Transportation Research Part B 86 (2016) 163–180

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

The electric vehicle touring problem

Chung-Shou Liao[a], Shang-Hung Lu[a], Zuo-Jun Max Shen[b,*]

[a] Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, Taiwan
[b] Department of Civil and Environmental Engineering, and Department of Industrial Engineering and Operations Research, University of California-Berkeley, Berkeley, CA 94720, USA

ARTICLE INFO

Article history:
Received 4 June 2014
Revised 5 February 2016
Accepted 5 February 2016
Available online 27 February 2016

Keywords:
Approximation algorithm
Battery swap
Electric vehicle
Shortest path
Traveling salesman problem
Vehicle routing

ABSTRACT

The increasing concern over global warming has led to the rapid development of the electric vehicle industry. Electric vehicles (EVs) have the potential to reduce the greenhouse effect and facilitate more efficient use of energy resources. In this paper, we study several EV route planning problems that take into consideration possible battery charging or swapping operations. Given a road network, the objective is to determine the shortest (travel time) route that a vehicle with a given battery capacity can take to travel between a pair of vertices or to visit a set of vertices with several stops, if necessary, at battery switch stations. We present polynomial time algorithms for the *EV shortest travel time path problem* and the *fixed tour EV touring problem*, where the fixed tour problem requires visiting a set of vertices in a given order. Based on the result, we also propose constant factor approximation algorithms for the *EV touring problem*, which is a generalization of the traveling salesman problem.

1. Introduction

Transportation is one of the fastest-growing sources of greenhouse gas emissions that contribute to climate change. In the United States, transportation accounts for approximately 25 percent of total greenhouse gas emissions (U.S. EPA Environmental Protection Agency, 2012). Consequently, during the last decade, the automobile industry has developed an increasing number of electric (battery) vehicles or hybrid electric vehicles to deal with the rising cost of energy. Electric vehicles (EVs), which release almost no air pollutants, could make a significant contribution to maintaining the quality of the environment. The Electric Power Research Institute estimates that EVs will account for 6%–30% of the vehicles in use by 2030 (Electric Power Research Institute, 2009).

An efficient EV routing service would obviously encourage the transition to electric vehicle use. The U.S. Department of Energy has developed an online service (U.S. DOE Department of Energy, 2012) that provides a route map interface, as well as information about EV charging facilities for EV owners. However, it is very difficult to design an optimal EV route planning service because EVs have some serious limitations. The first is the low energy capacity of batteries. Currently, their range is only 150 to 200 kilometers; hence, EVs are used primarily in urban areas. The second problem is that EV batteries require a long charging time. At the moment, they can be fully recharged from empty in 2 to 6h, depending on the level of charging available at the station. These factors have delayed the growth of the EV market; however, EVs can now be refueled in a

* Corresponding author. Tel.: +1 5106432392; fax: +1 5106421403.
E-mail addresses: csliao@ie.nthu.edu.tw (C.-S. Liao), u9634017@oz.nthu.edu.tw (S.-H. Lu), shen@ieor.berkeley.edu, maxshen@berkeley.edu (Z.-J.M. Shen).

http://dx.doi.org/10.1016/j.trb.2016.02.002
0191-2615/

matter of minutes through a system called battery-swaps. Recently, Tesla Motors (Tesla Motors, 2013) provided the solution via a network of battery switch stations. The state-of-the-art technology leads to a new model of EV route planning.

In this paper, we explore some interesting models that incorporate the battery capacity constraint when an electric vehicle is driven. First, we begin with the *EV shortest travel time path problem*. In this problem, we determine a route from a source to a destination that an electric vehicle with a given battery capacity U can travel along so that the total time including traveling and battery-swaps is minimized. If necessary, the vehicle can stop at several battery switch stations on the route to maintain its movement. Note that we measure our objective in terms of time. That is, the weight of an edge represents the time required for the vehicle to travel through the edge, and the capacity represents the length of time the vehicle can travel with a full battery. Similar to the traveling salesman problem (TSP), the *EV touring problem* involves organizing a tour of a set of cities so that the total time required is minimized. The vehicle visits each city and returns to the origin, stopping at battery switch stations whenever necessary. We consider two scenarios: the *on-site station* and the *off-site station* models in which each city has an *on-site* battery switch station and an *off-site* battery switch station within an acceptable distance, respectively.

Our contribution. The main results obtained in this paper are summarized as follows:

1. We consider the EV shortest travel time path problem and present a simple dynamic programming algorithm that runs in $O(kn^2)$ time, where k is a given upper bound of the number of battery-swaps and n is the order of the graph.
2. We develop efficient polynomial time algorithms for the fixed tour EV touring problem, where the fixed tour constraint requires visiting a set of cities and returning to the origin in a given order. This result extends the previous studies of the *fixed path gas station problem* reported in Khuller et al. (2011), Lin (2008), Lin et al. (2007) by using graph-theoretic techniques.
3. We propose two approximation algorithms within a $\frac{9}{4}$-factor and a $\frac{9}{2}$-factor, respectively, for the uniform and non-uniform cost on-site station EV touring problem. Moreover, if the battery capacity is sufficiently large, the approximation ratio is the same as that of the well-known Christofides algorithm for the TSP, i.e., $\frac{3}{2}$.
4. We also study the off-site station EV touring problem and propose a $\frac{3}{2}(\frac{3+2\alpha}{1-2\alpha})$-approximation algorithm to solve the problem, where α is a given acceptable distance between a city and its nearest battery switch station.

2. Preliminaries

A great deal of research has been devoted to the shortest route planning problem; and many variations and extensions of the problem have been proposed. One related problem is the well-known *capacitated vehicle routing problem*, which involves finding a set of routes that begin at a depot, visit multiple customers and deliver goods, and return to the depot such that the number of vehicles, each of which has a limited carrying capacity, is minimized or the total distance is minimized. Readers may refer to Laporte's survey (Laporte, 2009) and Pillac et al.'s review (Pillac et al., 2013) for further details on the constraints and conditions.

Another related work is the *orienteering problem* where the objective is to find a path of a fixed length from a single source that visits as many locations as possible (Bansal et al., 2004; Blum et al., 2007; Campbell et al., 2011). The EV touring problem can be regarded as an extension of this problem because the goal is to visit as many cities as possible under a fixed (i.e., full) battery capacity.

Compared with the widely studied routing problems, there is a dearth of research on the optimal *refueling problem* (Khuller et al., 2011; Lin, 2008; Lin et al., 2007; Suzuki, 2008; 2009), where the objective is to minimize the total cost of the fuel used. Lin (2008), Lin et al. (2007) investigated the shortest path problem with optimal refueling policies. They proposed a linear time algorithm for the fixed route version and polynomial time algorithms for other variations. Suzuki (Suzuki, 2008; 2009) developed a more comprehensive model that incorporates many operating costs, and conducted numerical studies. Recently, Khuller et al. (2011) proposed the *gas station problem* where the price of gas may vary at every station, so the owner of a petrol-powered vehicle must decide the amount of gas he/she will purchase (i.e., a fraction of the tank's capacity) at a particular gas station in order to minimize the total cost of gas required. They also study the *tour gas station problem*, where the objective is to find the cheapest tour that can visit a set of vertices and return to the origin, so that the total cost of the gas required is minimized.

Subsequently, Erdoğan and Miller-Hooks presented the *green vehicle routing problem* (Erdoğan and Miller-Hooks, 2012) and Schneider et al. proposed the similar *electric vehicle routing problem* (Schneider et al., 2014). They combined the *vehicle routing problem* with the possibility of filling alternative fuels or charging a vehicle's battery at stations along the routes. Both works provided meta-heuristic algorithms and an analysis of numerical experiments. The setting of the problems is similar to that of the above optimal refueling problem; the only difference is that the objective is to minimize the total distance instead of the total fuel cost. Recently, Adler and Mirchandani (2014) considered routing of many electric vehicles through a network of battery switch stations and developed an algorithm to control battery swap loads across the stations such that the average delay of every vehicle is minimized. In addition, a number of previous studies (Artmeier et al., 2010; Eisner et al., 2011; Kobayashi et al., 2011; Sachenbacher et al., 2011; Siddiqi et al., 2011) discussed energy efficient routing of electric vehicles.

To facilitate the understanding of the difference between our models and the models studied in the literature, we next formally introduce the gas station problem.

Given an undirected complete graph $G = (V, E)$ with a source v_s and a destination v_t in V, where $V = \{v_1, v_2, \ldots, v_n\}$, i.e., $1 \le s,\ t \le n$, let each edge $e = (v_i, v_j) \in E$ be associated with a distance cost $W_{ij} = W_{ji}$. A binary variable x_{ij} indicates if a path passes through the edge e from v_i to v_j. Each vertex $v_i \in V$ is associated with two variables, r_i and g_i, which refer to the amount of gas left in the tank and the amount of gas purchased to refuel the vehicle at v_i, respectively. In addition, let the gas price C_i represent the cost of a unit of gas when the vehicle is filled with gas at v_i. Given a tank capacity U and the initial U_s units of gas at v_s, the integer programming model of the gas station problem (Problem GSP) can be formulated as follows. The objective is to minimize the cost of the total amount of gas purchased subject to the six types of constraints below. The first set of constraints forms a path from s to t. The second set of constraints guarantees gas conservation, and the third one is the tank capacity constraint. The last three types of constraints are given based on the assumption and the definition of the variables.

Before we discuss the EV shortest travel time path problem, let us first take a look at the *charging station problem* (Problem CSP). If we allow an electric vehicle to partially recharge its battery at a charging station instead of swapping the battery, we can show that this *charging station problem* is actually the same as the above model. The slight difference between the models' objectives is within a constant factor.

$$\textbf{Problem GSP}: \quad \text{Minimize} \quad \sum_{i=1}^{n} C_i g_i$$

$$\sum_{j: j\neq i} x_{ij} - \sum_{j:\ j\neq i} x_{ji} = \begin{cases} 1, & \text{if } \ i = s; \\ -1, & \text{if } \ i = t; \quad i = 1, 2, \ldots, n \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

$$(r_i + g_i - W_{ij} - r_j)x_{ij} = 0, \quad (v_i, v_j) \in E \tag{2}$$

$$r_i + g_i \le U, \quad i = 1, 2, \ldots, n \tag{3}$$

$$r_i, g_i \ge 0, \quad i = 1, 2, \ldots, n \tag{4}$$

$$x_{ij} \in \{0, 1\}, \quad (v_i, v_j) \in E \tag{5}$$

$$r_s = U_s \tag{6}$$

To model the charging station problem, we assume that each vertex $v_i \in V$ is associated with two variables, r_i and g_i, which indicate the amount of power left in the battery and the amount of power used to recharge the vehicle at v_i, respectively. The distance cost for each edge $(v_i,\ v_j)$ in E is represented by W_{ij} in terms of the time required to traverse the edge; that is, W_{ij} is the time required when the vehicle drives along the edge. Let the battery-charging time B_i represent the time cost of a unit of battery power when the vehicle recharges its battery at v_i. Given a battery's capacity U, which is measured in terms of the amount of time a vehicle with a full battery can remain on the route, all the constraints of the integer programming model for this charging station problem are the same as those in Problem GSP.

The problem's objective, to minimize the total time cost including traveling and battery-charging, can be rewritten as follows:

$$\begin{aligned} \sum_{(v_i,v_j)\in E} W_{ij}x_{ij} + \sum_{i=1}^{n} B_i g_i &= \sum_{i=1}^{n} g_i + \sum_{i=1}^{n} B_i g_i \\ &= \sum_{i=1}^{n} (1 + B_i) g_i \end{aligned}$$

Because we measure battery power in terms of the amount of time an EV can remain on a route, the time required to traverse an edge $(v_i,\ v_j)$, i.e., W_{ij}, can be considered to be the power needed when the EV drives along the edge $(v_i,\ v_j)$. Therefore, the total amount of time expended to traverse an optimal path from s to t is the same as the total power needed on the path. That is, the optimum solution implies that the vehicle has no remaining power when it reaches the destination v_t, i.e., $r_t = 0$. Hence, the first equality holds, and the algorithms proposed by Khuller et al. (2011) for the gas station problem can be applied directly to Problem CSP.

