Free Rides in Dockless, Electric Vehicle Sharing Systems

Bobby S. Nyotta, Fernanda Bravo
UCLA Anderson School of Management, bobby.nyotta.1@anderson.ucla.edu, fernanda.bravo@anderson.ucla.edu

Jacob Feldman
Olin School of Business, jbfeldman@wustl.edu

We study free-ride policies as a mechanism to incentivize users of a "dockless" or "free-floating" electric vehicle sharing system (EVSS) to park vehicles at charging stations in order to maintain a charged fleet. A balanced system has a fleet that is adequately charged and evenly dispersed throughout the city. If left to unfold naturally, the system would fall out of balance, and revenue and customer experience might suffer. Most sharing systems use manual repositioning to achieve this balance, but we consider pricing incentives as an alternative method. We develop an infinite horizon dynamic program to analyze free-ride policies. We focus on an EVSS that offers free rides to customers if they return vehicles to charging stations. We build on this initial formulation to construct a mixed-integer program that outputs intuitive, battery-threshold rules for when to offer free rides. We also extend the model to accommodate more general discount-based policies. In a discrete-event simulation model using real data from an EVSS, we compare the performance of this simple policy against other sophisticated policies, including the commonly used fine-based policy. We first find that the simple threshold-based policy performs close to a more sophisticated, black-box policy in terms of revenue. We also discover that the free-ride policies generate customer utilities that are ten times higher than fine-based policies, but also generate less revenue. However, free-ride policies can be less costly to implement since they rely on manual repositioning up to 65-75% less than the benchmarking policies. Our simulation reveals this three-dimensional trade-off between customer satisfaction, revenue, and operational complexity. Furthermore, we find that the cost of repositioning and the customer heterogeneity in the likelihood to accept a discount are major drivers of the frequency of free-ride offers. Our results are robust under many demand patterns and under a variety of network settings.

Key words : vehicle sharing; electric vehicles; dynamic programming; revenue management

1. Introduction

In "dockless" or "free-floating," electric vehicle sharing systems (EVSSs), a fleet of electric rechargeable vehicles (i.e. cars, vespas, bicycles, or scooters) are scattered throughout a city. Users of the system can rent or check-out these vehicles for a small fee that is generally proportional to trip time or distance. When the user is finished with the vehicle, she can park it in any legal parking spot throughout the city. Included among these parking zones are charging stations, where the vehicle can be plugged in to regain charge. Users who end their trips at charging stations help the system, since as a vehicle charges, it gains the potential to serve a broader class of trips. One potential mechanism to incentivize users to end their rides at charging stations is to offer them a discount on their ride if they do so. In such a scenario, the system operator forgoes immediate revenue to ensure that the vehicle at hand is sufficiently charged for future rides. There is no guarantee, however, that the user will agree to take this discounted ride, since the proposed charging station could be far away from the user's desired end location. In this way, when the user decides whether or not to accept the offered discount, she trades off the price reduction with the potential inconvenience of concluding her ride far away from her desired end location.

In this paper, we study the trade-offs described above by considering how a system operator of a dockless EVSS can optimally offer discounted rides to incentivize users to end their trips at a charging station. The models that we develop are able to incorporate a variety of discount structures, but we focus our analysis mainly on free-ride policies in which users are strategically given the option to take a free trip to a charging station in lieu of a full-priced ride to their desired destination. This simplified discount structure allows us to develop implementable policies, which we show via extensive simulations perform quite well on a wide variety of performance metrics related to revenue, operational costs and customer satisfaction.

This work builds on an extensive body of research on vehicle sharing systems (VSSs). The increase in VSS-related literature can be attributed to the rapid expansion of such systems and the intriguing operational challenges that accompany this growth. In what follows, we briefly describe the rise of VSSs before detailing the classical operational problems that accompany such systems and that motivate our work on discount rides in EVSSs.

The make-up and growth of VSSs. The first VSSs were comprised entirely of gasoline-powered vehicles, which are still present in many VSSs, i.e. car2go and Zipcar. Soon after, bike sharing systems (BSSs), i.e. CitiBike, were introduced to handle shorter trips, and most recently, cities have witnessed the adoption of EVSSs, i.e. Bird Scooters, which have documented environmental and financial benefits over gasoline-powered vehicles (U.S. Department of Energy 2016). The gain in popularity of all three VSSs is without question, as membership is slated to exceed 12 million by 2020 and revenue is projected to reach $6.5 billion in 2024 (Navigant Research 2016b,a). As these systems continue to grow, so too does the necessity to develop efficient solutions to the many daily operational challenges.

Operational problems in VSSs. The primary operational problem in VSSs revolves around balancing supply with demand. In a perfect system, there would always be an available vehicle in close proximity to every inquiring user. In practice, achieving this service level is nearly impossible due to the stochasticity in demand and the limited number of vehicles. In fact, if left to run entirely on its own, most VSSs would inevitably experience an extremely lopsided dispersion of vehicles because supply and demand rarely match up perfectly. To combat this issue, VSSs have resorted to two main operational levers: rebalancing and pricing. The former refers to manually moving vehicles between stations in anticipation of future demand, which is effective but costly (Fishman et al. 2014).

Later generation VSSs became more sophisticated and started to offer dockless parking, which allows riders to park on any street in a pre-defined service region. While dockless systems provide users with more convenience and flexibility in terms of where users are permitted to finish rides, they also bring new flavors of operational problems. For one, merely keeping track of each vehicle's location is a more complex task in free-float systems since the pre-defined parking regions generally span the entire city. This is in stark contrast to traditional docked systems, in which the system's state can simply be described by the number of vehicle at each of the docking stations. This inherent difference significantly complicates the aforementioned rebalancing problem; manual rebalancing in dockless systems is a more tedious task since vehicles are not confined to docking stations. With this in mind, many of the dockless VSSs have flipped the rebalancing problem on its head; instead of manually rebalancing the system themselves, they attempt to incentivize users to accomplish this task for them. For example, LimeBike offers ride credits to users who check-out bikes that have sat idle for an extended period of time.

For the dockless EVSS that we consider, the system's state is characterized by the current location and charge level of each vehicle. In this setting, a balanced system not only has vehicles in close proximity to inquiring users, but it also ensures that these vehicles are adequately charged. In what follows, we describe how EVSS's have attempted to achieve this latter, more elementary notion of a balanced network.

Maintaining a charged fleet in EVSSs. For dockless EVSSs, there is perhaps an even more fundamental issue than that of balancing supply and demand. Paramount to operational efficiency and the profitability of such systems is the ability to keep the electric vehicle fleet adequately charged so that users do not forgo a ride because they cannot find a vehicle with enough charge. In some existing dockless EVSSs, the current practice is to aggressively fine riders who street-park vehicles with low battery. This approach, however, does not appear to be ideal since it often results in users choosing between the lesser of two evils when they unexpectedly finish a ride with a low-battery vehicle. In such scenarios, users are forced to either navigate out of their way to drop a low-battery vehicle at an open charging station or to park near their desired destination and pay a hefty fine, both of which negatively impact the user experience. Moreover, it is not obvious how exactly to choose this aforementioned "low-battery" threshold, which will have a dramatic effect on the day-to-day dynamics of the system. Choosing this threshold to be too low may result in many vehicles stranded on the street because they do not have enough battery to fulfill any rides. On the other hand, a threshold that is too high may lead to an overwhelming number of fines and hence an unhappy and frustrated customer base. In contrast, the pricing discount incentives that we consider have the potential both to keep the fleet charged while only improving the user experience, since any offered discounted ride can be turned down.

