Social-Aware Optimal Electric Vehicle Charger Deployment on Road Network

Qiyu Liu, Yuxiang Zeng, Lei Chen and Xiuwen Zheng

June 12, 2019

Social-Aware Optimal Electric Vehicle Charger Deployment on Road Network (technical report)

Qiyu Liu
HKUST
Hong Kong SAR
qliuau@cse.ust.hk

Yuxiang Zeng
HKUST
Hong Kong SAR
yx.zeng@cse.ust.hk

Lei Chen
HKUST
Hong Kong SAR
leichen@cse.ust.hk

Xiuwen Zheng
UCSD
La Jolla, CA
xiz675@eng.ucsd.edu

ABSTRACT

With the increasing awareness towards protecting environment, people are paying more attention to the electric vehicles (EVs). Accompanying the rapid growing number of EVs, challenges raise at the same time, about how to place EV chargers (EVC), within a city, to satisfy multiple types of charging demand. To provide a better EVC station deployment plan to benefit the whole society, we propose a problem called <u>S</u>*ocial-Aware* <u>O</u>*ptimal Electric Vehicle* <u>C</u>*harger* <u>D</u>*eployment* (SOCD) on road network. The SOCD problem is hard and different from existing work in three aspects, 1) we assume that the charging demand should be satisfied not only in urban areas but also in relatively rural areas; 2) our work is the first one that considers an EVC station should have multiple types of charging plugs, which is more reasonable in real world; 3) different from the regional deployment solutions in previous literature, our SOCD directly works on a real road network and EVC stations are placed at appropriate POIs laying on the road network. We show that the SOCD problem is NP-hard. To deal with the hardness, we design two heuristic algorithms whose efficiency and effectiveness can be experimentally demonstrated. Furthermore, we investigate the incremental case, that is, given an existing EVC station deployment plan and extra more budget, we need to decide where and how many to place more chargers. Finally, we conduct extensive experiments on real road network of Shanghai to demonstrate both effectiveness and efficiency of our algorithms.

KEYWORDS

Electric Vehicle, Road Network, Combinatorial Optimization

1 INTRODUCTION

Nowadays, the transportation sector accounts for a large proportion of total energy consumption. And the rapid growth of energy demand, especially fossil fuel, will lead to massive CO_2 emission [2]. As one of the solutions to alleviate the environmental pressure, electric vehicles (EVs) have been planned to replace or partially replace fossil fuel vehicles, and government incentives to increase adoptions were also introduced, such as the ones in the United States [18] and China [1]. However, as the predictable increase of total number of EVs, the explosive demand of accessing EV chargers (EVCs) in public zones becomes a new challenge at the same time. A survey [4] points out that, although the number of public EVC stations has grown from less than 1000 in April 2011 to 4153 in August 2012, it is still limited compared to the 160,000 gasoline stations in the US (US Department of Energy, 2012). Moreover, [4] also indicates that anxiety caused by too few public chargers and long charging time is one of the deterrents to intent for purchasing an EV. Thus, appropriate deployment of EVCs becomes a fundamental problem for the popularization of the electric vehicles.

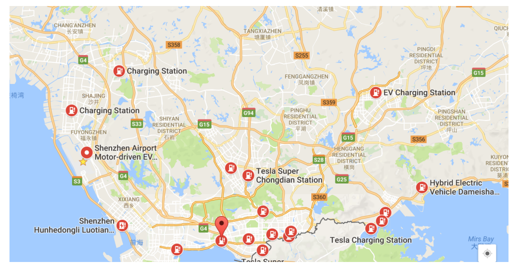


Figure 1: EVC distribution in Shenzhen.

In this paper, to nicely answer the optimal EVC deployment problem considering cost from a social scope, we formulate the problem called <u>S</u>*ocial-Aware* <u>O</u>*ptimal Electric Vehicle* <u>C</u>*harger* <u>D</u>*eployment* (SOCD) on a large-scale road network. Based on the road network information of a city, analysis over historical trajectory data and other relevant features, SOCD provides an EVC deployment plan such that the total social cost is minimized and the whole city's charging demand is satisfied. Note that, the deployment plan includes the location we should place an EVC station and the number of EVCs need to be installed to satisfy the nearby EVs' demand. The social cost contains two parts: (1) the investment from government and EVC providers, and (2) the measurement of the total anxiety and discomfort of EVC users among the whole society. Recently, EVC positioning related problems have been investigated by research community from interdisciplinary backgrounds. However, our work is totally different from the previous works [8, 11, 13, 15–17, 20]. The differences are basically threefold.

1. Satisfying Demand in Rural Areas. First of all, we study the problem of placing *public EVCs* instead of *private EVCs*, which can only be accessed by their owners. In addition, we require the whole road network of a city can be covered by the service region of all EVC stations. Most works like [15, 16] return a deployment plan such that regional charging demand is satisfied, where the charging demand is estimated via historical EV driving trajectory data. However, such strategy will lead to 0 EVC distributed in the region whose charging demand is very low, which does no good for popularizing EV among the whole city. To illustrate this point, Figure 1 shows the current EVC distribution of Shenzhen, one of the biggest city of China. EVCs are mainly distributed in downtown areas such as Luohu; however, for the rural area such as Fenggang and Pinghu, there is no any EVC station, which prevents EV driving into these areas. Although the charging demand in rural areas is low, it is still an important detrimental factor for potential EV drivers.

2. Multiple Types of Charger Plug. Besides, previous works have not taken plug types of a charger into consideration and they assume the charging capability is identical among all the EVCs. However, we find that plug types do influence the EVC deployment result significantly. Table 1 [1] lists some existing plug types.

Table 1: Table of the some charger plugs and charging power.

Charging Plug	Power	Charging Duration
Type-2$^{(1)}$	7.3 kW	8h14min
Type-2$^{(2)}$	16.5 kW	3h8min
CHAdeMO	50 kW	1h40min
Tesla Supercharger	120 kW	57min

We can find that the charging capability (charging power) varies much from different plug types and also the cost of installing different types of chargers will be different. It is obviously to see that, chargers with low-power plugs should not be installed in some areas with high parking fee, such as shopping centers and superior office buildings. On the other hand, these trickle chargers are suitable for somewhere long-time parking is allowed, such as airport parking and other long-term parking lots for hotels or apartment.

3. Solution Granularity. Most of current works on placing EVC returns regional result [15, 16], that is to say, these algorithms determine the necessary number of EVC within each region or grid cell partitioned in advance. Although some other researches like [11] work on road network, they use a simplified or highly extracted version, which loses much information of points of interest (POIs). Instead, we allow EVC station can be placed nearby *any POI* among a city. Here, *any* means that, given a road network, any node, representing a POI in a city, can be potential location of a EVC station. The reason of such setting is that, although we can know how many chargers are needed within some region, we must further determine where and how many we should place these chargers. For example, there might be some green land, a lake, a shopping mall and a large hotel. It is more reasonable to place EVC station at the latter two places rather than first two since shopping mall and hotel are POIs to EV driver, whereas green land and lake are not.

Contribution: We list our main contribution as follows.

(1) We first formulate the problem *Social-Aware Optimal Electronic Vehicle Charger Deployment* on a real road network. SOCD is the first work that considers the social benefit of rural areas, the multiple charger plug types and the influence of POIs on the real road network.
(2) We prove our SOCD problem is NP-hard and also hard to find any constant approximation, and then we devise several efficient heuristic algorithms, which are novel greedy based algorithms to solve the complex non-linear optimizing problem.
(3) Based on the proposed solution for SOCD problem, we further investigate the extendibility of our algorithms on the incremental case, that is, given an existing EVC deployment and more budget, how to place more chargers in a way contributes to the whole society as much as possible.

The rest parts of our paper is organized as follows. In Section 2, we give the definitions of some key concepts, formulate the SOCD problem and give the proof of hardness. In Section 3, we give the solutions to the SOCD problem. More specifically, Section 3.2 introduces the *Bounding&Optimizing* framework; Section 3.3 presents a more efficient algorithm called *Region Partition Based Deployment*; and Section 3.4 discusses about how to extend our algorithm to the incremental scenario. Section 4 presents extensive experimental results of our proposed algorithms under various parameter settings. Section 5 reviews previous works on the EVC related optimization problems. Finally, we conclude in Section 6.

[1] The data is crawled from https://leccy.net.

Table 2: Table of notations.

Notation	Description
G	a road network with POIs
S	a multi-plug EVC charging station
x_i	the number of chargers with i^{th} type plug of charging station S
$R(S)$, r_S	the influence region of S and its radius
$C(S)$	total charging capacity of EVC station S
P	an EVC station deployment plan
$w(p)$	the rural degree of location p
$f(S)$	the total installment fee of EVC station S
$D(S)$	the total number of EVs choosing station S for charging
$W(S)$	the expected waiting time at EVC station S
Benefit	the total social benefit of a given deployment plan P
Cost	the total social cost of a given deployment plan P
$Cost_t$	the total travel cost of a given deployment plan P
$Cost_b$	the total boring time of a given deployment plan P
$Social(P)$	the social influence (score) of a given deployment plan P

2 PRELIMINARIES

In this section, we formally introduce our Social-Aware Optimal Electronic Vehicle Charger Deployment on road network, which aims at determining an optimal EVC deployment plan such that the total social score is maximized. For quick reference, all the notations used in this paper are listed in Table 2.

