Optimizing the Relocation Operations of Free-Floating Electric Vehicle Sharing Systems*

Zulqarnain Haider† Hadi Charkhgard‡ Sang Won Kim§ Changhyun Kwon¶

June 2020

Abstract

Free-floating electric vehicle sharing (FFEVS) systems require nightly relocation and recharging operations to better meet the next day's spatial demand with sufficient battery levels. Such operations involve not only a crew of drivers to move the shared electric vehicles (EVs), but also a fleet of shuttles to transport those drivers. Comprehensive studies for relocating/recharging EVs and routing shuttles are limited in the literature, and many important operational questions such as an optimal mix of shuttles and drivers remain unanswered. Thus motivated, we first formulate mixed integer programs to model the relocation operations in two different approaches: i) the sequential approach where relocation/recharging decision is made first followed by the shuttle routing decision; and ii) the synchronized approach where all decisions are made simultaneously and synchronously. To solve large-scale problems, we also devise an efficient algorithm, called the exchange-based neighborhood-search method (EBNSM). A case study using real-life data from car2go in Amsterdam shows that the EBNSM-based approaches are capable of solving those large-scale instances within 10 minutes on a generic computer, and the synchronized approach saves the operational costs up to 15% compared to our sequential approach. Comprehensive numerical experiments show that when the service area is large, increasing the number of shuttles is more cost efficient than increasing the number of drivers. On the contrary, when the service area is small, the charging infrastructure is scarce, or the recharging requirements are low, increasing the number of drivers becomes more beneficial.

Keywords: shared mobility; electric vehicles; free-floating car sharing; vehicle routing

*Acknowledgement: This work was supported by the INFORMS Transportation Science and Logistics (TSL) Society Cross-Regional grant.

†Industrial and Management Systems Engineering, University of South Florida, Email: zulqarnain@usf.edu

‡Industrial and Management Systems Engineering, University of South Florida, Email: hcharkhgard@usf.edu

§KAIST College of Business, Korea Advanced Institute of Science and Technology, Email: sw.kim@kaist.ac.kr

¶Industrial and Management Systems Engineering, University of South Florida, Email: chkwon@usf.edu

1 Introduction

An important goal in the development of smart cities is improving efficiency and flexibility in resource utilization. In transportation sector, with the rapid advancements of mobile technologies and devices, on-demand vehicle sharing platforms have emerged as a viable alternative to the notion of car ownership, potentially leading to an efficient utilization of vehicles and urban land. Currently, several key players, such as car2go and DriveNow, are gaining traction with their on-demand free-floating car sharing services, leading fast expansion of the industry. As of 2020, Share Now, the world's largest carsharing company based in Germany which was formed from the merger between car2go and DriveNow, operates in 18 cities across Europe with over four million registered members and a total fleet of over 14,000 vehicles.

Unlike station-based car sharing models like ZipCar, free-floating car sharing systems let users pick an available car and drop it off at any legally permissible parking location within a large designated service area. As a result, users can make customized one-way trips in conjunction with other mobility options, directly addressing the first-mile last-mile problem in urban mobility.[1] For this reason, free-floating car sharing systems have the potential to be a promising alternative to private car ownership, with much higher adoption than traditional car sharing (Formentin et al., 2015). In North America, for example, it is estimated that a single car sharing vehicle can potentially reduce the need for 6 to 23 cars, substantially reducing the total number of vehicles held by households (Shaheen and Cohen, 2007; Martin et al., 2010). Moreover, free-floating car sharing systems can also facilitate more efficient use of urban infrastructure and land by reducing the need for perennially occupied or "locked" parking spots in the city centers with 36–84 m^2 of public spaces freed-up per vehicle (Loose, 2010).

Car sharing systems with electric vehicle (EV) fleets are also projected to play a crucial role in making urban transportation systems more sustainable (Firnkorn and Müller, 2011; Le Vine and Polak, 2019). Although EVs are often proposed as one of the most promising solutions to curbing green house gas emissions from the transportation sector, their mass adoption has yet to come—in 2017, EVs make up only 1.15% of total U.S. car sales (Bellan, 2018). In particular, their short driving range and high fixed costs create psychological concerns to the potential drivers, known as range anxiety and resale anxiety, imposing barriers to the adoption (Lim et al., 2015). By relieving the burden of car ownership, free-floating EV sharing (FFEVS) systems can effectively mitigate these psychological barriers to EV adoption: With EVs owned by the service provider rather than the individuals, FFEVS systems relieve the drivers' concerns about technological risks, future resale value, as well as maintenance (He et al., 2017). In fact, car2go has already deployed full EV fleets in four major cities in Europe with more than 2,000 EVs, and DriveNow operates mixed fleets of EVs and combustion engine cars in over ten European cities.

Despite these potential social benefits, the free-floating nature of the FFEVS systems also engender operational challenges. In FFEVS systems, customers' rental activities may place the vehicles at less fa-

[1] It refers to the problem of unavailability of public transit for the first and last mile of commute, forcing commuters to opt for private modes of transit over public ones.

vorable locations and cause a spatial and temporal mismatch between supply and demand, and users are only willing to walk up to 500 meters to the available vehicle (Weikl and Bogenberger, 2015). Indeed, the distance to an available vehicle is shown to be an important determining factor in a booking decision for car2go users (Herrmann et al., 2014). Hence, for successful operations of FFEVS systems, it is essential to have the fleets available in the right place at the right time, which raises two key operational challenges to the service providers: i) accurate prediction of the the spatial and temporal demands; and ii) timely and efficient relocation of the fleets to the predicted demand locations.

To deal with the first challenge, FFEVS service providers have been putting a great deal of efforts into improving demand prediction accuracy. For instance, in their white paper, car2go explains that "With the data which car2go has collected over the years, the company is able to predict demand extremely accurately using complex, proprietary algorithms" (car2go, 2017). The second challenge, however, still demands careful academic attention. To address that, one would need a comprehensive modeling and computational framework applicable to the optimization problem of relocation/recharging operations of EVs in FFEVS systems, but studies on relocating EVs are limited in the literature. This is the key motivation as well as the main focus of this paper: We develop an optimization framework for timely and efficient EV relocation/recharge operations in FFEVS systems. In particular, we aim to develop algorithms that are capable of solving real size problems.

In practice, the relocation operation in FFEVS systems are typically carried out with shuttles to transport the drivers, and therefore the relocation decisions should be made at two levels: i) EV relocation decision; and ii) Service shuttle routing decision.[2] Given the input of predicted demand level, current EV locations, and their fuel levels, the EV relocation decision determines which EVs should be moved to which target locations and how much and where they should be recharged. The shuttle routing decision is to determine how the shuttles should pick-up and drop-off the drivers to fulfill the EV relocation plan. Ideally, it would be desirable to develop a framework where the two decisions are made jointly and synchronously. Most of the existing studies in this context approach the two decision problems separately.

In this paper, we focus on the relocation, recharging, and routing decisions to serve the given demand for the next day in the static environment during the night. Accordingly, we develop a comprehensive modeling and computational framework applicable to the joint optimization of nightly relocation/recharging operations and shuttle routing. Then, based on the computational framework, we conduct a case study using actual data to address important operational questions regarding the relocation operations of FFEVS systems. Since the two levels of decisions for EV relocation and shuttle routing have been studied separately in the literature, it is natural to first construct a *sequential* approach that solves the EV relocation problem first and then solves the shuttle routing problem with the fixed EV relocation decision. This paper devises formu-

[2] In practice, different FFEVS systems use service shuttles of different size and type to carry out the relocation operation. Therefore, we use service shuttle or "shuttle" as an all encompassing term, which could describe a van with a large capacity, or a car with relatively limited capacity, or even single person mobility options like scooters and foldable bicycles that can be loaded into an EVs trunk (Weikl and Bogenberger, 2015). Our flexible modeling approach works for shuttles of various types and capacities by varying the maximum number of drivers allowed on board each shuttle.

lations and algorithms for faster computation that are suitable for handling the nature and the complexity of the two levels of decisions in the sequential approach. Furthermore, as a main goal, we also develop a joint and simultaneous decision making method, which we call the *synchronized* approach. Such a truly joint decision has rarely been studied in the literature. Our unique sequential approach and algorithm is shown to be capable of providing solutions to real size problems, and our synchronized approach reveals much enhanced performance both in terms of solution quality and computation time relative to our sequential approach.

We believe that the decision tool presented in this paper would be of great academic and practical significance because of the following reasons. First, vehicle relocation operations can be very costly and typically, they are a bottleneck that limits service efficiency as well as asset utilization. The relocation must be planned well in advance and be carried out efficiently as delays in relocation operations often results in lost demand. Unlike other shared mobility services such as bikes or scooters, the vehicle relocation operation in FFEVS systems cannot be done by moving multiple vehicles at a time, making it complex and costly—the cost of vehicle relocation in a free-floating car sharing system can be as much as 15 euros per movement (Chianese et al., 2017).

Second, the vehicle relocation problem in FFEVS systems is computationally very hard as it frequently involves charging the EVs during the relocation. Studies show users exhibit range anxiety even for short rental decisions in FFEVS systems (Weikl and Bogenberger, 2015), and there is a strong evidence that demand for FFEVS service is sensitive to the vehicles' battery levels (Kim et al., 2020). This additional layer of recharging the vehicles to a sufficient level, which requires available charging stations nearby and at least a few hours of charging time, drastically increases the complexity of the relocation problem. Indeed, vehicle unavailability and range anxiety were cited as main reasons when car2go had to cease its EV operations in San Diego by replacing the EV fleets with combustion engine vehicles of the same model (Garrick, 2016).

Third, there are many important yet unanswered operational and managerial questions for FFEVS systems. For example, how would operational resource allocation decisions impact the efficiency of EV relocation operations in FFEVS systems? More specifically, with a given number of available drivers, which would be a better operational strategy: running more shuttles or having more drivers available for relocation? Although FFEVS systems have already been operated in many cities for years, comprehensive studies for the EV relocation problem considering the actual relocation operations are limited in the literature. In this paper, we apply our computational framework to conduct a case study and address these operational questions using actual data.

Overall, our key findings and contributions are summarized as follows.

- We present a combined MIP formulation for the problem of EV relocation for given demand and charging level considerations and the subsequent shuttle routing problem to carry out the recommended relocation. The MIP formulation can be directly applied to small-scale problems and solved by off-the-shelf optimization solvers such as CPLEX and Gurobi. To present practical optimization methods for large-scale problems, this paper makes the following important methodological contributions.

1. To solve the standalone EV relocation problem, we present a path-based formulation that models each EV route as a separate variable and improves the computability of the EV relocation problem significantly, whereas the existing link-based formulations for EV relocation problem have suffered from increasing complexity as the problem size increases.
2. For large-scale problems (as in our case study), the exact solution method cannot even provide feasible solutions. We develop an efficient heuristic algorithm for large-scale problems that can obtain a quality synchronized solution in a few minutes. To derive synchronized decisions, our heuristic algorithm integrates the path-based EV relocation problem with the shuttle routing problem, which is a unique variant of dial-a-ride-problem (DARP) (Psaraftis, 1983) involving charging stops for EVs. The heuristic algorithm, which we call the exchange-based neighborhood-search method (EBNSM), draws upon clustering, exchange based neighborhood search, and a customized exchange algorithm for multiple precedence-constrained DARP.

- This work is the first in the literature that considers a synchronized EV routing and shuttle routing problem with general cases of charging infrastructure availability. Our extensive numerical study using real FFEVS data shows that the synchronized decision making improves the total shuttle route length up to 15% relative to the sequential approach. This is when both the approaches use our efficient path-based formulation. Therefore, it is expected that the performance gap could be higher if compared with a sequential approach using other computationally less efficient formulation such as link-based ones. When accumulated over time, this improvement can be a significant benefit for the operation.

- We conduct a case study using real data from the car2go service in Amsterdam, the Netherlands. Our findings from extensive full-scale numerical experiments, summarized below, inform several important operational decisions.

1. Our results suggest that, in most cases, increasing the number of shuttles is more cost effective than increasing the number of drivers on each shuttle. In particular, it is especially the case when the service area is large or when the charging requirements are low (e.g., the initial battery levels are high). In many cases, we find that pairing a separate shuttle with each driver and have the shuttle only to support that driver's movement, is optimal.
2. Our results also show that reducing the number of shuttles, and therefore, increasing the number of drivers per each shuttle can be beneficial when the service area is small, and the charging requirement is high. In particular, when the charging requirement is high (e.g., the initial battery levels are low), adding more shuttles may not be cost effective as it can only increase each shuttle's wait time to pick up the drivers.

2 Literature Review

In this section, we focus mainly on reviewing the literature on operational aspects of the relocation problem in vehicle-sharing systems to highlight our contributions. We refer the reader to Laporte et al. (2015) for a more comprehensive review of other relevant operational problems.

The problem of relocation has been well recognized in one-way car sharing and bike sharing systems. The current vehicle relocation strategies fall into two broad categories. The user-based relocation strategies include incentives, pricing mechanisms and policy interventions to influence user behavior (Weikl and Bogenberger, 2013). In one-way car sharing systems, repositioning can be carried out either through operator intervention, e.g., using relocation personnel (Kek et al., 2009; Jian et al., 2016; Bruglieri et al., 2014) and using a trip choice mechanism (de Almeida Correia and Antunes, 2012) or through customers by controlling their actions, e.g., through incentives (Pfrommer et al., 2014). Similarly, in case of bike sharing systems, the problem of system-wide imbalance is further compounded by the two-sided demand for bicycles to rent and empty racks to return. The focus of repositioning is to achieve certain desirable inventory levels either through manual rebalancing using trucks (Raviv et al., 2013) or through incentive mechanisms designed to influence customer behavior (Fricker and Gast, 2016; Haider et al., 2018).

In an initial conceptual paper, Weikl and Bogenberger (2013) present and evaluate several user-based and operator-based relocation strategies for free-floating car sharing systems. In a subsequent paper, Weikl and Bogenberger (2015) propose a practice ready six step relocation model for a mixed free-floating car sharing system with traditional and electric vehicles. Based on historical data, the area is categorized into macro zones and an optimization model is used to achieve desired macro level relocation. Rule based methods are used for making intra zone micro-level relocation and refueling/recharging decisions. A similar model for demand-based relocation in free-floating car sharing systems is presented by Schulte and Voß (2015) and Herrmann et al. (2014). Caggiani et al. (2017) propose dynamic clustering method to identify the size and number of flexible zones in which to perform repositioning operations. He et al. (2020) study robust repositioning strategies in dynamic environments.