We consider a dockless EVSS consisting of n vehicles and m charging stations. The vehicles in our setting should be thought of as cars or Vespas, and so manually moving these vehicles is quite tedious and costly in relation to moving bikes. At any given time, the state of each vehicle can be described by its location and its current charge level. As time progresses, users rent vehicles and ride them to their desired location. Our goal is to develop and characterize simple conditions under which a dockless EVSS operator should offer free rides to users who end their trip at a charging station. The hope is that these free rides will ensure that the system has sufficiently charged vehicles to serve future demand.

As hinted at above, we seek free-ride policies that are straightforward for the system operator to explain to the user, easy for the user to interpret, and simple to put into action. All three of these characteristics are satisfied by what we call **single-offer range** (SOR) policies. Under such policies, for each region in the network, there is a single, continuous, battery charge level interval that dictates when a free-ride will be offered. Upon rental, users who select a vehicle whose charge level falls within this interval will be offered a free ride, and those who select a vehicle with a charge level outside of this interval will not be offered a free ride. As we go on to show, these region-specific battery charge level intervals can be computed offline in an efficient manner, and hence implementing these so-called SOR policies is relatively straightforward compared to a nuanced dynamic pricing scheme.

1.1. Contributions

We first consider an infinite horizon dynamic program that maximizes the total discounted expected revenue when there are no restrictions on the structure of an optimal policy. The state space of the resulting dynamic program gives the current location and battery level of each vehicle, and the Bellman recursion encodes the trade-off between offering a free ride to a charging station and letting the user take a full-priced ride to her desired end location. This initial formulation has two central issues that hinder its practical use. First, the state space grows exponentially in n and hence the dynamic program is rendered intractable for realistic instances in which the EVSS system contains hundreds of vehicles. Second, even if we could derive an optimal policy from this dynamic program, there is no guarantee that it will be an easily implementable policy, let alone an SOR policy. In fact, it is not clear if it is possible to formulate this dynamic program so that only SOR policies are feasible.

To side-step the first issue, we focus on single-vehicle networks. In this setting, we can easily find the optimal policy, but again, there is no guarantee that this policy will be of the SOR variety. One somewhat counter-intuitive insight that we derive from this simplified one vehicle setting is that the value of a vehicle in a given location does not necessarily increase with its battery level. In Section 3, we describe how this observation is directly related to the second issue of deriving SOR policies. We eventually overcome this second difficulty by reformulating the initial infinite horizon dynamic program as a linear program (LP). By adding binary variables and a set of auxiliary constraints to this LP, we arrive at a mixed-integer program whose optimal solution gives the optimal SOR policy.

We then consider the problem at full-scale, where there are n vehicles and m charging stations in the network. In this setting, we develop free-ride policies based on the approximate dynamic programming technique developed in Desai et al. (2012) in which the value functions in the original dynamic programming formulation of the problem are approximated via a linear combination of basis functions. The optimal weights on each basis function within the approximation are generated from the optimal solution to specially crafted LP.

After deriving the optimal SOR policies from the single vehicle formulation of the problem and the approximate multi-vehicle policies that result from our approximate dynamic programming solution, we test their efficacy within a large scale simulation that uses real data from a U.S.-based EVSS. We benchmark the performance of these free-ride policies against our EVSS partner's current practice, in which users who street-park low battery vehicles are fined. Our simulation results reveal that the free-ride policies generate slightly less revenue than the fine-based policy, but provide a significantly better customer experience, which is critical for the long term success of the system.

2. Literature Review

We begin by reviewing the previous work on BSSs and VSSs, which both pre-date EVSSs. Then, we summarize the past work on EVSSs, which is limited since these systems have only recently come into existence.

Bike Sharing Systems (BSSs). Past BSS research has predominantly focused on network design. For instance, Lin and Yang (2011) determine where to build stations to maximize coverage, Sayarshad et al. (2012) examine how fleet size affects demand, utilization, and rebalancing costs, and Kabra et al. (2018) study the effect of increasing station density on ridership. Freund et al. (2017) develop a procedure to optimally redistribute bicycle docks across stations. All of these papers consider one-way BSSs, where riders can take bikes on one-way trips, which must end at a docking station. Our setting is less restrictive since users can take one-way trips, but they are not forced to end at a docking station.

Rebalancing in one-way BSSs has also been well-researched. This work involves efficiently designing truck routes that minimize the time and cost of moving bikes between docking stations (Raviv et al. 2013, O'Mahony and Shmoys 2015, Schuijbroek et al. 2017). Pricing has also been studied as a mechanism to rebalance a BSS. Chemla et al. (2013) propose a pricing strategy in which the fare is based on the availability at each station. Others have focused on minimizing underused stations by incentivizing riders to return bikes to these stations (Pfrommer et al. 2014, Singla et al. 2015, Fricker and Gast 2014).

While the existing BSS research can serve as a starting point for tackling operational problmes in EVSSs, there are two features of our problem that have not yet been considered in the BSS literature. First, to the best of our knowledge, the dockless component has not been studied. Second, the charging element of EVSSs presents a new challenge that does not exist in BSSs, since bikes are human-powered. However, as BSSs grow to include electric-assisted bicycles, maintaining a charged fleet will require attention and we hope that work in this area will draw inspiration from our research.

Car or Vehicle Sharing Systems (VSSs). While the VSS literature is not as vast, problems related to both system design (Chang et al. 2017, Lu et al. 2017) and rebalancing (Nair and Miller-Hooks 2011, Weikl and Bogenberger 2013) have been studied. Rebalancing in a BSS is inherently simpler than a VSS, since several bicycles can be placed on a truck and manually redistributed across a city on a single route. The same can obviously not be said for cars, so VSS rebalancing requires additional planning. We note that the existing rebalancing approaches tend to be costly, resource-intense, and time-consuming.

Several researchers have also explored how to rebalance a VSS through pricing discounts. Marecek et al. (2016) propose a destination-based pricing scheme in dockless VSSs, in which the fare depends on the distance between the drop-off location and the nearest parked vehicle, and Waserhole and Jost (2016) develop a queuing model for setting prices in a one-way VSS to maximize the number of trips. While both papers capture the spirit of the pricing policies that we analyze, neither of these models accounts for a vehicle's remaining battery, which is critical in an EVSS. Banerjee et al. (2016) provide a general framework for pricing in any mobility sharing system, but it is not obvious if their approach is able to capture the additional complexity of keeping the fleet sufficiently charged. For an overview of system design and rebalancing in VSSs, see Jorge and Correia (2013).