2.1 EVC Station on Road Network

First, since we assume that EVC stations are distributed on a given road network, we give the formal definition of the road network of a city as follows.

Definition 1: Road Network. A road network of a city is defined as a quadruple $G = (V, E, \tau, \delta)$, where V is the set of points of interests (POIs), E is the set of roads bridging nodes in V, $\tau : V \to \mathbb{R}^2$ is the function mapping vertices in V to 2D spatial space. For a given edge $e = (u, v) \in E$, $\delta(e)$ can be regarded as the travel cost from the start point u to the end point v of road e.

Note that, in some applications, the travel cost is modeled by driving time instead of road distance. The computation of driving time is more complicated because we need to consider about the real-time traffic condition. Since the main purpose of this paper is not to model the traveling cost on road network, we just use road distance to compute travel cost for simplification.

Then, we give the definition of Multi-plug EVC Station located on the road network, which has been discussed in our introduction.

Definition 2: Multi-plug EVC Station. Given a road network $G = (V, E, \tau, \delta)$, an EVC station with multiple charger plugs S has two attributes, $S.pos$ and $S.x$, where $S.pos$ is the location of station S and possible values of $S.pos$ are in $\{\tau(v)|v \in V\}$, and $S.x = \{x_S^{(1)}, x_S^{(2)}, \cdots, x_S^{(k)}\}$ is an array of size k denoting the numbers of k types of chargers, where $x_S^{(i)}$ is the number of chargers with i^{th} type plug. Besides, for a station S, the following constraint should be satisfied, $\sum_{i=1}^{k} x_S^{(i)} \leq K$, which means the total number of chargers installed at EVC station S should be bounded by K due to the limitation of space.

Definition 3: Installment Fee. Given an EVC station, the cost of installment fee, which is denoted by $f(S)$, is calculated as follows,

$$f(S) = estate\_price(S) + \sum_{i=1}^{k} x_S^{(i)} f_i,$$

where $estate\_price(S)$ is the cost of deploying an EVC station at location $S.pos$ and f_i is the fee of installing one charger with i^{th} type plug.

To measure the influence of setting a new EVC station S, we introduce the concept of *Influence Region*.

Definition 4: Influence Region. Given an EVC station, its influence region $R(S)$ is a circle centering at $S.pos$ with radius r_S, where r_S is defined as

$$r_S = r_{max} \cdot \left(\frac{2}{1 + \exp(-\sum_{i=1}^{k} x_S^{(i)} p_i)} - 1 \right), \tag{1}$$

where r_{max} is the maximum influential radius, and p_i is the charging power of i^{th} type charger plug and thus, $C(S) = \sum_{i=1}^{k} x_S^{(i)} p_i$ is the total charging capacity of EVC station S.

Note that, when the total charging capability of a station S_v, $C(S) = \sum_{i=1}^{k} x_i p_i$, increases from 0 to $+\infty$, the radius of S's influence region r_S increases from 0 to r_{max}, which is reasonable in real applications since the maximum influence region should be bounded by some distance constraint. That is to say, even an EVC station may have a very large charging capacity, it still cannot attract users far away from it (e.g., 50km).

Next, we give the definition of EVC Deployment Plan.

Definition 5: EVC Deployment Plan. Given a road network $G = (V, E, \tau, \delta)$, an EVC deployment plan is a set of Multi-plug EVC Stations. We use symbol $P = \{S_1, S_2, \cdots, S_m\}$ to denote it.

2.2 Social Influence of EVC Deployment

In this subsection, suppose that we are given an EVC deployment plan P, we discuss the social influence caused by this plan, which is the core optimization goal of our problem. We divide the social influence into two main parts, Social Benefit and Social Cost, denoted by *Benefit* and *Cost* respectively.

Before formally introducing the definition of *Benefit* and *Cost*, we first need to estimate the charging demand d_v of each node in road network. Here, the semantics of d_v is the number of EVs located near $\tau(v)$ that need to be refilled within a unit time interval (eg., 2 hours). To get d_v, we collect historical trajectory data of various types of cars. The details of how to estimate d_v via trajectory data are discussed in the Section 4.

2.2.1 Social Benefit. Given an EVC deployment plan P, the benefit (i.e., positive social influence) gained from placing chargers as plan P is the coverage of EVC stations' influence regions over the whole city, including both urban and rural areas. Formally, we have the definition of social benefit as follows.

Definition 6: Social Benefit. Given an EVC deployment plan P, for any $S \in P$, the corresponding influence region $R(S)$ and the set of nodes covered by $R(S)$ can be calculated respectively. The total social benefit, denoted by *Benefit*, can be calculated as:

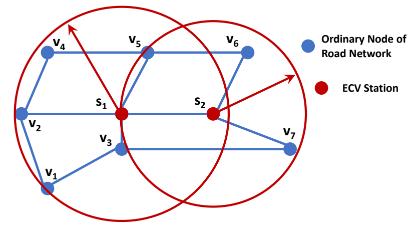


Figure 2: An example of calculation of social benefit.

$$Benefit(P) = \sum_{S \in P} \left(\frac{2}{1 + \exp\{-w(S.pos) I_1(S)\}} - 1 \right), \tag{2}$$

where $I_1(S)$ is the number of nodes in the road network covered by $R(S)$. Besides, $w(S.pos)$ is a weight parameter to measure the "rural degree" of the location of S. The higher the value of $w(S.pos)$, the more rural of the location of S.

Figure 2 illustrates a part of road network and a simple ECV deployment plan that S_1 and S_2 are two newly installed EVC stations. The two circles in the figure are the corresponding influence regions of S_1 and S_2. Suppose that, $w(S_1.pos) = 2$ and $w(S_2.pos) = 1$. Thus, according to Eq. (2) the total social benefit of such deployment plan can be calculated as: $(2/(1 + \exp(-2 \times 7)) - 1) + (2/(1 + \exp(-1 \times 5)) - 1) = 1.987$.

2.2.2 Social Cost. Meanwhile, an EVC deployment plan also requires some cost to implement, which we call "Social Cost". Similar to the assumptions in works [15, 16, 20], we take travel cost and the boring time elapse of waiting for EV getting fully charged into consideration when we calculate the total social cost.

Travel cost $Cost_t$. The first factor we consider that contributes to the social cost is the total travel cost, which is defined as the total driving distance from all EVs having charging demand to their nearest EVC stations within a time period ΔT. Given a road network G, an EVC deployment plan P and charging demand d_v for every node in road network, we can calculate the travel cost $Cost_t$ as follows,

$$Cost_t(P) = \sum_{S \in P} \sum_{v \in V} d_v dist(v, S) \cdot y(v, S), \tag{3}$$

where $dist(v, S)$ is the length of the shortest path from v to S on road network, and $y(v, S)$ is an indicator function. If EVs at v choose S for charging, $y(v, S) = 1$, otherwise, $y(v, S) = 0$.

Boring time $Cost_b$. Since we cannot install unlimited number of chargers in an EVC station, which means the total charging capacity of a station is limited, *queueing* naturally happens for all EVC stations. And long waiting time for available chargers significantly increases the boringness of EV drivers, which produces the social cost. Besides, as we have already shown that the charging power varies much from different types of plugs, the total charging time is also considered into boring time. Thus, we define the social cost caused by long boring time, denoted by $Cost_b$, as the sum of *waiting time* and the *charging time*.

For an EVC station S, to analyze the *waiting time* and the *charging time* at S, we first estimate the total number of EVs coming S for charging within a unit time interval as the following formula,

$$D(S) = \sum_{v \in V} \frac{1}{dist(v,S)} d_v y(v,S), \tag{4}$$

which is a weighted sum over the charging demand $d_v y(v,S)$ where the weight value $1/dist(v,S)$ implies the attraction of S to EVs at location $\tau(v)$. Note that, the attraction can be quantified by any decaying function of $dist(v,S)$ and here we adopt the inverse of $dist(v,S)$.

