for the system to picture the optimal solution based on the results derived by the greedy algorithm. The given counter example shows that the algorithm can not compete with the optimal solution since the duration of the trip is not considered such that the longest trip is picked at the beginning. Therefore, to have theoretically guaranteed performance bound in any scenario, an approximation algorithm is proposed.

The approximation algorithm first picks the shortest trip from the current set of pending trips and check whether condition Eq. (23) is satisfied or not. If the condition is no longer satisfied, then the algorithm terminates and outputs the chosen trip set; otherwise, all the trips overlapping with the chosen shortest trip are removed from the set of pending trips. These whole processes are repeated iteratively until the condition Eq. (23) is not satisfied, or the set of pending trips have been depleted. The algorithm is named Shortest Trip Algorithm (STA) and summarized in Algorithm 4. The approximation ratio of the algorithm is proved in the following.

**Theorem 2.** The Shortest Trip Algorithm has an approximation ratio of three compared with the optimal.

*Proof.* The sets of trips derived by STA algorithm and the optimal solution are $\mathcal{U}$ and $\mathcal{U}^*$. We can first build a map from $\mathcal{U}^*$ to $\mathcal{U}$ following the rules: 1) If the trip is in both sets, then the trip will be mapped to itself; 2) If the trip $I$ is in $\mathcal{U}^*$ but not in $\mathcal{U}$, but $I$ overlaps with a trip in $\mathcal{U}$, then $I$ will be mapped to the shortest overlapped trip in $\mathcal{U}$; 3) If the trip is in $\mathcal{U}^*$ but not in $\mathcal{U}$, and $I$ does not overlap with any trip in $\mathcal{U}$, then $I$ will be mapped to the longest trip $L$ in $\mathcal{U}^*$. These are all the three possible cases of trips in $\mathcal{U}^*$

Through these mapping functions, any trip in $\mathcal{U}^*$ is mapped to a trip in $\mathcal{U}$. From the perspective of the trips in $\mathcal{U}$, there are only two cases: 1) the trip $J$ in $\mathcal{U}$ that is not the longest trip 2) the longest trip $L$ in $\mathcal{U}$. For case 1, there are at most two trips in $\mathcal{U}^*$ can intersect with $J$, otherwise the third one in $\mathcal{U}^*$ intersecting with $J$ must be contained by $J$, which contradicts the rule that $J$ is the shortest interval chosen from the set of pending trips since we can always switch it with the shorter trip without any cost. For case 2, there are at most $|\mathcal{U}|$ trips in $\mathcal{U}^*$ that can be mapped to $L$. These trips all have a longer duration than $L$ since $L$ is the last chosen shortest trip. If there are $|\mathcal{U}| + 1$ such trips, we can always find one more trip adding to $\mathcal{U}$ without violating the time constraint, which is a contradiction to the truth that the algorithm terminates at generating $\mathcal{U}$. Therefore, we can conclude that $|\mathcal{U}^*| \leq 3|\mathcal{U}|$. □

*Time complexity.* The finding of the shortest interval takes time of $\mathcal{O}(n)$, and removing the overlapping trips also takes $\mathcal{O}(n)$. There are at most $\mathcal{O}(T)$ iterations due to the lower bound of each trip. Therefore the time complexity is in order of $\mathcal{O}(Tn)$. The approximation algorithm takes the same complexity as the earliest starting time algorithm, but guarantees that the achieved utility by one AV always choosing the shortest trip is at $1/3$ of the optimal solution.

### 7.4 Scheduling Trips for Multiple AVs

We have discussed the scheduling of one AV to maximize its utility. However, the shared AVs are public assets whose total utilities are more reasonable metrics to maximize. In a more realistic setting, multiple shared AVs operate in a region to serve the people there. Maximizing the sum of utility of all AVs brings maximum social benefits. Based on this new objective, the Maximizing Utility of k Shared Autonomous Vehicles (k-MUSAV) problem is formulated in the following,

$$\mathbf{P5}: \quad \max_{z_i^l} \sum_{1 \leq l \leq k} |\mathcal{U}^l| \tag{24}$$

**Subject to**

$$z_i^l = 0 \text{ or } 1, \forall i, l, \tag{25}$$

$$t_{ch} \cdot P_{ch} + E_{AV}^l \geq \sum_{i \in \mathcal{U}^l} (f_i^l - b_i^l) \cdot \overline{P}_{tr} + E_{cs}, \forall l \tag{26}$$

$$t_{ch} + \sum_{i \in \mathcal{U}^l} (f_i^l - b_i^l) + t_{cs} \leq T. \tag{27}$$

$$b_i^l + \overline{w} \leq f_i^l, \forall i, l, \tag{28}$$

$$\mathcal{U}^l = \{i | z_i^l = 1, \forall i\}, \forall l \tag{29}$$

$$f_i^l \leq b_j^l, \text{ or } f_j^l \leq b_i^l, \forall i, j \in \mathcal{U}, \forall l. \tag{30}$$

P5 is similar to P4 except that each AV $l$ should satisfy its own time constraint,

$$\sum_{i \in \mathcal{U}^l} (f_i^l - b_i^l) \leq \frac{T \cdot P_{ch} + E_{AV}^l - E_{cs}}{\overline{P}_{tr} + P_{ch}}, \forall l. \tag{31}$$

P5 aims to maximize the sum of the utility of all $k$ AVs.