Assume the vehicle starts at v_s with an empty battery, i.e., $U_s = 0$, where there is a battery-charging station; otherwise, if $U_s \neq 0$, replace v_s by a new source vertex $v_{s'}$ with $W_{s'j} = W_{sj} + (U - U_s)$, for each $j \neq s$, $B_{s'} = 0$ and $U_{s'} = 0$. Then, the reduction shows that the problem of starting from v_s with a non-empty U_s is equivalent to that of starting from $v_{s'}$ with an empty battery when there is a charging station at $v_{s'}$ (Khuller et al., 2011).

Table 1 summarizes the problems considered in this study. For all the problems listed in the table, we assume an undirected complete graph $G = (V \cup F, E)$ with a battery-swap time function $b : F \to R^+$ (or a gas price function $c : F \to R^+$), a battery capacity U (or a capacity of the gas tank U), and a distance edge weight function $w : E \to R^+$.

Table 1
The problems we considered in this study

Problem	Criteria	Objective
The EV shortest travel time path problem (Sections 2 and 3)	Finding a path P from s to t along vertices in F, where $V = \{s, t\}$ and $w(e) \leq U$, for each $e \in P$	$\min \sum_{e \in P} w(e) + \sum_{v \in P \cap F} b(v)$
The gas station problem	Finding a path P from s to t along vertices in F and deciding the amount of gas purchased at those vertices, where $V = \{s, t\}$ and $w(e) \leq U$, for each $e \in P$	$\min \Sigma_{v \in P \cap F} c(v)g(v)$, where $g(v)$ is the amount of gas purchased at v
The EV touring problem (Sections 4 and 5)	Finding a tour P that visits all vertices in V and returns to the origin, stopping at a set of battery switch stations $F^* \subseteq F$ when needed	$\min \sum_{e \in P} w(e) + \sum_{v \in F^*} b(v)$
The tour gas station problem	Finding a tour P that visits all vertices in V and returns to the origin, and deciding the amount of gas purchased along the vertices in P	$\min \Sigma_{v \in P \cap F} c(v)g(v)$, where $g(v)$ is the amount of gas purchased at v

3. Electric vehicle shortest travel time path

To define the EV shortest travel time path problem, we need some new notation. We define a binary variable y_i to indicate if the vehicle swaps a battery at station v_i. In addition, parameter B_i is defined as the time required for a vehicle to replace a battery at v_i. The integer programming model of the EV shortest (travel time) path problem (Problem EVSPP) can be described as follows.

$$\textbf{Problem EVSPP}: \text{Minimize} \quad \sum_{(v_i, v_j) \in E} W_{ij} x_{ij} + \sum_{i=1}^{n} B_i y_i$$

$$\sum_{j:\ j \neq i} x_{ij} - \sum_{j:\ j \neq i} x_{ji} = \begin{cases} 1, & \text{if } i = s; \\ -1, & \text{if } i = t; \quad i = 1, 2, \ldots, n \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

$$(((1 - y_i) r_i + y_i U) - W_{ij} - r_j) x_{ij} = 0, \quad (v_i, v_j) \in E \tag{2}$$

$$0 \leq r_i \leq U, \quad i = 1, 2, \ldots, n \tag{3}$$

$$y_i \in \{0, 1\}, \quad i = 1, 2, \ldots, n \tag{4}$$

$$x_{ij} \in \{0, 1\}, \quad (v_i, v_j) \in E \tag{5}$$

$$r_s = U_s \tag{6}$$

Note that the second set of constraints ensures battery power conservation, which is different from that of Problem GSP. More precisely, the vehicle will stop at some stations to swap its battery if the power left is insufficient to reach the next stop; that is, the electric power left in the battery might be wasted. The total power needed to traverse the optimal path could be more than the total distance cost in terms of the time expended. Hence, its objective cannot be rewritten in a similar manner to that of Problem CSP.

Thus, the key difference between the EV shortest (travel time) path problem (Problem EVSPP) and Problem GSP is that the EVSPP's objective must determine if an electric vehicle requires a 'battery-swap' (i.e., a fully recharged battery) at a battery switch station so that the total time cost including traveling and battery-swaps is minimized.

We use the dynamic programming (DP) technique for solving Problem EVSPP. The reason we do so is that this problem cannot be solved by straightforward greedy algorithms, even when its uniform battery-swap time cost model is considered. By contrast, the gas station problem (Problem GSP) can be simply solved by greedy approaches in the uniform cost model as follows (Khuller et al., 2011). Given a graph $G = (V \cup F, E)$, where $V = \{s, t\}$ and for each $u \in F$, $c(u)$ is a constant c, we assume the vehicle starts with no gas at the vertex s ($U_s = 0$), where there is a gas station with $c(s) = c$; that is, the vehicle must be filled with gas at the beginning of the journey. Then, the *uniform cost gas station problem* is equivalent to the original s, t-shortest path problem and can be solved easily by Dijkstra's algorithm (Dijkstra, 1959). Otherwise, if $U_s \neq 0$, we apply Dijkstra's algorithm from the destination t instead. By using Dijkstra's algorithm beginning at t, we can derive the shortest distance between t and every other vertex. Next, we apply Dijkstra's algorithm from the source s. Then, for every vertex v whose shortest distance from s is within U_s, i.e., $w(s, v) \leq U_s$, we select the minimum of the shortest distance from v to t minus the gas left at v, i.e., $\min\{w(v, t) - (U_s - w(s, v)) \mid w(s, v) \leq U_s\}$. Thus, the uniform cost gas station problem can also be solved by Dijkstra's algorithm.

As mentioned earlier, we let k be an upper bound of the number of battery-swaps. Given a graph $G = (V \cup F, E)$, where $V = \{s, t\}$ and each edge $e \in E$ is associated with a distance weight $w(e)$, we prove that an EV shortest travel time path from s to t with initial U_s units of power in G can be optimally solved by using dynamic programming.

Theorem 3.1. *The EV shortest travel time path problem with at most k battery-swaps can be optimally solved in $O(kn^2)$ time.*

Proof. First, suppose the vehicle starts with an empty battery $U_s = 0$ and there is a battery switch station at s. If there is no battery-switch station at s or $0 < U_s \le U$, the problem can be reduced to the EV s', t-shortest travel time path problem in $G = (V' \cup F, E')$ with $V' = V \setminus \{s\} \cup \{s'\}$, $b(s') = 0$, $U_{s'} = 0$, and $E' = E \setminus \{(s, u) \mid \forall u \in F\} \cup \{(s', u) \mid \forall (s, u) \in E)\}$, where $w(s', u) = w(s, u) + (U - U_s)$. Then, Problem EVSPP can be solved based on the following DP formulation:

$OPT[v, q]$ = The minimum time for the vehicle to travel from a vertex v to the destination t with exactly q battery-swaps; the first battery-swap of these q operations is performed at v.

For every v in $F \cup \{s\}$, each entry of the form $OPT[v, q]$, $1 \le q \le k$, can be computed by the following recurrence:

$$OPT[v, q] = \min_{u \in F}\{OPT[u, q-1] + b(v) + w(v, u) \mid w(v, u) \le U\}, \tag{1}$$

where the boundary condition of the recurrence is given as follows:

$$OPT[v, 1] = \begin{cases} b(v) + w(v, t), & \text{if } w(v, t) \le U; \\ \infty, & \text{otherwise.} \end{cases} \tag{2}$$

In the above DP recursion, let u be the first battery switch station after v when the vehicle travels from v through u to t and swaps a battery at u. We have to keep track of at most $|F|$ different values for all the stations to compute each entry of the table. The optimal solution is derived in the form of $\min_{1 \le \ell \le k}\{OPT[s, \ell]\}$, which can be obtained in $O(kn^2)$ time in a naive way by filling the table whose size is $O(kn)$, where n is the order of G. Note that k is at most the diameter of the given graph G, where a diameter of a graph is the largest distance of a shortest path from one vertex to another, and in the worst case, k could be close to n. $\square$

The major difficulty in solving the recursion (1) is that each update of the above DP recursion does not have good properties, even $b(v) + w(v, u)$, for every u, is known in advance. Thus, it is difficult to find a more efficient way for computing every entry of the table in sublinear time (rather than the $O(n)$-time operation) based on such a recurrence form, even using advanced data structures such as the Fibonacci heap (Driscoll et al., 1998; Fredman and Tarjan, 1987).

We remark that the notion of *communication graph*, which has been used in the *regenerator location problem* (Chen et al., 2010), can be applied to Problem EVSPP. The regenerator location problem involves finding the minimum set of regenerators located in the shortest paths that connect all pairs of vertices such that the distance between every consecutive regenerators in these paths is not larger than a given parameter $d_{\max}$, which specifies the transmission limit of an optical signal in telecommunication networks. Chen et al. (2010) provided a transformation procedure to construct a *communication graph* by adding an edge (u, v) of weight equal to the shortest path distance between u and v, for each pair (u, v) of vertices, into the original graph, if the edge weight is not bigger than $d_{\max}$. This procedure can be done by using the all-pairs shortest path algorithm. Based on the concept, for every pair (u, v) of vertices, we insert an edge (u, v) of weight equal to u, v-shortest path distance plus the battery-swap time $b(u)$ into the input graph in Problem EVSPP, if the u, v-shortest path distance is not larger than the battery capacity U. That is, the newly-added edge (u, v) in the transformed graph represents a shortest path from u to v for the electric vehicle to travel along with a full battery. Therefore, the EV shortest travel time path problem can be reduced to the original shortest path problem in the transformed graph. However, when U is large, e.g., half of the diameter, the computational complexity of the transformation procedure is the same as that of the all-pairs shortest path problem (APSP), which takes essentially cubic time. The currently best algorithm proposed by Williams (2014) can solve the APSP in $\tilde{O}(n^3/2^{\Omega(\sqrt{\log n})})$ time, compared with our dynamic programming approach that runs in $O(kn^2)$ time for Problem EVSPP.

In the following sections, we introduce our main results and investigate the touring problem that incorporates the battery capacity constraint for an electric vehicle. Similar to the EV shortest travel time path problem, it ensures that the vehicle never runs out of power during the shortest (travel time) tour that visits every city and returns to the origin.

4. Fixed tour EV touring

Given a complete graph $G = (V \cup F, E)$ with a set of cities V, a set of battery switch stations F and the vehicle's battery capacity U, where each station $v \in F$ is associated with a battery-swap time $b(v)$, $b : F \to R^+$, and each edge $e \in E$ is associated with a distance weight $w(e)$, $w : E \to R^+$, the goal is to find a tour that enables the vehicle to visit each city in V and return to the origin, and stop at battery switch stations in F when needed, such that the total time cost including traveling and battery-swaps is minimized. We call this generalization of the TSP the *EV touring problem*.

Before discussing the EV touring problem, we consider the fixed tour model, in which the vehicle visits cities that have battery-switch stations (i.e., $V = F$) in a given order during the tour. This problem is an extension of the *fixed path gas station problem* proposed in Khuller et al. (2011), Lin (2008), Lin et al. (2007).

We introduce some new notation: A path P from u to v, or shortly, a u, v-path, is denoted by P: $u \sim v$, and the path can also be represented by a sequence $P : u = v_1 - v_2 - \ldots - v_m - v_{m+1} = v$ if it is of length m. The u, v-path P is called a tour or a cycle of length m if $P: u = v_1 - v_2 - \ldots - v_m - v = u$. The distance weight of a path or a tour P is defined as $w(P) = \sum_{e \in P} w(e)$.

4.1. Uniform cost

Suppose the battery-swap time is identical at every station, i.e., $b(v_i) = c$ for each $v_i \in F$. First, consider a path in a given order $P: v_1 - v_2 - \ldots - v_n$, which consists exclusively of battery switch stations. Without loss of generality, assume the battery capacity U is larger than the distance weight of each edge in P in terms of time spent, i.e., $U > w(v_i, v_{i+1})$, $1 \le i \le n-1$. The vehicle starts at v_1 with an empty battery.