Charging time. Note that, the total charging capacity as $C(S) = \sum_{i=1}^{k} x_S^{(i)} p_i$, where p_i is the power of charger with i^{th} type plug. Thus, the expected charging time of station S, here, can be calculated as the inverse of the total charging capacity, that is, $1/C(S)$. Then, we evaluate the total *charging time* among all the stations in a given deployment plan P as:

$$\textit{charging time} = \sum_{S \in P} C(S)D(S) = \sum_{S \in P} \sum_{v \in V} \frac{1}{C(S)dist(v,S)} d_v y(v,S)$$

Waiting time. For the *waiting time*, we first model the waiting time at an EVC station as an M/D/1 queue [5], where "M" means the coming event of EV follows a Poisson process, "D" means the service time (i.e., the charging time) is a deterministic function and "1" stands for that there is one queue for a station. The expected value of waiting time at station S is given by Pollaczek-Khintchine formula [5] as follows,

$$W(S) = \frac{\rho_S \tau_S}{2(1-\rho_S)}, \quad \text{if } \rho_S \leq 1 \tag{5}$$

where τ_S is the average charging time and $\rho_S = \theta_S \tau_S$, θ_S being the EV arrival rate of EVC station S. Here, we estimate τ_S as $1/C(S)$ and estimate θ_S as the summed charging demand within a circular region, denoted by $R_{max}(S)$, which is centering at $S.pos$ with radius r_{max}, that is, $\rho_S = \sum_{\tau(v) \in R_{max}(S)} d_v / C(S)$. Note that, ρ_S must be less than 1, otherwise, the length of queue at station S will go to infinity. Then, we can estimate the total waiting time at all stations in a given plan P as follows,

$$\textit{waiting time} = \sum_{S \in P} D(S)W(S) = \sum_{S \in P} \sum_{v \in V} \frac{W(S)}{dist(v,S)} d_v y(v,S).$$

Thus, the total *boring time*, denoted by $Cost_b$, over the whole society can be calculated as the sum of total *waiting time* and total *charging time*, which is shown in Eq. (6),

$$\begin{aligned} Cost_b(P) &= \textit{waiting time} + \textit{charging time} \\ &= \sum_{S \in P} \sum_{v \in V} \frac{d_v y(v,S)}{dist(v,S)} \cdot \left(W(S) + \frac{1}{C(S)} \right). \end{aligned} \tag{6}$$

Definition 7: Social Cost. Suppose that we are given a road network $G = (V, E, \tau, \delta)$ and the charging demand d_v for each nodes in G, for an EVC deployment plan P, the total social cost of P is defined as,

$$\begin{aligned} Cost(P) &= \alpha Cost_t(P) + (1-\alpha) Cost_b(P) \\ &= \sum_{S \in P} \sum_{v \in V} d_v y(v,S) \left(\alpha dist(v,S) + \frac{1-\alpha}{dist(v,S)} \left(W(S) + \frac{1}{C(S)} \right) \right) \end{aligned}$$

where α is the parameter tuning the relative importance among these two kinds of social cost.

2.3 Problem Definition

With all the concepts defined above, we can formulate our Social-Aware Optimal Electric Vehicle Charger Deployment problem.

Definition 8: Social-Aware Optimal Electric Vehicle Charger Deployment (SOCD). Given road network $G = (V, E, \tau, \delta)$, charging demand $\{d_v | v \in V\}$, and the total budget B for deploying EVC stations, SOCD solves the optimization problem as follows,

$$\max_{P,y} \quad Social = \lambda Benefit - (1-\lambda) Cost \tag{7}$$

subject to:

$$\sum_{S \in P} f(S) \leq B \tag{8a}$$

$$\sum_{S \in P} y(v,S) = 1 \qquad \text{for } \forall v \in V \tag{8b}$$

$$\sum_{i=1}^{k} x_S^{(i)} \leq K \qquad \text{for } \forall S \in P \tag{8c}$$

$$\frac{\sum_{\tau(v) \in R_{max}(S)} d_v}{C(S)} \leq 1 \qquad \text{for } \forall S \in P \tag{8d}$$

where λ is the parameter tuning the relative importance between social benefit and social cost, $f(S)$ is the installment fee of S which is shown in Definition 3, $x_S^{(i)}$ is the number of i^{th} type of charger at station S. Eq. (8a) is the constraint on total installment fee not exceeding the expected budget B; Eq. (8b) requires that one node with charging demand can only choose one station for charging; Eq. (8c) gives a upper-bound K to the number of chargers an EVC station can install; and Eq. (8d) is for avoid waiting queue at each EVC station increasing to infinity.

Note that, similar to the well-known *facility location* problem [10], the decision variables in our SOCD can be divided into two sets, one is the optimal EVC station deployment plan P, and the other is the demand assignment y representing the EVs' choices of EVC station at each location $\tau(v)$.

Hardness Analysis. Our SOCD problem can be proved as NP-hard by using a reduction from the *KNAPSACK* problem.

Theorem 2.1. *(Hardness of the SOCD problem) The problem of Social-Aware Optimal Electric Vehicle Charger Deployment (SOCD) is NP-hard.*

Due to the NP-hardness, it is impossible to solve the SOCD problem in polynomial time. Besides, designing heuristics or greedy algorithms for SOCD also differs from the classical combinatorial optimization problems since in SOCD, we not only determine where to place an EVC station, but also should give the numbers of each type of chargers. Besides, our SOCD problem cannot be solved by common-used LP solvers such as LINDO since the optimization objective and the constraints such as the queuing constraint in Eq. (8d) are complex and non-linear.

3 METHODOLOGY

As stated in the last section, it is very hard to design exact and approximated algorithms for SOCD, in this section, we propose several efficient and effective heuristics to solve the problem. We introduce two algorithms for solving SOCD, *Bounding & Optimizing*

Greedy Deployment and *Region-Partitioning-Based Group Deployment*. Before formally introducing the algorithms, we first discuss how to assign charging demand given the incumbent EVC station deployment plan P. Note that, for a given EVC deployment plan P, to evaluate its total social influence defined in Eq. (7), it is necessary to determine which station will be chosen by an EV for charging (namely, $y(v, S)$). Since retrieving the optimal solution to $y(v, S)$ is intractable, we first give an heuristic algorithm solving the problem called *EVC Station Seeking Algorithm*.

3.1 EVC Station Seeking Algorithm

As shown the Definition 8, SOCD determines not only charger deployment plan P, but also the optimal charging demand assignment $y(v, S)$ for $\forall v \in V$ and $\forall S \in P$. The first problem we want to answer is that, given an existing EVC station deployment plan P, how the nodes in the road network with charging demand will choose EVC stations, that is, evaluating $y(v, S)$ for $v \in V, S \in P$. We first re-investigate some similar problems. In the greedy algorithm for *facility location* problem in [10], nodes with demand always choose their nearest facility in each greedy iteration. However, in our SOCD problem, we consider multiple social influence factors and travel cost is only one of them. Besides, some other works like [12, 16] formulate this procedure as a bi-level linear programming. They regard demand assignment as a sub-problem and iteratively invoke LP solver to solve it. Unfortunately, we cannot borrow this idea either because as we show in Definition 8, the formation of social influence is very complex and decision variables are coupled together, which prevents us from using all currents LP solvers.

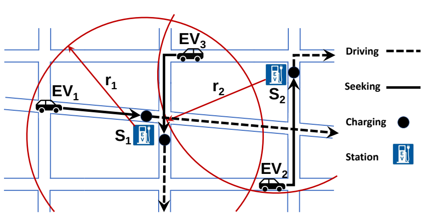


Figure 3: An example of EVC station seeking.

Now, we formally introduce the sub-problem, called *Station Seeking*, of SOCD. Given the current deployment plan P, since *Benefit*, in the SOCD optimization objective shown in Eq. (7), is fixed when P is fixed, maximizing *Social* is equivalent to minimizing *Cost*. Thus, we have the following definition of *Station Seeking* problem.

Definition 9: Station Seeking. Given current deployment plan P, the *Station Seeking* problem is formulated as follows,

$$\min_{y} Cost = \sum_{S \in P} \sum_{v \in V} d_v y(v, S) \left(\alpha dist(v, S) + \frac{1 - \alpha}{dist(v, S)} \left(W(S) + \frac{1}{C(S)} \right) \right)$$

subject to:

$$\sum_{S \in P} y(v, S) = 1, \qquad \text{for } \forall v \in V.$$

To solve this problem, we propose a *StationSeeking* algorithm, which is a greedy algorithm but yields the optimal solution. The algorithm is shown in Algorithm 1, in each iteration, for a node in road network v, we calculate the assignment cost for v choosing station S, denoted by $Cost_a(v, S)$, as:

$$Cost_a(v, S) = d_v \left(\alpha dist(v, S) + \frac{1 - \alpha}{dist(v, S)} \left(W(S) + \frac{1}{C(S)} \right) \right), \quad (9)$$

and we assign v to station S (i.e., let $y(v, S) = 1$) such that $Cost_a(v, S)$ is minimized.

Algorithm 1: *StationSeeking*

Input: EVC station deployment plan P, road network $G = (V, E, \tau, \delta)$, charging demand $\{d_v | v \in V\}$
Output: demand assignment $\{y(v, S) | y(v, S) \in \{0, 1\}, v \in V, S \in P\}$

```
for v in V do
    calculate Cost_a(v, S) as Eq. (9) for all S ∈ P;
    S' ← arg min_{S∈P} Cost_a(v, S);
    y(v, S') ← 1;
return {y(v, S) | y(v, S) ∈ {0, 1}, v ∈ V, S ∈ P};
```

We give an example in Figure 3. There are two EVC stations in the example, S_1 and S_2, suppose that EV_1, EV_2 and EV_3 are three EVs that need to be refilled soon. By following the greedy EVC Station Seeking manner, EV_1, EV_2 and EV_3 are assigned to S_1, S_2 and S_1 respectively. The following theorem indicates that such greedy algorithm yields the optimal solution for *Station Seeking* problem.

Theorem 3.1. *The greedy algorithm shown in Algorithm 1 yields the optimal solution for the Station Seeking problem.*

3.2 Bounding & Optimizing Based Greedy

In this subsection, we introduce an algorithm based on greedily selecting a location to build an EVC station such that the gain of *Social* is maximized in every step. However, as we have mentioned in Section 2.3, SOCD problem cannot borrow ideas from common combinatorial optimization problems since we need to determine both where to deploy EVC stations and how many chargers are needed. Thus, very different from the common greedy algorithm design pattern, which is to make the locally optimal choice at each stage, we devise a strategy called *Bounding & Optimizing Based Greedy*.