In order to address the k-MUSAV problem with guaranteed performance bound, a new algorithm named $k$ Vehicles Shortest Trip Algorithm (k-VSTA) is proposed based on the previously proposed STA algorithm. For the AVs numbering from 1 to $k$, the system starts from the first vehicle, choosing the shortest interval from the set of pending trips, especially for the first vehicle. Then it checks whether the time constraint Eq. (31) is satisfied or not. It will only assign the chosen trip to the first vehicle if the condition is satisfied. It removes the chosen trip from the sets of pending trips for the remaining $k - 1$ vehicles and the overlapping trips from the set of the first vehicle. It will continue this process sequentially until the $k$-th vehicle form one iteration.

**Algorithm 5:** $k$ Vehicles Shortest Trip Algorithm

**Input**: Number of AVs $k$, $n$ pending trips $\mathcal{Z} = \{z_i\}$ in a certain period $T$, $z_i = (b_i, f_i)$.
**Output**: The set of chosen trips $\mathcal{U}$ for one AV.
$\mathcal{U} \leftarrow \phi$
**while** $\exists\ l$ *s.t.* $\sum_{i \in \mathcal{U}^l}(f_i^l - b_i^l) \leq \frac{T \cdot P_{ch} + E_{AV}^l - E_{cs}}{\overline{P}_{tr} + P_{ch}}$ **do**
  **if** $\exists\ l$ *s.t.* $\widehat{\mathcal{Z}}^l \neq \phi$ **then**
    **for** $l = 1, 2, 3, \ldots, k$ **do**
      $z_h^l \leftarrow \arg\min_{z_i \in \widehat{\mathcal{Z}}^l} |f_i^l - b_i^l|, \mathcal{U}^l \leftarrow \mathcal{U}^l \bigcup \{h\}$
      $\widehat{\mathcal{Z}}^l \leftarrow \widehat{\mathcal{Z}}^l \setminus \{z_j : z_j \cap z_h^l \neq \phi, \forall j\}$
      $\widehat{\mathcal{Z}}^{-l} \leftarrow \widehat{\mathcal{Z}}^{-l} \setminus z_h^l$
      **if** $\sum_{i \in \mathcal{U}^l}(f_i^l - b_i^l) > \frac{T \cdot P_{ch} + E_{AV}^l - E_{cs}}{\overline{P}_{tr} + P_{ch}}$ **then**
        $l$-th vehicle will be not be eligible for choosing trips anymore.
k AVs fulfill the trips according to the time sequence of $\mathcal{U}^l$ respectively.

The processes will be executed iteratively until either the time constraint is not satisfied for any AV or the set of all pending trips is depleted. This algorithm is summarized in Algorithm 5, and it can be proved that this algorithm also has an approximation ratio compared with the optimal.

**Theorem 3.** The k Vehicles Shortest Trip Algorithm has an approximation ratio of $2k+1$ compared with the optimal.

*Proof.* $\mathcal{U}^*$ and $\mathcal{U}$ are the unions of the chosen sets of all $k$ AVs for the optimal and the k-VSTA algorithm, i.e. $\mathcal{U}^* = \cup_l\{\mathcal{U}^{l*}\}$, $\mathcal{U} = \cup_l\{\mathcal{U}^l\}$. A mapping from $\mathcal{U}^*$ to $\mathcal{U}$ is built similarly as the proof of Theorem 2. Note that the trips served by one AV cannot overlap. However, any two trips served by two different AVs can overlap. The longest trip $L$ is the longest one among all $\mathcal{U}^l$'s.

For any trip $J \in \mathcal{U}$, there are still two cases: 1) either $J$ is not the longest trip in $\mathcal{U}$; 2) or the longest trip $L$ in $\mathcal{U}$. For the first case, $J$ served by $i$-th vehicle cannot intersect with more than two trips served by $i$-th vehicle in the optimal solution; otherwise, $J$ can always be replaced by that shorter interval with no cost. $J$ can contain some trips served by other $k-1$ vehicles; however, $i$-th vehicle does not choose them since they have already been chosen by other vehicles. According to the mapping function, these contained trips are mapped to the same trips in $\mathcal{U}$ directly. Therefore, for each trip in $\mathcal{U}$ satisfying case 1, at most $2k$ trips in $\mathcal{U}^*$ are mapped to them. For the second case, it can be easily proved that the number of trips mapped to $L$ is no greater than $|\mathcal{U}|$. Summing them together, it is proved that $|\mathcal{U}^*| \leq (2k+1)|\mathcal{U}|$ □

When $k = 1$, the bound is 3, which coincides with Theorem 2. The time complexity can be analyzed similarly, which is $\mathcal{O}(kTn)$.

**Discussion.** The MUSAV and the generalized k-MUSAV problems are first proposed in this paper. Hence the k-VSTA algorithms are proposed here to ensure a guaranteed performance bound of the found solution. Otherwise, the solutions could be arbitrarily worse in some scenarios. However, being the first approximation algorithms, the solutions found by k-VSTA are also competitive to the optimal solution.