An intuitive strategy for the vehicle is to go as far as possible unless the battery power left is insufficient to reach the next vertex on the path. The next lemma shows that the greedy concept can be used to optimally select appropriate stations for battery-swaps for a given path in the uniform cost model.

Given a fixed path $P: v_1 - v_2 - \ldots - v_n$ consisting exclusively of battery switch stations, each of which has an identical battery-swap time, the optimal strategy for a vehicle to select the minimum number of stations to enable its movement and visit all the vertices sequentially can be described as the following lemma:

Lemma 4.1. *The EV would not stop at a station for a battery-swap unless the battery power left is insufficient to reach the next vertex in P.*

Proof. Let O be the optimal solution of battery switch stations whose total battery-swap time is the minimum, and F^* be the set of battery switch stations derived by the greedy strategy. Assume $F^* \neq O$, which implies that $|O| \le |F^*|$ in the uniform cost model.

Let v_j be the first vertex in $O \backslash F^*$, i.e., with the smallest index j, in P. Let the station v_i be selected immediately before v_j, that is, $v_i \in O \cap F*$, where $i < j$. Suppose the station v_ℓ, $\ell \neq j$, is selected by the greedy strategy immediately after v_i. It implies that $\ell > j$; otherwise, the vehicle can reach v_j from v_i without any battery-swaps and it would drive through v_ℓ. However, this contradicts the greedy choice of v_ℓ. Thus, we replace v_j by v_ℓ, $\ell > j$ to obtain full battery power at v_ℓ and $O = O \setminus \{v_j\} \cup \{v_\ell\}$. Repeating this argument leads to an optimal solution that contains all the battery switch stations of F^*. □

Subsequently, we extend the straightforward greedy concept to the fixed tour model. The major difference is that we have to determine the start vertex for the EV such that the number of battery-swaps is minimized in the fixed tour model. We refer to the result reported by Hsu and Tsai (Hsu and Tsai, 1991), who studied several optimization problems in *circular-arc graphs*. Based on Lemma 4.1, we present a linear time algorithm using graph-theoretic approaches for optimally selecting battery switch stations in a given tour in the uniform cost model. An intersection graph G is called a *circular-arc graph* if its vertices can be put into a one-to-one correspondence with a set of arcs on a circle, such that two vertices are adjacent in G if and only if their corresponding arcs have nonempty intersections. A circular ordering $v_1, v_2, \ldots, v_n$ of G is represented by $b_1 \preceq b_2 \preceq \ldots \preceq b_n \preceq b_1$, where b_i is the right endpoint of the arc v_i; and $b_i \preceq b_j$ means that b_j follows b_i in a clockwise direction.

Given a fixed tour of length n, $P: v_1 - v_2 - \ldots - v_n - v_1$; similarly, $v_i \prec v_j$ means that v_j follows v_i circularly in the ascending order of P. For any vertex v_i, $1 \le i \le n$, we denote a vertex v_ℓ as $\mathrm{FAR}(v_i)$ if the vehicle can drive with a full battery from v_i to v_ℓ circularly such that $w(v_i \sim v_\ell)$ is maximized, i.e., $w(v_i \sim v_\ell) \le U$ and $w(v_i \sim v_\ell - v_{\ell+1}) > U$. Moreover, it can be proved that the relationship between a vertex v_i and $\mathrm{FAR}(v_i)$ has the following *interleaving property*.

Lemma 4.2. *For any two vertices v_i and v_j in a given tour P with $v_i \prec v_j$, we have $FAR(v_i) \preceq FAR(v_j)$.*

Proof. Assume there exist two vertices v_i and v_j with $v_i \prec v_j$, such that $\mathrm{FAR}(v_j) \prec \mathrm{FAR}(v_i)$. Because the vehicle can drive with a full battery from v_i through $\mathrm{FAR}(v_j)$ to $\mathrm{FAR}(v_i)$, it contradicts the fact that $w(v_j \sim \mathrm{FAR}(v_j))$ is maximized. □

Based on the interleaving property, the computation of $\mathrm{FAR}(v_i)$ for each vertex v_i can be solved in linear time (Lin et al., 2007); that is, if $\mathrm{FAR}(v_i) = v_\ell$, then we can compute $\mathrm{FAR}(v_{i+1})$ by starting at v_ℓ. Repeating this argument from v_1 to v_{n-1} circularly to derive every $\mathrm{FAR}(v_i)$ in linear time.

We use a similar idea to that of Hsu and Tsai (Hsu and Tsai, 1991) and construct a directed graph $D = (V, E_D)$, where $V = \{v_1, v_2, \ldots, v_n\}$ and a directed edge $\overrightarrow{(v_i, v_j)} \in E_D$ if and only if $v_j = \mathrm{FAR}(v_i)$, $v_i \prec v_j$. First, we assume that every vertex $v_i \in V$ has its $\mathrm{FAR}(v_i)$; otherwise, the vehicle can begin with a full battery and return to the origin. Consequently, there is at least one directed cycle in D because V is of finite cardinality. Besides, two directed cycles cannot share a common vertex since every vertex has out-degree exactly one in D.

Next, we define $F_{v_i} = \{v_i^{(0)}, v_i^{(1)}, \ldots, v_i^{(m-1)}\}$, where $v_i^{(j+1)} = \mathrm{FAR}(v_i^{(j)})$, $v_i^{(0)} = v_i$, and $v_i \preceq \mathrm{FAR}(v_i^{(m-1)})$. By Lemma 4.1, F_{v_i} is a feasible solution of stations containing v_i. Moreover, for any feasible solution F' containing v_i, we have $|F_{v_i}| \le |F'|$. Let O be the optimal solution of stations and a vertex v_i be called *valid* if $|F_{v_i}| = |O|$. We have the following lemma based on the interleaving property.

Lemma 4.3. *Every directed cycle in the directed graph D consists exclusively of valid vertices.*

Proof. First, we prove that there exists one directed cycle comprised exclusively of valid vertices. Because F_v is the minimum solution of stations containing v, there is a vertex v_i that is valid; i.e., $|F_{v_i}| = |O|$. By assumption, v_i has its own $\mathtt{FAR}(v_i)$ and $\mathtt{FAR}(v_i)$ is contained in F_{v_i}. Again, $\mathtt{FAR}(v_i)$ is also valid because $|F_{\mathtt{FAR}(v_i)}| \le |F_{v_i}| = |O|$. Repeat this argument until there are two indices, a and b, such that $v_i^{(a)} = v_i^{(b)}$, where $a < b$, because V is of finite cardinality. Thus, these valid vertices, starting from $v_i^{(a)}$ and ending at $v_i^{(b)}$, form a directed cycle. Next, we let C^* be the directed cycle consisting exclusively of valid vertices. For a distinct directed cycle C, if any, and an arbitrary vertex v_j in C, we select the vertex v_ℓ in C^* to the immediate left of v_j. Consider $F_{v_j} = \{v_j = v_j^{(0)}, \ldots, v_j^{(m-1)}\}$ and F_{v_ℓ}. By the interleaving property, there must exist a vertex in F_{v_ℓ} that lies between $v_j^{(k)}$ and $v_j^{(k+1) \bmod (m)}$, $0 \le k \le m-1$. Consequently, $|F_{v_j}| \le |F_{v_\ell}| = |O|$ and v_j is valid. We repeat a similar argument as above and show that C consists exclusively of valid vertices. □

Thus, for the uniform cost fixed tour model, in which the vehicle visits all the cities that have battery switch stations in a given order during the tour, we can optimally select appropriate stations for battery-swaps in linear time based on the above lemmas by incorporating the greedy concept of Lemma 4.1 with a similar idea to that of Hsu and Tsai (Hsu and Tsai, 1991) (see Algorithm 1).

Algorithm 1 Select battery switch stations for the fixed tour uniform cost model.

Input: a fixed tour $P: v_1 - v_2 - \ldots - v_n - v_1$ consisting of battery switch stations, the distance of each edge in P and the battery capacity U;
Output: a set F^* consisting of battery switch stations selected;
1: Find $\mathtt{FAR}(v_i)$ for each $v_i \in P$;
2: Let each vertex be marked 'unvisited' and let $v^* = v_1$ be the first vertex;
3: **while** v^* is 'unvisited' **do**
4: Mark v^* as 'visited' and let $v^* = \mathtt{FAR}(v^*)$;
5: **end while**
6: **return** $F^* = F_{v^*}$;

Theorem 4.4. *For the uniform cost fixed tour model, the minimum total battery-swap time as well as the corresponding battery switch stations can be optimally determined in linear time.*

Proof. Based on the above lemmas, when a vertex v^* is visited twice, there exists a directed cycle from v^* to v^*. Thus, v^* is valid and F_{v^*} is the optimal solution of stations.

Regarding the time complexity analysis, $\mathtt{FAR}(v_i)$, for each v_i, can be totally derived in linear time in the initial step. The number of iterations in the while loop is at most $O(n)$; hence, the running time is linear in the order of G. □

4.2. Non-uniform cost

Next, we investigate the non-uniform cost model in which each vertex $v_i \in F$ has a different battery-swap time $b(v_i)$. Similarly, we consider this case in a fixed path, and then extend it to the fixed tour model. The assumptions of the previous subsection still hold; we define $\mathtt{FAR}(v_i)$ for each vertex v_i in a similar way and have the interleaving property as well.

For the fixed path gas station problem, Lin et al. (Lin et al., 2007) proposed a linear time algorithm that solves the problem in a greedy manner. More precisely, when arriving at a vertex v_i, the petrol-powered vehicle refills its gas tank at v_i if there are no stations with cheaper gas price lying between v_i and $\mathtt{FAR}(v_i)$; otherwise, the vehicle would partially refuel to be able to just reach the first station whose gas price is lower than that of v_i. Repeating the argument can derive the optimal solution in a given fixed path. However, this greedy manner cannot work in the *fixed path EV touring problem* which incorporates battery-swap operations, as discussed earlier. Thus, we refer to the result reported by Chang (Chang, 1998) who explored weighted optimization problems in circular-arc graphs, and solve the non-uniform cost model.

Given a fixed path $P: v_1 - v_2 - \ldots - v_n$ consisting of battery switch stations, each of which has a different battery-swap time, we prove that an optimal solution of stations required for traversing this path in a given order can be determined in linear time.

Theorem 4.5. *The non-uniform cost model of the fixed path EV touring problem can be optimally solved in linear time.*

Proof. We show that the problem can be solved based on the DP formulation below:

$\mathtt{MBS}\ (v_j) =$ The minimum time of total battery-swaps for the vehicle to travel from v_1 to $\mathtt{FAR}(v_j)$, and the last battery-swap is performed at v_j.

Suppose the vehicle starts at v_1 with an empty battery. For every $v_i \in F$, $\mathtt{FAR}(v_i)$ can be obtained in linear time as mentioned earlier; similarly, $v_i \prec v_j$ means that v_j follows v_i in the ascending order of P. Then, each entry of the form $\mathtt{MBS}(v_j)$, $1 \le j \le n$, can be computed by the following recurrence:

$$\mathtt{MBS}(v_j) = \begin{cases} b(v_1), & \text{if } j = 1; \\ b(v_j) + \min\{\mathtt{MBS}(v_i) \mid v_i \prec v_j \preceq \mathtt{FAR}(v_i)\}, & \text{if } 2 \le j \le n. \end{cases} \tag{3}$$

In the above DP recursion, note that the vehicle performs a battery-swap at v_i immediately before v_j when traveling from v_1 to $\mathtt{FAR}(v_j)$; therefore, station v_j lies between v_i and $\mathtt{FAR}(v_i)$. The optimal solution is derived in the form of $\min\{\mathtt{MBS}(v_i) \mid v_i \prec v_n \preceq \mathtt{FAR}(v_i)\}$.