Only a few papers, however, have considered a unique and critical component of EV relocation operations: shuttle routing. Gambella et al. (2018) present a time-space-network-based formulation for relocating the vehicles in car sharing systems given some demand and battery considerations, but they only consider *station-based* car sharing systems. In their work, the relocation is carried out by the so called relocators (drivers) who are on board the vehicle itself when traveling between a pair of stations. Kypriadis et al. (2018) propose a minimum-walking car repositioning problem for FFEVS systems. In their model, the drivers *walk* between the relocation assignments as opposed to traveling on board a shuttle. The problem of shuttle routing to carry out the recommended relocation for free-floating car sharing systems is also underserved. Maintaining a dedicated fleet of shuttles and drivers can be expensive and minimizing the cost of relocation operation is one of the key system objectives. For an free-floating car sharing system, Santos et al. (2017) consider the problem of shuttle routing given a fleet of shuttles and drivers and provided a set of pre-determined relocation assignments; hence their approach is sequential, rather than joint or synchronized.

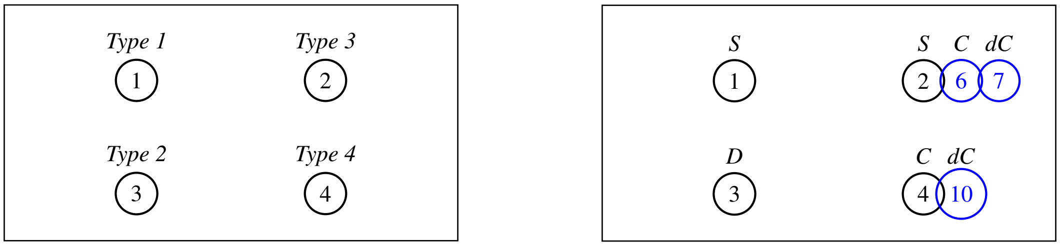
Closely related to our work, Folkestad et al. (2020) consider joint decision making for EV relocation and shuttle routing. The modeling approach, formulation, and algorithms in our paper, however, are distinct. In Folkestad et al. (2020), EVs are moved to charging stations rather than close to actual demand points, assuming a situation with ubiquitous charging infrastructure whereby a charging station can be blocked indefinitely. In case of non-availability of charging stations, postponement of charging is considered. This makes their relocation model similar to the one for station-based systems rather than free-floating systems, especially when charging infrastructure is scarce. Many cities and free-floating car sharing systems can have saturated charging infrastructure and station blocking may not be an option. Considering access to limited charging infrastructure in recharging planning is a key factor for success of FFEVS systems (He et al., 2019). Our work models the charging process more explicitly and flexibly. Specifically, our model can handle partial recharging, and therefore the operator can achieve different charging levels in different service zones, to better respond to battery level sensitive customers. The operator can also choose to partially charge all fleet vehicles up to desired charging levels as opposed to charging only a subset of vehicles fully while postponing the charging operation for others. We also note that Folkestad et al. (2020) only relocate EVs with battery levels below a certain threshold whereas our model is flexible to different charging / relocation requirements: Our model can handle purely demand based relocation (i.e., a fully charged EV may need to be relocated to fulfill demand requirements) or a purely recharging based relocation (i.e., a low charged EV needs charging but must stay in its current neighborhood). Also, whereas a hybrid genetic algorithm is proposed in Folkestad et al. (2020), we employ the exchange-based neighborhood search method to address computational challenges in joint decision making. Lastly, our case study provides important managerial insights for running the relocation operation more efficiently.

3 The Model

In this section, we present a MIP formulation for our problem. Our MIP formulation is divided into three distinct, but related, sub-problems: namely, the EV relocation and recharging problem, the shuttle routing problem, and the synchronization problem. These sub-problems are put together into a single model in Section 3.5.

3.1 Modeling Demand and Charging Satisfaction for Each Neighborhood

The relationship between location and demand is incorporated into our model to ensure the optimal placement of the EV fleet across the study area. To incorporate the demand information into our relocation decision, we divide the service area into small neighborhoods $h \in \mathcal{H}$. The size of the neighborhoods is small enough—less than $250\,000\,\mathrm{m}^2$—so that assumptions of demand uniformity and similarity of demand



(a) Original Network **(b) Extended Network**

Figure 1 – Conversion of an original network to an extended network with dummy nodes shown in blue. In the extended network, nodes are shown with their respective designations as supplier, demander and charger nodes

characteristics throughout the neighborhood hold reasonably well. We find the neighborhood level status data at 12 am to find the initial inventory levels, I_h^0. The average values of historical demand data from 6 am to 9 am the subsequent day are calculated for the neighborhood specific desired inventory levels, I_h^d.

Besides the requirements for demand fulfillment, we also consider the relationship between charging levels and the location of a vehicle. We posit that neighborhoods located in different areas of a town may have different charging level requirements for the vehicles located therein. For example, users in downtown could be more sensitive to the vehicles battery levels than those in the suburbs possibly because of the uncertainty in road traffic conditions. The system manager can thus have neighborhood specific charging requirements. The initial battery level of EVs, denoted by c^0, can be found using system status data. We use historical trip data and average the battery levels at the beginning of all trips originating in a neighborhood h to find the desired charging level c^f for all nodes located in the neighborhood. The system manager can, however, update the desired charging level for any neighborhood or decide to charge all vehicles fully. Later, these charging levels are used in determining crucial parameters for our path based electric vehicle relocation and recharging problem.

Let us consider a set of permissible parking spots in the service area. We associate two boolean characteristics with every node in the original network: the occupancy of the node (occupied or unoccupied) and the availability of charging infrastructure at the node (Yes or No). An occupied node without charging infrastructure, called *Type 1* node, is designated as a *supplier* node and represents current EV locations. An unoccupied node without charging infrastructure, called *Type 2* node, is designated as a *demander* node. An unoccupied node with charging infrastructure, called *Type 3* node, is designated as a *charger* node. In the special case where an occupied node may also be a charging node, called *Type 4* node, we create two nodes at the same location; a supplier node for the occupancy and a charger node for the charging station. A demand relocation happens when an EV moves from one of the occupied spots, a *supplier* to one of the empty spots, or *demander*. In case the EV needs to be charged, the EV must first visit a charging node. Since the charging process takes time, we create a dummy node for each charging node called a *dummy charger* node. The EV movement between a charger and its sister dummy charger represents the charging process. In Figure 1, we show the process of conversion to an extended network.

Algorithm 1: Procedure for removing nodes from the network

Input: $\widehat{\mathcal{S}}$, $\widehat{\mathcal{C}}$, $\widehat{\mathcal{C}}^d$, $\widehat{\mathcal{D}}$, I_h^0, I_h^d,
Output: $\mathcal{S}, \mathcal{C}, \mathcal{C}^d, \mathcal{D}$

for $h \in \mathcal{H}$ **do**
  $\texttt{removeCount}_h = 0$;
  for $i \in \widehat{\mathcal{S}}_h$ **do**
    if $c_i^0 > c_h^f$ **then**
      $\texttt{removeCount}_h = \texttt{removeCount}_h + 1$;
  if $I_h^0 \leq I_h^d$ **then**
    $\mathcal{S}_h^r = \{i \in \widehat{\mathcal{S}}_h \mid c_i^0 > c_h^f\}$;
    for $k \leftarrow 1$ **to** $\texttt{removeCount}_h$ **do**
      $\widehat{\mathcal{S}}_h \leftarrow \widehat{\mathcal{S}}_h \setminus \mathcal{S}_h^r[k]$, $\widehat{\mathcal{C}}_h \leftarrow \widehat{\mathcal{C}}_h \setminus \widehat{\mathcal{C}}_h[1]$, $\widehat{\mathcal{D}}_h \leftarrow \widehat{\mathcal{D}}_h \setminus \widehat{\mathcal{D}}_h[1]$;
  if $I_h^0 > I_h^d$ **then**
    $\mathcal{S}_h^r = \{i \in \widehat{\mathcal{S}}_h : c_i^0 > c_h^f\}$;
    for $k \leftarrow 1$ **to** $\min\{\texttt{removeCount}_h, I_h^d\}$ **do**
      $\widehat{\mathcal{S}}_h \leftarrow \widehat{\mathcal{S}}_h \setminus \mathcal{S}_h^r[k]$, $\widehat{\mathcal{C}}_h \leftarrow \widehat{\mathcal{C}}_h \setminus \widehat{\mathcal{C}}_h[1]$, $\widehat{\mathcal{D}}_h \leftarrow \widehat{\mathcal{D}}_h \setminus \widehat{\mathcal{D}}_h[1]$;
$\mathcal{S} \leftarrow \bigcup_{h \in \mathcal{H}} \widehat{\mathcal{S}}_h,\ \mathcal{D} \leftarrow \bigcup_{h \in \mathcal{H}} \widehat{\mathcal{D}}_h$;
$\mathcal{C} \leftarrow \bigcup_{h \in \mathcal{H}} \widehat{\mathcal{C}}_h$;

In the extended network, the nodes are designated as *supplier*, *demander*, *charger* or *dummy charger* based on their functions. Let us call the initial sets of these nodes $\widehat{\mathcal{S}}_h$, $\widehat{\mathcal{D}}_h$, $\widehat{\mathcal{C}}_h$ and $\widehat{\mathcal{C}}_h^d$ for each neighborhood h and $\widehat{\mathcal{S}}$, $\widehat{\mathcal{D}}$, $\widehat{\mathcal{C}}$ and $\widehat{\mathcal{C}}^d$ for the whole network, respectively. The supplier nodes correspond to the initial location of electric vehicles in each neighborhood, i.e., $|\widehat{\mathcal{S}}| = \sum_{h \in \mathcal{H}} I_h^0$. Since street level parking spots are very close and virtually indistinguishable, we use the central measure of all the spots in a neighborhood and create as many demander nodes in a neighborhood as the size of desired inventory in each neighborhood, i.e., $|\widehat{\mathcal{D}}| = \sum_{h \in \mathcal{H}} I_h^d$. It is worth noting that we assume no new demand arrives during the night as we model the nighty static relocations. This ensures that the current EV locations and the desired inventory in each neighborhood stay unchanged throughout the relocation operation. Indeed, many FFEVS systems in practice close their operations at night, and even when one stays operational, the demand levels are typically insignificant relative to day time demands. For instance, in the case study we consider, the night time demand is only 4% of the peak demand. Finally, each neighborhood can also have dozens of closely located charging stations. If the number of charging stations is greater than $\max\left(|\widehat{\mathcal{S}}_h|, |\widehat{\mathcal{D}}_h|\right)$, we use central measure of all charging stations to create as many charging nodes as the larger of the number of demanders or the number of suppliers in a neighborhood, i.e., $|\widehat{\mathcal{C}}_h| = \max\left(|\widehat{\mathcal{S}}_h|, |\widehat{\mathcal{D}}_h|\right)$. In case the number of these charging stations is less than $\max\left(|\widehat{\mathcal{S}}_h|, |\widehat{\mathcal{D}}_h|\right)$, number of charging nodes created is equal to the number of charging stations for each neighborhood.

(a) A traditional Relocation Decision **(b) An EV Relocation and Recharging Decision**

Figure 2 – In a traditional demand-based relocation problem, a single relocation operation involves movement between two nodes. In a demand and recharging based relocation model, a single relocation operation can involve a charging stopover at a pair of intermediate charger-dummy charger nodes

We can further reduce the size of this network by deleting certain nodes depending on neighborhood-level desired inventory and desired charging level using Algorithm 1. In all neighborhoods, only suppliers with initial battery level greater than the desired charging level can be removed, since the rest requires charging. For neighborhoods where $I_h^0 \leq I_h^d$, suppliers that do not require charging can be removed. An equal number of chargers and demanders can also be removed, while the rest are kept for suppliers which require charging and suppliers incoming from other neighborhoods to satisfy the leftover demand. On the other hand, for neighborhoods where $I_h^0 > I_h^d$, $I_h^0 - I_h^d$ suppliers are retained to satisfy demand in other neighborhoods. Out of I_h^d suppliers left, those that require charging are retained, while the rest are removed. An equal number of chargers and demanders are also removed while the rest are kept for suppliers that require charging.

3.2 Modeling the EV Relocation and Recharging (EVRR) Problem

In second part of our modeling approach, we describe the relocation and recharging decision for electric vehicles (EVs). We call this problem the EV relocation and recharging (EVRR) problem. The traditional relocation problem in car sharing and bike sharing systems involves the decision to relocate vehicles from current node i to a future node j to cater to shifting demand needs as shown in Figure 2-a). In case of electric cars, node j can be a charging station. The relocation scenario we consider is decidedly different than traditional setting. In our model, a vehicle can visit up to four nodes during its journey as shown in Figure 2-b). From its current node i, a supplier node, it can go to a pair of charger-dummy charger nodes k and l, one each for receiving and dispatching the vehicle, and once recharged to the required level, it can move to a demander node j. Dummy charger nodes are introduced to allow for two shuttle visits to the charging station since the driver does not wait while the vehicle is being charged. Finally, a vehicle may also choose to stay put at its current node if it is sufficiently charged and does not have to fulfill a demand-based relocation. Our model only specifies the neighborhood a vehicle needs to be relocated to and the required charging level. Our model, therefore, chooses an optimal route for an EV, not only deciding the terminal node j, if any, of its journey but also deciding which pair of charger nodes, if any, it should visit for recharging purposes.

Let $\mathcal{N}$ be the set of permissible parking spots in the reduced network, i.e., $\mathcal{N} = \mathcal{S} \cup \mathcal{D} \cup \mathcal{C}$. Let $\mathcal{N}'$ be the network with addition of dummy nodes, $\mathcal{C}^d$. The notation for EVRR problem is given in Table 1. Given

Sets	
$\mathcal{N}$	Set of permissible parking spots
$\mathcal{N}'$	Set of permissible parking spots plus the dummy nodes
$\mathcal{C}$	Set of charger nodes
$\mathcal{D}$	Set of demander nodes
$\mathcal{S}$	Set of supplier nodes
$\mathcal{C}^d$	Set of dummy charger nodes
$\mathcal{H}$	Set of demand neighborhoods indexed by $h \in \mathcal{H}$
$\mathcal{P}$	Set of all feasible paths indexed by $p \in \mathcal{P}$ where each path $p = (i_0, i_1, \ldots, i_s)$
$\mathcal{A}$	Set of all possible arcs indexed by $a \in \mathcal{A}$ where each arc $a = (i, j)$
$\mathcal{A}(p)$	Set of arcs in path p, where $\mathcal{A}(p) = \mathcal{A}^c(p) \cup \mathcal{A}^t(p)$, i.e., charging and travelling arcs on path p
$\mathcal{N}(p)$	Set of nodes in EV path p, where $\mathcal{N}(p) = \{\mathcal{S}(p), \mathcal{C}(p), \mathcal{C}^d(p), \mathcal{D}(p)\}$, i.e., supplier, charger, dummy charger and demander nodes of a particular path p
$\phi(i)$	Set of paths p that contain node i, i.e., $\{p \in \mathcal{P} : i \in p\}$
Variables	
x_p	Binary variable; 1 when a vehicle is to be relocated along a path p, 0 otherwise
Parameters	
t_{ij}	travel time alongside a travel arc (i, j)
w_p	charging time between charger and dummy charger nodes for path p
c_p^0	initial battery level at supplier node for path p
c_p^f	desired charging level at demander node for path p
β_1	rate of depletion; the decrease in battery level of a vehicle per unit of travel time
β_2	rate of charging; the increase in battery level of a vehicle per unit of time

Table 1 – Mathematical notation for EVRR

$\mathcal{S}$, $\mathcal{C}$, $\mathcal{C}^d$, and $\mathcal{D}$, We can enumerate all the possible EV paths and associate a binary decision variable x_p with each path $p \in \widehat{\mathcal{P}}$. The path variable is 1 if an electric vehicle moves on path p and 0, otherwise. An EV may or may not require charging. In the former case, the number of possible paths is equal to $|\mathcal{S}| \times |\mathcal{D}|$. In case of charging, the number of possible EV paths is $|\mathcal{S}| \times |\mathcal{C}| \times |\mathcal{D}|$. The total number of paths $|\widehat{\mathcal{P}}|$ is equal to $|\mathcal{S}| \times |\mathcal{D}| (1 + |\mathcal{C}|)$.