Electric Vehicle Sharing Systems (EVSSs). To date, the EVSS literature has primarily focused on system design. Boyacı et al. (2015) and Brandstatter et al. (2017) respectively study where to place charging stations in one-way systems and parking locations in dockless systems. In the presence of uncertain adoption, He et al. (2017) use robust optimization to define the service area for car2go's dockless EVSS operation in San Diego, CA. Unfortunately, car2go replaced the electric vehicles with gas-powered vehicles and later ceased their San Diego service, confirming that operating a dockless EVSS is challenging (The San Diego Union-Tribune 2016). For rebalancing an EVSS, Bruglieri et al. (2014) consider how to dispatch cyclists on folding bikes to low-battery electric vehicles. Upon reaching the vehicle, the cyclist places the collapsed bike in the trunk and drives to a relocation point. In contrast, we focus on ensuring that the EVSS fleet has enough battery to complete rides by offering a direct pricing discount to customers if they end rides at charging stations.

Related Research. Lim et al. (2014) examine the behavioral factors behind electric vehicle adoption. Battery swapping, where users can go to stations to exchange depleted batteries for recharged ones, has also been studied (Avci et al. 2014, Mak et al. 2013). Separately, Bellos et al. (2017) study how VSSs operated by auto manufacturers affects the firm's profit and decision to design more fuel efficient vehicles. This recent research related to car sharing and electric vehicle usage suggests that both will continue to grow, motivating our goal of effectively managing a system at the intersection of VSSs and EVs.

3. Dockless EVSS Models and Free-Ride Policies

We begin by describing our model of the EVSS and then move to detailing our approaches for deriving the free-ride policies discussed above. More specifically, in Section 3.1 we describe our model of the EVSS that we consider as well as its underlying dynamics that govern how the system evolves over time. The model that we develop is highly realistic and accounts for battery recharging of idle vehicles at charging stations, uncertainty rooted in demand, manual repositioning movements by the system operator, and the utility trade-off faced by customers who must choose to accept or decline a free ride. In Sections 3.2 and 3.3, we analyze a single-vehicle setting and develop a mixed-integer program to find the optimal SOR policies in this setting. Finally, in Section 3.4, we summarize our approximate dynamic programming approach for tackling the multi-vehicle problem.

3.1. Model Primitives

We partition the service area into r regions indexed by the set $\mathcal{R} = \{1, \dots, r\}$. Each region $i \in \mathcal{R}$ could represent a street, block or neighborhood depending on the desired granularity of the system. There is a subset $\mathcal{Z} \subset \mathcal{R}$ of regions that house a charging station. The system consists of n vehicles. We assume that a vehicle's state can be fully described by the tuple (i, w) where $i \in \mathcal{R}$ and $w \in \mathcal{W} = \{0, \delta, 2\delta, \dots, 1\}$ respectively represent the vehicle's current location and battery charge level. We use the convention that $w = 1$ corresponds to a full charge, and $\delta \in [0, 1]$ gives the granularity at which we keep track of battery charge.

We discretize time into disjoint time periods, whose length can be interpreted as the mean time between customer arrivals to the system. In each period, we assume that there is exactly one ride request. We let λ_i be the probability of seeing a vehicle request in region $i \in \mathcal{R}$ during each time period. Given a request for a vehicle in region i, we let p_{ij} be the probability that the user's desired end location is region $j \in \mathcal{R}$. We use $b_{ij} \in \mathcal{W}$ to denote the battery consumption of a trip from region i to region j. Further, we let d_{ij}, f_{ij} and t_{ij} respectively be the distance, fare collected, and duration for rides between regions i and j. A ride can only take place if the requested vehicle has sufficient battery to deliver the user to her desired destination. Thus, we let the set $\mathcal{R}(i,w)=\{j\in\mathcal{R}: w\geq b_{ij}\}$ give all reachable regions of a vehicle whose state is (i,w). Finally, we assume that vehicles located at charging stations gain $\gamma\in\mathcal{W}$ charge in each time period.

Next, we discuss how we incorporate manual vehicle repositioning events by the system operator into our model. Each vehicle is deemed eligible for a manual move to a nearby charging station if its remaining battery is below a predefined battery move threshold b_m. In each time period during which there exists at least one move-eligible vehicle, we assume that a repositioning event is initiated with probability p_m. We model a repositioning event as a "dummy" ride, in which a move-eligible vehicle is uniformly selected to be moved to the closest charging station over a random duration of t_m time periods. The dummy ride reflects the efforts of a crew member and hence comes at a cost of c_m to the system.

For each user that rents a vehicle, the system operator has the option to offer a free ride to a charging station if the vehicle has enough battery to reach at least one charging station. If a free ride is offered, the user decides whether to accept or reject this free ride based on her realized utility for each of these two options. To formalize this notion, we let $u(d,f)$ be the random utility that a user associates with a ride that leaves her a distance of d from her desired location and whose cost is f. The randomness in the utility arises due to the assumption that there is heterogeneity in each user's sensitivity to price and walking distance. We refer the reader to Section 4.1 for the explicit form of the utility function that we use in our simulations. If a free ride is offered and accepted, we assume that the user parks the vehicle at the charging station closest to her desired destination. Finally, we define $\mathbb{P}(Accept_{ijz})=\mathbb{P}\big[u(d_{zj},0)\geq u(0,f_{ij})\big]$ to be the probability that the user accepts a free ride to charging station z. If the user accepts the free ride, she pays nothing and must walk a distance of d_{zj} after dropping off the vehicle at z. On the other hand, if she rejects the offer, she commutes directly to j and pays f_{ij}, which occurs with probability $\mathbb{P}(Decline_{ijz})=1-\mathbb{P}(Accept_{ijz})$. All of the notation introduced above is summarized in Table A.1.

3.2. Single Vehicle and Single Charging Station (1V1C)

We begin by studying a setting with a single vehicle and a single charging station, so $n=1$ and $\mathcal{Z}=\{z\}$. We model the system's dynamics as a discrete time, infinite horizon dynamic program. The state space is given by the tuples $(i,w)\in\mathcal{R}\times\mathcal{W}$, which represent the possible locations and battery levels of the vehicle. The value function $V(i,w)$ gives the maximum total discounted expected revenue that can be derived from a vehicle in state (i,w). The per-period discount factor is $\beta \in (0,1)$ and we define $\beta_{ij} = \beta^{t_{ij}}$ to be the discount rate for a ride between regions i and j, which takes place over t_{ij} periods.