Consider the performance of the DP technique. When computing each entry $\mathtt{MBS}(v_{j+1})$, $1 \le j \le n-1$, we have to keep track of all the possibilities for every station v_i satisfying $v_i \prec v_{j+1} \preceq \mathtt{FAR}(v_i)$. More precisely, at each recursion step $j+1$, we need to *insert* $\mathtt{MBS}(v_j)$ into the current pool of possible solutions, and *delete* every station v_k whose $\mathtt{FAR}(v_k)$ precedes v_{j+1}; besides, the minimum value in the pool has to be determined. By using the Fibonacci heap (Fredman and Tarjan, 1987), which allows each of the *insert* and *return-minimum* operations to take $\Theta(1)$ time and the *delete* operation to take $O(\log n)$ amortized time, the recurrence can be solved in $O(n\log n)$ time.

Furthermore, for any v_i and v_j with $v_i \prec v_j$, if $\mathtt{MBS}(v_i) \ge \mathtt{MBS}(v_j)$, then $\mathtt{MBS}(v_i)$ can be directly removed from the solution pool based on the interleaving property; that is, it is impossible to select $\mathtt{MBS}(v_i)$ to comprise the optimal solution because $\mathtt{FAR}(v_i) \preceq \mathtt{FAR}(v_j)$. Therefore, when computing each entry of the recurrence, the pool of possible solutions can be maintained in a sorted list in increasing order. In other words, based on the interleaving property, each of the above *insert, delete* and *return-minimum* operations takes only $\Theta(1)$ time at every recursion step. The optimal solution $\min\{\mathtt{MBS}(v_i) \mid v_i \prec v_n \preceq \mathtt{FAR}(v_i)\}$ can thus be derived in linear time. We remark that the optimal solution of stations required can be obtained by backtracking the computation of the recurrence. □

We extend the DP technique to the fixed tour model where a tour $P: v_1 - v_2 - \ldots - v_n - v_1$ in a circular order is given. Recall that when computing each entry of the DP recursion for the fixed path model, a pool of possible solutions needs to be considered. For each v_j, $1 \le j \le n$, let the pool be defined by $S_j = \{v_i \mid v_i \prec v_j \preceq \mathtt{FAR}(v_i)\}$. Based on the interleaving property, every vertex set S_j can be obtained in linear time. Note that each S_j is nonempty because of the natural assumption that the battery capacity U is larger than the distance weight of each edge in P. One observes that in any feasible solution, there is at least one vertex in each S_j. The following theorem can be derived from Theorem 4.5.

Theorem 4.6. *The non-uniform cost model of the fixed tour EV touring problem can be optimally solved in $O(\delta n)$ time, where for every station v_j, δ is the minimum number of stations at which the vehicle can start with a full battery and reach v_j.*

Proof. For every v_j, $1 \le j \le n$, we let $S_j = \{v_i \mid v_i \prec v_j \preceq \mathtt{FAR}(v_i)\}$ and $S_{\min}$ be the one of the minimum cardinality; that is, the size of $S_{\min}$ is δ. As mentioned above, based on the interleaving property, $S_{\min}$ can be obtained in linear time and an optimal solution of battery-switch stations contains at least one vertex in $S_{\min}$. Then, we use the DP technique in the fixed path model δ times; each starts at a different vertex in $S_{\min}$. After performing the DP procedure for the δ fixed paths, the one with the minimum time of total battery-swaps is the optimal solution for the fixed tour model. Since the optimal DP procedure for each fixed path can be done in linear time by Theorem 4.5, the optimal solution for a fixed tour in a given order can be obtained in $O(\delta n)$ time. Note that in addition to the minimum total battery-swap time, the corresponding battery-switch stations can also be determined. Moreover, it could be assumed that δ is sufficiently smaller than n in practice. □

5. Electric vehicle touring

In this section, we divide the EV touring problem into two models. In the first model, called the *on-site station model*, every city has an *on-site* battery switch station, i.e., $V = F$. The integer programming model of the on-site EV touring problem (Problem OEVTP) can be described as follows.

Problem OEVTP involves determining a tour that visits all the vertices in $V = F = \{v_1, v_2, \ldots, v_n\}$. The objective is to minimize the total time expended on the tour. Assume the vehicle starts with an empty battery. We define a binary variable s_i to indicate if the vehicle starts from station v_i, and the definitions of other variables are similar to those in Problem EVSPP. Note that the first two sets of constraints ensure that the tour visits each city once. The third constraint guarantees that the vehicle starts from one station, i.e., the origin, and based on the fourth set of constraints, i.e., the subtour elimination constraints, and the first two ones, the vehicle returns to the origin. In addition, the fifth and sixth sets of constraints ensure that the vehicle swaps a battery in the origin and begins to travel with a full battery. The seventh and eighth sets of constraints incorporates the concept of the origin into the second one in the integer programming model of Problem EVSPP.

$$\textbf{Problem OEVTP}: \text{Minimize} \quad \sum_{(v_i,v_j)\in E} W_{ij}x_{ij} + \sum_{i=1}^{n} B_i y_i$$

$$\sum_{i:\ i\neq j} x_{ij} = 1, \qquad j = 1, 2, \ldots, n \tag{1}$$

$$\sum_{j:\ j\neq i} x_{ij} = 1, \qquad i = 1, 2, \ldots, n \tag{2}$$

$$\sum_{i=1}^{n} s_i = 1 \tag{3}$$

Table 2
Comparison of the results of the EV touring problem with those reported in Khuller et al. (2011)

Model	Cost	Ratio	Condition	Under condition	(Khuller et al., 2011)
On-site	Uniform	$\frac{3U+6c}{2U+2c}$	$U \geq c \geq 0$	9/4	$\frac{3}{2}$
			$U \gg c \geq 0$	3/2	
	Non-uniform	$\frac{3U+6b_{\max}}{2U+2b_{\min}}$	$U \geq b_{\max}, b_{\min} \geq 0$	9/2	$\frac{3c_{\max}}{2c_{\min}}$
			$U \gg b_{\max}, b_{\min} \geq 0$	3/2	
Off-site	Uniform	$\frac{3(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)}$	$U \geq c \geq 0$	$\frac{3}{4}(\frac{3+2\alpha}{1-2\alpha})$	$\frac{3}{2}(\frac{1+2\alpha}{1-2\alpha})$
			$U \gg c \geq 0$	$\frac{3}{2}(\frac{1+2\alpha}{1-2\alpha})$	
	Non-uniform	$\frac{3(U+2\alpha U+2b_{\max})}{2(1-2\alpha)(U+b_{\min})}$	$U \geq b_{\max}, b_{\min} \geq 0$	$\frac{3}{2}(\frac{3+2\alpha}{1-2\alpha})$	$\frac{3c_{\max}}{2c_{\min}}(\frac{1+2\alpha}{1-2\alpha})$
			$U \gg b_{\max}, b_{\min} \geq 0$	$\frac{3}{2}(\frac{1+2\alpha}{1-2\alpha})$	

$$\sum_{v_i, v_j \in S} x_{ij} \leq |S| - 1, \qquad S \subset V \tag{4}$$

$$s_i \leq y_i \qquad i = 1, 2, \ldots, n \tag{5}$$

$$s_i U \leq r_i \qquad i = 1, 2, \ldots, n \tag{6}$$

$$(1 - s_j)(((1 - y_i)r_i + y_i U) - W_{ij} - r_j)x_{ij} = 0, \quad (v_i, v_j) \in E \tag{7}$$

$$s_j(((1 - y_i)r_i + y_i U) - W_{ij})x_{ij} \geq 0, \qquad (v_i, v_j) \in E \tag{8}$$

$$0 \leq r_i \leq U, \qquad i = 1, 2, \ldots, n \tag{9}$$

$$s_i \in \{0, 1\}, \qquad i = 1, 2, \ldots, n \tag{10}$$

$$y_i \in \{0, 1\}, \qquad i = 1, 2, \ldots, n \tag{11}$$

$$x_{ij} \in \{0, 1\}, \qquad (v_i, v_j) \in E \tag{12}$$

Please align all the above constraints ((1) to (12))

Khuller et al. (2011) proposed the *tour gas station problem*, which involves finding the cheapest tour that visits all the cities in V and possibly some gas stations in F. They proved that the uniform cost on-site station model of this problem can be reduced to the original TSP, where the gas price is identical at each station. In contrast, the uniform cost on-site station model of the EV touring problem cannot be transformed into the TSP directly because of the 0–1 recharging operations, i.e., battery-swaps. We provide an illustration to demonstrate the difference in the Section A.1.

Moreover, the EV touring problem is simply seen to be NP-hard by reducing from the TSP: set the battery capacity U equal to the summation of total edge (distance) weights; now the electric vehicle can travel without any battery-swaps. Therefore, we propose an approximation algorithm for Problem OEVTP by using the graph-theoretic approaches for solving the fixed tour EV touring problem. We also design a branch-and-bound exact algorithm for Problem OEVTP and conduct numerical studies to compare the performance of the approximation algorithm with that of the exact solution methodology (see the Section A.2). The computational result shows the effectiveness of the approximation algorithm.

The second model presents a more interesting scenario in that $V \nsubseteq F$, but there is at least one battery switch station within an acceptable distance of every city. This is called the *off-site station model*. According to the natural assumption in Khuller et al. (2011),Li et al. (1992), we let $\alpha = \max_{v \in V} \min_{u \in F}\{w(v, u)\}/U$ and let the acceptable distance be at most αU, $0 \leq \alpha < 1/2$. In practice, a vehicle cannot visit a city if the nearest station is more than the distance $U/2$ from the city. Note that the vehicle is allowed to visit a city multiple times, if necessary, in the off-site station model, because the city might have an on-site battery switch station that is also the off-site station of some other nearby cities. We also assume the distance weights satisfy the *triangle inequality* (Khuller et al., 2011).

Table 2 summarizes the results of the EV touring problem, where $b_{\min} = \min_{v \in F}\{b(v)\}$ and $b_{\max} = \max_{v \in F}\{b(v)\}$. Note that comparisons between the results of the EV touring problem and those of the tour gas station problem (Khuller et al., 2011) are presented in the last two columns. For the latter problem, although it was claimed that $\frac{c_{\max}}{c_{\min}}$ was around 1.2 in practice, where $c_{\min} = \min_{v \in F}\{c(v)\}$ and $c_{\max} = \max_{v \in F}\{c(v)\}$, the ratios derived in this study can be regarded as constants directly under such reasonable conditions in the on-site station model.

5.1. On-site station EV touring

The approximation algorithm is implemented in two phases. First, we exploit Christofides algorithm (Christofides, 1976) to derive a route plan, i.e., a permutation of all the vertices in V. Christofides algorithm is a combination of the minimum spanning tree of a complete graph G with the minimum weight perfect matching on the vertices with odd degree in the tree. The result of this algorithm is a Hamiltonian tour with a 1.5-approximation ratio if the distance function satisfies the triangle inequality property. Then we use the algorithms derived in the previous section to optimally identify appropriate battery switch stations in both the uniform and non-uniform cost cases and ensure that a vehicle's movement can be maintained. The steps of the approach for solving the on-site EV touring problem are detailed in Algorithm 2.

Algorithm 2 Finding an approximation to the on-site EV touring problem.