We reduce the size of $\widehat{\mathcal{P}}$ by removing infeasible or unnecessary paths. Associated with each path p are parameters c_p^0, c_p^f, and w_p representing initial battery level at the supplier, desired charging level at demander, and the charging time required to achieve c_p^f, respectively. Since we have already enumerated all the paths, then for a path $i \to k \to l \to j$, knowing the initial battery level at the supplier, the final charging level at the demander and the discharging on arcs (i,k) and (l,j), one can determine path dependent charging time w_p on charging arc (k,l) as follows:

$$w_p = \frac{1}{\beta_2}\left(c_p^f - c_p^0 + \beta_1 \sum_{(i,j)\in\mathcal{A}^t(p)} t_{ij}\right) \qquad \forall p \in \widehat{\mathcal{P}}. \tag{1}$$

In essence, w_p represents the service time at a charging station. Our model for charging process assumes that charging time has a linear relationship with charging levels. Others have considered non-linear charging process and used piece-wise linearization to model different charging speeds (Pelletier et al., 2018). However, a linear charging process sufficiently models the reailty for our case when the focus is on operational problem and the total time spent on recharging rather than the non linearities of the charging process itself. If $w_p \leq 0$ for a path, charging stop is not needed and hence the corresponding path $p : i \to k \to l \to j$ is removed and only direct path $p : i \to j$ is kept. Conversely, if $w_p > 0$, direct path $p : i \to j$ is removed since a charging station must be visited. The path set $\widehat{\mathcal{P}}$ is reduced to $\mathcal{P}$:

$$\mathcal{P} = \widehat{\mathcal{P}} \setminus \{p = i \to k \to l \to j : i \in \mathcal{S}, j \in \mathcal{D}, k \in \mathcal{C}, w_p \leq 0\} \setminus \{p = i \to j : i \in \mathcal{S}, j \in \mathcal{D}, w_p > 0\}. \tag{2}$$

The full enumeration of paths and the subsequent reduction of path set ensures that all paths p will fulfill charging requirements at their respective demander node because of charging time w_p associated with them. Furthermore, for each path $p \in \mathcal{P}$, the total path time l_p can be calculated as $l_p = t_{ik} + w_p + t_{lj}$.

Given the notation in Table 1, we can write the EVRR feasibility problem as follows:

$$\text{(EVRR)} \quad \sum_{p\in\phi(i)} x_p \leq 1 \qquad \forall i \in \mathcal{N}, \tag{3}$$

$$\sum_{p\in\mathcal{P}} x_p = \min\{|\mathcal{S}|, |\mathcal{D}|\}, \tag{4}$$

$$x_p \in \{0,1\} \qquad \forall p \in \mathcal{P}. \tag{5}$$

In the feasibility problem described above, the objective is to find a path for every electric vehicle. Constraint (3) makes sure that each node can only be visited once by an EV while it treads a path p. Constraint (4) ensures that the total number of paths chosen must be equal to total number of EV movements. In case $|\mathcal{S}| \geq |\mathcal{D}|$, the number of EV paths must be equal to $|\mathcal{D}|$. Conversely, if $|\mathcal{D}| \geq |\mathcal{S}|$, the number of EV paths should equal $|\mathcal{S}|$.

Sets	
$\mathcal{N}'$	Set of permissible parking spots plus the dummy nodes
$\mathcal{N}'_0$	Set of permissible parking spots, the dummy nodes and the depots
$\delta^-(j)$	Set of shuttle arcs entering node j
$\delta^+(j)$	Set of shuttle arcs leaving node j
Variables	
z_{ij}	Binary variable which equals one if a shuttle travels directly from node i to node j, and zero otherwise
y_i	Number of drivers carried on a shuttle when it leaves node i.
τ_i	Continuous variable representing the arrival time for shuttle at node i.
τ^k_{N+1}	Continuous variable representing the arrival time for shuttle k at depot node $N+1$.
Parameters	
K	The number of shuttles available
q	The maximum number of drivers available for each shuttle

Table 2 – Mathematical notation for SR Problem

3.3 Modeling the Shuttle Routing (SR) Problem

The second part of our combined operational problem is a shuttle routing (SR) problem. The network for SR, $\mathcal{N}'_0$, is same as the extended network of Section 3.2, with two dummy nodes for depot for beginning and termination of shuttle route added. Let z_{ij} be 1 if a shuttle travels between nodes i and j as part of its route and 0, otherwise. Similarly, let y_i be the number of drivers on a shuttle while it is leaving node i.

The detailed notation for SR feasibility problem is given in Table 2, while the problem itself is described henceforth:

$$\text{(SR)} \quad \sum_{j\in\delta^+(0)} z_{0,j} \le K, \tag{6}$$

$$\sum_{i\in\delta^-(N+1)} z_{i,N+1} \le K, \tag{7}$$

$$\sum_{i\in\delta^-(j)} z_{ij} \le 1 \qquad \forall j \in \mathcal{N}', \tag{8}$$

$$\sum_{i\in\delta^-(j)} z_{ij} - \sum_{i\in\delta^+(j)} z_{ji} = 0 \qquad \forall j \in \mathcal{N}', \tag{9}$$

$$z_{ij} = 1 \implies \tau_j \ge \tau_i + t_{ij} \qquad \forall j \in \mathcal{N}',\ i \in \delta^-(j), \tag{10}$$

$$z_{i,N+1} = 1 \implies \tau^k_{N+1} \ge \tau_i + t_{i,N+1} \qquad \forall\, i \in \delta^-(N+1), \forall\, k \in \mathcal{K} \tag{11}$$

$$z_{ij} = 1 \implies y_j = y_i - 1 \qquad \forall j \in S \cup C',\ i \in \delta^-(j), \tag{12}$$

$$z_{ij} = 1 \implies y_j = y_i + 1 \qquad \forall j \in C \cup D,\ i \in \delta^-(j), \tag{13}$$

$$0 \leq y_i \leq q \qquad \forall i \in \mathcal{N}', \tag{14}$$

$$z_{ij} \in \{0, 1\} \qquad \forall (i,j) \in A. \tag{15}$$

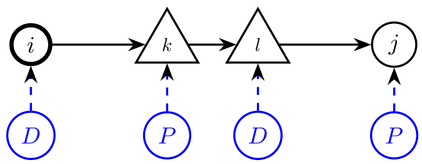


Figure 3 – Drop off (D) and pickup of (P) of drivers by shuttle at different nodes for a particular EV path p: Drivers do not wait at the charging station. The shuttle must visit the Charger node k to pick a driver and Dummy Charger node l to drop a driver.

The SR feasibility problem (6)–(15) is then to ensure that every shuttle route begins at the starting dummy node (0) and ends at the terminal dummy node $(N+1)$. Constraints (6)–(7) ensure that total arcs leaving depot node (0) or entering depot node $(N+1)$ must be less than or equal to the total shuttles K. All the other nodes must only be visited once by one of the shuttles (8). Constraint (9) is flow conservation constraint. Constraint (10) updates the arrival time of a shuttle at a node j when it visits arc (i,j) while constraint (11) finds arrival times at terminal depot node $N+1$ for multiple shuttles. Constraints (12) and (13) update the number of drivers on the shuttle as it drops off and picks up the drivers at node j. As shown in Figure 3, a driver is dropped off at supplier and dummy charger nodes while one is picked up at charger and demander nodes. Finally, constraint (14) ensures the number of drivers on each shuttle must not exceed shuttle capacity q. In the SR feasibility problem described in this section, a feasible shuttle route is any route that begins at a depot, ends at a depot while visiting any number of intermediary nodes and loading and unloading an indeterminate number of drivers. Next section synchronizes the feasibility problems described in Sections 3.2 and 3.3.

3.4 Synchronizing EVRR and SR Decision Models

A feasible synchronized EV and shuttle routing problem (f-SYNC) admits only those shuttle routes that include visits to the nodes supplied by the EVRR problem, as opposed to any number of nodes in a feasible SR. Similarly, an f-SYNC solution makes sure that drivers are available to drive the electric vehicles on the routes supplied by the EVRR problem, as opposed to an indeterminate number of drivers loaded and unloaded in a feasible SR solution. Finally, an f-SYNC solution only admits those shuttle routes that respect the time windows E_i of EV routes for the EVRR problem. The following constraints (16)–(19) serve the purpose of synchronizing τ_j and z_{ij} variables, representing the shuttle route, with E_j and x_p variables,

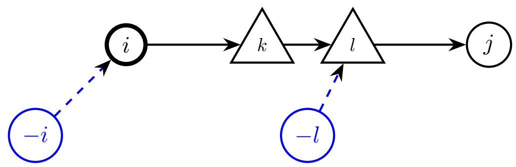


Figure 4 – The arrival times of EV at each node i on a path p depend only on shuttle arrival times at Supplier node i (driver drop-off) and Dummy Charger node l (driver drop-off) for a particular path p

representing the EV paths.

$$\sum_{i\in\delta^-(j)} z_{ij} = \sum_{p\in\phi(j)} x_p \qquad \forall j \in \mathcal{N}', \tag{16}$$

$$x_p = 1 \implies \tau_j \geq \tau_i + l_p \qquad \forall p \in \mathcal{P}, i = \mathcal{S}(p), j = \mathcal{D}(p), \tag{17}$$

$$x_p = 1 \implies \tau_j \geq \tau_i + t_{ij} \qquad \forall p \in \mathcal{P},\ (i,j) \in A(p), \tag{18}$$

$$z_{ij} \in \{0,1\} \qquad \forall (i,j) \in \mathcal{A}, \tag{19}$$

where $l_p = t_{ik} + w_p + t_{lj}$ is the length of path $i \to k \to l \to j$. Figure 4 illustrates such a path.

Consider a path $p = i \to k \to l \to j$ where $i = \mathcal{S}(p), k = \mathcal{C}(p), l = \mathcal{C}^d(p)$ and $j = \mathcal{D}(p)$. The electric vehicle arrival time at nodes belonging to path p can be uniquely determined as follows:

$$E_i = \tau_i, \tag{20}$$

$$E_k = \tau_i + t_{ik}, \tag{21}$$

$$E_l = \tau_i + t_{ik} + w_p, \tag{22}$$

$$E_j = \max(\tau_i + t_{ik} + w_p + t_{lj}, \tau_l + t_{lj}). \tag{23}$$

For a direct path $p = i \to j$ where $i = \mathcal{S}(p)$ and $j = \mathcal{D}(p)$:

$$E_i = \tau_i, \tag{24}$$

$$E_j = \tau_i + t_{ij}. \tag{25}$$

The set of equations can be used to determine the path dependent arrival times of each electric vehicle once the arrival times of shuttle at node i and node l are known. As a repositioning shuttle completes its tour, it drops and picks the drivers, who in turn relocate the EVs between nodes. Given a shuttle route, at different nodes, we may have a shuttle, an EV or a driver waiting for some time. Equation (20) states that arrival of EV at supplier node i only occurs after a driver is dropped off by the shuttle at time τ_i. Therefore, the EV wait time is simply equal to the arrival time of the shuttle at the node, i.e., $EV_{\text{wait}} = E_i$. Given τ_i, the charging process for EV begins at $\tau_i + t_{ik}$ and ends at $\tau_i + t_{ik} + w_p$. At charger node k, a driver may wait for a shuttle after dropping the EV, i.e., $D_{\text{wait}} = \tau_k - E_k$. The charging process for EV begins at $\tau_i + t_{ik}$, therefore, it does not wait on the node. EV arrival on dummy charger node l occurs at the end of charging process at time $\tau_i + t_{ik} + w_p$. At each dummy charger node, if the charging process is over before shuttle arrives, the EV must wait for the driver, i.e, $EV_{\text{wait}} = \tau_l - E_l$. In this case, EV arrival time at demander j is $\tau_l + t_{lj}$. Conversely, if the charging process is not over when the shuttle arrives, the driver must wait for the EV, i.e., $D_{\text{wait}} = E_l - \tau_l$. the EV arrival at demander j occurs at $E_l + t_{lj}$. At demander node j, if the driver has already dropped the EV before shuttle's arrival, the driver must wait, i.e., $D_{\text{wait}} = \tau_j - E_j$. In some cases, however, a shuttle may arrive at a demander before the driver brings the EV, i.e., $SH_{\text{wait}} = E_j - \tau_j$; hence, the shuttle must wait for driver's arrival.

3.5 Final Formulation for the Synchronized Approach

Thus far, we have presented three disjoint feasibility problems. The EVRR problem finds feasible EV routes and only admits the EV routes that fulfill the demand and charging requirements of the system. Similarly, the SR problem finds feasible shuttle routes, and the f-SYNC admits only those feasible shuttle routes that are synchronized with the requirements outlined in the EVRR problem. In this section, we connect all three disjoint feasibility problems into a combined MIP formulation with an overarching objective function.

$$
\begin{aligned}
\text{(SYNC)} \quad \text{minimize} \quad & \sum_{k \in K} \tau^k_{N+1}, && && (26) \\
\text{subject to} \quad & (3)\text{–}(5), && \text{[EV Relocation and Recharging (EVRR)]} \\
& (6)\text{–}(15), && \text{[Shuttle Routing (SR)]} \\
& (16)\text{–}(19). && \text{[Synchronizing EV and Shuttle Routing (f-SYNC)]}
\end{aligned}
$$

The objective function (26) of the combined problem minimizes the travel time or makespan of shuttle routes. Thus the combined MIP problem makes sure that only those electric vehicle relocations are carried out that are promising for the overall system objective of minimizing the physical cost of carrying out that relocation using shuttles and drivers.