Recall that in each time period, there is a customer arrival at region i with probability λ_i, and this request is for a ride to region j with probability p_{ij}. If the vehicle has enough battery to reach the destination, that is $w \geq b_{ij}$, then the inquiring user will rent the vehicle. Otherwise, the user leaves the system and the vehicle remains at state (i,w). Further, if the vehicle has enough battery to reach the charging station z, i.e., $w \geq b_{iz}$, then the system operator must choose whether or not to offer a free ride to z. Finally, we note that if the vehicle's remaining battery satisfies $w \leq b_m$, then the vehicle is manually repositioned to the charging station with probability p_m. In what follows, we present the value functions of our dynamic program for the cases in which the vehicle is not at the charging station (i.e., $i \neq z$) and has enough battery to reach the charging station (i.e. $w \geq b_{iz}$). Thus, the recursion in (1) illustrates the cases in which $w \geq \max\{b_{iz}, b_m\}$, and the recursion in (2) corresponds to the case in which $b_m \geq w \geq b_{iz}$. The remaining cases are presented in Appendix B.

$$\begin{aligned}
V(i,w) = \max\Big\{ & \underbrace{\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij} + \beta_{ij}V(j,w-b_{ij})\big)}_{DoNotOffer}, \\
& \underbrace{\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\Big(\mathbb{P}(Decline_{ijz})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \mathbb{P}(Accept_{ijz})\cdot\beta_{iz}V(z,w-b_{iz})\Big)}_{Offer}\Big\} \\
& + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w).
\end{aligned} \tag{1}$$

$$\begin{aligned}
V(i,w) = & p_m\cdot\underbrace{\big(-c_m+\beta_m V(z,w)\big)}_{MoveOccurs} + (1-p_m)\cdot\Bigg(\max\Big\{\underbrace{\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)}_{DoNotOffer}, \\
& \underbrace{\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\Big(\mathbb{P}(Decline_{ijz})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \mathbb{P}(Accept_{ijz})\cdot\beta_{iz}V(z,w-b_{iz})\Big)}_{Offer}\Big\} \\
& + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w)\Bigg).
\end{aligned} \tag{2}$$

The maximization in (1) weighs the trade-off between offering and not offering a free ride, which is only relevant when the user's desired end location is reachable, i.e., $j \in \mathcal{R}(i,w)$.

If the vehicle does not have enough battery to reach region j, then the system stays in the same state, but the value of the vehicle is discounted one period. The $DoNotOffer$ term corresponds to the scenario in which the system operator does not offer a free ride. In this case, the system accrues f_{ij} in revenue and the vehicle moves to j, ending this trip in t_{ij} periods (hence the discount factor β_{ij}) with a charge of $w - b_{ij}$. The $Offer$ term captures the value of offering a free ride and the recursion considers the probability that this offer will be accepted by the user. If the offer is accepted, the vehicle reaches the charging station z in t_{iz} periods with $w - b_{iz}$ remaining battery. If the offer is declined, then the user rides to her desired destination and pays the full fare.

Equation (2) models a scenario in which the vehicle is in state (i, w) and has enough battery to complete short trips, but is still eligible for a manual reposition to the charging station. The structure of (2) is similar to (1), but the $MoveOccurs$ term accounts for the possibility of a manual repositioning event, which occurs with probability p_m. If a manual repositioning event occurs, the system incurs a cost of c_m and the vehicle is moved to the charging station z in t_m periods. If the vehicle is not moved, then the value function takes the form of (1). We note that the formulations in (1) and (2) can be modified to incorporate more flexible discounts, in addition to or in lieu of the free-ride discounts. For instance, offering a $(1-\sigma)$-discount for some $\sigma \in [0,1]$ is possible by adding an additional term into the maximization. This term would have the same structure as the $Offer$ term, but the system would realize a revenue of σf_{ij} and the utility gained from accepting the $(1-\sigma)$-discounted ride would be $u(d_{z_j j}, \sigma f_{ij})$. This generalization would allow the system operator to offer a full-fare ride, a $(1-\sigma)$-discounted ride, or a free ride.

Free-ride policies. For free-ride policy $\pi : \mathcal{R} \times \mathcal{W} \mapsto \{DoNotOffer,\ Offer\}$, we define $S_\pi = \{(i,w) : \pi(i,w) = Offer\}$ to be the set of states in which a free-ride is offered . A free-ride policy π is a single-offer range (SOR) policy if for each region i, there exists battery charge levels $w_2^i \geq w_1^i \geq b_{iz}$, such that if the vehicle is in state (i,w), then $\pi(i,w) = Offer$ if and only if $w \in [w_1^i, w_2^i]$. We let Π and $\Pi_{\text{SOR}} \subset \Pi$ respectively denote the set of all free-ride policies and all SOR policies. Further, let $\pi^* \in \Pi$ be the optimal free-ride policy, which can easily be derived via value function iteration since the dynamic program given in (1)-(2) has only $r \cdot |\mathcal{W}|$ states, however there is no guarantee that $\pi^* \in \Pi_{\text{SOR}}$.

Next, we present conditions that would guarantee that $\pi^* \in \Pi_{\text{SOR}}$. At first glance, these conditions seem to be trivially satisfied for any reasonable network, however we are able to present simple counter-example to break this intuition. First, note that by re-arranging the $DoNotOffer$ and $Offer$ terms in (1) and (2), we see that $\pi^*(i,w) = Offer$ if

$$\sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\mathbb{P}(Accept_{ijz})\cdot\left(\beta_{iz}V(z,w-b_{iz})-\beta_{ij}V(j,w-b_{ij})-f_{ij}\right)\geq 0, \tag{3}$$

and $w \geq b_{iz}$. Hence an SOR policy will be optimal if $\beta_{iz}V(z,w-b_{iz}) \geq f_{ij}+\beta_{ij}V(j,w-b_{ij})$ for a single continuous battery charge level interval. A sufficient pair of conditions for this to hold are (i) the value functions $V(i,w)$ are increasing in the battery level w and (ii) the marginal value of each percentage of charge is larger at charging stations than at standard regions. With respect to (i), it seems intuitive that a vehicle with more charge should be able to generate more revenue than a vehicle with less charge, since vehicles with more charge can serve a broader collection of ride requests. However, we quickly discovered that it is not difficult to construct a system in which more battery is not always beneficial. An example of such a network is provided in Appendix A of the Online Appendix. Consequently, (3) can be satisfied for several, disjoint battery ranges and hence π^* is not guaranteed to be an SOR policy. In what follows, we show how to use the above dynamic program to obtain optimal SOR policies in a setting with a single vehicle and multiple charging stations.

3.3. Single Vehicle and Multiple Charging Stations (*1VMC*)

In this section, we consider an EVSS with a single vehicle and m charging stations indexed by the set $\mathcal{Z} = \{1,\dots,m\}$. We assume that if a user whose desired destination is region j accepts a free ride, then she will only park her vehicle at the charging station closest to her destination, which we define as $z_j = \arg\min_{z\in\mathcal{Z}} d_{jz}$. We define $\bar{b}_i = \min_{z\in\mathcal{Z}} b_{iz}$ as the minimum battery level required to reach a charging station from region i, and note that the system operator will only consider offering a free ride if the vehicle's battery level satisfies $w \geq \bar{b}_i$, so at least one charging station can be reached. Since we assume that each user's destination is unknown to the system, it is possible for a free ride to be offered to a user who cannot reach the charging station closest to her desired location. In the case, we assume that the user will reject the free ride.