Input: $G = (V, E)$ of order n; a distance function $w : E \to R^+$, a battery-swap function $b : V \to R^+$ and the vehicle's battery capacity U;

Output: A tour P that visits all the vertices in V with a set of stations $F^* \subseteq V$ for battery-swaps;

1: Use Christofides algorithm to determine the visiting order of all cities in V, denoted by $P : v_1 - v_2 - \ldots - v_n - v_1$;

2: Select a set of stations F^* for battery-swaps by Algorithm 1 in the uniform cost fixed tour model or by the DP recursion (3) in the non-uniform cost fixed tour model;

Assume the vehicle in the on-site station model starts with an empty battery. Without loss of generality, the distance weight of every edge is not larger than the given battery capacity U. Note that once the permutation over all the vertices in a tour is determined, Algorithm 1 can optimally select stations for battery-swaps in linear time in the uniform cost on-site model. Similarly, the DP recursion (3) can also be applied to the non-uniform cost on-site model and optimally solve the problem in $O(\delta n)$ time as mentioned earlier. Thus, the solution derived by Algorithm 2 is feasible, which represents the total time cost, denoted as SOL. Then, we analyze its approximation ratio. Note that the total time cost includes traveling and battery-swaps; therefore, let $SOL = SOL_{travel} + SOL_{swap}$ represent the respective time required. Let OPT be the minimum time needed to complete the EV tour, as shown by

$$OPT = OPT_{travel} + OPT_{swap}. \tag{4}$$

Consider the traveling time SOL_{travel}, i.e., $w(P)$. Let OPT_{TSP} be the minimum time required for the original TSP. It is trivial that $OPT_{TSP} \le OPT_{travel}$. Because the vehicle is allowed to visit a city multiple times, Phase 1 combines the minimum spanning tree of G with the minimum weight perfect matching on the vertices with odd degrees in the tree to obtain an Euler tour instead of a Hamiltonian tour. Based on a similar analysis in (Christofides, 1976), the $\frac{3}{2}$-approximation ratio can be derived, and the equation follows immediately.

$$SOL_{travel} \le \tfrac{3}{2} OPT_{TSP} \le \tfrac{3}{2} OPT_{travel} \tag{5}$$

Let $b_{\min}$ and $b_{\max}$ represent the smallest and largest battery-swap time respectively, i.e., $b_{\min} = \min_{v \in F}\{b(v)\}$ and $b_{\max} = \max_{v \in F}\{b(v)\}$. The vehicle starts with an empty battery and needs to stop at a minimum of $\lceil \frac{OPT_{travel}}{U} \rceil$ stations. Hence, the relationship between OPT_{travel} and OPT_{swap} can be formulated as follows:

$$b_{min}\left(\left\lceil \frac{OPT_{travel}}{U} \right\rceil\right) \le OPT_{swap} \Rightarrow \; OPT_{travel} \le \frac{U}{b_{\min}} OPT_{swap}. \tag{6}$$

Note that b_{min} can be replaced by a constant c in the uniform cost case.

Given the fixed tour P obtained in Phase 1 of Algorithm 2, let $F^* = \{v_{s_1}, v_{s_2}, \ldots, v_{s_k}\}$ be the set of stations derived in Phase 2, where $|F^*| = k$ is the number of battery-swaps required. For $1 \le i \le k-1$, a path $p_i : v_{s_i} \sim v_{s_{i+1}}$ represents the subpath from v_{s_i} to $v_{s_{i+1}}$ along the tour P. Then, we let $v_{s_{(k+j)}}$ be $v_{s_{((k+j) \bmod k)}}$, e.g., $v_{s_{k+1}} = v_{s_1}$, so that $p_k : v_{s_k} \sim v_{s_1}$ and $p_{k+1} : v_{s_1} \sim v_{s_2}$ are circularly along P, i.e., $p_{k+1} = p_1$. The next lemma follows:

Lemma 5.1. *For both uniform and non-uniform cost models, we have $w(p_i) + w(p_{i+1}) > U$, $1 \le i \le k$.*

Proof. In the uniform cost model, we have $v_{s_{i+1}} = \mathtt{FAR}(v_{s_i})$, $1 \le i \le k-1$, by Algorithm 1. Hence, for $1 \le i \le k-1$, it implies $w(p_i) + w(p_{i+1}) \ge w(p_i - v_{((s_{i+1}+1) \bmod n)}) > U$ because $v_{((s_{i+1}+1) \bmod n)} \preceq \mathtt{FAR}(v_{s_{i+1}})$ trivially. In addition, we use proof by contradiction on the case $i = k$. Suppose $w(p_k) + w(p_{k+1}) = w(p_k) + w(p_1) = w(v_{s_k} \sim v_{s_1}) + w(v_{s_1} \sim v_{s_2}) \le U$; it implies that $v_{s_2} \preceq \mathtt{FAR}(v_{s_k})$. Then, $F_{v_{s_2}} = F^* \setminus \{v_{s_1}\}$ is a feasible solution, which contradicts that F^* is optimal. Thus, the statement holds in the uniform cost model.

In the non-uniform cost model, we have $v_{s_i} \prec v_{s_{((i+1) \bmod k)}} \preceq \mathtt{FAR}(v_{s_i})$, for $1 \le i \le k$, by the DP recursion (3). Similarly, we use proof by contradiction. Assume there exists j, $1 \le j \le k$, such that $w(p_j) + w(p_{j+1}) = w(v_{s_j} \sim v_{s_{((j+1) \bmod k)}}) + w(v_{s_{((j+1) \bmod k)}} \sim v_{s_{((j+2) \bmod k)}}) \le U$, which implies that $v_{s_{((j+2) \bmod k)}} \preceq \mathtt{FAR}(v_{s_j})$. So, $F^* \setminus \{v_{s_{((j+1) \bmod k)}}\}$ is a feasible solution with lower total battery-swap time, which contradicts that F^* is optimal. The proof is complete. □

Then, by summing the equations for every i from Lemma 5.1, we have the following inequality:

$$2SOL_{travel} = 2w(P) = \sum_{i=1}^{k} (w(p_i) + w(p_{i+1})) > kU$$

$$\Rightarrow k < \frac{2SOL_{travel}}{U}$$
$$\Rightarrow SOL_{swap} < \frac{2SOL_{travel}}{U} b_{max}. \tag{7}$$

The last inequality holds because $SOL_{swap} \le kb_{\max}$. Based on the above properties, the next theorem follows.

Theorem 5.2. *The on-site EV touring problem can be approximated within a $\frac{3U+6b_{max}}{2U+2b_{min}}$-ratio for the non-uniform cost case.*

Proof. According to Eqs. (4)–(7), we have:

$$\begin{aligned}
SOL &= SOL_{travel} + SOL_{swap} \\
&< SOL_{travel} + \frac{2SOL_{travel}}{U} b_{max} = SOL_{travel}\left(1 + \frac{2b_{max}}{U}\right) \\
&\le \frac{3}{2} OPT_{travel}\left(1 + \frac{2b_{max}}{U}\right) = \frac{3U + 6b_{max}}{2U} OPT_{travel} \\
&= \frac{3U + 6b_{max}}{2U}\left(\frac{U}{U + b_{min}}\right) OPT_{travel} + \frac{3U + 6b_{max}}{2U}\left(\frac{b_{min}}{U + b_{min}}\right) OPT_{travel} \\
&\le \frac{3U + 6b_{max}}{2U + 2b_{min}} OPT_{travel} + \frac{3U + 6b_{max}}{2U + 2b_{min}} OPT_{swap} \\
&= \frac{3U + 6b_{max}}{2U + 2b_{min}} (OPT_{travel} + OPT_{swap}) = \frac{3U + 6b_{max}}{2U + 2b_{min}} OPT
\end{aligned}$$

□

It is reasonable to assume that the battery capacity U is larger than b_{max} in terms of time expended, which leads to a constant approximation ratio $\frac{9}{2}$. For the uniform cost case, the ratio is within $\frac{3U+6c}{2U+2c}$, for a constant c; and it is $\frac{9}{4}$ based on the assumption that $U \ge c$. In addition, if U is significantly larger than b_{max} and c, the approximation factor is $\frac{3}{2}$, which is the same as that of Christofides algorithm for the TSP, as compared with the ratio $\frac{3c_{\max}}{2c_{\min}}$ derived in (Khuller et al., 2011).

5.2. Off-site station EV touring

Consider the off-site station model in which the distance between every city in V and its nearest battery switch station is at most αU, $0 \le \alpha < \frac{1}{2}$; when $\alpha = 0$, it is the on-site station model. A function *near*: $V \to F$ is defined as the nearest station to each city $v \in V$; thus, $\alpha = max_{v_i \in V}\{w(v_i, near(v_i))\}/U$. To ensure a vehicle's movement, assume the distance between any two vertices in $V \cup F$ is less than $U/2$. The assumption can be considered as route planning in urban areas. In practice, $U/2 \approx$ 100 kilometers is sufficient for the diameter of a metropolitan region.

For the off-site EV touring problem, suppose the vehicle begins with a full battery because the vehicle might start from a vertex without any battery switch station. The rationale behind the approach for this problem is similar to the greedy concept of Lemma 4.1; however, this straightforward approach cannot produce an optimal solution, even for the uniform cost case in the off-site station model. Because the distance between a vertex v and $near(v)$ is not identical, the set of stations F^* selected by the greedy concept is only a feasible solution. Given a fixed tour $P: v_1 - v_2 - \ldots - v_n - v_1$ derived in Phase 1 of Algorithm 2, the vehicle will start from v_1 and stop at the nearest battery switch station of a vertex v_i, i.e., $near(v_i)$, unless it can reach $near(v_{i+1})$ of the vertex v_{i+1} (see Fig. 1). The steps of the approach for solving the off-site EV touring problem are detailed in the GreedyHeuristic procedure. In addition, the following analysis based on F^* is similar to that of the on-site station model.

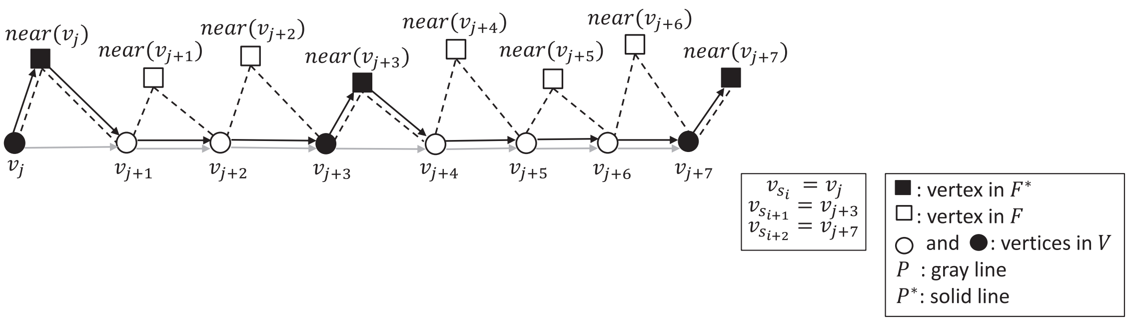

Fig. 1. An illustration of a route between a vertex v_{s_i} and $near(v_{s_i})$ on P^*

Lemma 5.3. Given a fixed tour that traverses each vertex in V for the off-site EV touring problem, the GreedyHeuristic*procedure* can derive a feasible solution of stations F^* that satisfies the greedy concept of *Lemma 4.1*, and incorporate additional routes for battery-swaps in linear time.

```
1: procedure GreedyHeuristic(G, P)
2:     Let v_1 be the start vertex and let F* = ∅;
3:     for i = 1 to n do
4:         if the vehicle cannot reach v_{i+1} and then near(v_{i+1}) with remaining power at v_i, where v_{n+1} = v_1 then
5:             leave v_i for near(v_i), and do a battery-swap there;
6:             F* = F* ∪ {near(v_i)};
7:         end if
8:     end for
9:     return a subset of battery switch stations F* and a tour P* that incorporates extra routes into P for battery-swaps;
10: end procedure
```

Proof. It is obvious that $w(v_1 - v_2 - near(v_2)) \le U$ under the assumption that the distance between any two vertices in $V \cup F$ is less than $\frac{U}{2}$. The initial case holds because the vehicle starts from v_1 with a full battery in the off-site station model. We denote the amount of power left at a vertex v_i as r_i while the vehicle follows the tour P^*. We prove that F^* is a feasible solution by induction on index i, and the hypothesis assumes that $r_i \ge w(v_i, near(v_i))$; that is, the vehicle with r_i units of power left at v_i can at least reach its nearest station $near(v_i)$. Consider the case when the vehicle arrives at v_i, $i \ge 2$. It is trivial that $r_{i+1} \ge w(v_{i+1}, near(v_{i+1}))$ if $r_i > w(v_i - v_{i+1} - near(v_{i+1}))$; otherwise, if $r_i < w(v_i - v_{i+1} - near(v_{i+1}))$, the GreedyHeuristic procedure would select $near(v_i)$ for a battery-swap. The vehicle can reach $near(v_i)$ from v_i by the induction hypothesis, and the amount of power left at v_{i+1} is $r_{i+1} = U - w(near(v_i), v_{i+1}) \ge w(v_{i+1}, near(v_{i+1}))$ under the assumption. Hence, the procedure can guarantee that the vehicle will never run out of battery power on the tour P^*; i.e., F^* is a feasible solution.