The basic idea is that, in the *Bounding Stage*, we evaluate the upper-bound of the gain to *Social* for setting one EVC station S_i at every possible location $\tau(v_i)$ and pick the location with the highest upper-bound to deploy an EVC station in this step; then, in the *Optimizing Stage*, assuming that the location of station has been decided in the *Bounding Stage*, we determine the numbers of each types of chargers to try to reach the upper-bound; and then, we repeat the above greedy picking procedure until there is no budget left to build a new EVC station.

Framework. The framework of *Bounding & Optimizing Based Greedy Deployment* is shown in Algorithm 2. We start from an empty EVC station deployment plan P and B, the initial total budget. The *Bounding Stage* are shown in lines 4-5, where we greedily select the location to place an EVC station maximizing the upper-bound to the gain of *Social*. Denoting the station newly placed as S_i, line 6 updates the demand assignment $y(v, S)$. Then, in line 7, we invoke the *Knapsack Based Optimizing* in Algorithm 3 to determine the number of each type of chargers at station S_i. Then, S_i will be inserted into current plan P and remained B will be updated. The algorithm will terminate if budget is exhausted. In the sequel, we will introduce the *Bounding Stage* and *Optimizing Stage* in details.

Bounding Stage. Given the incumbent EVC station deployment plan P, let $Social(P)$ be the total social influence of P, which is calculated by Eq. (7). For deploying an EVC station S_i at location

Algorithm 2: *Bounding&Optimizing*

Input: road network $G = (V, E, \tau, \delta)$, charging demand $\{d_v | v \in V\}$
Output: an EVC station deployment plan P

1 $P \leftarrow \phi$;
2 $B \leftarrow$ initial total budget;
3 **while** $B > 0$ **do**
   /* Bounding Stage */
4   calculate social efficiency upper-bound $ub\_g(v)$ for every node v in road network;
5   pick the location $\tau(v_i)$ with highest $ub\_g(v)$ to build an EVC station S_i;
6   invoke *StationSeeking* (Algorithm 1) to update demand assignment $y(v, S)$;
   /* Optimizing Stage */
7   invoke *KnapsackBasedOpt* (Algorithm 3) to get $\{x_{S_i}^{(1)}, x_{S_i}^{(2)}, \cdots, x_{S_i}^{(k)}\}$;
8   $S_i.pos \leftarrow \tau(v_i)$; $S_i.x \leftarrow \{x_{S_i}^{(1)}, x_{S_i}^{(2)}, \cdots, x_{S_i}^{(k)}\}$;
9   $P \leftarrow P \cup \{S_i\}$;
10   $B \leftarrow B - f(S_i)$;
11 **return** P;

$S_i.pos \in \{\tau(v) | v \in V\}$, we define its social efficiency (i.e., the gain of *social* value per budget cost), denoted by $g(S_i)$, as follows,

$$g(S_i) = \frac{Social(P \cup \{S_i\}) - Social(P)}{f(S_i)}, \tag{10}$$

where $f(S_i)$ is the installment fee of station S_i, which is defined in Definition 3. Then, we evaluate the upper-bound of social efficiency when we place station S_i at location $\tau(v)$, which is denoted by $ub\_g(v)$. Note that, $ub\_g(v)$ indicates the potential social efficiency of setting S_i at $\tau(v)$. $ub\_g(v)$ is given by Lemma 3.2.

LEMMA 3.2. *(Upper-Bound) Suppose that we are deploying an EVC station S_i at location $S_i.pos = \tau(v)$, the upper-bound of S_i's social efficiency $g(S_i)$, denoted by $ub\_g(v)$, is given by*

$$g(S_i) \leq ub\_g(v) = \frac{\lambda \Delta Benefit^*}{estate\_price(S_i)}, \tag{11}$$

where $\Delta Benefit^*$ is:

$$\Delta Benefit \leq \frac{2}{1 + \exp\{-w(S_i.pos)[I_1^*(S_i)]\}} - 1, \tag{12}$$

where $I_1^*(S_i)$ is the number of nodes in the road network covered by the circular region centering at $S_i.pos$ with radius r_{max}.

Optimizing Stage. Suppose that we have decided to deploy an EVC station S_i at location $\tau(v)$, then, we discuss how many chargers of different types are needed to increase the social efficiency $g(S_i)$ to the greatest extent. The problem can be formulated as:

$$\begin{aligned} \min g(S_i) &= \frac{Social(P \cup \{S_i\}) - Social(P)}{f(S_i)} \\ &= \frac{\lambda Benefit(P \cup \{S_i\}) - (1-\lambda)Cost(P \cup \{S_i\}) - Social(P)}{estate\_price(S_i) + \sum_{i=1}^{k} f_i x_{S_i}^{(i)}} \end{aligned} \tag{13}$$

such that,

$$\sum_{i=1}^{k} x_{S_i}^{(i)} \leq K \tag{14a}$$

$$estate\_price + \sum_{i=1}^{k} x_{S_i}^{(i)} f_i \leq B \tag{14b}$$

$$\sum_{i=1}^{k} x_{S_i}^{(i)} p_i \geq \sum_{\tau(v) \in R_{max}(S_i)} d_v \tag{14c}$$

Note that, the optimization goal shown in Eq. (13) is fractional. According to [9], the linear fractional programming (LFP) problems are usually transformed to standard linear programming to use LP solver. But unfortunately, Eq. (13) is non-linear fraction due to the term *Benefit*, which currently has no effective solution. Thus, we propose a heuristic algorithm called *KnapsackBasedOpt* to solve it.

The motivation of *KnapsackBasedOpt* is that, we start from an initial deployment $\{x_{S_i}^{(1)}, x_{S_i}^{(2)}, \cdots, x_{S_i}^{(k)}\}$ and repeatedly add chargers with i^{th} type plug such that the social efficiency $g(S_i)$ is maximized. To do that, we first generate a feasible solution $\{x_{S_i}^{(1)}, x_{S_i}^{(2)}, \cdots, x_{S_i}^{(k)}\}$ such that the total installment fee is minimized satisfying the total charging capacity constraint in Eq. (14c). We describe this problem, which is an unbounded knapsack problem (UKP), as follows,

$$\begin{aligned} \min \quad & \sum_{i=1}^{k} x_{S_i}^{(i)} f_i \\ \text{s.t.} \quad & \sum_{i=1}^{k} x_{S_i}^{(i)} p_i \geq \sum_{\tau(v) \in R_{max}(S_i)} d_v \end{aligned} \tag{15}$$

Algorithm 3: *KnapsackBasedOpt*

Input: road network $G = (V, E, \tau, \delta)$, charging demand $\{d_v | v \in V\}$, station selected in *Bounding Stage* S_i, current deployment plan P
Output: numbers of each type of chargers $\{x_{S_i}^{(1)}, x_{S_i}^{(2)}, \cdots x_{S_i}^{(k)}\}$

/* get initial solution via unbounded knapsack */
1 using dynamic programming to solve the unbounded knapsack problem shown in Eq. (15) and denote the result as $x[1 \cdots k]$;
/* start adding chargers */
2 **while** $\sum_{i=1}^{k} x[i] \leq K$ **do**
3   $G(j) \leftarrow$ difference of $g(S_i)$ after and before adding one j^{th} type charger;
4   $j^* \leftarrow \arg\max_j G(j)$;
5   **if** *budget B is enough for adding* j^{*th} *charger* ***and*** $G(j^*) \geq 0$ **then**
6     $x[j^*] \leftarrow x[j^*] + 1$;
7   **else return Fail** ;
8 **return** $x[1], x[2], \cdots, x[k]$;

The pseudo code of *KnapsackBasedOpt* is shown in Algorithm 3. Line 1 solves the knapsack problem in Eq. (15) to get an initial feasible solution. Line 3 defines a value $G(j)$ to denote the difference of social efficiency $g(j)$ after and before increasing one charger with j^{th} type plug at station S_i. In line 4, we pick the j^{*th} type charger such that it can increase $G(j)$ to the greatest extent. Lines 5-8 further determines whether we can add one j^*. If current budget is enough for deployment of one more charger with j^{*th} type plug and there is positive gain of $g(S_i)$ if adding j^{*th} charger (i.e., $G(j^*) > 0$), we add 1 to $x[j^*]$; otherwise, the algorithm terminates. Note that, there are totally three stop conditions of *KnapsackBasedOpt* algorithm, and when we invoke it in our *Bounding&Optimizing* framework, we should check which reason leading to termination of *KnapsackBasedOpt*. If total budget B is exhausted, the whole loop in *Bounding&Optimizing* terminates; whereas, if the other two stop conditions are triggered, we only break *KnapsackBasedOpt* and continue to select another site to build an EVC station in *Bounding&Optimizing*.

Complexity Analysis. We analyze the worst case time complexity of our *Bounding&Optimizing* algorithm as follows. The worst run time corresponds to the case that initial total budget is very large, which means the algorithm will terminate after traversing all the possible locations (namely, $O(|V|)$ nodes in road network) to build an EVC station. The *Bounding Stage*, which is shown in lines 4-5 of Algorithm 2, takes time $O(|V|)$ since we need to evaluate all the social efficiency. After deciding where to place an EVC station, in the *Optimizing Stage*, solving the unbounded knapsack problem via dynamic programming takes time $O(KD^*)$, where K is the upper-bound of total number of chargers at one station and D^* is the total demand within a circular region centering at a station with radius r_{max}. Note that K and D^* are constant, which means line 7 takes time $O(1)$. Besides, lines 8-10 take time $O(1)$. Thus, the time complexity of the worst case is $O(|V|^2)$.