Denote the optimal solution found for $k$ vehicle system as $\mathcal{U}_k^*$, hence the average utility for each vehicle equals to $\frac{\mathcal{U}_k^*}{k}$. According to Theorem 3, the solution found by k-MUSAV is no worse than $\frac{\mathcal{U}_k^*}{2k+1}$, inducting the average utility is no worse than $\frac{\mathcal{U}_k^*}{k(2k+1)}$. Similarly, the average utility for $k+1$ vehicle system achieved by the optimal solution and k-VSTA algorithm are $\frac{\mathcal{U}_{k+1}^*}{k+1}$ and $\frac{\mathcal{U}_{k+1}^*}{(k+1)(2k+3)}$. Taking the ratio between the average utility of $k+1$ and $k$ vehicle system, it is found that ratios for the optimal solution and the k-VSTA solution are $\frac{\mathcal{U}_{k+1}^* \cdot k}{\mathcal{U}_k^*(k+1)}$ and $\frac{\mathcal{U}_{k+1}^* \cdot k}{\mathcal{U}_k^*(k+1)} \cdot \frac{2k+1}{2k+3}$. Despite of the same factor $\frac{\mathcal{U}_{k+1}^* \cdot k}{\mathcal{U}_k^*(k+1)}$, the utility of k-VSTA only times an extra factor of $\frac{2k+1}{2k+3}$, which is very close to 1 for very large $k$. Therefore, the proposed k-VSTA algorithm can maintain similar increase of utility as the optimal solution as the system size scales up (i.e. $k$ increases).

## 8 Performance Evaluations

The goal of performance evaluations is to integrate existing datasets to validate the designs of algorithms. In particular, we first evaluate the prediction of the machine learning algorithm on time series data. Based on these predictions, we evaluate the deployment strategies of charging stations, assignment algorithms and operating range of the AVs and compare them with competitive algorithms from the previous literatures or existing approaches. As a significant application of AVs, the proposed mechanisms for shared AVs are also simulated afterwards.

### 8.1 Simulation Setup

We acquire solar irradiance data from *SolarAnywhere* [35], which provides data for the continental U.S. with a resolution of 10 km. A field of $200km \times 200km$ in the metropolitan area of El Paso, TX is chosen in our simulation. 5 years (2009-2014) hourly solar irradiance data are extracted from the dataset and used in our prediction models, among which the first 3 years are used as the training data and the next 2 years are used as testing. As a proof of concept, we do not consider the complex traffic/road conditions, but set the source and destinations of the AVs in a 2D plane measured in Manhattan distance. The system schedules all the AVs to move at a constant speed of $60km/h$ to avoid congestions and the fully-charged battery can support a total mileage of $200km$ (as we set 10-30% lower than normal mileage of electrical vehicles to account for the intensive computations on-board).

Each station has a number of charging piles that can host a total 10 vehicles at the same time if the residual energy is enough for the station. The charging process takes 60 minutes from empty to full. Due to the resolution of the data source, we divide the field into grids of equal length ($1km$). Similar to Section 3, we analyze the traffic data from [28] to obtain the volume at different locations. The patterns exhibit a drastic spatial-temporal difference, based on which, we apply a conversion ratio (set to 10%) of the number of AVs in that traffic volume. This ratio can be adjusted according to the number of possessions of AVs. We

average the simulation results over 100 runs. The prediction engine is developed with Tensorflow and tested on Nvidia Tesla P100 GPU in HPC and the backend algorithms are developed in Python and MATLAB.

### 8.2 Comparison of Prediction Schemes

The accuracy of energy prediction determines the performance of the backend algorithms. To this end, we first evaluate different machine learning algorithms. Solar energy and demand represent a perfect example of time series study at each geographical locations. The objective is: given the solar energy income or the number of the charging requests for the previous one or few hours, predict these values for the next time periods. We apply the Long Short Term Memory (LSTM) network [37] to forecast the future values. LSTM is the state-of-the-art recurrent neural network that surpasses traditional structures. Each LSTM cell consists of a set of gates to remember and forget relevant information towards minimizing the loss objective. In our evaluation, a total of 64 LSTM cells are stacked as the hidden layer and the depth of the network is extended by adding the number of layers. The hourly solar income data and charging requests data in one week are fed into the prediction engine for training. The hourly data in the next two consecutive days are used for testing, which compares the predicted hourly data with the actual values.

Fig. 5 shows the results of predictions for solar energy and charging requests. For two very different solar income distribution (sunny and cloudy), LSTM can predict with a high accuracy. The highest solar income mainly comes from 10 a.m. to 18 p.m.. For the charging requests, they mainly concentrate on the working hours and reduce rapidly. There are strong demands of charging from 7 a.m to 10 a.m. and 17 p.m to 20 p.m., however the solar income encounters a deficit due to sunset. This situation has been captured by the proposed framework as discussed in the next subsections.

To show different methods of prediction, statistical methods of Moving Average (MA), and Auto-Regressive Integrated Moving Average (ARIMA) [38] are compared with LSTM. Root Mean Square Error (RMSE) is applied here as the metric to measure the distance between predicted and actual values,

$$RMSE(z^*) = \sqrt{E\big[(z^* - z)^2\big]}. \tag{32}$$

$z^*$ is the predicted value. $z$ is the actual value and $E\big[(z^* - z)^2\big]$ is the expectation of the square error. An LSTM with 2 hidden layers is evaluated with different backward time steps (back). MA is evaluated with different window sizes (wz), and ARIMA of 0 degree of differencing is evaluated with different lag order ($p$). LSTM with 3 backward hours has the overall minimum RMSE of 27.5 compared with the other two schemes, which improves 29.1 % and 26.9 % compared with the best prediction performance of MA (wz=1) and ARIMA (p=6).