In addition, for any $near(v_\ell) \in F*$, we have $r_\ell < w(v_\ell - v_{\ell+1} - near(v_{\ell+1}))$ from the condition in line 4 in the GreedyHeuristic procedure. If the vehicle stops at $near(v_j)$ right before $near(v_\ell)$, it implies that $w(near(v_j) - v_{j+1} \sim v_\ell - v_{\ell+1} - near(v_{\ell+1})) > U$, which satisfies the greedy concept of Lemma 4.1. Moreover, the decision-making at each iteration takes $O(1)$ time and the number of iterations in the for loop is n. Thus, the total running time is linear in the order of G. □

In Algorithm 2, we replace Phase 2 by the GreedyHeuristic procedure. Consider the uniform cost case; that is, $b(v) = c$ for every vertex v, where c is a constant. Note that the GreedyHeuristic procedure not only selects stations, but also devises a tour P^* that incorporates routes between vertices and their nearest stations, if necessary. Similarly, let $SOL = SOL_{travel} + SOL_{swap}$ and $OPT = OPT_{travel} + OPT_{swap}$ by Eq. (4). Given a fixed tour $P : v_1 - v_2 - \ldots - v_n - v_1$ derived in Phase 1 of Algorithm 2, the following also holds for the off-site station model.

$$w(P) \le \frac{3}{2} OPT_{TSP} \le \frac{3}{2} OPT_{travel} \tag{8}$$

Suppose $F^* = \{near(v_{s_1}), near(v_{s_2}), \ldots, near(v_{s_k})\}$ and the tour derived by the GreedyHeuristic procedure can be represented by $P^* : v_1 \sim v_{s_1} - near(v_{s_1}) \sim v_{s_2} - near(v_{s_2}) \sim \ldots \sim v_{s_k} - near(v_{s_k}) \sim v_1$, where k is the number of stations required (see Fig. 2). In addition, let $\triangle_i : v_{s_i} - near(v_{s_i}) - v_{s_i+1}$ represent the subpath from v_{s_i} to v_{s_i+1} along the tour P^* (see Fig. 1). We have $w(near(v_{s_i}), v_{s_i+1}) < w(v_{s_i}, near(v_{s_i})) + w(v_{s_i}, v_{s_i+1})$ because of the triangle inequality. Thus, for $1 \le i \le k$,

$$\begin{aligned} w(\triangle_i) &< 2w(v_{s_i}, near(v_{s_i})) + w(v_{s_i}, v_{s_i+1}) \\ &\le 2\alpha U + w(v_{s_i}, v_{s_i+1}). \end{aligned}$$

Then, sum the equations for every i and the next inequality follows.

$$\begin{aligned} \sum_{i=1}^{k} w(\triangle_i) &= SOL_{travel} - w(P) + \sum_{i=1}^{k} w(v_{s_i}, v_{s_i+1}) < 2k\alpha U + \sum_{i=1}^{k} w(v_{s_i}, v_{s_i+1}) \\ &\Rightarrow SOL_{travel} < w(P) + 2k\alpha U \le \frac{3}{2} OPT_{travel} + 2k\alpha U \end{aligned} \tag{9}$$

Moreover, for $1 \le i \le k-2$, let $p_i : v_{s_i+1} \sim v_{s_{i+1}} - v_{s_{i+1}+1} \sim v_{s_{i+2}}$ be the subpath from v_{s_i+1} to $v_{s_{i+2}}$ on the tour P (see Fig. 1; e.g., the path from v_{j+1} to v_{j+7} of the gray line represents a subpath p_i, for some i); and let $p_0 : v_1 \sim v_{s_1} - v_{s_1+1} \sim v_{s_2}$ and $p_{k-1} : v_{s_{k-1}+1} \sim v_{s_k} - v_{s_k+1} \sim v_1$. We have the following lemma.

Lemma 5.4. *For every* $1 \le i \le k-1$, $w(p_i) + w(v_{s_i}, v_{s_i+1}) > (1 - 2\alpha)U$*; in addition,* $w(p_0) > (1 - \alpha)U$.

Proof. From the proof of Lemma 5.3, we have $w(near(v_{s_i}) - v_{s_i+1} \sim v_{s_{i+1}} - v_{s_{i+1}+1} - near(v_{s_{i+1}+1})) > U$, for $1 \le i \le k-1$. Besides, the vehicle starts at v_1 with a full battery and does the first battery-swap at v_{s_1}, so $w(v_1 \sim v_{s_1} - v_{s_1+1} - near(v_{s_1+1})) > U$ by the GreedyHeuristic procedure.

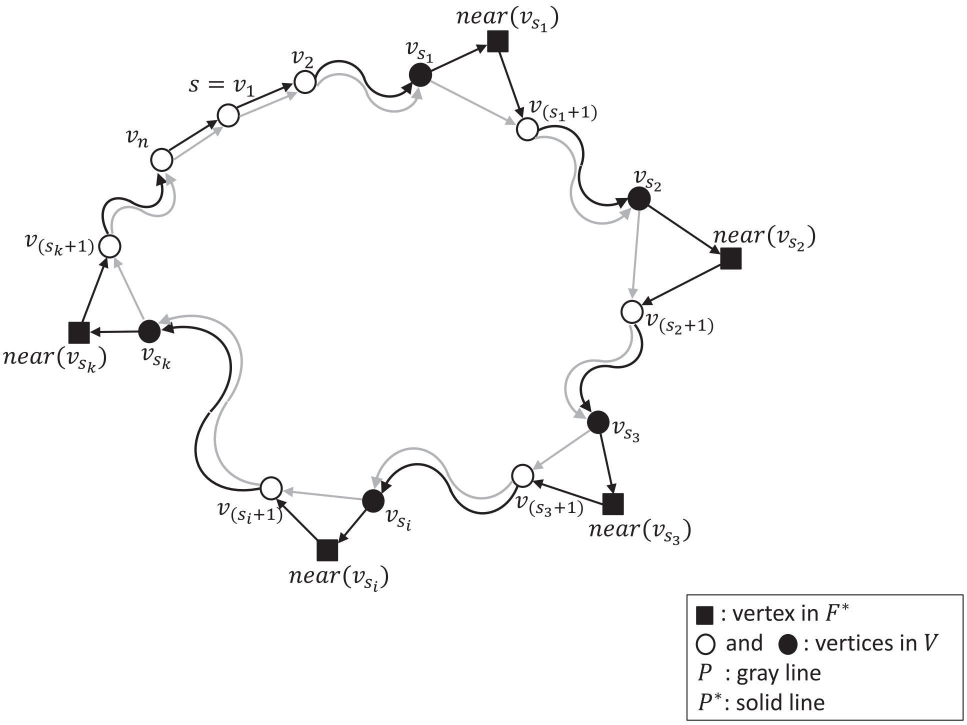

Fig. 2. An illustration of the tour P^*, which incorporates battery-swaps into P

In addition, $near(v_i)$ is the nearest station to a vertex v_i by definition; for $0 \le i \le k-2$, it implies that $w(v_{s_{i+1}+1}, near(v_{s_{i+1}+1})) \le w(v_{s_{i+1}+1}, near(v_{s_{i+2}})) \le w(v_{s_{i+1}+1} \sim v_{s_{i+2}} - near(v_{s_{i+2}}))$ because of the triangle inequality. It follows that $w(p_0) + w(v_{s_2} - near(v_{s_2})) > U$, and $w(near(v_{s_i}), v_{s_i+1}) + w(p_i) + w(v_{s_{i+2}}, near(v_{s_{i+2}})) > U$, for $1 \le i \le k-2$. Obviously, $w(p_0) > (1-\alpha)U$. Moreover, since v_1 follows v_{s_k} circularly ($v_1 = v_{s_k+1}$ possibly), the inequality also holds for the case $i = k-1$ by letting $v_{s_{k+1}} = v_1$. Therefore, for $1 \le i \le k-1$,

$$\begin{aligned}
&w(p_i) + w(near(v_{s_i}), v_{s_i+1}) + w(v_{s_{i+2}}, near(v_{s_{i+2}})) > U \\
&\Rightarrow w(p_i) + (w(v_{s_i}, near(v_{s_i})) + w(v_{s_i}, v_{s_i+1})) + w(v_{s_{i+2}}, near(v_{s_{i+2}})) > U \\
&\Rightarrow w(p_i) + w(v_{s_i}, v_{s_i+1}) > U - w(v_{s_i}, near(v_{s_i})) - w(v_{s_{i+2}}, near(v_{s_{i+2}})) \ge (1-2\alpha)U.
\end{aligned}$$

Thus, the proof is complete. □

Similarly, we sum the equations for every i, $0 \le i \le k-1$, and derive the following formulation (by Eq. (8) and $SOL_{swap} = kc$ in the uniform cost model).

$$2w(P) > w(p_0) + \Sigma_{i=1}^{k-1}(w(v_{s_i}, v_{s_i+1}) + w(p_i)) > k(1-2\alpha)U$$

$$\Rightarrow k < \frac{3OPT_{travel}}{(1-2\alpha)U} \tag{10}$$

$$\Rightarrow SOL_{swap} < \frac{3cOPT_{travel}}{(1-2\alpha)U} \tag{11}$$

The next theorem follows from the above discussion.

Theorem 5.5. *The uniform cost model of the off-site EV touring problem can be approximated within a* $\frac{3(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)}$*-ratio, where* $0 \le \alpha < \frac{1}{2}$.

Proof. According to Eqs. (9) and (11), we have:

$$\begin{aligned}
SOL &= SOL_{travel} + SOL_{swap} \\
&< \frac{3}{2}OPT_{travel} + 2k\alpha U + \frac{3cOPT_{travel}}{(1-2\alpha)U} \\
&< \frac{3}{2}\left(\frac{U + 2\alpha U + 2c}{U - 2\alpha U}\right)OPT_{travel}
\end{aligned}$$

The last inequality holds by Eq. (10). Because the vehicle begins with a full battery in the off-site station model, Eq. (6) is modified as follows:

$$c\left(\left\lceil \frac{OPT_{travel}}{U} \right\rceil - 1\right) \le OPT_{swap}$$
$$\Rightarrow OPT_{travel} \le \frac{U}{c} OPT_{swap} + U.$$

Thus, it implies that

$$\begin{aligned}\frac{3}{2}\left(\frac{U+2\alpha U+2c}{U-2\alpha U}\right)OPT_{travel} &= \frac{3}{2(1-2\alpha)}\left(\frac{U+2\alpha U+2c}{U+c}\right)OPT_{travel} + \frac{3}{2(1-2\alpha)}\left(\frac{U+2\alpha U+2c}{U+c}\right)\left(\frac{c}{U}\right)OPT_{travel}\\ &\le \frac{3(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)}(OPT_{travel}+OPT_{swap}) + \frac{3c(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)}\\ &= \frac{3(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)}OPT + \frac{3c(U+2\alpha U+2c)}{2(1-2\alpha)(U+c)},\end{aligned}$$

where $0 \le \alpha < 1/2$. □

The scenario where U is larger than c leads to the approximation ratio $\frac{3}{4}(\frac{3+2\alpha}{1-2\alpha})$. In addition, if U is significantly larger than c, the approximation ratio is $\frac{3}{2}(\frac{1+2\alpha}{1-2\alpha})$, the same as that of the tour gas station problem (Khuller et al., 2011). When $\alpha = 0$, the ratio is exactly equal to the result of the on-site station model. For the non-uniform cost case, we refer to (Khuller et al., 2011) and replace c with b_{max}/b_{min} in the approximation ratio based on a similar reduction scheme in the uniform cost case. The approximation ratio is $\frac{3}{2}(\frac{3+2\alpha}{1-2\alpha})$ when U is larger than $b_{\max}$.