3.3 Region Partition Based Algorithm

Although our *Bounding&Optimizing* framework shown in Algorithm 2 can return an EVC deployment plan with high *Social* value, the time complexity, $O(|V|^2)$, is still high. The reason is that, in each iteration, such a greedy algorithm suffers from $|V|$ times comparisons in lines 4-5 in Algorithm 2. To reduce the total time complexity, an intuitive way is to partition the road network into m sub-regions and independently conduct the *Bounding&Optimizing* framework within each sub-region. Finally, we integrate all the results on each sub-region to get an EVC station deployment plan.

Figure 4: Illustration of Voronoi-based region partition.

We use Voronoi diagram [3] to partition the road network. Specifically, we select m major nodes (e.g., center points in administrative districts among a city) in road network G with each corresponding to one Voronoi cell. The distance from any location in a Voronoi cell to the corresponding seed is less than that to any other seed. Thus, we can partition the original road networks into several sub-regions represented by different Voronoi cells. An example of Voronoi diagram based region partition is shown in Figure 4 by using Shanghai road network, where blue markers are seeds of Voronoi cells.

The pseudo code of our region partition based algorithm is presented in Algorithm 4. Line 1 partitions the input road network into m sub-regions based on the selected m seeds $\{S_1, S_2, \cdots, S_m\}$. Lines 2-4 are initialization steps, where line 3 sets the total budget of each sub-region as the total budget B divided by m and line 4 initializes the EVC station deployment plans in each sub-region as empty sets. Then, for each sub-region G_i, we conduct the *Bounding&Optimizing* in Algorithm 2 to get regional deployment plan P_i and then update the remained budget B_i. After getting all regional plans, we evaluate the total remained budget (i.e., $\sum_{i=1}^{m} B_i$) which is used for deploying more extra stations on the whole road network G, whose corresponding deployment plan is denoted by P'. Finally, we return the whole deployment plan P, which is the union of all regional plans $P_1, \cdots, P_m$ and the plan P' which utilizes the total remained money.

Time complexity. For Algorithm 4, given m Voronoi seeds $\{S_1, S_2, \cdots, S_m\}$, line 1 partitions the whole space into m parts within time $O(m \log m)$ by using Fortune's algorithm [7]. Denote the number of nodes in i^{th} sub-region as $|V_i|$. Then, lines 6 takes time

Algorithm 4: *RegionPartition*

Input: road network: $G = (V, E, \tau, \delta)$, set of m seeds of Voronoi diagram: $\{S_1, S_2, \cdots, S_m\}$, charging demand $\{d_v | v \in V\}$
Output: an EVC station deployment plan P
/* Divide original region into m sub-regions */
1 using Voronoi diagram to partition the road network G according to seeds $\{S_1, \cdots, S_m\}$ and let $G_1, \cdots, G_m$ be the m sub-regions after partitioning;
/* Initialization */
2 **for** $i = 1$ *to* m **do**
3     $B_i \leftarrow B/m$;
4     $P_i \leftarrow \phi$;
/* place stations in each sub-region */
5 **for** $i = 1$ *to* m **do**
6   invoke *Bounding&Optimizing* framework with total budget B_i to determine EVC deployment plan P_i over sub-region G_i;
7   update the remained budget B_i;
8 use total remained budget $\sum_{i=1}^{m} B_i$ to place extra stations among road network G according to the same greedy algorithm and denote the result as P';
9 **return** $P_1 \cup \cdots \cup P_m \cup P'$;

$O(\sum_{i=1}^{m} |V_i|^2)$. Suppose that in the partition step, we evenly divide the whole region, that is to say, $|V_1| = |V_2| = \cdots = |V_m| = |V|/m$. Thus, the run time of lines 5-7 is $O(\sum_{i=1}^{m} |V_i|^2) = O(|V|^2/m)$. For line 8, the time complexity is influenced by the remained total budget and the number of nodes without placing any charger. Note that, in real applications, the remained budget should be so small that we cannot place many extra stations. Thus, the total time complexity of our region partition based algorithm is $O(|V|^2/m)$.

Note that, the total number of sub-regions m makes a tradeoff between total *Social* value and run time of algorithm. Large m leads to faster termination of the *RegionPartition* algorithm with some loss of the *Social* value. This is natural to understand since we conduct greedy station placing over each partitioned sub-region independently, that is to say, interaction between different sub-regions is ignored.

3.4 Extend to Incremental Case

Above we have discussed the solutions to the SOCD problem on a real road network, however, there exists another kind of EVC station deployment problem where the budget will not be totally disbursed at initial time and extra more budget will be available some day in the future. We call such special case the *Incremental SOCD* problem. To avoid ambiguity, we call the original SOCD problem "Static SOCD" and without specific clarification, "SOCD" only means "Static SOCD" but not "Incremental SOCD". The following is an example that illustrates such an application scenario.

Example: Incremental SOCD. *Shanghai government is putting efforts on promoting the development of electric vehicles and they have already granted funding to place some EVC stations. However, with the increasing number of EVs, the current EVC stations cannot provide enough charging service, which leads to the negative social influence. Thus, after careful investigation, Shanghai government decides to give more extra budget on deploying more EVC stations. The incremental SOCD problem is that, based on the extra budget and the previous EVC station deployment plan, how to place more EVC stations such that the total Social value is maximized?*

For the incremental SOCD problem, [15] investigated a relevant problem, that is, determine how to arrange the chargers based on a historical deployment plan and a number K which is the number of extra EVC stations we want to install. Our incremental SOCD problem is different from that of [15] since we add constraint on total budget instead of number of EVC stations. And comparing with [15], the most distinguished point of our SOCD problem under incremental setting is that, again, we maximize the total influence from a whole social perspective.

Algorithm 5: *IncrementalSOCD*

```
Input: road network: G = (V, E, τ, δ), charging demand: {d_v | v ∈ V},
       previous EVC station deployment plan: P, extra budget: B^i
Output: the incremental EVC station deployment plan: P^i
1 curr_pos ← {S.pos | S ∈ P};
2 V' ← V − {v | τ(v) ∈ curr_pos};
3 invoke Algorithm 2 or 4 on the remained candidate node set V' to get the
    incremental EVC station deployment plan P^i;
4 return P^i;
```

Fortunately, the algorithms we proposed, *Bounding&Optimizing* in Algorithm 2 and *RegionPartition* in Algorithm 4, can both be extended to the incremental case naturally since these two algorithms are based on greedy strategy where in each time we pick one best location to build a station. To retrieve an incremental deployment plan based on the existing deployment, we invoke either *Bounding&Optimizing* or *RegionPartition* on the road network without nodes that have been deployed a station previously. We denote this road network as V'. The framework for solving the incremental SOCD problem is shown in Algorithm 5. Note that, the time complexity of Algorithm 5 depends on the selection of Algorithm 2 or Algorithm 4 in line 3. If we select Algorithm 2, it takes $O(|V'|^2)$; however, if Algorithm 4 is selected, time complexity is $O(|V'|^2/m)$, where m is the partition number in Algorithm 4.

4 EXPERIMENTAL STUDY

In this section, we conduct experiments on both real and synthetic datasets under various parameter settings. To demonstrate the efficiency and effectiveness, we report both the CPU time and the *Social* value of algorithms introduced in Section 3. All of the experiments were conducted on a server with Intel(R) Xeon(R) CPU E5-2650 @ 2.60GHz and 32GB main memory, and all the algorithms were implemented in C++ and executed on Ubuntu 16.04.

4.1 Experiment Setup

We first introduce experiment configurations, including data preparation, parameter setting, and competitor algorithms.

Data preparation. We conduct all the experiments on Shanghai road network data[2], which contains 20,337 nodes and 106,870 edges. For convenience, we pre-calculate all the pairwise shortest distances (i.e., $dist(u,v)$ for any $u, v \in V$) via a distributed Dijkstra's algorithm. To estimate the rural degree which is used to evaluate social benefit in Eq. (2), we select 17 major center points, denoted by $p_1, p_2, \cdots, p_{17}$, from 17 administrative districts of Shanghai. Then, for any node v in the road network, we estimate the rural degree of this node by,

$$w(\tau(v)) = g(\min_{i=1\cdots17} ||\tau(v) - p_i||^2), \qquad (16)$$

where $||\cdot||$ is 2-norm and $g(\cdot)$ is a function used for normalization.

To calculate *Benefit* and *Cost*, we collect massive trajectory data[3] to estimate the charging demand d_v of all nodes in road network. Different from some works like [15, 16] using taxi trajectories, we collect trajectories of various types of vehicles which can better simulate the real traffic condition and demand of charging. We assume that the charging demand d_v is proportion to the volume of traffic flow nearby location $\tau(v)$. First, for each node v, we retrieve trajectories which have location records whose distance from $\tau(v)$ is less than 1 km. Note that EV drivers will not travel too far to seek a station for charging and we assume 1 km is an appropriate value. Then, for the retrieved trajectories, assuming that $t_j(v)$ is the time stamp of j^{th} trajectory travelling to somewhere nearby $\tau(v)$, we set a time window whose length is 5 min to filter out all the trajectories with $t_j(v)$ out of the window. We select 5 min as the length of time window since the GPS sampling interval in the raw trajectory data is 4-6 min. To smooth the result, we set 10 different time windows, count the number and regard the average number as the estimation of d_v.

