Functional Design of Cluster-Based Resource Distribution for Electric Vehicle Sharing Systems

Junghoon Lee¹, Young-In Cho¹, Misook Oh¹, Yongtae Ko¹, and Sooyoung Kim¹

¹ Dept. of Computer Science and Statistics, Jeju National University, Republic of Korea
jhlee@jejunu.ac.kr

Abstract. This paper presents a cluster-based resource distribution scheme for electric vehicle sharing systems, aiming at overcoming time complexity and enhancing the service ratio. A clustering plan is encoded by an integer-valued vector to apply genetic algorithms. Specific vector elements are dedicated to intermediary stations while negative numbers separate stations. The number of intermediary stations through which vehicles move across cluster domains, is given in priori and genetic operators make better the quality of the solution generation by generation. The fitness function gives precedence to a clustering plan having even load for each cluster. Next, the local relocation scheduler also runs a genetic algorithm to match overflow and underflow stations, trying to minimize the relocation distance, or the distance taken by a service vehicle.

Keywords: electric vehicle, resource relocation, genetic algorithm, station clustering, local schedule

1 Introduction

EV (Electric vehicle) sharing systems can potentially cope with the high cost of EVs, reduce the number of vehicles in cities, and enhance the parking lot availability especially in downtown areas [1]. However, uneven demand patterns, that is, rent-outs and returns, on each sharing station bring the stock imbalance problem, which makes some station unable to serve sharing requests. It is necessary to explicitly move EVs from overflow stations to underflow stations [2]. However, if a sharing system gets larger, for example, to the entire city level, the number of sharing stations also increases. Then, EV relocation between stations becomes an extremely time-complex problem. Clustering is one of the most widely used strategies to decompose a big problem into smaller manageable ones. The relocation is carried out in each cluster, so the number of stations participating in the cluster-level relocation is also significantly cut down.

* This work was supported by the research grant from the Chuongbong Academic Research Fund of Jeju National University in 2013.

ISSN: 2287-1233 ASTL
Copyright © 2014 SERSC
How to divide the whole space into clusters is also a very complex task dependent on many problem-specific factors such as clustering goals, imposed constraints, and the like. There have been many spatial clustering techniques including the well-known k-means method [3]. Hence, we can just employ one of them for station clustering in EV sharing systems. The clustering goals may be the reduction in the number of EVs, the reduction of inter-station distance, and so on. Meantime, it is not possible to find a set of clusters which need no inter-cluster relocation. Hence, clustering comes with an efficient inter-cluster relocation schedules. As this scheduling issue cannot be simultaneously taken into account in the clustering phase, it is necessary to first allocate potential overflow and underflow stations as evenly as possible.

Moreover, to make it possible for EVs to move between different clusters, intermediary stations, which belong to more than one cluster, must be designated. EVs in an overflow cluster are moved to a common intermediary station first and sent to an underflow station in the underflow cluster [4]. Here, it is necessary to match overflow and underflow clusters as well as to determine local schedules within respective clusters. In addition, we assume that intermediary stations have enough parking lot space. Otherwise, it will impose another constraint in creating a set of clusters. Local schedules are required to synchronize with each other. For example, if EVs are not ready in the intermediary station of an underflow cluster, the relocation team must wait for EVs to arrive from the overflow cluster.

Prior to intensively challenging the above-mentioned problem, this paper focuses on a two-cluster relocation system to develop a preliminary design for clustering and relocation scheduling as shown in Fig. 1. Essential requirements for them are listed while how to make them work will be addressed. To begin with, it is assumed that each cluster is managed by a single relocation team. For clustering, the distance of each station is calculated in advance. To apply genetic algorithms, a clustering plan is encoded by an integer-valued vector. Next, our design defines a fitness function capable of measuring inter-station distance within a cluster, degree of biased load between two clusters, and other geographic constraints [5]. The local scheduler also employs a genetic algorithm to generate a relocation schedule consisting of station pairing for surplus EVs and subsequent pair ordering.

![Fig. 1. 2-cluster sharing systems](image-url)
2  Cluster-based Relocation

2.1  Clustering

Fig. 1 also depicts an example of a sharing system consisting of 6 stations numbered from \( S_0 \) to \( S_5 \), and an instance of clustered sets. A station having surplus EVs is an overflow station. In Fig. 1, \( S_0, S_2, \) and \( S_5 \) are overflow stations. Basically, EVs in overflow stations are relocated to underflow stations. If the number of surplus EVs in a cluster is larger than 0, it is an overflow cluster. Cluster 1, having 3 surplus EVs in total, is an overflow cluster. At the cluster level, EVs are required to move from overflow clusters to underflow clusters through intermediary stations, as just the intra-cluster relocation cannot distribute EVs as ordered by a relocation plan. Here, the intermediary station, be it an overflow or underflow station, is counted so as to make each cluster less different. Hence, +1 EV in \( S_1 \) is counted in Cluster 2.

To apply genetic operators such as selection, reproduction, and mutation, it is necessary to represent a clustering plan by an integer-valued vector. Fig. 2 shows the clustering plan for Fig. 1. Here, number \( i \) corresponds to \( S_i \). A predefined location (or locations) is reserved for an intermediary station. In this example, the first location, occupied by \( S_5 \), is such a location, indicating that it is the intermediary station. In addition, negative numbers separate clusters. \((n-1)\) negative numbers are needed for \( n \) clusters. One negative number, namely, -1, is enough for the 2-cluster case. The order in a cluster is meaningless. Hence, \((5, 0, 2, 3, -1, 4, 1)\) and \((5, 0, 3, 2, -1, 4, 1)\) are logically equivalent. Specifically, after reproduction, duplicated elements will be replaced by missing ones [6].

![Fig. 2. Encoding for clustering](image)

Genetic iterations continuously evaluate how well a solution fits for the scheduling goal. Basically, short inter-station distance in each cluster is desirable, as it can reduce the relocation distance. The terrain effect can be counted in inter-station distance. In addition, the information on whether a station is likely to be overflowed or not according to the time axis will be very useful to evenly distribute relocation load for each cluster. Moreover, a relocation team may consist of more than 2 members, making it possible to simultaneously move more than 2 EVs for identical pairs [6].
2.2 Local scheduling

For local scheduling with genetic algorithms, each schedule is represented by $2 \times m$ entries, where $m$ is the number of EVs to move. In an overflow cluster, the intermediary station acts as an underflow station, and vice versa. The first part consisting of $m$ elements denotes an unordered set of relocation pairs, while the second part indicates the relocation order for the pairs. Being aware of whether its cluster is overflowed or not, the local scheduler iterates genetic operators. For an overflow cluster, it is possible for the move to the intermediary station to be placed before the regular relocation pairs, if this allocation does not significantly extend the relocation distance. For the underflow cluster, we can estimate the idle time spent waiting for EVs to arrive at the intermediary station.

3 Concluding remarks

Our future work will definitely extend the relocation scheme to the n-cluster case. At this setting, how to select intermediary station is the first to consider. Practically, based on reachability from other stations, parking lot space, and charging facility availability, the candidate stations are limited, or sometimes fixed. Moreover, EVs in sharing systems can possibly participate in a V2G program and the role of intermediary stations will become more important. On the contrary, an intermediary station can belong to 3 or more clusters, making the problem more complex. On the other extreme, multiple stations are shared by all potential clusters. If not, the set of stations sharing an intermediary station can possibly constitute a group, forming a clustering hierarchy. However, the physical world does not allow complex solutions, so it is necessary to simplify the problem, permitting a certain degree of unbalanced distribution of EVs among clusters.

References