To find the optimal SOR policy, we first consider the equivalent linear programming of the dynamic program for the 1VMC instance, which is provided in Appendix B. For simplicity, in the LP that follows, we only include the constraints for the cases in which $i \notin Z$ and $w \geq b_m$, but note that the analysis holds when the constraints are added for all cases. Let

$$V^{no}(i,w) = \lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w)$$

and

$$\begin{aligned} V^{o}(i,w) =& \lambda_i \sum_{\substack{j\in\mathcal{R}(i,w):\\ w\geq b_{iz_j}}} p_{ij}\cdot\Big(\mathbb{P}(Decline_{ijz_j})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \mathbb{P}(Accept_{ijz_j})\cdot\beta_{iz_j}V(z_j,w-b_{iz_j})\Big) \\ &+ \lambda_i \sum_{\substack{j\in\mathcal{R}(i,w):\\ w< b_{iz_j}}} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w), \end{aligned}$$

where these two expesions respectively correspond to the case in which the system operator does not and does offer a free ride. The linear program of interest in given below

$$\begin{aligned} Z^* = \min_{V(\cdot)} & \sum_{i\in\mathcal{R}}\sum_{w\in\mathcal{W}} V(i,w) && (LP\ Full) \\ & V(i,w) \geq V^{o}(i,w) && \forall i\notin\mathcal{Z}, w\geq\max\{b_m,\bar{b}_i\} \\ & V(i,w) \geq V^{no}(i,w) && \forall i\notin\mathcal{Z}, w\geq\max\{b_m,\bar{b}_i\} \\ & V(i,w) = \lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w) && \forall i\notin\mathcal{Z}, b_m\leq w<\bar{b}_i. \end{aligned}$$

From the optimal solution to *LP Full*, which we denote as $V^*(i,w)$, we can find the optimal free-ride policy by setting $\pi^*(i,w) = Offer$ if $V^*(i,w) = V^{o*}(i,w)$, and hence the set of states where the system operator should offer a free ride is $S_{\pi^*} = \{(i,w) : V^*(i,w) = V^{o*}(i,w)\}$. In fact, for any free-ride policy $\pi\in\Pi$, we can find the total discounted expected revenue from a vehicle in region i with battery level w by solving the following LP

$$\begin{aligned} Z(\pi) = \min_{V(\cdot)} & \sum_{i\in\mathcal{R}}\sum_{w\in\mathcal{W}} V(i,w) && (LP\ Policy) \\ & V(i,w) = \mathbb{1}_{(i,w)\in S_\pi} V^{o}(i,w) + \mathbb{1}_{(i,w)\notin S_\pi} V^{no}(i,w) && \forall i\notin\mathcal{Z}, w\geq\max\{b_m,\bar{b}_i\} \\ & V(i,w) = \lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) + \Big(1-\lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\Big)\cdot\beta V(i,w) && \forall i\notin\mathcal{Z}, b_m\leq w<\bar{b}_i. \end{aligned}$$

Building on the above two LPs, we can find the optimal SOR free-ride policy by solving the following mixed-integer linear program

$$\begin{aligned} \tilde{Z} = \max & \sum_{i\in\mathcal{R}}\sum_{w\in\mathcal{W}} V(i,w) && (Single\ Threshold) \\ & V(i,w) \leq V^{o}(i,w) + Mx^{no}(i,w) && \forall i\notin\mathcal{Z}, w\geq\max\{b_m,\bar{b}_i\} \\ & V(i,w) \leq V^{no}(i,w) + Mx^{o}(i,w) && \forall i\notin\mathcal{Z}, w\geq\max\{b_m,\bar{b}_i\} \\ & V(i,w) = \lambda_i \sum_{j\in\mathcal{R}(i,w)} p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big) \end{aligned}$$

$$+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,w) \qquad \forall i\notin\mathcal{Z}, b_m\le w<\bar{b}_i$$

$$x^o(i,w)+x^{no}(i,w)=1 \qquad \forall i\notin\mathcal{Z}, w\ge\max\{b_m,\bar{b}_i\}$$

$$x^o(i,w-\delta)\ge x^o(i,w)-y(i,w) \qquad \forall i\notin\mathcal{Z}, w-\delta\ge\bar{b}_i \tag{4}$$

$$\sum_{w\in\mathcal{W}}y(i,w)+x^o(i,\bar{b}_i)\le 1 \qquad \forall i\notin\mathcal{Z} \tag{5}$$

$$x^o(i,w),\ x^{no}(i,w),\ y(i,w)\in\{0,1\}\ \forall i\notin\mathcal{Z},\ w\ge\max\{b_m,\bar{b}_i\},$$

where M is a large constant. The binary decision variables $x^o(i,w)$ and $x^{no}(i,w)$ respectively denote whether or not a free ride is offered when the vehicle is in state (i,w). The binary variable $y(i,w)$ denotes whether or not battery w is the lower threshold battery level w_1^i for region i. The upper threshold w_2^i is equal to 1 if $x^o(i,1)=1$, otherwise it is equal to smallest w such that $x^o(i,w)-x^o(i,w+\delta)=1$. Furthermore, we call attention to two fundamental changes in *Single Threshold* in relation to *LP Full*: (1) the objective is a maximization and (2) the sign of the inequalities in the constraints is reversed. In *Single Threshold*, the "big-M" terms in the first two constraints force the one of the two right-hand sides of these constraints to be quite large, effectively making this constraint irrelevant. The constraint without the large right-hand side is the binding constraint and corresponds to the optimal action that maximizes the expected discounted reward.

In the optimal solution to *Single Threshold*, we refer to the optimal, binary decision variables as $\tilde{x}^o(i,w)$, $\tilde{x}^{no}(i,w)$ and $\tilde{y}(i,w)$. Based on this optimal solution, we define the free-ride policy $\tilde{\pi}(i,w)=Offer$ if $\tilde{x}^o(i,w)=1$, and therefore $S_{\tilde{\pi}}=\{(i,w):\tilde{x}^o(i,w)=1\}$. In the following proposition, we show that $\tilde{\pi}\in\Pi_{\text{SOR}}$. All proofs are provided in Appendix B of the Online Appendix.

PROPOSITION 1. *Let $\tilde{\pi}$ be the free-ride policy derived from Single Threshold. We have that $\tilde{\pi}\in\Pi_{\text{SOR}}$.*

Next, we build on Proposition 1 and show that $\tilde{\pi}$ is actually the optimal SOR policy.

THEOREM 1. *The policy $\tilde{\pi}$ that is derived from the optimal solution to Single Threshold is the optimal SOR free-ride policy. In other words, $\tilde{\pi}=\arg\max_{\pi\in\Pi_{\text{SOR}}}Z(\pi)$.*

Theorem 1 shows that we can recover the optimal SOR policy by solving *Single Threshold* and constructing $\tilde{\pi}$.

3.4. Multiple Vehicle and Multiple Charging Stations (NVMC)

For the multiple vehicle setting, our dynamic program must keep track of the location and battery level of every vehicle in the network. Further, since each ride can last multiple time periods, we also must account for vehicles that are currently in use. More formally, the state of each vehicle can be represented by the 4-tuple (i, w, τ, j), where $i \in \mathcal{R}$ gives the vehicle's current (or last) location, $w \in \mathcal{W}$ gives the vehicle's battery level when it was at region i, and τ gives the number of time periods until the vehicle reaches the desired destination j. We assume access to each user's end destination j only after deciding whether or not to offer them a free ride. If the vehicle is idle, we set $\tau = 0$ and $j = 0$. Finally, we let $\mathcal{S}$ be all possible 4-tuples for the n vehicles of the system, and for state $s \in \mathcal{S}$, we let $V(s)$ be the optimal value functions. We do not give the explicit form of these value functions since they are simply more cluttered versions of those presented for the 1VMC problem instance. Naturally, computing the optimal policy in this setting, much less characterizing its structure, is quite a difficult task. As such, we elect to employ the approximate dynamic programming approach of Desai et al. (2012). In what follows, we summarize this approach and describe how we apply it to our setting.