Besides, for the *estate_price* at each location $\tau(v)$, here, we use a Gaussian distribution to generate samples. Specifically, we assume that $estate\_price \sim \mathcal{N}(\mu, \sigma^2)$ where μ is the expected estate price of Shanghai and σ^2 is fixed to 400,000 which is achieved by analyzing Shanghai estate price data. Note that, in real application, decision makers can manually modify the distribution of estate price to adapt to different real world applications.

Algorithm 6: *baseline*

```
Input: road network: G = (V, E, τ, δ), charging demand: {d_v | v ∈ V}
Output: an EVC station deployment plan P
P ← φ;
B ← initial total budget;
sort all the nodes v ∈ V by d_v in descending order;
while B > 0 do
    pop v with highest d_v from V;
    S.pos ← τ(v);
    start adding chargers in S from the chargers with highest power to lower
      ones if budget is sufficient;
    P ← P ∪ {S} update remained budget B;
return P;
```

SOCD Approaches and Baseline. We implement the two main algorithms for solving SOCD problem, one is *Bounding&Optimizing* in Algorithm 2 and another is *RegionPartition* in Algorithm 4, which are denoted by B&O and RP respectively for brevity. Specifically, in algorithm RP, to partition the whole region via Voronoi diagram, we select seeds as $p_1, p_2, \cdots, p_{17}$, which are the major center points in 17 districts of Shanghai used for estimating the rural degree. The partition results are already shown in Figure 4.

For the baseline algorithm, as our work is the first one taking comprehensive social influence into consideration, and the optimization goal is too complex to use existing LP solvers, here, we propose a demand-first greedy baseline algorithm (denoted by "baseline" in short) shown in Algorithm 6. Since in the worst case, the baseline algorithm scans all the nodes to set EVC stations, which leads to highest running time $O(|V|)$. However, intuitively, it is easy to see that baseline algorithm will exhaust all the budget much faster than B&O and RP, which produces low *Social* value since we

[2] Download from https://figshare.com/articles/Urban_Road_Network_Data.

[3] All the trajectory data is provided by SAIC Motor Co. Ltd.

lose the chance to investigate many possible locations to build an EVC station in very early stage.

Parameter Setting. There are mainly 6 parameters in our solution: 1) λ: the relative importance between *Benefit* and *Social*; 2) α: the relative importance between $Cost_t$ and $Cost_b$; 3) B: initial total budget; 4) K: the maximal number of chargers that an EVC station can install; 5) r_{max}: the maximal radius of influence region; 6) μ: expected value of Shanghai real estate price. The parameters settings are shown in Table 3. Each time, we vary one parameter, while other parameters are set to the underlined default values.

Table 3: Table of parameter settings.

Parameter	Value
λ	[0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
α	[0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
B	[30, 35, 40, 45, 40] (million)
K	[2, 4, 6, 8, 10]
r_{max}	[500, 1000, 1500, 2000, 2500]
μ	[1.2, 1.3, 1.4, 1.5, 1.6] (million)

4.2 Effectiveness Demonstration

As we have proved in Section 2.3, exact evaluation of SOCD problem is extremely costly due to the NP-hardness. Thus, it is impossible to compare our heuristic algorithms with the optimal solutions on large-scale data. To demonstrate the effectiveness of our SOCD approaches, we compare the results achieved by our solutions to SOCD (i.e., B&O and RP) with the optimal one (OPT) which is calculated via brute enumeration on a small-scale SOCD instance with 20 major nodes sampled from the real road network. The results are shown in Table 4. The optimal *Social* value is 0.345253 and our B&O algorithm can achieve 0.322983, which is very close to the optimal. In addition, the *Social* value of the RP algorithm is 0.157776, which is nearly half of that of OPT. Note that, since there are only 20 nodes in the small-scale SOCD instance, the region partition based approach RP cannot achieve relative good result since it is very hard to find a reasonable cut on the small road network. Particularly, B&O and RP run nearly 10^5 times faster than the brute enumeration based algorithm. The seed up ratio of our heuristics will be much higher than 10^5 when the data size increases.

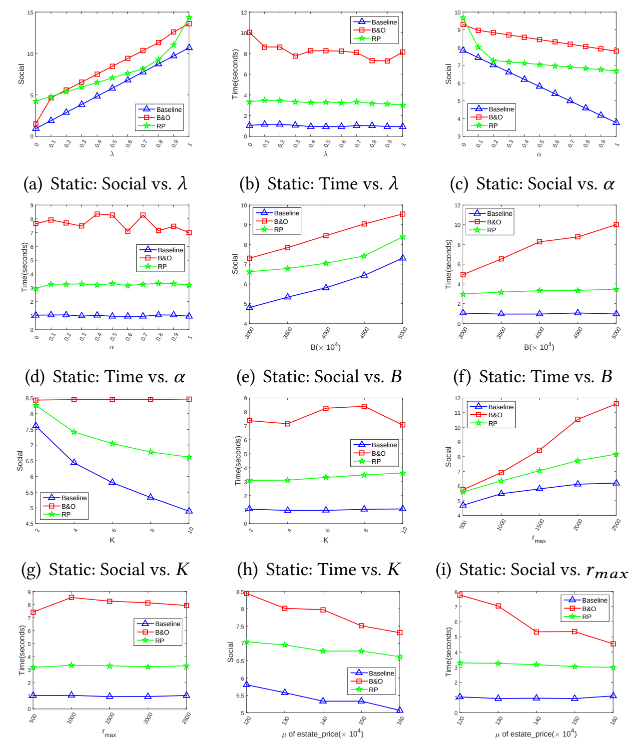
Table 4: Results on a small-scale SOCD instance.

Algorithms	Social value	CPU time
B&O	0.322983	<0.001
RP	0.157776	<0.001
OPT	0.345253	73.930

4.3 Experimental Result of Static SOCD

In this section, we report the *Social* value and CPU time of our solutions to SOCD problem, B&O, RP and baseline, on real dataset and synthetic dataset. All the experimental results under different parameter settings are shown in Figure 5 (results of incremental case are shown in Figure 6).

Result Overview. Figures 5(a), 5(c), 5(e), 5(g), 5(i) and 5(k) report the *Social* value of different parameters shown in Table 3; on the other hand, Figures 5(b), 5(d), 5(f), 5(h), 5(j) and 5(l) report the CPU time under different parameter settings. We can see that, B&O always has the highest *Social* comparing with baseline and RP. However, B&O takes much more time than the other two algorithms, where the run time of RP is very close to that of baseline, which is the most efficient algorithm. The reason is that, RP is based on sub-region partition in which we regard each sub-region as an independent part and conduct greedy placing strategy.



(a) Static: Social vs. λ (b) Static: Time vs. λ (c) Static: Social vs. α
(d) Static: Time vs. α (e) Static: Social vs. B (f) Static: Time vs. B
(g) Static: Social vs. K (h) Static: Time vs. K (i) Static: Social vs. r_{max}
(j) Static: Time vs. r_{max} (k) Static: Social vs. μ (l) Static: Time vs. μ

Figure 5: Results of static SOCD w.r.t. λ, α, B, K, r_{max} and μ.

Effect of B. B is the initial total budget which are granted for building EVC stations among a city. Figures 5(e) and 5(f) show the *Social* value and CPU time of three algorithms by varying $B = 30, 35, 40, 45, 40$ (million RMB). With the increase of B, *Social* values of all the three algorithms increase. The reason is natural, more initial budget means more chargers, which leads to the increase of social *Benefit* and the decrease of both $Cost_t$ and $Cost_b$. Besides, for the running time, we can see in Figure 5(f), the CPU time of algorithm B&O increases when B increases. According to the complexity analysis in Section 3.2, the more initial budget, the more iterations are needed before the termination of B&O. In the worst case, if $B \rightarrow +\infty$, the time complexity will be $|V|^2$ where $|V|$ is the total number of nodes in the road network.

Effect of K. K denotes the maximal number of chargers can be installed per station. Figures 5(g) and 5(h) report the results by setting $K = [2, 4, 6, 8, 10]$. *Social* value of RP and baseline decreases when K increases; whereas that of B&O remains stable even slightly increases. We give the reasons as follows. For baseline, higher value of K means that we can set more chargers at a single EVC station, indirectly leading to faster spending of budget, which is one of the factors leading to low value of *Social* according to the analysis in the discussion of baseline algorithm. For RP, since each sub-region is independent with each other, it is easy to fall into local optimal and not taking good use of K. However, B&O does not suffer from this point which makes it robust to K. Besides, for the run time, it remains stable for all of three algorithms since K is not the influence factor of time complexity.