3.6 Formulation for the Standalone EVRR Problem

EVRR feasibility problem in Section 3.2 describes a relocation problem complete with demand and charging requirements. This relocation model can be used in a standalone manner by the decision makers, interested in relocation alone, using an appropriate objective function.

$$\text{(P1)} \quad \text{minimize} \quad \sum_{p \in \mathcal{P}} l_p x_p \tag{27}$$

$$\text{(P2)} \quad \text{minimize} \quad \max_{p \in \mathcal{P}} (l_p x_p) \tag{28}$$

We recommend two different objective functions for relocation problem as given by (27) and (28). The former minimizes the total length of the paths selected for EVs while the latter minimizes the maximum length of paths selected. The second objective can be linearized by introducing an auxiliary variable m and adding an extra constraint.

$$\text{(P3)} \quad \text{minimize} \quad m, \tag{29}$$

$$\text{subject to} \quad m \geq l_p x_p \qquad \forall p \in \mathcal{P}. \tag{30}$$

It is easy to show that (P3) with constraint (30) is equivalent to (P2). The relocation problem gives a set of EV routes $\bar{x}$ as an optimal solution. On its own, the solution of EVRR problem for the full city-wide network provides decision makers with optimal relocation decision. The problem will also be used in the heuristic approaches for solving the synchronized EV relocation and shuttle routing problem.

3.7 Formulation for the Sequential Approach

We can use the standalone relocation model presented in Section 3.6 and a subsequent shuttle routing model to formulate the sequential approach.

$$\text{(SEQ-A)} \quad \text{minimize} \quad \sum_{p \in \mathcal{P}} l_p x_p, \tag{31}$$

$$\text{subject to} \quad \text{(3)–(5).} \qquad \text{[EV Relocation and Recharging (EVRR)]}$$

Let $\bar{x}$ be an optimal solution for (SEQ-A). Given $\bar{x}$, solve (SEQ-B).

$$\text{(SEQ-B)} \quad \text{minimize} \quad \sum_{k \in K} \tau^k_{N+1}, \tag{32}$$

$$\text{subject to} \quad \text{(6)–(15),} \qquad \text{[Shuttle Routing (SR)]}$$

$$\text{(16)–(19).} \qquad \text{[Synchronizing EV and Shuttle Routing (f-SYNC)]}$$

The sequential approach uses same modeling components as synchronized approach. It is easy to see that the sequential approach gives us solutions which are feasible for synchronized approach but are sub-optimal. Therefore, solutions for sequential approach can be used as initial solutions for synchronized approach.

4 Computational Methods

Owing to large size of our problems for the full city-wide network, we devise heuristic approaches to solve the sequential (SEQ) and the synchronized (SYNC) problems. The heuristic approaches for SEQ and SYNC rely on cluster-relocate-route approach to solve the EV relocation and shuttle routing problems. The algorithmic components for the two approaches are described in this section. In Section 4.1, we describe the heuristic approach for solving the sequential problem. The best solution from sequential approach serves as an initial solution for our main algorithm, the Exchange-Based Neighborhood-Search Method (EBNSM), for solving the SYNC problem as described in Section 4.2.

4.1 Setting the Benchmark: the Sequential Approach

To show the system wide benefits of the synchronized (SYNC) approach to EV relocation and shuttle routing problem, we compare it with the sequental (SEQ) approach. Since we want to compare the two approaches in terms of their relocation decisions, we use a similar cluster-relocate-route approach for the SEQ problem. The steps to solve the SEQ problem are described as Algorithm 2.

Algorithm 2: The Sequential Approach

Input: $\mathcal{N}'_0, K, q$

Output: Best EV paths: $\mathcal{X}_k$, Best shuttle route: **z**

Given number of shuttles K, find K-centers in the network using greedy approximation approach mentioned in Section 4.1.1;

Solve the MIP problem (NCA) described in Appendix B for creating K clusters of nodes from K-centers. Put the objective coefficient $d_{ik} = t_{ik}, \forall i \in \mathcal{N}, k \in \mathcal{K}$. Let $\mathcal{N}_k, \mathcal{S}_k, \mathcal{C}_k, \mathcal{D}_k$, and $\mathcal{P}_k$ be the sets of nodes, suppliers, chargers, demanders and paths for each cluster k, respectively;

foreach $1 \leq k \leq K$ **do**

  Given $\mathcal{N}_k, \mathcal{S}_k, \mathcal{C}_k, \mathcal{D}_k$, and $\mathcal{P}_k$, solve (SEQ-A). Let $\mathcal{X}_k$ be the set of best EV paths;

  Given $\mathcal{X}_k$, obtain an initial route r_k for the shuttle using greedy approach;

  Given an initial shuttle route (r_k), run the customized 2-interchange algorithm to get the best shuttle route. Let $\text{ub}^k \leftarrow$ obj(2-int);

$\text{bestSoln} \leftarrow \sum_{k \in \mathcal{K}} \text{ub}^k$;

$\text{bestRoutes} \leftarrow \bigcup_{k \in \mathcal{K}} r_k$;

The clusters for the SEQ approach are formed using minimum cost assignment problem NCA with objective coefficient $d_{ik} = t_{ik}$ while the relocation decision is achieved by solving minimum path cost EV relocation and recharging problem (SEQ-A). The MIP formulation SEQ-B can be used to get optimal shuttle routes for moderate sized instances but fails for large instances owing to precedence constraints and loose time windows. Therefore, we use a greedy algorithm to construct an initial shuttle route from optimal EV path set and use the customized 2-interchange algorithm to improve the route.

4.1.1 Finding K-centers

As first step for sequential method, we solve a K-center problem to find K centers in the network. The K-center problem is a well known NP-hard problem (Megiddo and Supowit, 1984). We use a 2-opt greedy approximation algorithm (Plesník, 1987) to solve the problem. We let t_{ij} be the traversal time between nodes i and j and assume the time matrix to be symmetric.

4.1.2 Creating K Clusters

Once K-centers have been found, we can assign to each center k a set of nodes which form the k-th cluster. The clustering process is described in detail in Appendix B.

4.1.3 Finding Optimal EV Routes

Once K clusters have been formed, optimal EV paths in each cluster can be found by simply solving the minimum path cost EV relocation and recharging problem (SEQ-A). The output of the problem is an optimal set of paths for each cluster. The set of paths is subsequently used to find best shuttle routes for each cluster.

4.1.4 Finding Optimal Shuttle Routes

We present a customized 2-interchange algorithm which draws from a 2-interchange procedure for Dial-A-Ride Problem (DARP) presented in Psaraftis (1983). A DARP involves a vehicle picking up and dropping off multiple customers. In our problem, the EVs are equivalent to customers in DARP.

Contrary to a customer in DARP, each EV moves multiple times through the nodes on its route. Therefore, a shuttle must satisfy multiple precedence constraints for each EV. However, given a set of EV paths, a shuttle route r can be constructed trivially using a greedy procedure whereby the suppliers are visited first, followed by chargers, dummy chargers and demanders, in this order, while maintaining capacity feasibility for drivers. Given r, a new route r_{new} can be constructed by swapping two arcs $(i, i+1)$ and $(j, j+1)$ with two new arcs (i, j) and $(i+1, j+1)$. Since direction of segment $(i+1 \to \cdots \to j)$ is now reversed, it is necessary to check precedence feasibility and ensure driver availability on the shuttle. The proposed 2-interchange procedure is presented as Algorithm 4 in Appendix A. The feasibility and improvement checks are customized for our problem and their algorithmic descriptions are also presented, in relative detail, in Appendix A.

4.2 Solving the SYNC Problem using EBNSM

In this section, we describe in relative details the steps of EBNSM procedure for finding solutions to synchronized EV relocation and shuttle routing problem (SYNC). EBNSM is an iterative procedure, described as Algorithm 3, for solving EV relocation and shuttle routing problem synchronously. It relies on solution for SEQ method and improves it by iteratively adding neighborhood paths and updating the shuttle routes. Here, we expand on the individual steps.

EBNSM improves the solution obtained from sequential method by iteratively adding new EV paths and updating the shuttle route. The exchange procedures, `ExchangeSuppliers()`and `ExchangeChargers()`, are used to replace a pair of old paths in path set $\mathcal{X}$ with a pair of new paths by exchanging their supplier and charger/dummy charger nodes, respectively. No new nodes are added in the process. However, the visiting order of exchanged nodes must be changed in the shuttle route to ensure that precedence feasibility

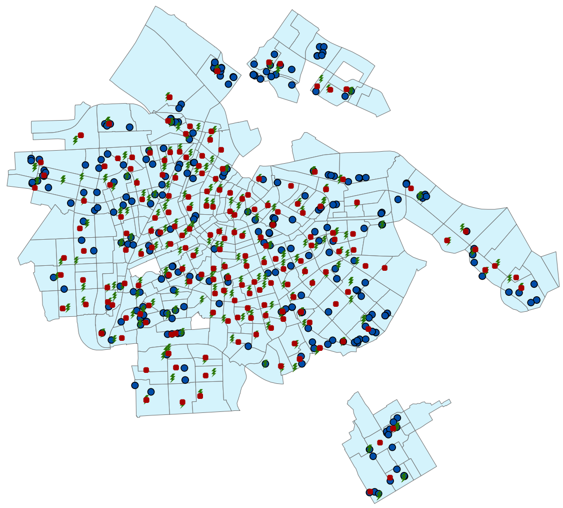
Algorithm 3: Exchange-Based Neighborhood-Search Method (EBNSM)

Input: $\mathcal{N}_0', \mathcal{S}, \mathcal{C}, \mathcal{C}^d, \mathcal{D}$, K = Number of Shuttles = Number of Clusters, q = Number of Drivers per Shuttle, SNSIt: Number of iterations for Small Neighborhood Search

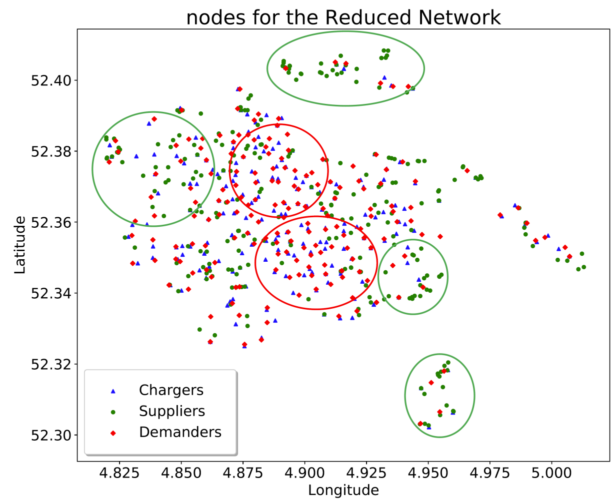
Output: Best shuttle routes: $\mathbf{z}$

Given number of shuttles K, use Algorithm 2 to find initial paths $\mathcal{X}_k$ and initial shuttle routes r_k for each cluster k ;
for $k \leftarrow 1$ **to** K **do**
    for $\mathrm{it} \leftarrow 1$ **to** SNSIt **do**
        $\Delta \leftarrow 0$;
        for $i \leftarrow 1$ **to** $|\mathcal{X}_k|$ **do**
            for $j \leftarrow i+1$ **to** $|\mathcal{X}_k|$ **do**
                $\overline{\mathcal{X}}_k^i, \overline{\mathcal{X}}_k^j \leftarrow \texttt{ExchangeSuppliers}(\mathcal{X}_k^i, \mathcal{X}_k^j)$; // $\overline{\mathcal{X}}_k$ is the updated path set
                $r^k_{\text{new}} \leftarrow \texttt{UpdateRoutes}(\overline{\mathcal{X}}_k)$;
                $\Delta_{ij} \leftarrow \texttt{RouteImprovement}(r^k, r^k_{\text{new}})$;
                if $\Delta_{ij} > \Delta$ **then**
                    $\Delta \leftarrow \Delta_{ij}$;
        if $\Delta = 0$ **then**
            break ;
        $r^k \leftarrow r^k_{\text{new}}, \mathcal{X}_k \leftarrow \overline{\mathcal{X}}_k$;
    for $\mathrm{it} \leftarrow 1$ **to** SNSIt **do**
        $\Delta \leftarrow 0$;
        for $i \leftarrow 1$ **to** $|\mathcal{X}_k|$ **do**
            for $j \leftarrow i+1$ **to** $|\mathcal{X}_k|$ **do**
                $\overline{\mathcal{X}}_k^i, \overline{\mathcal{X}}_k^j \leftarrow \texttt{ExchangeChargers}(\mathcal{X}_k^i, \mathcal{X}_k^j)$; // $\overline{\mathcal{X}}_k$ is the updated path set
                $r^k_{\text{new}} \leftarrow \texttt{UpdateRoutes}(\overline{\mathcal{X}}_k, r^k)$;
                $\Delta_{ij} \leftarrow \texttt{RouteImprovement}(r^k, r^k_{\text{new}})$;
                if $\Delta_{ij} > \Delta$ **then**
                    $\Delta \leftarrow \Delta_{ij}$;
        if $\Delta = 0$ **then**
            break ;
        $r^k \leftarrow r^k_{\text{new}}, \mathcal{X}_k \leftarrow \overline{\mathcal{X}}_k$;
    $\mathrm{ub}^k = \mathrm{length}(r^k)$
Given route r^k, generate a vector $\mathbf{z}_k$;
ub $\leftarrow \sum_{k \in K} \mathrm{ub}^k$;
$\mathrm{bestSoln} \leftarrow$ ub, $\mathrm{bestRoutes} \leftarrow \mathbf{z}_k$;

is maintained. The route update step in EBNSM, `UpdateRoutes()`, swaps the positions of the pair of exchanged nodes in the shuttle route according to the updated path set $\overline{\mathcal{X}}_k$. In doing so, since a supplier is swapped with a supplier, and a charger with a charger, the capacity feasibility of shuttle route is also maintained. Finally, the route improvement check for the new route r_{new} is done using the same procedure



(a) Map for the Neighborhoods



(b) All Nodes for Reduced Network

Figure 5 – Left: Map for the Neighborhoods together with Suppliers (blue circle), Demanders (red square) and Chargers (green bolt sign), Right: All Nodes for Reduced Network

`RouteImprovement()` as used in Algorithm 4 and described in Appendix A.