### 8.3 Comparison of Charging Station Deployment

Next, we compare the proposed *offline* and stochastic *online* charging station placement algorithms with the optimal result. To provide more insights, we also benchmark our

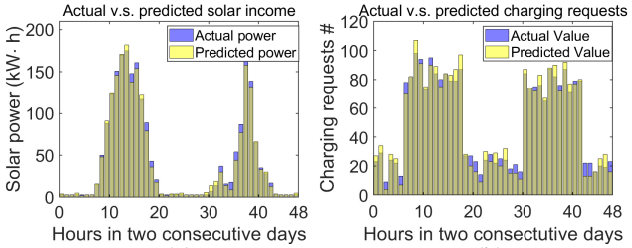



Fig. 5: Actual solar power and charging requests vs. prediction (a) solar power (b) charging requests.

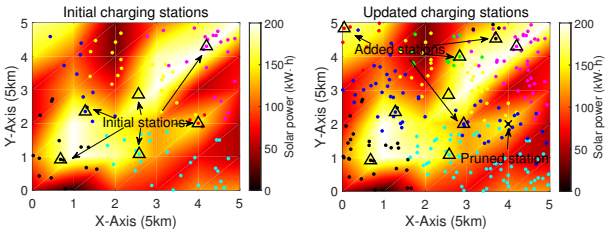
TABLE 2: Comparison of average RMSE of different prediction algorithms for solar income and charging requests (back - # backward time steps in hrs, wz - time window, $p$ - lag order, $d$ - degree of differencing.)

| LSTM | back=24 | back=12 | back=6 | back=3 | back=1 |
|---|---|---|---|---|---|
| 2-layer | 45.7 | 41.5 | 35.8 | **27.5*** | 32.1 |
| MA | wz=1 | wz=2 | wz=3 | wz=4 | wz=5 |
| | **38.8** | 43.2 | 46.5 | 51.0 | 55.8 |
| ARIMA | $p$=2 | $p$=4 | $p$=6 | $p$=8 | $p$=10 |
| $d$=0 | 50.3 | 45.2 | **37.6** | 42.5 | 47.3 |

algorithm with the classic online $k$-means clustering algorithm [36], in the similar application context. As a demonstration, we plot the energy distributions and the results in an $25 \times 25\text{km}^2$ area, with the charging requests depending the fractional volume of the traffic at that time. A total number of 150 synthetic charging requests per hour are distributed in the chosen field, whose distribution coincides with the distribution of the traffic volume by analyzing the data from [28]. The brighter color indicates higher solar energy depending on the terrains and surroundings. Different integer number $k$'s ($k \in [1, 20]$) of charging stations are plugged into different schemes, and the minimum $k$ value satisfying the energy constraint Eq.(5) is used in the optimization. Larger allowed $k$ values may decrease the derived min-max distance, which also involves more infrastructure cost of deploying more stations. However, in order to compare the performance of different schemes if adequate budget of station deployment is allowed, the experiment is also conducted for more station numbers with $\min(k) + 2$ and $\min(k)+4$, where $\min(k)$ denotes the minimum $k$ value satisfying the energy constraint.

Fig. 6 (a) shows the placement of charging stations based on historical solar distributions. The triangles and dots represent the charging stations and requests respectively, where the same color means that they are served at the same station. Instead of being simply placed at the energy-richest places, the stations tend to be more scattered to accommodate the charging requests from different locations while ensuring there are sufficient harvestable energy.

Once there is a pattern change in the distributions of traffic volume or energy distribution, the stochastic online algorithm can capture such shift and continuously update the deployments as demonstrated in Fig. 6 (b). 4 new charging stations are added to the current group to satisfy the emerging new requests around these areas because of the increase of demand. As a validation of our algorithm designs, these new locations all enjoy sufficient solar energy,




(a) (b)

Fig. 6: Comparison of offline and online station placement mechanisms (a) offline placement (b) online placement. (Best view in color)

thus can successfully handle the additional demands.

Finally, we compare the performance of different mechanisms of charging station deployment based on the metrics of *power generation*, *min-max traveling distance* of the AVs, *number of charging stations established* and *number of requests fulfilled* during the day. The optimal solution is derived if both ambient energy profiles and charging requests are known in advance. We also compare our algorithm with a competitive online $k$-means clustering algorithm [36], which finds clusters of comparable spatial extent in an online setting.