We develop and test free-ride policies by solving the smoothed approximate linear program (SALP) introduced by Desai et al. (2012). In this approach, the dynamic program is formulated as an equivalent linear program and the value functions are approximated by a linear combination of L basis functions, which capture key properties of the state of the system. More specifically, we approximate the value functions $V(s)$ by $\tilde{V}(\mathbf{s}) = \sum_{l=1}^{L} r_l \cdot \phi_l(\mathbf{s})$, where $\phi_l : \mathcal{S} \mapsto \mathbb{R}$ is the l-th basis function and r_l is its weight. These weights are the decision variables within the linear program. Once the optimal weights have been derived, free-ride policies can be developed by using the value function approximations $\tilde{V}(s)$ to approximate the revenue trade-off between offering a free ride or letting the user take her desired ride.

This approximation reduces the number of variables in the LP formulation of the dynamic program (we have just L decision variables, one for each basis function), but there is still a constraint for every state-action pair, and hence the resulting linear program can be intractable when the state space is large, as is the case in our problem. In such scenarios, Desai et al. (2012) propose an approach in which the constraints are randomly sampled and then the linear program is solved using this subset of states. Further, the optimal solution to this linear program is permitted to violate the constraints up to a certain “budget”, whose magnitude reflects the extent to which the sampled linear program is further approximated. We explain how we sample constraints and choose the budget in Appendix F of the Online Appendix. Surprisingly, Desai et al. (2012) show both theoretically and numerically that the number of constraints that one must sample to arrive at reasonable[1] approximation of the original linear program that contains all possible constraints does not depend on the size of the underlying state space $\mathcal{S}$, but only on L, the number of basis functions.

4. Numerical Experiments

In this section, we present the details and results of a series of three distinct large-scale discrete event simulations, which we carry out on EVSS networks inspired by those of our collaborator. We crafted these experiments in an effort to study the efficacy of free-ride policies under varying demand patterns and system parameters. We benchmark the performance of the free-ride policies that we develop against our EVSS collaborator’s current fine-based practice, under which users are fined for street-parking low-battery vehicles. We present the details of our experiments in the following three sections. In Section 4.1, we begin by providing a high-level description and motivation for each of the three experiments that we conduct. Following this high-level summary of each experiment, we describe the key ingredients and assumptions that go into setting up and running each experiment. Next, in Section 4.2, we discuss the various policies that we test, and list the performance metrics that guide our assessment of each policy’s performance. Finally, in Section 4.3, we summarize the results of the three experiments, and in doing so, we provide high-level managerial insights that surface from our extensive series of simulations.

4.1. The Three Experiments: Motivation and Set-Up

We begin by summarizing the distinguishing features and high-level goals of our three experiments. This summary is followed by a description of how we design and set up each individual experiment to match these intentions. As part of this latter description, we explain how we use the historical ride data provided to us by our EVSS collaborator, which includes transactional data based on all rides from 9/20/15 to 11/21/15. Each vehicle in

[1] The exact theoretical guarantee is fairly technical and hence we leave our description of this guarantee at this high level.

the fleet is equipped with a device that continually transmits information every minute to the company's database. As a result, for each ride, we know the starting and ending timestamps, origin and destination coordinates, fare paid, battery consumed, distance traveled, and a variety of other readings. As suggested by the company, we removed rides greater than 15 miles, since it is likely that these rides do not reflect a commute, but rather a leisure trip without a pre-conceived destination. Further, we removed rides that end in regions that do not serve as an origin for any other rides, since these rides will create "sink" regions in our simulation. The data cleansing ultimately leaves us with ∼28,300 rides. About 85% of these rides are less than an hour, and the average distance traveled is 3.6 miles. It is important to note that over the two month period that our data set spans, the system was static and had no change in fleet, service area, or pricing. As we discuss later on, in each of the three experiments, we use this data set to varying degrees to guide our choices for the key system parameters within our simulations.

The three experiments that we conducted are respectively labeled True_EVSS, Parameter_Sensitivity, and Demand_Sensitivity and are summarized below. We note that we refer to the arrival probabilities λ_i and the transition probabilities p_{ij} as demand parameters, while any other system parameter is referred to as an operational parameter.

- Experiment True_EVSS: The intent of this first set of experiments is to measure the performance of each policy that we consider using a simulated EVSS network that most closely resembles that of our collaborator. For this purpose, we use the historical ride data to guide many of the underlying parameter values within our discrete event simulation. We carry out this simulation on only a single set of parameter values, which we refer to as the baseline parameters. We believe these values to be the most realistic based on the historical ride data and discussions with our EVSS collaborator.

- Experiment Parameter_Sensitivity: In this experiment, we fix demand to be uniform across the network, and then we study the overall impact of changing key operational parameters on both the performance of various policies, and on the specific structure of the optimal SOR policies. We delay a detailed description of the exact nature of this parameter sensitivity analysis until after fully formalizing the set-up of our simulations.

- Experiment Demand_Sensitivity: In our final experiment, we fix all of the operational parameters that were varied in the previous experiment to the baseline values, and instead only vary the two parameters related to demand. Our goal in this setting is to understand how the performance and structure of the optimal SOR policies is affected by shifting the demand towards or away from the charging stations in the network. Again, we delay a formal description of how we vary demand in this way until after fully formalizing the set-up of our simulations.

Next, we move to describing how we set up each of the three experiments. As should be evident from the summaries above, the three experiments only differ by the manner in which the demand and operational parameters are set and varied across simulations. As such, the experimental set-ups for each of experiment share quite a bit of common ground. In an effort to succinctly describe the set-up for each experiment, we first describe their shared features, before sequentially addressing the defining elements that make each of the experiments unique.

Common features of each experiment. In what follows, we detail the elements of our simulation that remain fixed as we move from experiment to experiment. First, among these common elements is the structure of the underlying EVSS network, which includes how we partition the service area into discrete regions, the number of vehicles in the system, and the number/location of the charging stations. Furthermore, throughout each of our experiments, we use the same random ride generator to determine each user's exact trip when they rent a vehicle, and we also use the same random utility model to capture customer preferences when a free ride is offered. Finally, the random process governing manual respositions by the system operator remains unchanged across the experiments. Next, we discuss each of these common aspects individually in greater detail.