Effect of r_{max}. We also test the influence of the maximal radius of influence region r_{max} and the results are presented in Figures 5(i) and 5(j) where r_{max} is set to [500, 1000, 1500, 2000, 2500]. For the three algorithms, *Social* increases when r_{max} increases. The reason is straightforward, there would be more nodes covered by influence region of a newly deployed EVC station when r_{max} increases. As for the run time, it is similar to result parameter K, total run time of all the three algorithms remains stable w.r.t. r_{max} since r_{max} is not influential to time either.

Effect of μ. In Figures 5(k) and 5(l), we experimentally study the effect of the expectation of estate price μ, which we have discussed above in the data preparation part. We find that, when μ increases, *Social* value of three algorithms, baseline, B&O and RP, decreases. That is because, high expected estate price will lead to large proportion of initial budget is spent for buying estate, instead of installing chargers, which will decrease the *Benefit* and increase the *Cost*, and finally decrease the *Social* value. Besides, CPU time of B&O also decreases as μ decreases since high estate price will increase the total budget cost for setting up one EVC station, which will use up all the initial budget very soon to end the iteration.

In summary, on the real road network data, B&O can always achieve the highest overall *Social* value, but it has the highest run time among all the approaches. The baseline which is based on demand-first greedy strategy is always the fastest one but suffers from relative low *Social* value. Instead, the region partition based algorithm RP is a good compromise between run time and *Social* value; namely, RP can reach high *Social* value within time close to baseline. Decision makers can select different solutions based on the realistic conditions. Note that, due to the space limit, we only report the experimental results w.r.t. the parameters shown in Table 3. Other parameter settings such as different distribution of charging demand d_v and variance of the distribution of estate price have similar results to Figures 5 and 6, and thus are omitted here.

4.4 Experiment Results of Incremental SOCD

In Section 3.4, we have discussed how to extend our solutions to static SOCD problem to solve the Incremental SOCD problem and the basic framework is shown in Algorithm 5. In this section, we conduct experiments to test the performance of our solutions under such case by varying different parameters as the same setting shown in Table 3. To simulate an incremental SOCD scenario, we divide the total budget into 4 parts: B million, 1 million, 1 million and 1 million. The B million budget is used to get an initial EVC deployment plan and 1 million extra budget is granted incrementally for three times to add new EVC stations. For the three SOCD approaches, we also denote their corresponding incremental version as B&O, RP and baseline respectively. In Figure 6, we report the experiment results of incremental SOCD. Since the performance and the trend w.r.t. each parameter is similar to that of the static case which has been analyzed in Section 4.3, we omit the redundant analysis. Besides, we also test the performance of the incremental SOCD by varying the extra budget in 1, 2, 3, 4 and 5 million and this result is shown in Appendix **??** due to the space limit.

5 RELATED WORKS

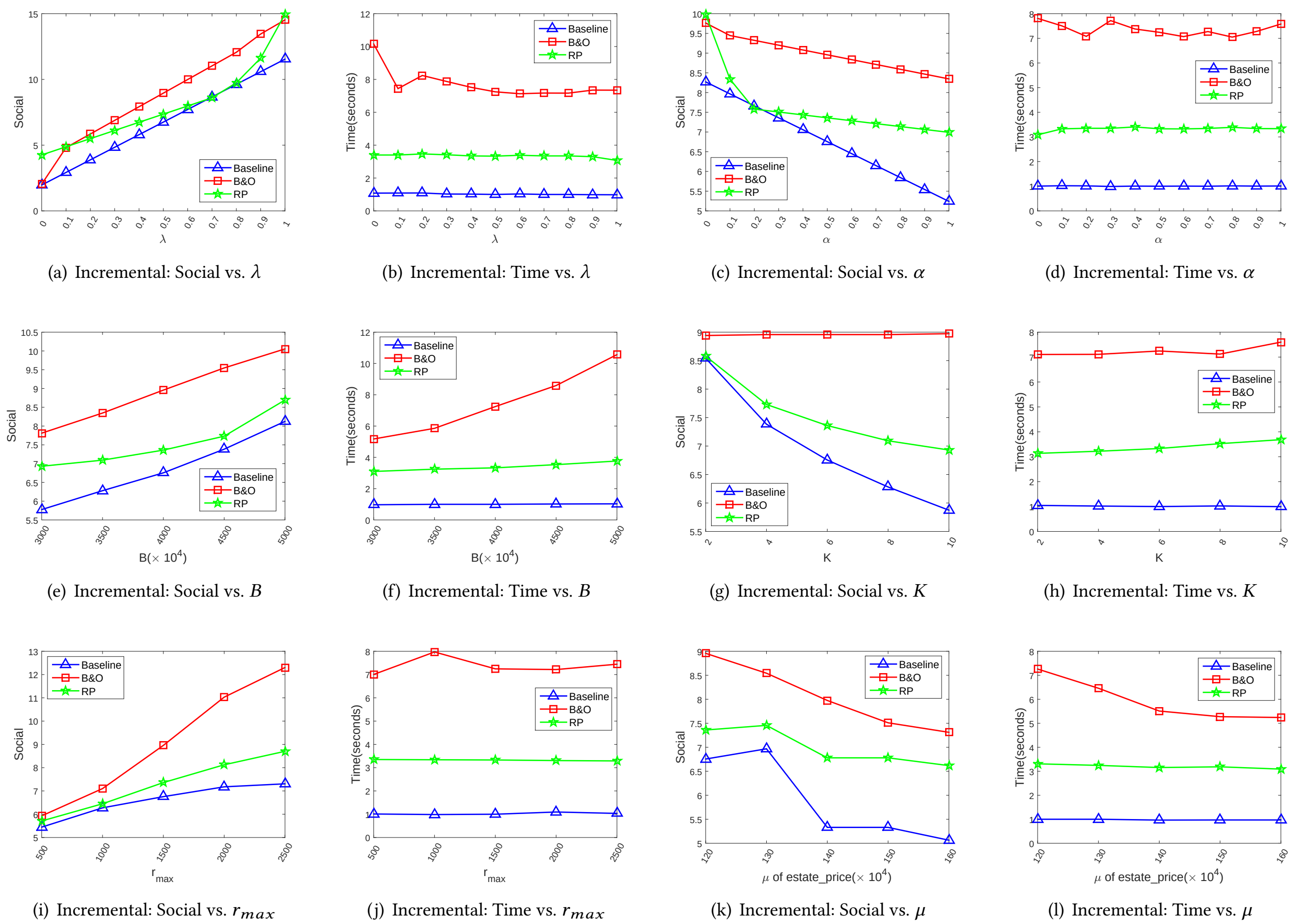
To the best of our knowledge, this paper is the first one considering about maximizing the social influence of EVC stations deployment plan. In this section, we investigate some previous literatures which are related to our topic.

Facility Location Problem. Facility location (FL) problem is one of the fundamental theoretical problems which has been investigated in [6, 10, 14]. Given a set of candidate facilities' locations, such as warehouses and gas stations, and a set of nodes with demand that can be satisfied by traveling to some facility, a general facility location problem is to decide the location of facilities, to minimize the total travel cost from nodes with demand to their selected facilities. Specifically, [10, 14] studied the case whose distances between facilities and nodes are in a metrics space, which is called "metric facility location" (MFL) problem. [10] pointed that it is hard to approximate within any constant ratio less than 1.463 and [14] achieves the current best ratio, which is 1.488. However, as we have discussed above, our SOCD problem is much more complex than FL both from optimization objective and constraints, which prevents us using current solutions to facility location problem.

EVC Related Optimization Problem. As we have mentioned in the Introduction, most current literatures about EVC related optimizing problem focus on partitioned regions of a city [8, 15, 20]. These works return the deployment of EVC stations within a region or a cell, instead of some concrete location. Note that, the meaning of "partitioned region" is totally different from what we use in the algorithm *Region Partition Based Deployment* in Section 3.3. Specifically, [8] estimates the optimal charger distribution within a region such that the total EV drivers' discomfort can be minimized. [15] considers how to place extra K EVC stations based on a given EVC distribution. Another perspective provided by [20] is using game theory to model the interaction between EVC deployment and EV's selection to EVC station. Note that, unfortunately, we cannot borrow ideas from these EVC related works due to the following reasons. First, we focus on deciding the concrete location on road network where an EVC station should be installed. Second, we propose more realistic assumption to an EVC station, where a station might have multiple types of charging plugs with different charging power and price. Third, we define the optimization objective as social influence, which is much more complex than any other previous works with similar topics. Besides, for the incremental SOCD, we consider extra budget instead of extra K stations, which is more reasonable since the number K is usually hard to decide.

6 CONCLUSION

With the continuously increasing charging demand of electric vehicles, how to place EV chargers (EVC), within a city, to achieve positive social influence is becoming urgent challenges. In this paper, we propose a new EVC station placing problem called *Social-Aware Optimal Electric Vehicle Charger Deployment* (SOCD) which considers multiple complex social influence of EVC arrangement. Since SOCD problem is both NP-hard and hard to approximate within any constant, we propose two efficient heuristic algorithms,



(a) Incremental: Social vs. λ (b) Incremental: Time vs. λ (c) Incremental: Social vs. α (d) Incremental: Time vs. α

(e) Incremental: Social vs. B (f) Incremental: Time vs. B (g) Incremental: Social vs. K (h) Incremental: Time vs. K

(i) Incremental: Social vs. r_{max} (j) Incremental: Time vs. r_{max} (k) Incremental: Social vs. μ (l) Incremental: Time vs. μ

Figure 6: Experimental results of incremental SOCD w.r.t. λ, α, B, K, r_{max} and μ.