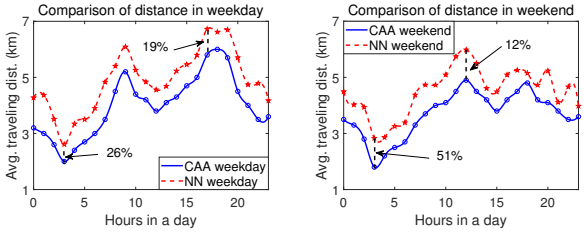
From Table 3, we can see that the online algorithm achieves the best performance as well as outperforms the offline algorithm in all four criteria. This is because the offline algorithm simply calculates the "optimal" solution based on historical data, which may be outdated. The online algorithm can capture such changes given that it generates the minimum number of stations (by pruning the low-utilized ones) while achieving the min-max traveling distance for requests served at the same station. It also generates the maximum solar energy output for higher energy storage and mitigate the impact from ambient dynamics. Compared with online $k$-means, the online algorithm generates 63% more electricity, reduces the min-max distance by 30%, saves the number of established stations by 37%, and increases the number of completed charging requests by 74%. Our online algorithm are also close to the optimal results by 10-15% off the optimal solution. For different $k$ values, the proposed online algorithm all achieves the best performance, and similar proportions of generated electricity, number of charging stations, and number of completed requests are achieved for the case of $\min(k)+2$ and $\min(k)+4$. Especially, the reduction of min-max distance increases to 40% compared with the benchmark in the case of $\min(k)+4$, since more stations allow better coverage of the field and reduce the traveling distance of vehicles significantly via the optimization.

### 8.4 Comparison of Charging Assignment Strategies

The proposed Charging Assignment Algorithm (denoted by CAA) jointly considers the energy status of charging stations and the distributions of charging requests. An intuitive strategy is to find the Nearest Neighboring (NN) station and fulfill the charging requests. In this subsection, we compare CAA with NN by measuring the average traveling distance for AVs in order to get charged by the designated station.

TABLE 3: Comparison of different mechanisms of charging station placement for different $k$ value of charging stations numbers.

| | opt.* | onl. k-means | off. | onl. | $k$ val. |
|---|---|---|---|---|---|
| solar power(kW·h) | 195.2 | 106.0 | 165.7 | **173.3** | $+0$ |
| | 246.8 | 122.1 | 206.2 | **220.2** | $+2$ |
| | 297.5 | 135.5 | 219.9 | **246.1** | $+4$ |
| min-max dist.(km) | 3.2 | 6.1 | 4.5 | **4.3** | $+0$ |
| | 3.0 | 5.7 | 4.3 | **3.9** | $+2$ |
| | 2.9 | 5.5 | 4.0 | **3.3** | $+4$ |
| # stations | 7.1 | 12.6 | 9.5 | **8.0** | $+0$ |
| | 9.1 | 14.6 | 11.5 | **10.0** | $+2$ |
| | 11.1 | 16.6 | 13.5 | **12.0** | $+4$ |
| # completed req. | 90.3 | 46.0 | 76.7 | **80.2** | $+0$ |
| | 119.2 | 55.7 | 99.8 | **111.2** | $+2$ |
| | 139.3 | 56.2 | 105.0 | **123.7** | $+4$ |




(a) (b)

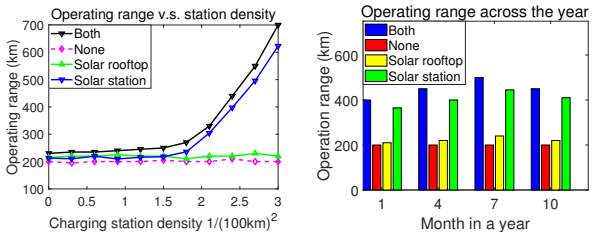
Fig. 7: Average traveling distance of AVs to get charged (a) weekdays (b) weekends.

Fig. 7 compares the two different strategies in weekdays and weekends. Through the analysis of traffic flows, we find that the traffic patterns are different for weekdays and weekends (different rush hours). In the weekday as shown by Fig. 7 (a), during the rush hours (6-9 a.m., 5-7 p.m.), the average traveling distance is the highest, since charging requests are high during these hours but the harvested solar energy is not enough. With the decline of charging requests and rising solar radiations, the traveling distance decreases around the noon time. The distance bottoms at about 3 a.m., during which the charging requests are usually the lowest (bedtime). CAA can save traveling distance from about 19% to 26% in weekday. Note that NN does not lead to less traveling distance. If the design follows the naive NN strategy, the tension between the energy income and demand is pronounced. Thus, the AVs would drain energy storage at some popular stations and push the rest customers to other stations with more distance.

Fig. 7 (b) indicates a slight shift of the busy hours from 8-12 p.m., (more sporadic during the daytime with several peaks in the afternoon). Even more than the weekdays, CAA saves the traveling distance from 12% to 51%. We find that CAA can save more traveling distance than NN during the night. Note that the charging stations requires surplus for nighttime operations, so is more likely to deplete its energy if the assignment is improperly scheduled. The proposed strategy can always result more energy storage by saving, thereby improving the system robustness.

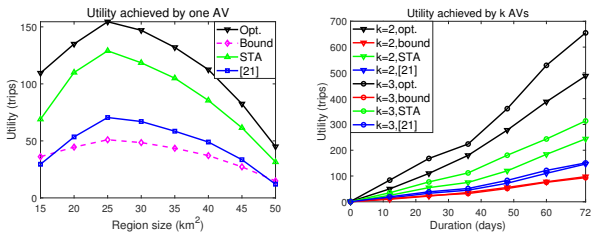
### 8.5 Operating Range of AVs

The operating range is a major concern severely affecting the wide adoption and promotion of electric AVs. In this subsection, we benchmark the proposed strategies against the traditional approaches [10], [11], [12], [16] with/without




(a) (b)

Fig. 8: Operation range of AVs. (a) Under different densities of charging station. (b) In different months across the year.