EVSS network. The service area that we use within the simulations is modeled off of the location of our EVSS collaborator. We create a discretized service region by partitioning our EVSS collaborator's service area into a collection of square regions that comprise $\mathcal{R}$, the set of feasible locations for vehicles and charging stations within our models and simulations. The size of each square was chosen so that it takes no more than four minutes to walk from end-to-end based on the assumption that people walk at 3.1 miles per hour (Browning et al. 2006). This partitioning scheme leaves us with 577 square regions, where the distance between opposite corners is approximately 0.2 miles. Our EVSS collaborator has charging stations positioned in 40 of these regions, and manages a total of 300 electric vehicles.

Generating trips. To generate the inter-arrival times of users, we fit the historical inter-arrival data to several distributions and find that the best fit is the inverted beta distribution with shape parameters $\alpha = 0.92$ and $\beta = 4.07$, location parameter $= 0.01$, and scale parameter $= 8.86$. This parameter set gives a mean inter-arrival time of 2.66 minutes, which we use as our period length when deriving the discount policies. If a customer arrives to a region with several vehicles available, we assume that she always rents the highest-charged vehicle [2]. In the event that a vehicle is rented in region $i \in \mathcal{R}$ and the user's desired location is $j \in \mathcal{R}$, we use a linear regression model fitted to the historical ride data to randomly generate the ride duration (in minutes) t_{ij} and battery consumption b_{ij} of the given user's trip. We refer the reader to Appendix C.1 of the Online Appendix for the full regression output and a detailed description of how the regressions are used to generate each ride in the simulation. The fare for the given trip is then set to be $f_{ij} = 1 + 0.15 \cdot t_{ij}$, which is a pricing scheme vetted by our collaborator and common in many modern dockless EVSSs.

Customer utility model. If the system operator offers a free ride, we assume that the user trades off the inconvenience of walking a distance of d to her desired destination and paying a fare of f for her ride. We capture this trade-off using a common linear structure for the utility function throughout all of our experiments. More specifically, we set $u(d, f) = -\alpha_d \cdot d - \alpha_f \cdot f$, where $\alpha_d \sim U(0, \mathcal{DM})$, $\alpha_f \sim U(0, 1)$ are generated randomly for each arriving user. The operational parameter $\mathcal{DM}$ captures the extent to which there is variability within the customer population with regards to the inconvenience of walking the extra leg to the desired destination of the user. Note that we only consider the disutility associated with each ride, however this utility function could easily be updated by simply adding a constant term μ, which reflects the utility gained from a successful commute. We discard this positive term simply for notational convenience.

Manual repositioning events. Finally, recall that a vehicle is deemed move-eligible if it has a battery level below b_m. Furthermore, in each time period, if there are any move-eligible vehicles in the network, then with probability p_m we uniformly select one to be moved to a charging station. The selected vehicle is then assumed to be unavailable for a random amount of time, generated from a truncated normal distribution with a mean of four hours and standard deviation of 30 minutes. Given that such moves could require an employee

[2] Our historical data set does reveal that users generally choose the highest charged vehicle when presented with a choice among many vehicles.

to go between opposite ends of the city, we felt this was a reasonable distribution for the time required for a manual move.

Battery recharging. When solving for the policies that we test, we discretize the set of battery levels $\mathcal{W}$ by $\delta = 0.02$. Further, we set the re-charging rate to be $\gamma = 0.02$, which corresponds to a charging time of 2.25 hours to replenish a completely depleted vehicle. While our collaborator's vehicles take 3-5 hours to fully recharge, we were not able to discretize the battery levels in any finer increments than 0.02 and still tractably solve *Single Threshold* to optimality in a reasonable amount of time. In the simulation however, we take a more conservative approach and assume that the battery takes 5 hours to replenish.

Distinguishing features for experiment True_EVSS. As noted above, the intention of this experiment is to recreate the EVSS network of our collaborator with as much accuracy and nuance as possible. For this purpose, we use the historical ride data to govern the demand parameters. More formally, for each region i, we set $\lambda_i = \frac{\text{\# of rides originating at } i}{\text{\# of total rides}}$, and for each pair of regions $i, j \in \mathcal{R}$, we set $p_{ij} = \frac{\text{\# of rides from } i \text{ to } j}{\text{\# of rides originating at } i}$. In total, the data set of ∼28,300 rides is spread across 16,720 unique origin-destination pairs out of a possible $r^2 = 332,929$ such pairs. The remaining set of operational parameters are fixed to the following values, which we denote as our *baseline parameter set.* We set the cost c_m of a manual move to be $25 since this is the fine our collaborator enacts for street-parking a low-battery vehicle that eventually must be moved to a charging station. We set the battery move-threshold $b_m = 0.2$, the probability of a move $p_m = 20\%$, and the dollar-to-mile sensitivity parameter in the customer utility function $\mathcal{DM} = 5$. These values are our best guesses at reality after several discussions with our EVSS collaborator.

Distinguishing features for experiment Parameter_Sensitivity. In this experiment, we assume uniform demand and transition probabilities across the entire network. As a result, rides between any pair of regions can occur, hence we have r^2 possible trips and we set $\lambda_i = p_{ij} = \frac{1}{r}$. The motivation behind this assumption is rooted in the historical arrival probabilities, which are reasonably uniform across each region as indicated in the heatmap of λ_i in Appendix G of the Online Appendix. For this fixed uniform demand pattern, we vary $p_m = \{5\%, 10\%, \ \mathbf{20}\%\}$ and use a battery move thresholds of $b_m \in \{0.05, \ 0.10, \ \mathbf{0.20}\}$. We also consider manual move costs of $c_m \in \{\$5, \ \$\mathbf{25}, \ \$50\}$ and dollar-to-mile sensitivity parameter $\mathcal{DM} \in \{0.5, \mathbf{5}, 20\}$. The baseline values introduced in our description above of experiment True_EVSS are bolded for reference.

Distinguishing features for experiment Demand_Sensitivity. In this final experiment, we fix the operational parameters at their baseline values and we vary the two demand parameters to capture scenarios in which demand is either generally close or far away from the charging stations. In total, we test nine different ride patterns that are generated by varying arrival and destination demand to reflect the following three scenarios: either clustered close (C) to charging stations, uniform (U) across the service area, or far (F) from charging stations. When demand is assumed to be clustered close to (far away from) charging stations, we assume that the arrival probability at region i and the probability that a user ends her trip at region i is linearly decreasing (increasing) in d_{iz_i}, the distance between region i and its closest charging station. Specifically, when demand is close to charging stations, we set $\lambda_i = \frac{d^* - d_{iz_i}}{\sum_{j\in\mathcal{R}} d^* - d_{jz_j}}$, where we define $d^* = 0.1 + \max_{j\in\mathcal{R}}\{d_{jz_j}\}$. Note that regions that are closer to charging stations will have larger arrival probabilities. When demand is assumed to be farther from charging stations, we set $\lambda_i = \frac{0.1 + d_{iz_i}}{\sum_{j\in\mathcal{R}} 0.1 + d_{jz_j}}$. In both cases, we use the additive factor of 0.1 to ensure that the relative magnitudes of the arrival probabilities are reasonable and non-zero. We set the transition probabilities for each of the three scenarios in a similar fashion.