Bounding&Optimizing Based Greedy Deployment and *Region Partition Based Deployment*. Finally, by conducting extensive experiments on a real road network, we demonstrate both efficiency and effectiveness of our proposed algorithms.

REFERENCES

[1] National Energy Administration. [n. d.]. Plan of Developing Clean-Energy Vehicles (2012-2020). http://www.nea.gov.cn/2012-07/10/c_131705726.htm. Accessed Oct 10, 2017.

[2] U.S. Energy Information Administration. [n. d.]. International Energy Outlook 2017. https://www.eia.gov/outlooks/ieo/. Accessed Oct 10, 2017.

[3] Franz Aurenhammer. 1991. Voronoi diagramsâĂŤa survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)* 23, 3 (1991), 345–405.

[4] Sanya Carley, Rachel M Krause, Bradley W Lane, and John D Graham. 2013. Intent to purchase a plug-in electric vehicle: A survey of early impressions in large US cites. *Transportation Research Part D: Transport and Environment* 18 (2013), 39–45.

[5] Robert B Cooper. 1981. *Introduction to queueing theory.* North Holland,.

[6] Gérard Cornuéjols, George L Nemhauser, and Lairemce A Wolsey. 1983. *The uncapacitated facility location problem.* Technical Report. Carnegie-mellon univ pittsburgh pa management sciences research group.

[7] Steven Fortune. 1987. A sweepline algorithm for Voronoi diagrams. *Algorithmica* 2, 1-4 (1987), 153.

[8] Yajing Gao and Yandong Guo. 2013. Optimal planning of charging station for phased electric vehicle. *Energy and Power Engineering* 5, 04 (2013), 1393.

[9] Michael Grant, Stephen Boyd, and Yinyu Ye. 2008. CVX: Matlab software for disciplined convex programming.

[10] Sudipto Guha and Samir Khuller. 1999. Greedy strikes back: Improved facility location algorithms. *Journal of algorithms* 31, 1 (1999), 228–248.

[11] Long Jia, Zechun Hu, Yonghua Song, and Zhuowei Luo. 2012. Optimal siting and sizing of electric vehicle charging stations. In *Electric Vehicle Conference (IEVC), 2012 IEEE International.* IEEE, 1–6.

[12] Hande Küçükaydin, Necati Aras, and I Kuban Altınel. 2011. Competitive facility location problem with attractiveness adjustment of the follower: A bilevel programming model and its solution. *European Journal of Operational Research* 208, 3 (2011), 206–220.

[13] Albert YS Lam, Yiu-Wing Leung, and Xiaowen Chu. 2014. Electric vehicle charging station placement: Formulation, complexity, and solutions. *IEEE Transactions on Smart Grid* 5, 6 (2014), 2846–2856.

[14] Shi Li. 2011. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Automata, languages and programming* (2011), 77–88.

[15] Yanhua Li, Jun Luo, Chi-Yin Chow, Kam-Lam Chan, Ye Ding, and Fan Zhang. 2015. Growing the charging station network for electric vehicles with trajectory data analytics. In *Data Engineering (ICDE), 2015 IEEE 31st International Conference on.* IEEE, 1376–1387.

[16] Chen Liu, Ke Deng, Chaojie Li, Jianxin Li, Yanhua Li, and Jun Luo. 2016. The Optimal Distribution of Electric-Vehicle Chargers across a City. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on.* IEEE, 261–270.

[17] Zhipeng Liu, Fushuan Wen, and Gerard Ledwich. 2013. Optimal planning of electric-vehicle charging stations in distribution systems. *IEEE Transactions on Power Delivery* 28, 1 (2013), 102–110.

[18] Department of Energy. [n. d.]. FACT SHEET: Obama Administration Announces Federal and Private Sector Actions to Accelerate Electric Vehicle Adoption in the United States. https://energy.gov/articles/fact-sheet-obama-administration-announces-federal-and-private-sector-actions//-accelerate. Accessed Oct 10, 2017.

[19] Vijay V Vazirani. 2013. *Approximation algorithms.* Springer Science & Business Media.

[20] Yanhai Xiong, Jiarui Gan, Bo An, Chunyan Miao, and Ana LC Bazzan. 2015. Optimal Electric Vehicle Charging Station Placement.. In *IJCAI*. 2662–2668.

A PROOF OF THEOREM 2.1

Proof. We reduce the 0-1 Knapsack problem [19], which is a well-known NP-complete problem, to our SOCD problem. An instance of the 0-1 Knapsack problem is as follows: given a set of m items associated with weights $\{w_i\}$ and values $\{v_i\}$ where $i = 1, \cdots, m$, we aim to decide whether there is a subset of items S with total weight $\leq W$, so that the total value is equal to a given value V.

For notational simplicity, let $maxv = \max(\{v_i\}_{i=1}^m)$ and obviously, $v_i \leq maxv$ holds for $\forall v_i \in V$. Then we can construct an instance of the SOCD problem from the instance of the 0-1 Knapsack problem as follows:

- For m items in the 0-1 Knapsack problem, the road network G contains m nodes and each node corresponds to an item. Note that, the edge information of G, including connection and length, is set randomly since it does no influence on the reduction.
- For each node i in the graph G, the rural degree in Eq. (2), $w(\tau(v_i))$, is set to $\log \frac{1+v_i/maxv}{1-v_i/maxv}$.
- We suppose the maximum number of chargers in an EVC station K is 1 and the cost of installing one charger is fixed to f. Then, for each node i in the graph G, the cost of deploying an EVC station at this node, which is $estate\_price + f$, is set to w_i, exactly the same as the weight of i^{th} item.
- The maximum influential radius r_{max} of the influence region in Eq. (1) is set to a value less than the minimum distance between all pairs m nodes in the graph G.
- The total budget B for deploying EVC stations is equal to the weight of the knapsack W.
- We set $\lambda = 1$ of the optimization objective shown in Eq. (7).

Given an instance of the above problem, we want to decide whether there exists an EVC deployment plan P such that $Social = \frac{V}{maxv}$ and $\sum_{s \in P} cost_s \leq W$.

Next, we prove that an instance of the 0-1 Knapsack problem is YES if and only if an instance of the decision version of SOCD problem is YES. Since $\lambda = 1$, then $Social = Benefit$. So we only consider the social benefit in Eq. (2). Besides, we set r_{max} less than the minimum distance between all pairs of distance. Then, it is obvious that any node in the graph can only be covered by the EVC station built at the same node. Since each item in the instance of 0-1 Knapsack problem corresponds to one node in the road network of SOCD problem, without loss of generality, let $\{v_1, v_2, \cdots, v_{|P|}\}$ denote the node(s) at which we deploy the EVC stations in the deployment plan P. Accordingly, the social influence of P, $Social$, in the instance of our SOCD problem is,

$$\begin{aligned} Social &= \lambda \cdot Benefit - (1-\lambda) \cdot Cost \\ &= 1 \cdot Benefit - 0 \cdot Cost \\ &= \sum_{v_i \in P} \left(\frac{2}{1 + \exp\{-\log \frac{1+v_i/maxv}{1-v_i/maxv}\}} - 1 \right) \\ &= \sum_{v_i \in P} \left(\frac{2}{1 + \frac{1-v_i/maxv}{1+v_i/maxv}} - 1 \right) \\ &= \sum_{v_i \in P} \left(1 + \frac{v_i}{maxv} - 1 \right) = \frac{\sum_{v_i \in P} v_i}{maxv} \end{aligned}$$

Therefore, given V, if there exists an EVC deployment plan P such that $Social = \frac{V}{maxv}$ and $\sum_{s \in P} cost_s \leq W$, then there should an subset of items $\{v_1, v_2, \cdots, v_{|P|}\}$ in the 0-1 Knapsack problem such that the total weight $\leq W$ and the total value is equal to the given value V.

From the justification above, the decision version of the SOCD problem is NP-complete and the optimization version of the SOCD problem is NP-hard. □

B PROOF OF THEOREM 3.1

Proof. Since each node v must select one and only one station S, the lower bound of the optimization goal shown in Definition 9 is every node taking the station with lowest assignment cost $Cost_a(v, S)$ defined in Eq. (9). Thus, greedy yields the optimal naturally. □

C PROOF OF LEMMA 3.2

Proof.

$$\begin{aligned} g(S_i) &= \frac{Social(P \cup \{S_i\}) - Social(P)}{f(S_i)} \\ &= \frac{\lambda \Delta Benefit - (1-\lambda)\Delta Cost}{estate\_price(S_i) + \sum_{i=1}^{k} x_{S_i}^{(i)} f_i} \\ &\leq \frac{\lambda \Delta Benefit}{estate\_price(S_i)}, \end{aligned} \tag{17}$$

where $\Delta Benefit$ is the difference of $Benefit$ after and before deploying EVC station S_i at location $\tau(v_i)$.

Then, for $\Delta Benefit$, the highest value of $\Delta Benefit$ corresponds to the case that the radius of S_i's influence region reaches r_{max}. Thus, let $I_1^*(S_i)$ be the number of nodes covered by the circular region centering at $S_i.pos$ with radius r_{max}, and the inequality shown in Eq. (12) always holds. □