(a) (b)

Fig. 9: Utility achieved by shared AVs. (a) Utility of one AV. (b) Utility of multiple AVs.

renewable energy and show additional benefits of cost savings.

We measure the operating range of AVs for different density of charging stations during different seasons across the evaluation period. In Fig. 8(a), we compare the operating range for the "none", "solar rooftop", "solar station" and "both" schemes, which represent the AVs solely depend on themselves, with solar-harvesting rooftop, with solar charging stations, and integration of both, respectively. The range of AVs equipped with solar rooftop enjoys an incremental improvement (about 5%) compared with the traditional AVs, which verifies our findings in Section 3 (about 3%). The extra 2% improvement of operation range is actually benefited from our energy-aware route planning scheme. The results also indicate that the previous approach of [16] does not have astonishing performance.

With low density of charging stations (lower than $1.6/(100\text{km})^2$), the range is not significantly improved since most of the AVs fail to reach a charging station before battery depletion. However, it increases rapidly when the density is above $2/(100\text{km})^2$. We found that most of the charging requests can reach the stations within $5km$ beyond this threshold, which is only about 5 mins drive to get charged. Moreover, solar rooftop further increases 10% operating range with solar stations. It effectively compensates the extra miles for the AVs to reach the charging station. Note that, if the station density reaches $3/(100\text{km})^2$, the operating range of AVs can be extended for more than 3 times, which is quite promising for large-scale applications in the future smart cities.

Due to the seasonal change of solar income, we also evaluate the operation range across the year. Shown in Fig. 8(b), the operating range is influenced by the climate. The simulation is conducted at Jan., Apr., Jul., and Oct.. During any month, "both" and "solar station" schemes perform better than single "solar rooftop", which achieve almost twice of the operating range. While applying "both" scheme for all $4$ considered months, we find that the summer months achieve almost 20% better performance than spring and fall, and even 30% better than winter. Since "both" schemes utilizing the advantages of both solar harvesting and charging at solar power stations, the harvested energy sometimes plays a crucial role, since the extra harvested energy fills the inevitable energy gap to reach the charging stations.

### 8.6 Utility of Shared AVs

We use the same traces of traffic by analyzing the traffic data from [28] while changes the traces to trips that wait for shared AVs to serve. The period $T$ of simulation is set to 48 hrs, the time spent for fully charging the battery is set to one hour with rapid charger technology [39], the service region is set initially to 25 $km^2$, and the average waiting time for each trip is derived based on the average distance between the finishing position and another beginning position within the region. [21] is used as the benchmark.

As shown in Fig. 9 (a), the utility achieved by one AV is compared between different mechanisms. "Opt." represents the optimal solution achieving the largest possible utility, "Bound" represents the theoretical bound derived for our algorithm (which is converted to $1/3$ utility automatically here), and "STA" is compared with the benchmark "[21]". The optimal solution is derived by going through all combinations of chosen non-overlapping trips which satisfy the energy requirement and choosing the combination with the maximum utility. However, its time complexity is in the order of all combinations of $n$ trips, i.e. $\mathcal{O}(2^n)$. The time complexity of the STA algorithm is $\mathcal{O}(Tn)$ according to Section 7.3. Therefore, our proposed polynomial-time algorithm dominates the exponential-time optimal solution in terms of time complexity.

We first compare the achieved utility under different sizes of the region which contains all the trips. All mechanisms tend to increase initially, then decrease rapidly after reaching the peak in the midst. For small regions, the number of trips is inadequate, which results in low utility due to the leisure time of AV; while the utility is also diminished as the average waiting time added to each trip increases and also the AV needs to travel a longer distance to reach the next trip which also costs more energy. Compared with the benchmark, our algorithm achieves an average of 98% increase. Though the theoretical approximation ratio is 3, our algorithm realizes an average of 76% utility of the optimal utility.

As shown in Fig. 9 (b), the utility achieved by $k$ AVs is also studied for different durations of time. Due to the space limit, only the cases of $k = 2$ and 3 are depicted here. The increment of utility per unit of duration is observed to increase through time due to more choices available for a longer time, enabling the AVs to choose more efficient trips. Here, efficiency means the average time spent for completing one trip. The utility achieved by $k = 3$ is less than 1.5 times of $k = 2$ because the third AV added can not find the trips as efficient as the previous two AVs. Our proposed algorithm maintains the scaling of utility due to the increment of the number of serving AVs while the improvement of the utility for the benchmark is limited. In our algorithm, though the third AV has to choose after

the first two AVs, it can choose once in every iteration, making it possible for the third AV to still choose many efficient trips. For $k = 2$, our algorithm achieves about 52% of the optimal and 167% of the benchmark; for $k = 3$, our algorithm achieves about 53% of the optimal and 205% of the benchmark.