6. Concluding remarks

In this study, we have considered several EV route planning problems that incorporate 0–1 battery recharging operations. We have presented a simple dynamic programming algorithm that optimally solves the EV shortest travel time path problem in polynomial time. We have further studied the fixed tour EV touring problem and used graph-theoretic techniques to develop optimal algorithms, which extend the prior work of the fixed path gas station problem. We have also investigated the on-site station and off-site station EV touring problem, and proposed approximation algorithms with constant factors and a $\frac{3}{2}(\frac{3+2\alpha}{1-2\alpha})$-factor, respectively, where each city has an off-site battery switch station within a given acceptable distance αU in the off-site station model.

We remark that the latest result in the literature reported by An et al. (2015) improved the approximation ratio of Christofides algorithm for the *s, t-path TSP*. Our approach can be combined directly with *s, t*-Hamiltonian path derived by An et al. (or any better approximation algorithms that may be proposed in the future) to obtain an improved performance ratio for variations of the EV touring problem.

We conclude the paper by suggesting the following future research directions:

1. It is worthwhile to develop a faster polynomial time algorithm for the EV shortest travel time path problem. Compared to the current dynamic programming technique used in this study, a different form of recursion might be necessary.
2. For the non-uniform cost fixed tour model of the EV touring problem, the algorithm proposed in this paper takes $O(\delta n)$ time. Although it could be assumed that δ is sufficiently smaller than n in practice, it would be of theoretical interest to find improvement on the running time.

Acknowledgments

We wish to thank the anonymous reviewers for valuable comments, which helped us improve the quality of the presentation of the paper. We also thank Shao-Chieh Lin and Shu-Hua Wang for conducting partial computational experiments in the appendix. This work was supported by the National Science Council of Taiwan under Grant NSC102-2221-E007-075-MY3, Sayling Wen Cultural and Educational Foundation, the National Science Foundation under Grant CMMI 1265671, and the National Science Foundation of China under Grants 71210002 and 71332005.

Appendix

This appendix presents an example to illustrate the hardness of our proposed problem model in Section 5, as well as some numerical studies of the EV touring problem.

A.1. Uniform cost EV touring

Khuller et al. (2011) showed that the uniform cost on-site station model of the tour gas station problem can be reduced to the original TSP. By contrast, the uniform cost on-site station model of the EV touring problem cannot be transformed into the TSP, as mentioned in Section 5. We provide an example to reveal the difference (see Fig. 3).

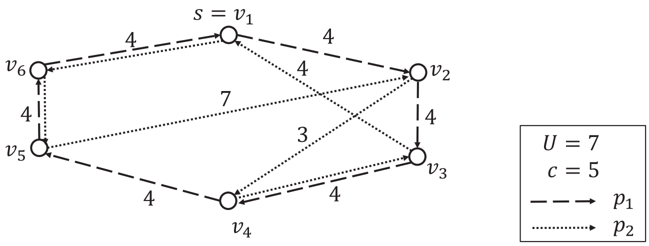

Fig. 3. An example that illustrates the subtleties between gas refueling (battery recharging) and battery-swapping for the EV touring problem in the setting where a vehicle visits every vertex and returns to the origin $s = v_1$ via two different routes: dashed-tour p_1 and dotted-tour p_2.

Suppose the vehicle begins with an empty battery. Each edge is associated with the time required for a vehicle to drive along it. Let the uniform battery-swap time be five and the battery capacity be seven. There are two feasible solutions for the on-site EV touring problem: $p_1 : v_1 - v_2 - v_3 - v_4 - v_5 - v_6 - v_1$ and $p_2 : v_1 - v_6 - v_5 - v_2 - v_4 - v_3 - v_1$. The traveling time for p_1 is 24, and p_1 needs six battery-swaps at v_1, v_2, v_3, v_4, v_5 and v_6. On the other hand, the total time for p_2 with five stops required at v_1, v_6, v_5, v_2 and v_3 is $26 + 5 \times 5 = 51$, compared with the total time cost $24 + 6 \times 5 = 54$ for p_1. However, for the *uniform cost tour gas station problem*, like the previous discussion, the total amount of gas required is the same as the sum of distance weights; thus, it can be reduced to the original TSP. The 0–1 recharging operations, i.e., the battery-swaps, are the main difference between these two optimal refueling problems. In the EV touring problem, the tour p_1 wastes $(7–4) \times 5 = 15$ units of power and the vehicle returns to the origin with three units of power left; however, the tour p_2 only wastes $(7–4) \times 2 = 6$ units of power and the vehicle returns to the origin with three units of power left. Although the vehicle takes the shorter p_1 tour, it wastes more power and requires one more battery-swap than p_2. The total time required for p_1 is longer. Hence, the EV touring problem with 0–1 recharging operations cannot be reduced to the original TSP.

A.2. Computational experiments

In this subsection we have carried out Algorithm 2 to obtain the results of some computational experiments for the on-site EV touring problem (Problem OEVTP). We have also provided a simple branch-and-bound exact algorithm for optimally solving Problem OEVTP and demonstrated the effectiveness of the proposed approximation algorithms. The experimental evaluation was conducted on the machine with Intel Core i5-4690(3.5 GHz) of CPU with 16GB DDR3 of RAM, running Windows 8.1 x64. Algorithm 2 and the branch-and-bound algorithm were compiled by GCC 4.6.1 64-bit, and running time was measured using one single thread. Table 3 shows the results of its uniform cost model.

There have been a considerable amount of studies on exact solution methodologies for many variations of the traveling salesman problem (TSP) (Braekers et al., 2014; Liu et al., 2015; Salazar-González and Santos-Hernández, 2015). The key idea of our branch-and-bound exact algorithm is to incorporate the proposed strategies for the fixed tour EV touring problem in Section 4 when traversing the branch-and-bound tree. More precisely, if no violated inequalities are found at a node of the search tree, a branching step on a decision variable, which indicates if the vehicle swaps a battery, creates two problems according to the greedy concept of Algorithm 1 in the uniform cost fixed tour model or the DP recursion (3) in the non-

Table 3
Computational results of the branch-and-bound exact algorithm and the proposed approximation algorithm for the uniform cost on-site EV touring problem

Test instance	Branch-and-bound			Algorithm 2			Upper
(Graph, U, c)	OPT	Time	Branches	Solution	Time	Ratio	Bound
(Star, 2, 1)	8	0.016	765	8	0.004	1	2
(Star, 2, 0.2)	5.6	0.013	765	5.6	0.005	1	1.636
(Star, 100, 0.2)	5.2	0.020	1545	5.2	0.003	1	1.503
(Diamond, 2, 1)	9	0.030	3498	9	0.005	1	2
(Diamond, 2, 0.2)	6.6	0.041	4002	6.6	0.006	1	1.636
(Diamond, 100, 0.2)	6.2	0.063	11,922	6.2	0.002	1	1.503
(Grid, 2, 1)	14.414	0.658	348,993	14.414	0.012	1	2
(Grid, 2, 0.2)	10.414	0.714	393,105	10.414	0.014	1	1.636
(Grid, 100, 0.2)	9.614	11.443	12,091,901	9.614	0.011	1	1.503
(3-cube, 2, 1)	12	0.483	241,656	13.732	0.008	1.144	2
(3-cube, 2, 0.2)	8.8	0.611	313,080	9.732	0.008	1.106	1.636
(3-cube, 100, 0.2)	8.2	3.034	2,663,448	8.932	0.007	1.089	1.503
(4-cube, 2, 1)	***n/a**	***10000**	***1.043E+11**	29.292	0.031	**†1.221**	2
(4-cube, 2, 0.2)	***n/a**	***10000**	***9.870E+10**	20.492	0.016	**†1.164**	1.636
(4-cube, 100, 0.2)	***n/a**	***10000**	***1.029E+11**	18.492	0.015	**†1.141**	1.503

*: Running within a time limit of 10,000 s; †: letting OPT = 24, 17.6, and 16.2 for the three conditions of 4-cube, respectively.

Table 4
Computational results of the branch-and-bound exact algorithm and the proposed approximation algorithm for the non-uniform cost on-site EV touring problem

Test instance	Branch-and-bound			Algorithm 2			Upper
(Graph, U, $[b_{\min}, b_{\max}]$)$_{\#}$	OPT	Time	Branches	Solution	Time	Ratio	Bound
(Star, 2, [0.5,2])$_1$	7.5	0.018	693	7.5	0.016	1	3.6
(Star, 2, [0.5,2])$_2$	6.5	0.012	677	6.5	0.017	1	3.6
(Star, 2, [0.5,2])$_3$	7.0	0.010	599	7.0	0.019	1	3.6
Average		0.013	657		0.017	1	
(Star, 2, [0.1,0.4])$_1$	5.5	0.014	757	5.5	0.016	1	2
(Star, 2, [0.1,0.4])$_2$	5.4	0.011	751	5.4	0.028	1	2
(Star, 2, [0.1,0.4])$_3$	5.3	0.019	759	5.3	0.017	1	2
Average		0.015	756		0.020	1	
(Star, 100, [0.1,0.4])$_1$	5.1	0.022	1547	5.1	0.003	1	1.511
(Star, 100, [0.1,0.4])$_2$	5.1	0.018	1531	5.1	0.005	1	1.511
(Star, 100, [0.1,0.4])$_3$	5.1	0.030	1577	5.1	0.003	1	1.511
Average		0.023	1552		0.004	1	
(Diamond, 2, [0.5,2])$_1$	8.5	0.031	3148	8.5	0.016	1	3.6
(Diamond, 2, [0.5,2])$_2$	7.5	0.036	3198	7.5	0.015	1	3.6
(Diamond, 2, [0.5,2])$_3$	7.5	0.023	2252	7.5	0.017	1	3.6
Average		0.030	2866		0.016	1	
(Diamond, 2, [0.1,0.4])$_1$	6.5	0.031	3788	6.5	0.016	1	2
(Diamond, 2, [0.1,0.4])$_2$	6.3	0.037	3802	6.3	0.020	1	2
(Diamond, 2, [0.1,0.4])$_3$	6.3	0.028	3394	6.3	0.018	1	2
Average		0.032	3662		0.018	1	
(Diamond, 100, [0.1,0.4])$_1$	6.1	0.059	11,082	6.1	0.002	1	1.511
(Diamond, 100, [0.1,0.4])$_2$	6.1	0.052	10,778	6.1	0.006	1	1.511
(Diamond, 100, [0.1,0.4])$_3$	6.1	0.060	12,478	6.1	0.008	1	1.511
Average		0.057	11,446		0.005	1	

#: Representing the type of random instances.

uniform cost fixed tour model, and the search continues in depth-first order. The traversing scheme can not only derive a good lower bound in an efficient way but also improve the effectiveness of the branch-and-bound algorithm.

We have tested the algorithms on the four types of graphs: stars, diamonds, grids, and hypercubes. The star graph has a center vertex and five outer vertices. The diamond graph is a complete graph with six vertices. The grid graphs is a 3×3 graph with nine vertices. We use two regular hypercube graphs: cube (3-cube) and tesseract (4-cube) with 8 and 16 vertices, respectively. Then, we refer to Table 2 and have conducted the experiments for both the branch-and-bound exact algorithm and Algorithm 2 under the following three conditions: capacity U larger than battery-swap time c ($U = 2,\ c = 1$), smaller battery-swap time c ($U = 2,\ c = 0.2$), and sufficiently large capacity U ($U = 100,\ c = 0.2$). In the following tables, the columns labeled by OPT represent the minimum time cost to complete the EV tour. The columns labeled by Solution mean the total time cost derived by Algorithm 2, and the columns labeled by Ratio show the fraction of the minimum time cost relative to the derived solution by Algorithm 2, i.e., OPT/Solution. The columns labeled by Time show the running time in second to execute the procedures. The columns labeled by Branches represent the number of branches required to perform the branch-and-bound exact algorithm. The columns labeled by Upper Bound show the theoretical worst-case upper bound on the approximation ratios presented in Table 2.