5 Case Study: car2go in Amsterdam

We apply our framework to a fully operational system of car2go in Amsterdam, the Netherlands, where the FFEVS service is provided using more than 300 EVs. From the actual data, we take the initial and target locations of EVs that need to be relocated and test the performance of our computational method. We use actual municipal boundaries of smallest size as neighborhoods as shown in Figure 5-a). For the computational experiment, we first construct a full network containing 339 neighborhoods, 829 nodes, 332 dummy nodes and 2 nodes representing the depot. We use rules in Section 3.1 to reduce the size of the network. Final network has 155 suppliers, 270 chargers, 270 dummy chargers and 155 demanders. The total number of possible EV paths is $|\mathcal{P}| = 6,267,580$, out of which 155 paths need to be selected to relocate the 155 EVs. The nodes are depicted in Figure 5-b), which clearly shows that the current EV locations (suppliers) and the desired locations (demanders) are concentrated in different areas, hence necessitating relocation operations.

5.1 Dataset and Parameters

In our numerical experiments, we use both neighborhood level and spot level data. Based on the two boolean characteristics, i.e., occupancy and availability of charging infrastructure at a node, each node can belong to either of the four node types described in Section 3.2. Each node is also located in a certain neighborhood. All the neighborhoods and the nodes therein have locations described through their coordinates. The locations are used to generate their inter node distances and travel times. We calculate Euclidean distances, and multiply them with a detour index of 1.4 to estimate driving distances (Boscoe et al., 2012). We use city speeds of 30 miles per hour to estimate travel times in minutes. The parameter value of β_1, the vehicle's charge depletion rate, is set to 2.5 minutes per unit of charge depletion. Similarly, β_2 representing a vehicle's charging rate and given in percentage gain in charging per minute is set at 0.4. The values for β_1 and β_2 were derived from actual system data. Barring a major technological shift, these values will likely stay fixed over the medium term. The starting battery level for all vehicles parked at the supplier nodes given by c^0 is found from the system status data at 11:59 am on May 8th, 2016. The desired charging level c^f for all nodes located in a particular neighborhood is found as an average of the initial battery levels for all the trips originating in that neighborhood. In our experiments, we allow for full recharging whereby all the EVs are recharged to 100%.

5.2 The Exact Approach

In this Section, we present the results for solving the standalone EV relocation and recharging problem (Section 3.6), sequential EV relocation and shuttle routing problem (Section 3.7), and synchronized EV relocation and shuttle routing problem (Section 3.5) using exact solution approaches. All the experiments were done on a machine with 3.6 GHz CPU clock speed, 16 GB RAM and 64-bit Windows 8 operating system using Java API of CPLEX 12.9.0.

For sequential approach, the relocation and recharging problem (SEQ-A) was solved to optimality for the full network with 829 nodes within 118 seconds. However, the subsequent shuttle routing problem (SEQ-B) is a DARP variant with each EV (customer) being served up to four times by a repositioning shuttle. The problem also involves multiple precedence constraints and loose time windows which depend on shuttle arrivals at the preceding nodes. CPLEX was only successful in solving SEQ-B exactly for up to 10 EVs and 50 nodes. For synchronized EV relocation and shuttle routing problem described in Section 3.5, CPLEX failed to solve the MIP model for the system sized instance discussed in this paper. For this instance, we could not even get a feasible solution for the model. In general, CPLEX was successful in solving the MIP model exactly when the problem size was limited to 15 neighborhoods and 50 total nodes.

Solution time is another important consideration. The operational nature of the problem necessitates solution methods which provide "reasonably good" solutions within a few minutes. In some cases, customized decomposition-based approaches have previously been used for similar integrated models in other industries. However, the instances solved were either small (Luo et al., 2019) or took many hours to achieve sufficient convergence (Cordeau et al., 2001). The structure of our problem combined with large size of our instances and the need for quick solutions makes our problem less suitable for decomposition based approaches. Therefore, we used heuristic approaches to solve the real-life instances of our problem.

The results comparing the exact approaches with the heuristic approaches for solving SEQ and SYNC problems for small instances are provided in Table 3. Generally, for the smaller sized problems, the dif-

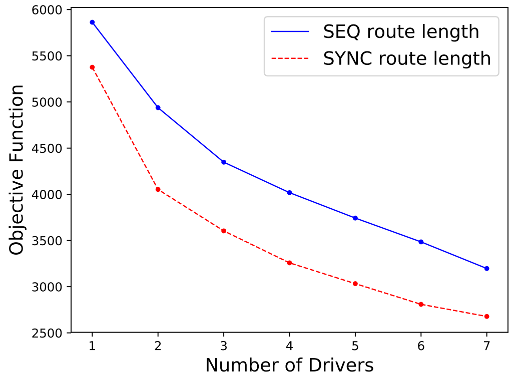
Problem Size				Synchronized Approach		Sequential Approach	
$\lvert\mathcal{N}'\rvert$	$\lvert\mathcal{H}\rvert$	$\lvert\mathcal{X}\rvert$	Algorithm	Exact Method	EBNSM	Exact Method	Our Heuristic
15	6	2	Objective	123.8	123.8	123.8	123.8
			CPU Time	10.65	2.23	1.80	2.07
25	10	4	Objective	156.7	155.3	155.3	155.3
			CPU Time	3600	2.45	470.40	2.28
45	15	9	Objective	184.6	184.8	184.8	184.8
			CPU Time	3600	2.76	3600	2.69

Table 3 – Performance comparisons (small scale problem case): Synchronized vs. Sequential approaches as well as Exact method vs. Our algorithms

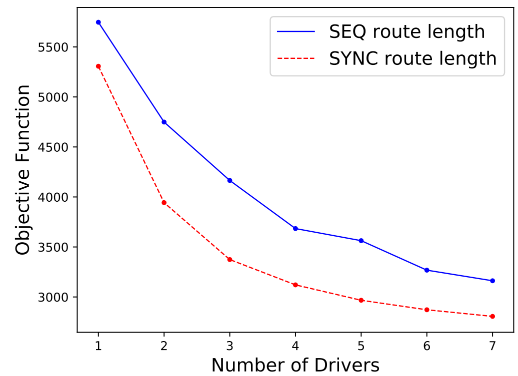
ference between synchronized and sequential approaches was not substantial owing to the fact that even in sequential approach, the relocation objective is to minimize the total EV path length of all EVs. In smaller networks, this is a good proxy for minimizing the total shuttle route. The results obtained using heuristic approaches were comparable to those with exact approaches for small instances while taking considerably smaller time. Therefore, for solving the system sized instance of the sequential and synchronized problems, we use the heuristic approaches presented in Section 4.

5.3 Computational Performance of EBNSM

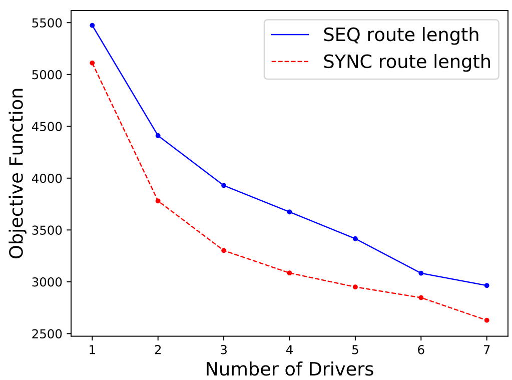
We run EBNSM for instances of the SYNC problem for various numbers of shuttles K and number of drivers q. Each shuttle services one cluster. We solve SEQ-A to get an initial path set $\mathcal{X}_k$ and use the 2-interchange procedure to get shuttle route r_k. Since we do not have a lower bound for the shuttle route length, we use number of iterations to be the termination criterion. Moreover, if K is small, each shuttle will have larger route length. Therefore, the number of 2-interchange iterations are inversely proportional to K. We use 1000/K. The numbers are empirically chosen as when the 2-interchange procedure has stopped improving the objective. Given the size of the instances, we also want to limit the run time of the algorithm to 20 minutes. Since the size of clusters varies in inverse proportion to number of shuttles/clusters K, we run 500/K iterations of the neighborhood procedure for finding the "best" EV relocation decision. We use a depth-first strategy and for each iteration select the exchange with highest improvement. For each iteration of the neighborhood search, we update the shuttle route while maintaining precedence and capacity feasibility to get the best shuttle route given the EV paths selected. When the iterations have stopped giving improvement, we terminate the exchange procedure. Each instance of EBNSM for SYNC problem takes less than 10 minutes to terminate. Moreover, the number of iterations can be modified according to the size of the problem.



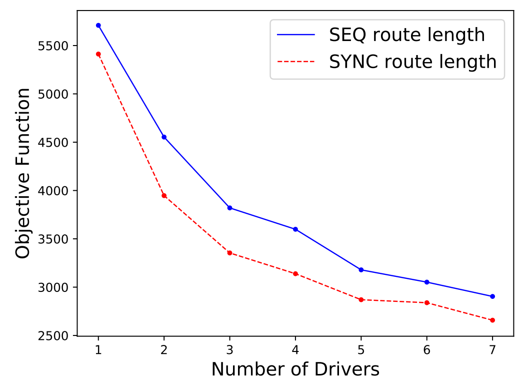
(a) K **= 4**



(b) K **= 5**



(c) K **= 6**



(d) K **= 7**

Figure 6 – The difference in objective function values for SYNC and SEQ approaches for different values of K and q.

5.4 Value of the SYNC Approach

We compare the results by the SYNC and SEQ approaches. The purpose of comparing sequential and syncrhonized approaches is not to claim equivalence between the two problems. Similar integrated models have been presented before to show the benefits of integration. For instance, the two aspects of location and routing have been "simultaneously" considered in location-routing literature. The combined problem, although more complex, offers the "promise of more effective and economical decisions" (Balakrishnan et al., 1987).

Given values of K and q, the output of each instance of algorithms is in terms of total route length in minutes denoted as L. We can also calculate following parameters: total number of personnel $(P = Kq+K)$ and average route time per shuttle in hours $(T = \frac{L}{60\times K})$. We also calculate total wait times for shuttles (SH_{wait}), EVs (EV_{wait}) and drivers (D_{wait}) for the best route. We compare the performance of SYNC and SEQ approaches for 100% recharging situation. For full recharging, the SYNC approach outperforms the SEQ approach in terms of total route length for all instances of the problem. On average, the route length for SYNC approach is 15% shorter as compared to the SEQ approach. The difference in objective function value varies considerably, in 5–28% range, across instances. Generally, the difference grows larger when the number of drivers on board each shuttle increases as can be seen in Figure 6. However. if the number of drivers is increased so much that some drivers are not utilized at all, the difference between the two approaches shrinks. As with the exact approach, when service area is very small, i.e., when the number of clusters/ shuttles is very large, the difference between the SYNC and SEQ approaches decreases.

We also compare the cumulative wait times for all EVs (155), all shuttles (K) and all drivers ($K \times q$) for SYNC and SEQ approaches. The SYNC approach outperforms the SEQ approach in terms of EV wait times. On average, the average improvement for full recharging is 21%. The improvement for driver wait times is also 24% on average. However, the difference decreases as number of drivers per shuttle increases. In case of shuttles, the wait times are negligibly small for small K values. As K increases, the wait times oscillate and SEQ approach has, on average, smaller shuttle wait times in case of full recharging.

6 Resource Allocation and Operational Efficiency

An important managerial decision in the operation of EV relocation for FFEVS systems is how operational resource allocation decisions impact the efficiency of operation. In this section, we study the relationship between the operational cost of EV relocation and the size of the shuttles and number of drivers. Specifically, using our computational method and car2go data, we conduct an extensive numerical experiment by calculating the relocation cost for different number of shuttles (clusters) and number of drivers on board each shuttle. We find that besides the per unit shuttle and personnel costs, the size of service area and the initial battery levels of EV fleet are important determining factors for an efficient operation of EV relocation in FFEVS systems.

6.1 Cost Parameters

Let Γ_{D} and Γ_{SH} be the hourly costs for personnel and shuttles, respectively. The total cost of relocation operation can be calculated as $C = P \times T \times \Gamma_{\mathrm{D}} + K \times T \times \Gamma_{\mathrm{SH}}$. It is the sum of personnel cost and the shuttle operating cost. In our experiments, we assume the labor cost to be $40/hr, and the per hour operating cost for a shuttle to be $24. We also limit the available time for relocation operation to 7 hours. Therefore, we only consider the instances which achieve the relocation operation within the time limit.

6.2 Sensitivity Analysis for the Number of Shuttles and Size of Shuttles

We consider the sensitivity to the number of repositioning shuttles K and the maximum number of drivers on board each shuttle q. The parameter q is also a proxy for the size and type of shuttle being used by the system. As mentioned earlier, different FFEVS systems may use a van, a car, or single-person mobility options like scooters and foldable bicycles to carry out the relocation. It is worthwhile to know which of these options is most cost effective for system operators and also results in least wait times for personnel.