## 9 Conclusions

This paper designs and optimizes a new energy provisioning framework to power electric autonomous vehicles in future smart cities. Our study starts from finding an optimal placement of solar-powered charging stations for optimal coverage and cost minimization for AVs. The strategy can be adjusted online stochastically to adapt to any shift in the new energy/demand distributions. Based on the locations of charging stations, the charging requests are assigned to maximize the satisfaction of users in terms of traveling distance under the budget of solar energy income. For each trip request, the framework schedules the optimal route for the AVs adaptively based on energy consumed and harvested, along with the solar-harvesting rooftop design. Benefiting from solar power, the scheduling of $k$ shared AVs is addressed using a $2k + 1$ approximation algorithm maximizing the overall utility. We demonstrate that the proposed framework can extend the operating scope of AVs, save energy depending on harvested solar energy, and maximize the utility of shared AVs efficiently. In the paper, we intend to construct a solar-powered AV network from none, while the case of potential competitors who can also deploy other such charging stations have not been discussed and could be our future work.

## 10 Acknowledgment

The work in this paper was supported by the U.S. National Science Foundation under grant number CNS-1513719.

## References

[1] TAMU Urban Mobility Information. Retrieved from "https://mobility.tamu.edu/", .

[2] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al. , "End to end learning for self-driving cars", *arXiv preprint arXiv*:1604.07316, 2016.

[3] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, "Autonomous vehicles: challenges, opportunities, and future implications for transportation policies", *J. Mod. Transp.*, vol. 24, no. 4, pp. 284-303, 2016.

[4] J. Stewart, "Self-Driving Cars Use Crazy Amounts of Power, and It's Becoming a Problem", 2018. Retrieved from https://www.wired.com/story/self-driving-cars-power-consumption-nvidia-chip/

[5] E. Cai, D. Juan, D. Stamoulis, and D. Marculescu, "NeuralPower: predict and deploy energy-efficient convolutional neural networks", *ACML*, 2017.

[6] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles", *J. Power Sources*, vol. 226, pp. 272-288, 2013.

[7] P. Wong, and M. Alizadeh, "Congestion control and pricing in a network of electric vehicle public charging stations", *IEEE Allerton Conf. on Comm., Control, and Computing*, 2017.

[8] E. Rigas, S. Ramchurn, N. Bassiliades, and G. Koutitas, "Congestion Management for Urban EV Charging Systems", *IEEE SmartGridComm*, 2013.

[9] P. Zhou, X. Wei, C. Wang, and Y. Yang, "Explore truthful incentives for tasks with heterogenous levels of difficulty in the sharing economy", *IJCAI*, 2019.

[10] B. Du, Y. Tong, Z. Zhou, Q. Tao, and W. Zhou, "Demand-Aware Charger Planning for Electric Vehicle Sharing", *ACM KDD*, 2018.

[11] Y. Xiong, J. Gan, B. An, C. Miao, and A. L. Bazzan, "Optimal Electric Vehicle Charging Station Placement", *IJCAI*, 2015.

[12] C. Liu, K. Deng, C. Li, J. Li, Y. Li, and J. Luo, "The optimal distribution of electric-vehicle chargers across a city", *IEEE ICDM*, 2016.

[13] K. Vatanparvar, S. Faezi, I. Burago, M. Levorato, and M. Faruque, "Extended Range Electric Vehicle with Driving Behavior Estimation in Energy Management", *IEEE Trans. Smart Grid*, vol. 10, no.3, pp.2959-2968, 2018.

[14] K. Vatanparvar, A. Faruque, and M. Abdullah, "Battery lifetime-aware automotive climate control for electric vehicles", *ACM DAC*, 2015.

[15] P. Zhou, C. Wang, and Y. Yang, "Self-sustainable Sensor Networks with Multi-source Energy Harvesting and Wireless Charging", *IEEE INFOCOM*, 2019.

[16] L. Jiang, Y. Hua, C. Ma, and X. Liu, "SunChase: Energy-efficient route planning for solar-powered EVs", *IEEE ICDCS*, 2017.

[17] N. Sharma, J. Gummeson and D. Irwin, "Cloudy computing: leveraging weather forecasts in energy harvesting sensor systems,", *IEEE SECON*, 2010.

[18] C. Wang, J. Li, Y. Yang and F. Ye,"Combining solar energy harvesting with wireless charging for hybrid wireless sensor networks", *IEEE Trans. Mob. Comput.*, vol. 17, no. 3, pp. 560-576, 2018.

[19] M. Magno, N. Jackson, A. Mathewson, L. Benini, and E. Popovici, "Combination of hybrid energy harvesters with MEMS piezoelectric and nano-Watt radio wake up to extend lifetime of system for wireless sensor nodes", *ARCS*, 2013.

[20] I. Rahman, P. Vasant, B. Singh, M. Abdullah-Al-Wadud, and N. Adnan, "Review of recent trends in optimization techniques for plug-in hybrid, and electric vehicle charging infrastructures", *Renew. Sust. Energ. Rev.*, vol.58, pp.1039-1047, 2016.

[21] N. Kang, F. M. Feinberg, and P. Y. Papalambros, "Autonomous electric vehicle sharing system design", *J. MECH. DESIGN*, vol. 139, no. 1, p.011402, 2017.

[22] C. Li, C. Liu, K. Deng, X. Yu, and T. Huang, "Data-driven charging strategy of PEVs under transformer aging risk", *IEEE IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp.1386-1399, 2017.