As shown in Table 3, the execution time of Algorithm 2 is significantly smaller than that of the branch-and-bound algorithm, taking only a few milliseconds in some cases. In particular, the computational load of the branch-and-bound algorithm (i.e., the execution time and the number of branches) increases exponentially with the size of the test graphs, especially under the third condition (where $U = 100$ and $c = 0.2$). Moreover, Algorithm 2 derived good approximation ratios, which are even equal to one in several cases.

Next, we consider the non-uniform cost model of Problem OEVTP, where each on-site station may have different battery-swap time. We have conducted the similar experiments. We replaced the uniform battery-swap time with a time interval $[b_{\min}, b_{\max}]$. For each case, we have generated three random instances and tested the procedures for Problem OEVTP on the instances. Tables 4 and 5 show the computational results in the non-uniform cost model. Obviously, Algorithm 2 ran much faster than the branch-and-bound algorithm, and approximated the optimal solutions well. Hence the approximation algorithm can achieve the consistent performance with the uniform cost model. Note that we set a limit of the execution time to 10,000 s for the branch-and-bound algorithm. The boldface numbers in Tables 3 and 5 represent that the exact solution methodology cannot find the optimum under the setting. This result also demonstrates the usefulness of the proposed approximation algorithms. We remark that the theoretical upper bound on the approximation ratios is associated with only two parameters: capacity and battery-swap time, rather than the size of graph instances.

Table 5
Computational results of the branch-and-bound exact algorithm and the proposed approximation algorithm for the non-uniform cost on-site EV touring problem.

Test Instance	Branch-and-bound			Algorithm 2			Upper
(Graph, U, $[b_{\min}, b_{\max}]$)$_{\#}$	OPT	Time	Branches	Solution	Time	Ratio	Bound
(Grid, 2, [0.5,2])$_1$	14.914	0.613	356,759	15.414	0.015	1.034	3.6
(Grid, 2, [0.5,2])$_2$	12.414	0.625	234,691	13.914	0.022	1.121	3.6
(Grid, 2, [0.5,2])$_3$	19.414	0.842	343,785	19.414	0.017	1	3.6
Average		0.693	311,745		0.018	1.052	
(Grid, 2, [0.1,0.4])$_1$	10.514	0.741	398,913	10.614	0.015	1.010	2
(Grid, 2, [0.1,0.4])$_2$	11.414	0.909	395,109	11.414	0.014	1	2
(Grid, 2, [0.1,0.4])$_3$	10.014	0.878	366,163	10.314	0.022	1.030	2
Average		0.843	386,729		0.017	1.010	
(Grid, 100, [0.1,0.4])$_1$	9.514	10.917	11,484,399	9.514	0.016	1	1.511
(Grid, 100, [0.1,0.4])$_2$	9.514	12.630	11,872,977	9.514	0.009	1	1.511
(Grid, 100, [0.1,0.4])$_3$	9.514	9.633	7,839,735	9.514	0.009	1	1.511
Average		11.060	10,399,037		0.011	1	
(3-cube, 2, [0.5,2])$_1$	11.5	0.368	195,460	13.232	0.010	1.151	3.6
(3-cube, 2, [0.5,2])$_2$	11.5	0.311	103,944	14.232	0.023	1.238	3.6
(3-cube, 2, [0.5,2])$_3$	10.0	0.375	146,112	11.732	0.021	1.173	3.6
Average		0.351	148,506		0.018	1.187	
(3-cube, 2, [0.1,0.4])$_1$	8.7	0.541	304,828	9.632	0.008	1.107	2
(3-cube, 2, [0.1,0.4])$_2$	8.4	0.702	295,248	9.332	0.022	1.111	2
(3-cube, 2, [0.1,0.4])$_3$	8.7	0.629	270,096	9.832	0.022	1.130	2
Average		0.624	290,058		0.017	1.116	
(3-cube, 100, [0.1,0.4])$_1$	8.1	2.834	2,538,088	8.832	0.012	1.090	1.511
(3-cube, 100, [0.1,0.4])$_2$	8.1	3.304	2,636,904	8.832	0.010	1.090	1.511
(3-cube, 100, [0.1,0.4])$_3$	8.1	2.602	2,049,240	8.832	0.010	1.090	1.511
Average		2.913	2,408,078		0.011	1.090	
(4-cube, 2, [0.5,2])$_1$	23.5	7465.542	8.100E+10	28.792	0.034	1.225	3.6
(4-cube, 2, [0.5,2])$_2$	23	7206.558	7.705E+10	28.292	0.037	1.230	3.6
(4-cube, 2, [0.5,2])$_3$	20	7681.881	8.517E+10	28.292	0.052	1.415	3.6
Average		7451.327	8.107E+10		0.041	1.290	
(4-cube, 2, [0.1,0.4])$_1$	***n/a**	***10000**	***9.751E+10**	20.392	0.049	**†1.165**	2
(4-cube, 2, [0.1,0.4])$_2$	***n/a**	***10000**	***9.736E+10**	20.292	0.043	**†1.166**	2
(4-cube, 2, [0.1,0.4])$_3$	***n/a**	***10000**	***1.032E+11**	20.292	0.034	**†1.208**	2
Average		***10000**	***9.937E+10**		0.042	**†1.180**	
(4-cube, 100, [0.1,0.4])$_1$	***n/a**	***10000**	***1.037E+11**	18.392	0.015	**†1.142**	1.511
(4-cube, 100, [0.1,0.4])$_2$	***n/a**	***10000**	***1.029E+11**	18.392	0.033	**†1.142**	1.511
(4-cube, 100, [0.1,0.4])$_3$	***n/a**	***10000**	***1.008E+11**	18.392	0.024	**†1.142**	1.511
Average		***10000**	***1.025E+11**		0.024	**†1.142**	

*: Running within a time limit of 10,000 s; †: letting OPT = 17.5, 17.4, and 16.8 for the three samples of the second condition of 4-cube, respectively, and OPT=16.1 for all the three samples of the third condition of 4-cube.

References

Adler, J.D., Mirchandani, P.B., 2014. Online routing and battery reservations for electric vehicles with swappable batteries. Transportation Research Part B 70, 285–302.

An, H.-C., Kleinberg, R., Shmoys, D.B., 2015. Improving christofides' algorithm for the s-t path TSP. Journal of the ACM 62 (5).34:1–28

Artmeier, A., Haselmayr, J., Leucker, M., Sachenbacher, M., 2010. The shortest path problem revisited: Optimal routing for electric vehicles. In: KI 2010: Advances in Artificial Intelligence, LNAI, Vol. 6359, pp. 309–316.

Bansal, N., Blum, A., Chawla, S., Meyerson, A., 2004. Approximation algorithms for deadline-TSP and vehicle routing with time-windows. In: The 36th ACM Symposium on Theory of Computing (STOC'04), pp. 166–174.

Blum, A., Chawla, S., Karger, D.R., Lane, T., Meyerson, A., Minkoff, M., 2007. Approximation algorithms for orienteering and discounted-reward TSP. SIAM Journal on Computing 37 (2), 653–670.

Braekers, K., Caris, A., Janssens, G.K., 2014. Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots. Transportation Research Part B 67, 166–186.

Campbell, A.M., Gendreau, M., Thomas, B.W., 2011. The orienteering problem with stochastic travel and service times. Annals of Operations Research 186 (1), 61–81.

Chang, M.-S., 1998. Efficient algorithms for the domination problems on interval and circular-arc graphs. SIAM Journal on Computing 27 (6), 1671–1694.

Chen, S., Ljubić, I., Raghavan, S., 2010. The regenerator location problem. Networks 55, 205–220.

Christofides, N., 1976. Worst-case analysis of a new heuristic for the traveling salesman problem. Tech. rep.. Graduate School of Industrial Administration, Carnegie-Mellon University.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numerische Mathematik 1 (1), 269–271.

Driscoll, J.R., Gabow, H.N., Shrairman, R., Tarjan, R.E., 1998. Relaxed heaps: an alternative to Fibonacci heaps with applications to parallel computation. Communication of the ACM 31 (11), 1343–1354.

Eisner, J., Funke, S., Storandt, S., 2011. Optimal route planning for electric vehicles in large networks. In: The Twenty-Fifth AAAI conference on Artificial Intelligence, pp. 1108–1113.

Electric Power Research Institute, 2009. The power to reduce CO_2 emissions: the full portfolio. Available from URL: http://portfolio.epri.com/ (accessed November 2012).

Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. Transportation Research Part E 48, 100–114.

Fredman, M.L., Tarjan, R.E., 1987. Fibonacci heaps and their uses in improved network optimization algorithms. Journal of the ACM 34 (3), 596–615.

Hsu, W.-L., Tsai, K.-H., 1991. Linear time algorithms on circular-arc graphs. Information Processing Letters 40 (3), 123–129.
Khuller, S., Malekian, A., Mestre, J., 2011. To fill or not to fill: the gas station problem. ACM Transactions on Algorithms 7 (3).No. 36
Kobayashi, Y., Kiyama, N., Aoshima, H., Kashiyama, M., 2011. A route search method for electric vehicles in consideration of range and locations of charging stations. In: The Intelligent Vehicles Symposium (IV'11) IEEE, pp. 920–925.
Laporte, G., 2009. Fifty years of vehicle routing. Tranportation Science 43 (4), 408–416.
Li, C.-L., Simchi-Levi, D., Desrochers, M., 1992. On the distance constrained vehicle routing problem. Operations Research 40 (4), 790–799.
Lin, S.-H., 2008. Finding optimal refueling policies in transportation networks. In: the 4th Algorithmic Aspects in Information and Management (AAIM'08), LNCS, Vol. 5034, pp. 280–291.
Lin, S.-H., Gertsch, N., Russell, J.R., 2007. A linear-time algorithm for finding optimal vehicle refueling policies. Operations Research Letters 35, 290–296.
Liu, M., Luo, Z., Lim, A., 2015. A branch-and-cut algorithm for a realistic dial-a-ride problem. Transportation Research Part B 81, 267–288.
Pillac, V., Gendreau, M., Guéret, C., Medaglia, A.L., 2013. A review of dynamic vehicle routing problems. European Journal of Operational Research 225 (1), 1–11.
Sachenbacher, M., Leucker, M., Artmeier, A., Haselmayr, J., 2011. Efficient energy-optimal routing for electric vehicles. the Twenty-Fifth AAAI Conference on Artificial Intelligence 1402–1407.
Salazar-González, J.-J., Santos-Hernández, B., 2015. The split-demand one-commodity pickup-and-delivery travelling salesman problem. Transportation Research Part B 75, 58–73.
Schneider, M., Stenger, A., Goeke, D., 2014. The electric vehicle routing problem with time windows and recharging stations. Tranportation Science 48 (4), 500–520.
Siddiqi, U.F., Shiraishi, Y., Sait, S.M., 2011. Multi-constrained route optimization for electric vehicles (EVs) using particle swarm optimization (PSO). In: the 11th International Conference on Intelligent Systems Design and Applications (ISDA'11), pp. 391–396.
Suzuki, Y., 2008. A generic model of motor-carrier fuel optimization. Naval Research Logistics 55, 737–746.
Suzuki, Y., 2009. A decision support system of dynamic vehicle refueling. Decision Support Systems 46, 522–531.
Tesla Motors Inc., 2013. Availble from URL: http://www.teslamotors.com/batteryswap (accessed October).
U. S. DOE Department of Energy, 2012, The alternative fuels and advanced vehicles data center(AFDC). Available from URL: http://www.afdc.energy.gov/afdc/locator/stations/ (accessed November 2012)
U. S. EPA Environmental Protection Agency, 2012. Available from URL: http://www.epa.gov (accessed November 2012).
Williams, R., 2014. Faster all-pairs shortest paths via circuit complexity. In: the 46th ACM Symposium on Theory of Computing (STOC'14), pp. 664–673.