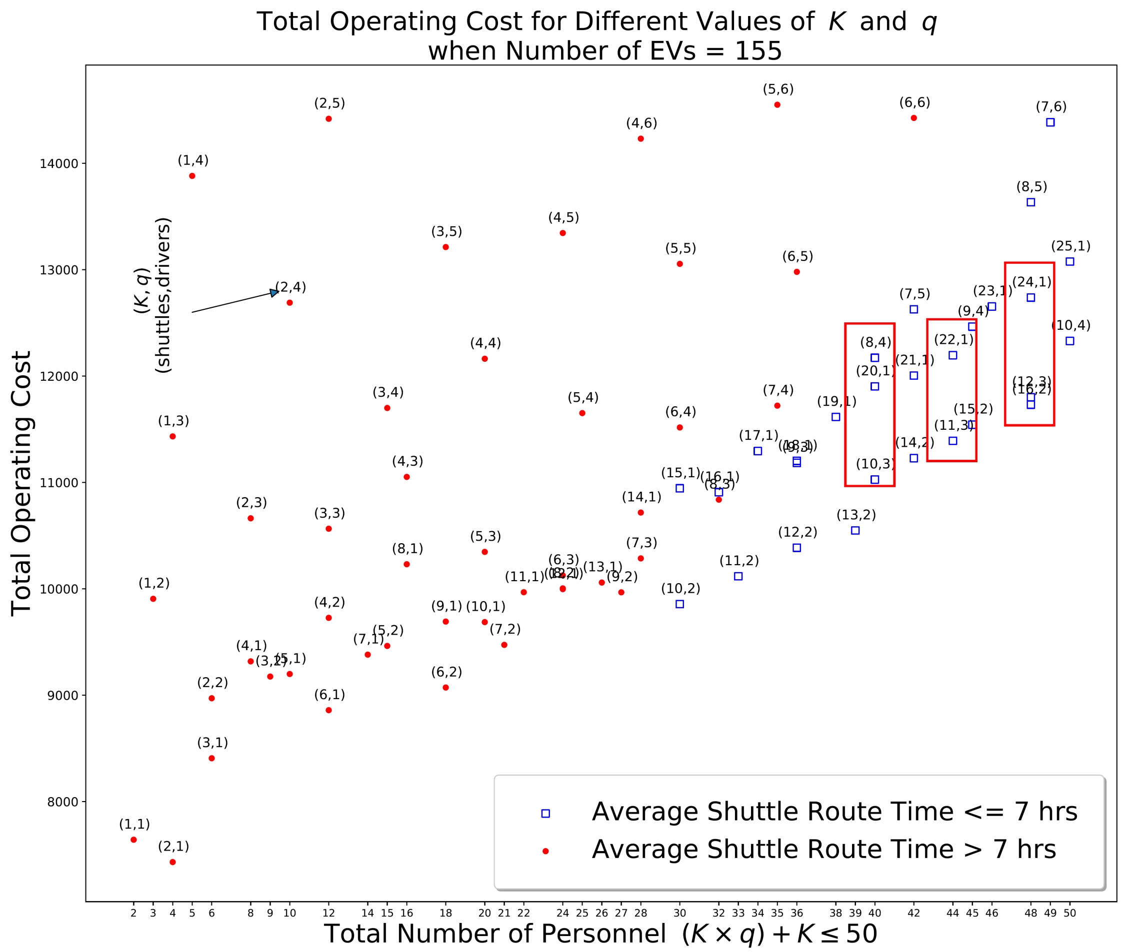


Figure 7 – For a Large Network with 155 EVs and given a certain number of personnel and per unit costs, it is advantageous to cluster more and increase the number of shuttles

To carry out the sensitivity analysis, we vary the value of K from 1 to 30 and q from 1 to 12. By varying K and q, we not only vary the extent of clustering, but also find out the best way to distribute manpower and shuttle resources to carry out system-wide relocation of EVs. Given a number of total personnel P, a manager may be interested in the best resource allocation in terms of shuttles and drivers. For instance if $P = 6$, the possible resource allocation combinations could be $(K, q) = \{(1, 5), (2, 2), (3, 1)\}$, since each shuttle also requires a driver. In Figure 7, we map the cost of all such combinations for P ranging from 2 to 50 with K ranging from 1 to 30 and q ranging from 1 to 12 when 155 EVs are to be relocated. We observe that in most cases the combination (K, q) with larger number of shuttles is also more cost effective. However, as shown in boxes in Figure 7, in some cases, for given P, increasing K actually increases the cost of operation and combinations with smaller number of shuttles and larger number of drivers per shuttle are more cost effective. When we repeat the same experiment with a smaller network of 10 EVs varying K from 1 to 5 and q from 1 to 12, we obtain similar results, shown in Figure C.1 of Appendix C, where choosing the highest K is not the best option. Details of the instances for which (K, q) combinations with

P	(K, q)	Average Hours per Shuttle	Total Operating Cost	Shuttle Wait Time	EV Wait Time	Driver Wait Time
6	(1,5)	3.7	972	4	716	589
	(2,2)	3.3	948	146	505	498
	(3,1)	3.5	1,092	328	738	331
8	(1,7)	3.3	1,122	8	500	681
	(2,3)	3.1	1,132	128	383	578
	(4,1)	3.2	1,345	488	647	489
10	(1,9)	3.3	1,388	0	502	676
	(2,4)	3.0	1,341	136	292	791
	(5,1)	3.1	1,603	611	499	614
12	(1,11)	3.3	1,667	3	451	746
	(2,5)	3.0	1,580	137	241	686
	(3,3)	3.1	1,700	295	196	869
	(4,2)	3.0	1,713	426	172	773
24	(2,11)	3.0	3,017	124	235	658
	(3,7)	3.1	3,179	300	179	944
	(4,5)	3.0	3,140	406	161	745

Table 4 – Comparison of Shuttle Route Length and Operating Costs for For Various (K, q) Combinations Given Number of Personnel for a Small Network when Number of EVs = 10. The Instances in the Table, Marked Red in Figure C.1 of Appendix C, Correspond to Situations when Increasing the Number of Drivers is Advantageous

smaller number of shuttles are more cost effective are given in Table 4.

Therefore, it is important for a decision maker to understand when it is advantageous to increase drivers as opposed to shuttles and vice versa, for given number of available personnel. Systems with relatively high labor cost should consider single person mobility options like foldable bicycles or scooters so a cluster of EVs could be assigned to an individual driver without any cars or vans moving him/her around. Changing the per unit cost parameters may also impact these results and the optimal strategy may shift to two drivers per shuttle rather than one driver. However, increasing the number of drivers per shuttle beyond two drivers invariably gives diminishing returns due to large driver wait times on the shuttle. This suggests that for most systems with high hourly personnel costs, using large vans may not be a cost effective option.

6.3 Analyzing the Shuttle, EV, and Driver Wait Times

For given Γ_{D} and Γ_{SH}, the average route time per shuttle given by T also impacts the total cost of operation. T is in turn inlfuenced by shuttle wait times. As K is increased, each shuttle is responsible for relocating

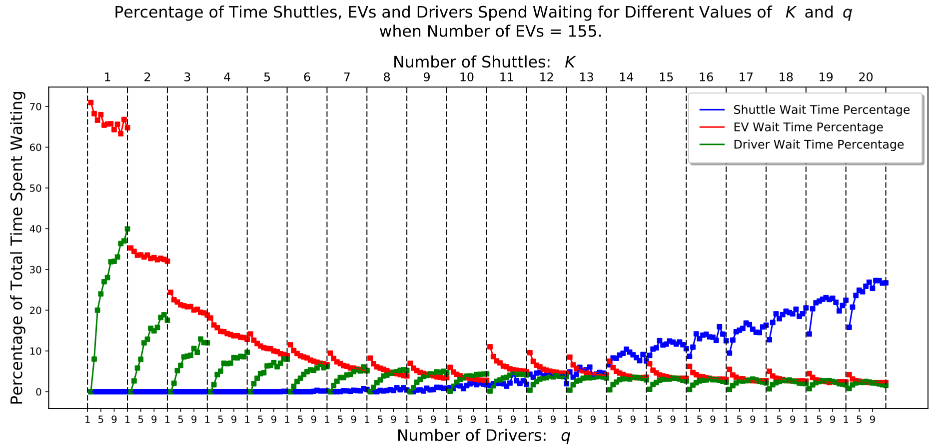


Figure 8 – Shuttle wait times increase as the number of shuttles increases

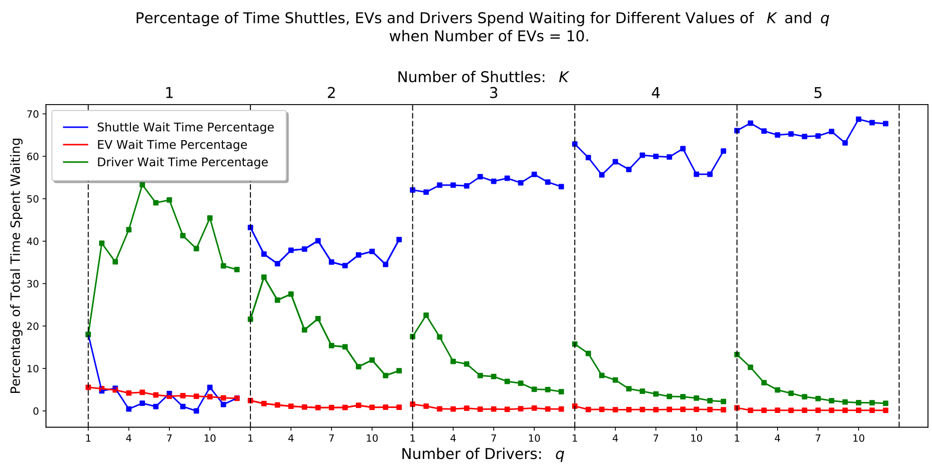


Figure 9 – Shuttle wait times increase as the number of shuttles increases

smaller number of EVs. For $K = 1$, a shuttle must relocate all 155 EVs and for $K = 14$, the number drops to less than 12 EVs. As the number of EVs per shuttle decreases, the corresponding shuttle wait times start increasing as can be seen in Figures 8 and 9. This is owing to the fact that for smaller number of EVs, shuttles must wait for EVs to get recharged before picking up the final batch of drivers from the demander nodes. Therefore, after some point, adding more shuttles only increases the wait times of the shuttles, and it becomes beneficial to increase the number of drivers instead, as shown for a small instance in Figure 10.

6.4 Analyzing the Impact of Initial Battery Levels and Charging Speed

The difference between initial battery levels and desired charging levels is also an important factor in relocation operations. A larger difference signifies longer recharging processes and larger wait times for shuttles and drivers. This impact is considerable especially for small-scale networks where an EV's charging process may delay the trip completion. For low initial battery levels, the charging process takes longer and adding

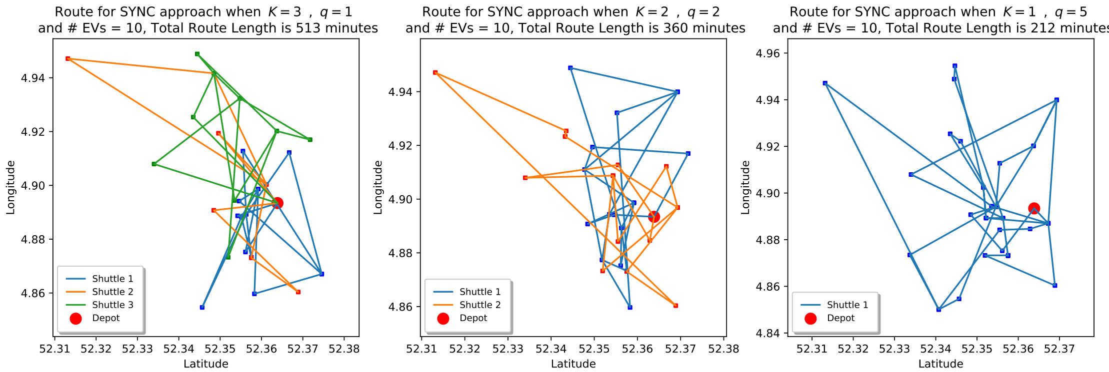


Figure 10 – Possible relocation operation alternatives when number of personnel = 6, number of EVs = 10. For $(K, q) = (3, 1)$, shuttles wait 52% of the time; for $(K, q) = (2, 2)$, shuttles wait 37% of the time; for $(K, q) = (1, 5)$ shuttles wait 1.8% of the time.

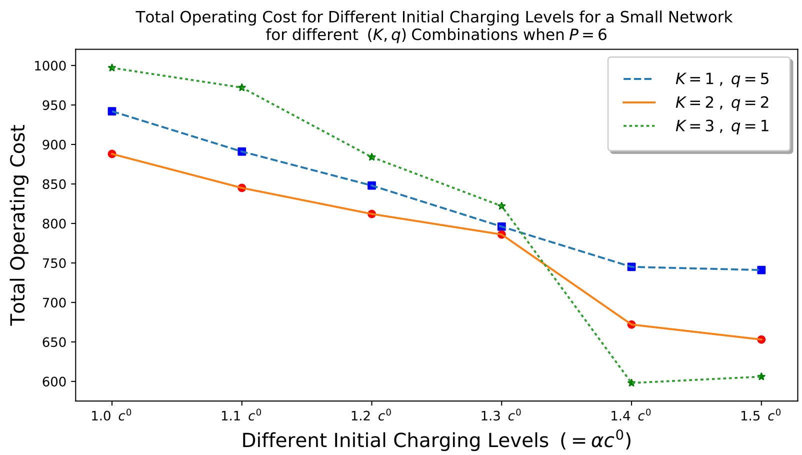


Figure 11 – As initial battery levels increase, it becomes beneficial to increase the number of shuttles

extra shuttles only contributes to shuttle wait times. However, if the initial battery levels get higher (so the charging requirements become lower), increasing the number of shuttles can still be beneficial. As shown in Figure 11, we increase the initial battery levels from c^0 to αc^0 uniformly. As α increases, the charging requirement decreases, and the cost of operation with larger number of shuttles (green asterisks) becomes more favorable.

We note here that, as shown in equation (1), increasing the initial charging levels c^0 decreases w_p, i.e., the time spent charging on the charging station. A similar effect will occur as we vary the charging speed, i.e., parameter β_2. As charging speeds become faster, due to technological advancement, it will result in a decrease in w_p. Therefore, the sensitivity analysis with different initial battery levels is a good proxy for varying the charging speed. As technology advances, and charging speed becomes faster, increasing the number of shuttles, i.e., using a larger number of smaller shuttles to carry out the relocation operation becomes beneficial.

7 Conclusion

This paper presents a mathematical model to solve the problem of nightly repositioning and recharging of electric vehicles in a FFEVS system. Since in many such systems, EVs are moved by a crew of shuttles and drivers, we propose that the relocation decision be made in synchronization with shuttle routing decision to minimize the cost of relocation operation. In contrast, most current approaches make the relocation and shuttle routing decisions sequentially. Our unique path based formulation can solve problems of moderate size. For system sized instances, we propose the exchange based neighborhood search method which draws from the mathematical model and solves all the instances within 10 minutes. Comparison of synchronized approach with sequential approach shows that the former improves the total length of shuttle routes and in turn the cost of relocation operation by 15% on average. FFEVS systems require an elaborate relocation operation and improving the cost of such an operation improves the bottom line of these systems. Moreover, our model achieves complete system wide repositioning and recharging, therefore improving the distribution of EVs and directly addressing the issues of demand imbalance and range anxiety in FFEVS systems.

The data for our experiments comes from a real life free-floating car sharing operator and the instances used in this paper represent the complexities of an actual relocation operation. This paper presents EBNSM which uses cluster-relocate-route approach to solve the system-sized relocation instances for relocating and recharging. For the largest instance solved in this paper, we relocate 155 EVs while increasing the system wide average charging levels from 42 % to 90 %. The model is also flexible to changes in system status, initial battery levels, and desired demand configuration. It allows for partial recharging of EVs and their relocation close to the actual demand points.

We conduct a variety of experiments with different numbers of shuttles and drivers per shuttle, a proxy for shuttle size, to find out the most cost effective human resource allocation. The results suggest that given a certain number of personnel, it is more cost effective to increase the number of shuttles rather than number of drivers on each shuttle, especially when the service area is large. This trend may be reversed for small service areas and small number of EV relocations per shuttle. In these cases, adding extra shuttles only adds to wait times and increasing the number of drivers may be beneficial. This implies that although increasing the number of drivers on shuttles improves the route length, the improvement is not justified due to extra cost. Therefore, systems with relatively high labor cost should consider single person mobility options like foldable bicycles or scooters as this will relieve the cost of an extra car and driver.