[23] E. Cela, *The quadratic assignment problem: theory and algorithms, Vol. 1*, Springer Science & Business Media, 2013.

[24] P. Zhou, C. Wang, and Y. Yang, "Static and Mobile Target k-Coverage in Wireless Rechargeable Sensor Networks", *IEEE Trans. Mob. Comput.*, vol. 18, no. 10, pp.2430-2445, 2018.

[25] F. Vignola, "GHI correlations with DHI and DNI and the effects of cloudiness on one-minute data", *ASES*, 2012.

[26] H. Mousazadeh, A. Keyhani, A. Javadi, H. Mobli, K. Abrinia, and A. Sharifi, "A review of principle and sun-tracking methods for maximizing solar systems output", *Renew. Sust. Energ. Rev.*, vol. 13, no. 8, pp.1800-1818 2009.

[27] Solar Dataset of EnvisionSolar. Retrieved from "http://www.envisionsolar.com".

[28] Traffic Count Map in El Paso, TX. Retrieved from "http://gis.elpasotexas.gov/trafficcounts".

[29] Solar Dataset of National Renewable Energy Laboratory. Retrieved from "https://midcdmz.nrel.gov".

[30] S. Hanley, "Hyundai & Kia To Begin Offering Solar Sunroofs After 2019", 2018. Retrieved from https://cleantechnica.com/2018/11/06/hyundai-kia-to-begin-offering-solar-sun-roofs-after-2019/

[31] D. Zou, L. Gao, S. Li, and J. Wu, "Solving 0–1 knapsack problem by a novel global harmony search algorithm", *Appl. Soft Comput.*, vol. 11, no. 2, pp.1556-1564, 2011.

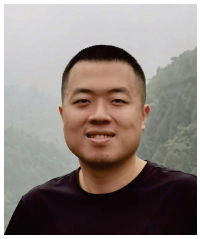
[32] Alternative Fuels Data Center, U.S. Department of Energy, "Electric Vehicle Charging Station Locations". Retrived from https://afdc.energy.gov/fuels/electricity_locations.html

[33] R. Fowler, M. Paterson, and S. Tanimoto, "Optimal packing and covering in the plane are NP-complete", *Inf. Process. Lett.*, vol. 12, no. 3, pp.133-137, 1981.

[34] T. Gonzalez, "Clustering to minimize the maximum intercluster distance", *Theor. Comput. Sci.*, vol. 38, pp.293-306, 1985.

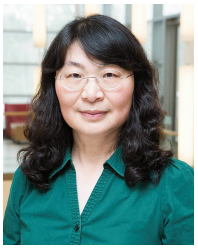
[35] Solar Dataset of Solar Anywhere. Retrieved from "http://www.solaranywhere.com/".

[36] E. Liberty, R. Sriharsha, and M. Sviridenko, "An algorithm for online k-means clustering", *SIAM ALENEX*, 2016.
[37] S. Hochreiter, and J. Schmidhuber, "Long short-term memory", *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[38] D. Asteriou, and S. Hall, "ARIMA models and the box–Jenkins Methodology", *J Forecast.*, vol. 16, no. 3, pp.147-163, 1997.
[39] Podpoint, "How Long Does It Take to Charge an Electric Car". Retrieved from https://www.kia.com/sg/discover-kia/ask/how-long-does-it-take-to-charge-an-electric-car.html.



**Pengzhan Zhou** received the B.S. degree in both Applied Physics and Applied Mathematics from Shanghai Jiaotong University, Shanghai, China. He received Ph.D. degree in Computer and Electrical Engineering from Stony Brook University, NY, USA. His research interests include machine learning, wireless sensor networks, performance evaluation of network protocols and algorithms.

**Cong Wang** received the B. Eng degree in Information Engineering from the Chinese University of Hong Kong in 2008, M.S. degree in Electrical Engineering from Columbia University in 2009, and Ph.D. in Computer and Electrical Engineering from at Stony Brook University, NY, in 2016. He is currently an Assistant Professor at the Department of Cybersecurity, George Mason University, Fairfax, VA. His research focuses on addressing security and privacy challenges in Mobile, Cloud Computing, IoT, Machine Learning and System. He is the recipient of IEEE PERCOM Mark Weiser Best Paper Award in 2018, Commonwealth Cyber Initiative Research and Innovation Award, and NSF CAREER Award in 2021.



**Yuanyuan Yang** received the BEng and MS degrees in computer science and engineering from Tsinghua University, Beijing, China, and the MSE and PhD degrees in computer science from Johns Hopkins University, Baltimore, Maryland. She is a SUNY Distinguished Professor of computer engineering and computer science at Stony Brook University, New York, and is currently on leave at the National Science Foundation as a Program Director. Her research interests include edge computing, data center networks, cloud computing and wireless networks. She has published over 400 papers in major journals and refereed conference proceedings and holds seven US patents in these areas. She is currently the Associate Editor-in-Chief for IEEE Transactions on Cloud Computing and an Associate Editor for ACM Computing Surveys. She has served as an Associate Editor-in-Chief and Associated Editor for IEEE Transactions on Computers and Associate Editor for IEEE Transactions on Parallel and Distributed Systems. She has also served as a general chair, program chair, or vice chair for several major conferences and a program committee member for numerous conferences. She is an IEEE Fellow.