References

Balakrishnan, Anantaram, James E Ward, Richard T Wong. 1987. Integrated facility location and vehicle routing models: Recent work and future prospects. *American Journal of Mathematical and Management Sciences* **7**(1-2) 35–61.

Bellan, R. 2018. The grim state of electric vehicle adoption in the U.S. URL https://www.citylab.com/transportation/2018/10/where-americas-charge-towards-electric-vehicles-stands-today/572857/.

Boscoe, F. P., K. A. Henry, M. S. Zdeb. 2012. A nationwide comparison of driving distance versus straight-line distance to hospitals. *The Professional Geographer* **64**(2) 188–196.

Bruglieri, M., A. Colorni, A. Luè. 2014. The vehicle relocation problem for the one-way electric vehicle sharing: an application to the milan case. *Procedia-Social and Behavioral Sciences* **111** 18–27.

Caggiani, L., R. Camporeale, M. Ottomanelli. 2017. A dynamic clustering method for relocation process in free-floating vehicle sharing systems. *Transportation Research Procedia* **27** 278–285.

car2go. 2017. The five conditions essential to successfully operate autonomous carsharing fleets in the future URL https://www.car2go.com/media/data/italy/microsite-press/files/car2go_white-paper_autonomous-driving_2017_en.pdf.

Chianese, Y.M., A. Avenali, R. Gambuti, L. Palagi. 2017. One-way free floating car-sharing: Applying vehicle-generated data to assess the market demand potential of urban zones. *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual*, vol. 2. IEEE, 777–782.

Cordeau, Jean-François, Goran Stojković, François Soumis, Jacques Desrosiers. 2001. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation science* **35**(4) 375–388.

de Almeida Correia, G. H., A. P. Antunes. 2012. Optimization approach to depot location and trip selection in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review* **48**(1) 233–247.

Firnkorn, J., M. Müller. 2011. What will be the environmental effects of new free-floating car-sharing systems? The case of car2go in Ulm. *Ecological Economics* **70**(8) 1519–1528.

Folkestad, C. A., N. Hansen, K. Fagerholt, H. Andersson, G. Pantuso. 2020. Optimal charging and repositioning of electric vehicles in a free-floating carsharing system. *Computers & Operations Research* **113** 104771.

Formentin, S., A. G. Bianchessi, S. M. Savaresi. 2015. On the prediction of future vehicle locations in free-floating car sharing systems. *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 1006–1011.

Fricker, Christine, Nicolas Gast. 2016. Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity. *Euro Journal on Transportation and Logistics* **5**(3) 261–291.

Gambella, C., E. Malaguti, F. Masini, D. Vigo. 2018. Optimizing relocation operations in electric car-sharing. *Omega* **81** 234–245.

Garrick, D. 2016. Car2go ceases San Diego operations. *San Diego Union-Tribune*. URL https://www.sandiegouniontribune.com/news/politics/sd-me-car2go-leave-20161230-story.html.

Haider, Zulqarnain, Alexander Nikolaev, Jee Eun Kang, Changhyun Kwon. 2018. Inventory rebalancing through pricing in public bike sharing systems. *European Journal of Operational Research* **270**(1) 103–117.

He, L., Z. Hu, M. Zhang. 2020. Robust repositioning for vehicle sharing. *Manufacturing & Service Operations Management* **22**(2) 241–256.

He, L., G. Ma, W. Qi, X. Wang. 2019. Charging an electric vehicle-sharing fleet. *Manufacturing and Service Operations Management* (forthcoming).

He, L., H.-Y. Mak, Y. Rong, Z.-J. M. Shen. 2017. Service region design for urban electric vehicle sharing systems. *Manufacturing & Service Operations Management* **19**(2) 309–327.

Herrmann, S., F. Schulte, S. Voß. 2014. Increasing acceptance of free-floating car sharing systems using smart relocation strategies: A survey based study of car2go hamburg. *International Conference on Computational Logistics*. Springer, 151–162.

Jian, S., D. Rey, V. Dixit. 2016. Dynamic optimal vehicle relocation in carshare systems. *Transportation Research Record: Journal of the Transportation Research Board* (2567) 1–9.

Kek, A. G.H., R. L. Cheu, Q. Meng, C. H. Fung. 2009. A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E: Logistics and Transportation Review* **45**(1) 149–158.

Kim, S. W., H.-Y. Mak, M. Olivares, Y. Rong. 2020. Empirical investigation on the range anxiety for electric vehicles. *Working Paper*.

Kypriadis, D., G. Pantziou, C. Konstantopoulos, D. Gavalas. 2018. Minimum walking static repositioning in free-floating electric car-sharing systems. IEEE, 1540–1545.

Laporte, G., F. Meunier, R. W. Calvo. 2015. Shared mobility systems. *4OR* **13**(4) 341–360.

Le Vine, S., J. Polak. 2019. The impact of free-floating carsharing on car ownership: Early-stage findings from london. *Transport Policy* **75** 119–127.

Lim, M. K., H.-Y. Mak, Y. Rong. 2015. Toward mass adoption of electric vehicles: Impact of the range and resale anxieties. *Manufacturing & Service Operations Management* **17**(1) 101–119.

Lin, S. 1965. Computer solutions of the traveling salesman problem. *Bell System Technical Journal* **44**(10) 2245–2269.

Loose, W. 2010. The state of european car-sharing. *Project Momo Final Report D* **2**.

Luo, Zhixing, Mengyang Liu, Andrew Lim. 2019. A two-phase branch-and-price-and-cut for a dial-a-ride problem in patient transportation. *Transportation Science* **53**(1) 113–130.

Martin, E., S. Shaheen, J. Lidicker. 2010. Impact of carsharing on household vehicle holdings: Results from north american shared-use vehicle survey. *Transportation Research Record: Journal of the Transportation Research Board* (2143) 150–158.

Megiddo, N., K. J. Supowit. 1984. On the complexity of some common geometric location problems. *SIAM Journal on Computing* **13**(1) 182–196.

Pelletier, Samuel, Ola Jabali, Gilbert Laporte. 2018. Charge scheduling for electric freight vehicles. *Transportation Research Part B: Methodological* **115** 246–269.

Pfrommer, J., J. Warrington, G. Schildbach, M. Morari. 2014. Dynamic vehicle redistribution and online price incentives in shared mobility systems. *Intelligent Transportation Systems, IEEE Transactions on* **15**(4) 1567–1578.

Plesník, J. 1987. A heuristic for the p-center problems in graphs. *Discrete Applied Mathematics* **17**(3) 263–268.

Psaraftis, H. N. 1983. k-interchange procedures for local search in a precedence-constrained routing problem. *European Journal of Operational Research* **13**(4) 391–402.

Raviv, T., M. Tzur, I. A. Forma. 2013. Static repositioning in a bike-sharing system: Models and solution approaches. *EURO Journal on Transportation and Logistics* **2**(3) 187–229.

Santos, A. G., P. G.L. Cândido, A. F. Balardino, W. Herbawi. 2017. Vehicle relocation problem in free floating carsharing using multiple shuttles. *Evolutionary Computation (CEC), 2017 IEEE Congress on*. IEEE, 2544–2551.

Schulte, F., S. Voß. 2015. Decision support for environmental-friendly vehicle relocations in free-floating car sharing systems: The case of car2go. *Procedia CIRP* **30** 275–280.

Shaheen, S., A. Cohen. 2007. Growth in worldwide carsharing: An international comparison. *Transportation Research Record: Journal of the Transportation Research Board* (1992) 81–89.

Weikl, S., K. Bogenberger. 2013. Relocation strategies and algorithms for free-floating car sharing systems. *IEEE Intelligent Transportation Systems Magazine* **5**(4) 100–111.

Weikl, S., K. Bogenberger. 2015. A practice-ready relocation model for free-floating carsharing systems with electric vehicles–mesoscopic approach and field trial results. *Transportation Research Part C: Emerging Technologies* **57** 206–223.

Appendix for

Optimizing the Relocation Operations of Free-Floating Electric Vehicle Sharing Systems

A Two-Interchange Algorithm for Multiple Precedence Constraints

We present a customised 2-interchange algorithm which draws from a 2-interchange procedure for Dial-A-Ride Problem (DARP) presented in Psaraftis (1983). A DARP involves a vehicle picking up and dropping off multiple customers. A DARP tour, unlike that of a Travelling Salesman Problem (TSP) (Lin, 1965), must satisfy precedence constraints since origin of each customer must precede his/her destination on the route. In our problem, the EVs are equivalent to customers in DARP. Contrary to a customer in DARP, each EV moves multiple times through the nodes on its route. Therefore, a shuttle must satisfy multiple precedence constraints for each EV. We borrow and extend the notation used in Psaraftis (1983).

A given shuttle route r of length $M = \mathrm{length}(r)$ can be described as a sequence $\left(\widehat{\mathcal{S}}_1, \widehat{\mathcal{S}}_2, \ldots, \widehat{\mathcal{S}}_i, \ldots, \widehat{\mathcal{S}}_M\right)$ where i represents the i-th stop of the route and $\widehat{\mathcal{S}}_i$ is defined using the following symbolic values:

$$\widehat{\mathcal{S}}_i = \begin{cases} 0 & \text{if } i = 1 \text{ or } i = M \text{ (Depots)} \\ n^{+} & \text{if shuttle visits supplier node of EV } n \text{ at node } i \\ n^{>} & \text{if shuttle visits charger node of EV } n \text{ at node } i \\ n^{<} & \text{if shuttle visits dummy charger node of EV } n \text{ at node } i \\ n^{-} & \text{if shuttle visits demander node of EV } n \text{ at node } i \end{cases}$$

$\forall \;\; i = 1, 2, \ldots, M$

Alternativelty, the shuttle route can be represented through matrix $[m\,(n,i)]$ where $m\,(n,i)$ is the status of EV n at the i-th stop of the shuttle tour.

$$m\,(n,i) = \begin{cases} 5 & \text{if supplier for EV } n \text{ has not been visited so far.} \\ 4 & \text{if charger for EV } n \text{ has not been visited so far.} \\ 3 & \text{if dummy charger for EV } n \text{ has not been visited so far.} \\ 2 & \text{if demander for EV } n \text{ has not been visited so far.} \\ 1 & \text{if route for EV } n \text{ has been completed.} \end{cases}$$

$(n = \text{EV number}, \; i = 1, 2, \ldots, M)\,.$

Given an initial route r, a 2-interchange swapping algorithm works by interchanging two links in the

Algorithm 4: A customized 2-interchange procedure for finding 2-optimal shuttle route

Input: q; set of EV paths $\mathcal{X}$; Iter = number of 2-interchange iterations
Output: $\mathbf{z}$

Given $\mathcal{X}$, use `GreedyProcedure(`$\mathcal{X}$`)`to obtain an initial route r as an array $r[1], r[2], \cdots, r[\text{end}]$;
for $k \leftarrow 1$ **to** Iter **do**
    $\Delta \leftarrow 0$;
    for $i \leftarrow 1$ **to** $\text{length}(r) - 3$ **do**
        for $j \leftarrow i + 2$ **to** $\text{length}(r) - 1$ **do**
            `// Swapping nodes` $i + 1$ `and` j `and reversing nodes between them`
            $r_{\text{new}} \leftarrow \text{copy}(r)$;
            $r_{\text{new}}[i + 1 : j] \leftarrow r_{\text{new}}[j : i + 1]$;
            if $\neg$`PrecedenceFeasibilityCheck(`r_{new}`)` **then** `// for synchronization`
                continue;
            if $\neg$`CapacityFeasibilityCheck(`r_{new}`)` **then** `// for driver availability`
                continue;
            $\Delta_{ij} \leftarrow$ `RouteImprovement(`r, r_{new}`)`;
            if $\Delta_{ij} > \Delta$ **then**
                $\Delta \leftarrow \Delta_{ij}$, $r_{\text{best}} \leftarrow r_{\text{new}}$;
    $r \leftarrow r_{\text{best}}$;
Given route r, generate a vector $\mathbf{z}$.

route with two other links. For a given route r and a proposed swap (i, j), a new route r_{new} can be constructed by substituting two links. Link $i \rightarrow i + 1$ is substituted with $i \rightarrow j$, while $j \rightarrow j + 1$ is substituted with $i + 1 \rightarrow j + 1$. Since direction of segment $(i + 1 \rightarrow \cdots \rightarrow j)$ is now reversed, it is necessary to check precedence feasibility. The shuttle also picks up and drops off the drivers at each node it visits. Therefore, a proposed 2-interchange must also be feasible in terms of number of drivers on the shuttle. Furthermore, a proposed 2-interchange must also improve (decrease) the length of the shuttle route. Given a route r and a proposed interchange (i, j), we next describe the steps for `PrecedenceFeasibilityCheck(`r_{new}`)`, `CapacityFeasibilityCheck(`r_{new} and `RouteImprovement(`r, r_{new}`)`. Let us also consider a small example to illustrate the steps of the 2-interchange procedure. We consider 4 EVs with the following paths: $11 \rightarrow 2$, $40 \rightarrow 15$, $23 \rightarrow 16 \rightarrow 47 \rightarrow 4$ and $24 \rightarrow 13 \rightarrow 46 \rightarrow 12$. EVs 1 and 2 have direct paths while EVs 3 and 4 visit interdmediate charging stations.

`GreedyProcedure(`$\mathcal{X}$`)`:

Given a set of EV paths, we use the procedure described in Algorithm 5 to construct an initial route r. The initial shuttle route for the example with 4 EV routes is shown in Table A.1.

Algorithm 5: `GreedyProcedure`($\mathcal{X}$): Greedy algorithm for finding initial shuttle route

Input: $\mathcal{X}$ = set of EV routes, q, y = current number of drivers, $\mathcal{V} \leftarrow \{\}$
Output: $r = [r[1], r[2], \cdots, r[\text{end}]]$

$j \leftarrow 1$;
$r[j] \leftarrow 0, y \leftarrow q$;
$j \leftarrow j + 1$;
foreach $x \in \mathcal{X}$ **do**
  while $y \geq 1$ **do**
    $r[j] \leftarrow \{\mathcal{S}(x)\}$;
    $y \leftarrow y - 1$;
    $j \leftarrow j + 1$;
  while $y \leq q - 1$ **do**
    $r[j] \leftarrow \{\mathcal{C}(x)\}$;
    $y \leftarrow y + 1$;
    $j \leftarrow j + 1$;
foreach $x \in \mathcal{X}$ **do**
  while $y \geq 1$ **do**
    $r[j] \leftarrow \{\mathcal{C}^d(x)\}$;
    $y \leftarrow y - 1$;
    $j \leftarrow j + 1$;
  while $y \leq q - 1$ **do**
    $r[j] \leftarrow \{\mathcal{D}(x)\}$;
    $y \leftarrow y + 1$;
    $j \leftarrow j + 1$;
end $\leftarrow j$;
$r[\text{end}] \leftarrow 0$;

`PrecedenceFeasibilityCheck`(r_{new}):

For a given shuttle route to be precedence feasible, the supplier of each EV route must be visited before the charger, which in turn must be visited before the dummy charger and finally the dummy charger must be visited before the demander. Therefore, for each EV n, there are three precedence constraints. We can use the matrix $m(n, i)$ to ensure the feasibility of all three precedence constraints. Given a proposed 2-interchange (i, j), assume that for an EV n, $m(n, i) = 5$ and $m(n, j) = 3$. This implies that the supplier for EV n has not been visited at node i but the charger has already been visited before node j. It follows then that the supplier and charger nodes for EV n lie within the shuttle route segment $(i + 1 \to \cdots \to j)$. Since performing the 2-interchange (i, j) will reverse the direction of traversal on segment $(i+1 \to \cdots \to j)$, the charger node will be visited before the supplier node. This will violate the first precedence constraint for EV n. Similarly, if $m(n, i) = 4$ and $m(n, j) = 2$, the proposed 2-interchange (i, j) will result in the dummy charger node being visited before the charger node of an EV n, violating the second precedence feasibility constraint.

Normally, performing the feasibility check will require $O(N^3)$ time since we must perform $O(N)$ checks for each 2-interchange. However, the computational complexity can be reduced to $O(N^2)$ by performing a customized version of *screening* procedure described in Psaraftis (1983). For a given tour r and a given stop i $(1 \leq i \leq M-2)$, we define FIRSTSTOP(i) to be the position of the first stop (charger) beyond node $(i+1)$ for which the corresponding EV supplier has not been visited including and up to node i. If no such stop exists, FIRSTSTOP(i) $= M$. Similarly, we define SECONDSTOP(i) to be the position of the second stop (dummy charger) beyond node $(i+1)$ for which the corresponding EV charger has not been visited including and up to node i. If no such stop exists, SECONDSTOP(i) $= M$. Finally, we define THIRDSTOP(i) to be the position of the third stop (demander) beyound node $(i+1)$ for which the corresponding EV dummy-charger has not been visited including and up to node i. If no such stop exists, THIRDSTOP(i) $= M$. Mathematically, FIRSTSTOP(i) = h if h is the smallest number above $(i+1)$ for which there exists an EV n so that $m(n,i) = 5$ and $m(n,h) = 3$. If no such EV exists, the $h = M$. Similarly, SECONDSTOP(i) = h if h is the smallest number above $(i+1)$ for which there exists an EV n so that $m(n,i) = 4$ and $m(n,h) = 2$. If no such EV exists, the $h = M$. Finally, THIRDSTOP(i) = h if h is the smallest number above $(i+1)$ for which there exists an EV n so that $m(n,i) = 3$ and $m(n,h) = 1$. If no such EV exists, the $h = M$. Theorem A.1 describes the test for precedence feasibility check for a proposed 2-interchange (i,j).

Theorem A.1. *The exchange* (i,j) *is precedence feasible if and only if* $j < \mathrm{FIRSTSTOP}(i)$ *and* $j < \mathrm{SECONDSTOP}(i)$ *and* $j < \mathrm{THIRDSTOP}(i)$.

Proof. This theorem is similar to the result of (Psaraftis, 1983). The proof is obvious and hence omitted. □

The screening part of the 2-interchange algorithm can be summarized as follows:

Algorithm 6: Screening Procedure for Prercedence Feasibility Check for a Proposed 2-Interchange

```
Input: FIRSTSTOP(i), SECONDSTOP(i), THIRDSTOP(i)
Output: TF(i, j)
for i ← 1 to M − 2 do
    for j ← i + 1 to M do
        if j < FIRSTSTOP(i) and j < SECONDSTOP(i) and j < THIRDSTOP(i) then
            TF(i, j) ← 'true' ;
        else
            TF(i, j) ← 'false' ;
```

The values for $m(n,i)$, $\widehat{\mathcal{S}}_i$, $\mathrm{FIRSTSTOP}(i)$, $\mathrm{SECONDSTOP}(i)$, $\mathrm{THIRDSTOP}(i)$, and the matrix TF(i,j) for the small example are calculated in Table A.1.

We use a depth first procedure to select the best interchange. The TF(i,j) matrix checks for precedence feasibility for all possible (i,j) swaps given an initial route r. The precedence feasible swaps are shown as

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Type	**	S	S	S	S	C	C	DC	DC	D	D	D	D	**
Node, $r[i]$	0	11	40	23	24	16	13	47	46	2	15	4	12	0
$\widehat{\mathcal{S}}_i$	0	1^+	2^+	3^+	4^+	$3^>$	$4^>$	$3^>$	$4^<$	1^-	2^-	3^-	4^-	0
$m(1,i)$	5	4	4	4	4	4	4	4	4	3	3	3	3	3
$m(2,i)$	5	5	4	4	4	4	4	4	4	4	3	3	3	3
$m(3,i)$	5	5	5	4	4	3	3	2	2	2	2	1	1	1
$m(4,i)$	5	5	5	5	4	4	3	3	2	2	2	2	1	1
FIRSTSTOP$[i]$	6	6	6	7	14	14	14	14	14	14	14	14	14	14
SECONDSTOP$[i]$	14	14	14	8	8	9	14	14	14	14	14	14	14	14
THIRDSTOP$[i]$	14	14	14	14	14	12	12	13	14	14	14	14	14	14
i|j			**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
1			T	T	T	F	F	F	F	F	F	F	F	F
2				T	T	F	F	F	F	F	F	F	F	F
3					T	F	F	F	F	F	F	F	F	F
4						T	F	F	F	F	F	F	F	F
5							T	F	F	F	F	F	F	F
6								T	F	F	F	F	F	F
7									T	T	T	F	F	F
8										T	T	T	F	F
9											T	T	T	F
10												T	T	F
11													T	F
12														F

Table A.1 – An Example Illustrating the Screening Procedure for Precedence Feasibility Check

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Node, $r[i]$	0	11	40	23	24	16	13	47	46	2	15	4	12	0
Type	**	S	S	S	S	C	C	DC	DC	D	D	D	D	**
y_i^{old}	4	3	2	1	0	1	2	1	0	1	2	3	4	4
i	1	2	3	4	5	6	7	10	9	8	11	12	13	14
Node, $r_{\text{new}}[i]$	0	11	40	23	24	16	13	2	46	47	15	4	12	0
Type	**	S	S	S	S	C	DC	DC	C	D	D	D	D	**
y_i^{new}	4	3	2	1	0	1	0	-1	0	1	2	3	4	4

Table A.2 – An Example Illustrating the Capacity Feasibility Check for interchange **(7, 10)**

T in the table. We next apply capacity feasibility check and improvement check on this set of routes.

`CapacityFeasibilityCheck`(r_{new}):

The capacity feasibility check is done to ensure that the proposed 2-interchange does not cause the number of drivers to drop below zero or jump above the capacity q. A given shuttle route r begins at the depot with q drivers. A driver is dropped at each supplier and dummy charger node and one is picked from each charger and demander node. Given a proposed 2-interchange (i, j), and the new route r_{new}, the capacity only changes for the route segment $(j \to \cdots \to i + 1)$. Given y_i^{old}, i.e., the number of drivers at node i on the current shuttle route r, one can easily determine y_i^{new}, i.e., the number of drivers at node i for new shuttle route r_{new}. If for a given node i on shuttle route, y_i^{new} falls below 0 or above capacity q, the proposed interchange (i, j) is deemed infeasible. For example, the proposed interchange $(7, 10)$ is capacity infeasible as shown in Table A.2.

`RouteImprovement`(r, r_{new}):

A proposed two interchange (i, j) involves sustitution of two links $(i, i + 1)$ and $(j, j + 1)$ with two new links (i, j) and $(i + 1, j + 1)$. In traditional TSP and DARP problems, an interchange is considered to be improving or favorable if $t_{i,i+1} + t_{j,j+1} > t_{i,j} + t_{i+1,j+1}$. The improvement for an interchange (i, j) can simply be calculated as $t_{i,i+1} + t_{j,j+1} - (t_{i,j} + t_{i+1,j+1})$. The updated arrival times τ_j for the new route can be calculated by iteratively adding the link traversal times, i.e., $\tau_1 = 0$, $\tau_j = \tau_i + t_{ij}, \forall(i, j)$. In case of the synchronized EV relocation and shuttle routing problem presented in this paper, the improvement check is computationally more challenging. The arrival time of a shuttle at a demander node on an EV path $p \in \mathcal{X}$ depends on both the EV path length (l_p) and the shuttle arrival time on corresponding supplier node. This dependency is given by Equations (17) and (18). Therefore for a shuttle moving on link (i, j) the arrival time τ_j at node j is calculated as:

$$\tau_j = \begin{cases} \max(\tau_i + t_{ij}, \tau_l + l_p) & \text{if } \ p \in \mathcal{X}, l = \mathcal{S}(p), j = \mathcal{D}(p) \\ \tau_i + t_{ij} & \text{otherwise.} \end{cases}$$

Since a proposed interchange (i, j) reverses the shuttle traversal on segment $(i + 1 \to \cdots \to j)$, a demander node contained in the segment may have its arrival time changed from $\tau_{i'} + t_{ij}$ to $\tau_{l'} + l_p$ or vice versa, owing to the change in its position. Similarly, a supplier node l' contained in the segment may have its arrival time changed due to change in its position on the route. If the supplier node belongs to path $p \in \mathcal{X}$, this change in τ_l' can also nonlinearly impact the shuttle arrival at the downstream demander node $j' = \mathcal{D}(p)$ and at any nodes after j' on the route. Therefore, improvement after a proposed 2-interchange can only be checked by fully calculating the shuttle arrival times at all nodes using the expression for τ_j given above. Let τ_M^{old} be the total route time for a given route r. Let τ_M^{new} be the total route time for the route r_{new}. The improvement Δ_{ij} can be calculated as: $\Delta_{ij} = \tau_M^{\text{new}} - \tau_M^{\text{old}}$.

B Steps to create K-clusters from K-centers

Once K-centers have been found, we can assign to each center k a set of nodes which form the k-th cluster. The assignment must fulfill certain high level requirements.

1. The number of nodes within each cluster should be approximately equal to form evenly sized clusters.

2. For each cluster, the number of demanders and chargers in the cluster should be greater than or equal to the minimum of the number of suppliers and demanders in the cluster to ensure that each electric vehicle has an available path within the cluster.

Given K clusters, these requirements are equivalent to fulfilling the following conditions:

$$|\mathcal{S}_k| \geq \min\left\{\frac{|\mathcal{S}|}{K}, \frac{|\mathcal{D}|}{K}\right\}, \quad |\mathcal{D}_k| \geq |\mathcal{S}_k|, \text{ and } \quad |\mathcal{C}_k| \geq |\mathcal{S}_k|.$$

The assignment of nodes to clusters is done using an assignment problem that minimizes the total *distance* from all nodes to the assigned center. Let r_{ik} be a binary variable which is 1 if node i is assigned to center k and 0, otherwise. Similarly let $d_{ik} = t_{ik}$ be the measure of *distance* from node i to center k. The node to center assignment problems (NCA) can be written as follows:

$$\text{(NCA)} \quad \text{minimize} \quad \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{K}} d_{ik} r_{ik}, \tag{B.1}$$

$$\text{subject to} \quad \sum_{k \in \mathcal{K}} r_{ik} = 1 \qquad \forall i \in \mathcal{N}, \tag{B.2}$$

$$\sum_{i \in \mathcal{S}} r_{ik} \geq \min\left\{\frac{|\mathcal{S}|}{K}, \frac{|\mathcal{D}|}{K}\right\} \qquad \forall k \in \mathcal{K}, \tag{B.3}$$

$$\sum_{i \in \mathcal{D}} r_{ik} \geq \sum_{i \in \mathcal{S}} r_{ik} \qquad \forall k \in \mathcal{K}, \tag{B.4}$$

$$\sum_{i \in \mathcal{C}} r_{ik} \geq \sum_{i \in \mathcal{S}} r_{ik} \qquad \forall k \in \mathcal{K}, \tag{B.5}$$

$$r_{ik} \in \{0, 1\} \qquad \forall i \in \mathcal{N}, k \in \mathcal{K} \tag{B.6}$$

C Additional Figures

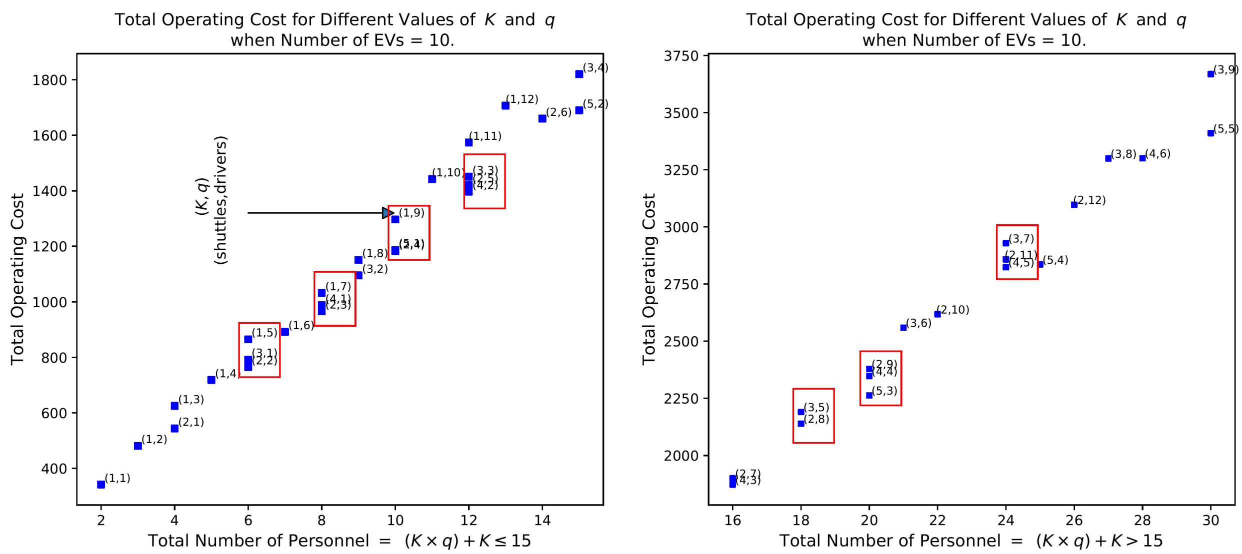


Figure C.1 – For a Small Network with 10 EVs and given a certain number of personnel and per unit costs, it is advantageous to increase the number of drivers