4.2. Policies Tested and Performance Metrics

In this section, we summarize the various free-ride and benchmark policies that we implement within the three experimental settings that we consider. We begin by describing the handful of free-ride-based policies that are derived from the models presented in Section 3. These policies are all computed using a discount factor of $\beta = 0.999$ and discretized battery levels in increments of $\delta = 0.02$. Next, we describe two benchmark policies; one of which is the current fine-based practice of our EVSS collaborator and the other is a “hands-off” policy in which the system operator lets the system unfold naturally. We then detail the numerous performance metrics that we use to measure the efficacy of each policy. The union of all policies that we consider across the three experiments are summarized below:

1VMC-SOR: This is the optimal SOR free-ride policy $\tilde{\pi}$ that is derived by solving *Single Threshold*. Recall that under the policy $\tilde{\pi}$, we offer a free ride each time that a user rents a vehicle in region i with battery level w and $(i, w) \in S_{\tilde{\pi}}$.

1VMC-50: This is the discount-ride policy that we derive from solving the *1VMC* dynamic program modified so that in addition to having the option to offer a free-ride, the system operator can also offer a half-priced, or 50%-discounted, ride.

NVMC-SALP: This is the policy derived from solving the smoothed-ALP described in Section 3.4. Due to the fact that this policy is computationally intensive to implement, we only consider its performance in experiment True_EVSS. We seed our value function approximation using ten basis functions, which we list in Appendix F of the Online Appendix along with other key details for implementing this policy.

Fine-Based (FB): Under the Fine-Based policy, users will be fined \$25 for street-parking a vehicle that has a charge level less than b_m, the battery threshold for a manual move. The user can avoid this fine by parking the low-battery vehicle at a charging station. This is the current policy implemented by our EVSS collaborator to alleviate "stranded" low-battery vehicles in their network. Hence the interesting scenario within the simulation arises when the user's preferred trip does not end at a charging station and leaves the rented vehicle depleted. In this case, the user will trade off the utility $u(0, \$25 + f_{ij})$ of incurring a fine, but getting to her desired location, with the utility $u(d_{z_j j}, f_{ij})$ of dropping the vehicle at a charging station. Ultimately the user will select the higher utility option.

Never-Offer (NO): Under this policy, the system operator lets the system unfold naturally and so the only way for a vehicle's battery to be replenished is if the customer's intended destination is a charging station or if the vehicle is selected for a manual repositioning to a charging station.

We evaluate the performance of these policies via Monte Carlo simulation. Each simulation begins with fully-charged vehicles assigned to regions according to the distribution of arrival probabilities, and then runs for 100 days. Since the network we consider is fairly large, we use the first 70 days in each simulation as a warm-up period. Using rides from the last 30 days, we compute a wide variety of performance metrics, which are all listed in Table 1. The values that we eventually report in our results are per-day averages of each metric over 30 distinct simulations of the 100 day time horizon. We note that we do not report profit in Table 1, but we do track revenue generated from fares and the number of manual repositioning moves, which is a proxy for operational costs. Both of these metrics can be used together to develop a sense of profit. Additionally, for experiments Parameter_Sensitivity and Demand_Sensitivity, we summarize the structure of the SOR policies that we derive from solving *Single Threshold* by reporting the average battery charge-level range that characterizes these threshold-based policies.

Table 1 Description of Performance Metrics.

Metric	Description
Revenue	Daily revenue accrued by the system.
Rides Completed	Rides taken per day.
Moves	Number of manual repositioning moves completed per day.
Offers	Number of discounted-ride offers extended per day. For the Fine-Based policy, this is the number of times per day that a customer decides between accepting or avoiding a fine.
Accepts	Number of discounted-ride offers accepted per day. For the Fine-Based policy, this is the number of times per day that a customer avoids a fine by ending their trip at a charging station.
Unmet Demand | Vehicle	Total unmet demand per day due to users not finding a vehicle at their origin region or at one of the neighboring regions.
Unmet Demand | Battery	Total unmet demand per day due to vehicles not having enough charge.
Utility per Ride	Average utility of users who were offered the discounted ride option. For the Fine-Based policy, this metric is computed only using rides where the user had to trade-off a fine and the inconvenience of dropping the vehicle at a charging station.
Average Battery	Average charge of the fleet (with and without the vehicles at charging stations) at the end of the day.
Proportion in $\mathcal{Z}$	Proportion of the fleet at a charging station (in $\mathcal{Z}$) at the end of the day.
Rides Fulfilled at i	Proportion of rides fulfilled at a customer's arriving region versus at one of the neighboring regions.

4.3. Results and Managerial Insights

In this section, we sequentially present the results of each experiment. In doing so, we concisely summarize the core trade-off between revenue earned and customer satisfaction that arises when designing incentive-driven (or penalty-driven) policies in dockless EVSS networks. Furthermore, we highlight the high-level managerial insights that we are able to glean from our simulations, which we believe to be impactful take-home points.

Results of experiment True_EVSS. The performance of all five policies listed above is presented in Table 2. What is immediately evident is that the Fine-Based policy garners more revenue that the discount-based policies, but users of the system under the Fine-Based policy experience at least double the disutility of users in the discount-based policies. More specifically, we observe that the revenue earned under the Fine-Based policy is 2% and 10% higher than the revenue earned under the NVMC-SALP and 1VMC-SOR policies respectively. This trend is likely a result of the stringent $25 fine that is enough to prevent users from leaving uncharged vehicles on the street, without having to discount their ride. Consequently, users will often forgo their desired ride in order to avoid a fine, and instead park the vehicle at a (potentially undesirable) charging station. This leads to far fewer manual moves per day, but an average utility that is ten times worse than the 1VMC-SOR policy. Hence the notion that the Fine-Based policy generates the most revenue should be taken with a grain of salt, as our simulation does not account for the long-term consequences of a dissatisfied user-base. In short, the potential negative impact of the Fine-Based policy on customer satisfaction could outweigh the short-term benefits of increased revenue.

Results of experiment Parameter_Sensitivity. Recall that in this experiment, we depart from the historical demand patterns and set the arrival and transition probabilities to be uniform across the entire network. For this fixed demand pattern, we consider eight configurations of the operational parameters, where each configuration is distinguished by a deviation from the baseline setting along one parameter. More specifically, we test $c_m \in \{\$5, \mathbf{\$25}, \$50\}, b_m \in \{0.05, 0.10, \mathbf{0.20}\}, p_m \in \{5\%, 10\%, \mathbf{20\%}\}$, and $\mathcal{DM} \in \{0.5, \mathbf{5}, 20\}$, where the bolded values indicate the baseline values. Again, we arrive at eight different parameter configurations by choosing one parameter whose value will deviate from the baseline, and then enumerating over all such combinations. For this second experiment, we only test the 1VMC-SOR and the Fined-Based policies, whose performance along all of the dimension listed in Table 1 is reported in Appendix D of the Online Appendix. While our primary focus of this experiment is to conduct a performance sensitivity analysis with regards to the many operational parameters, we first comment on the relative overall performance of the two tested policies. In general, we observe that the relative performance of these two policies matches that of experiment True_EVSS, however interestingly, when we consider the test case where the battery move threshold is set to its lowest value ($b_m = 0.05$), the FB policy is dominated by the 1VMC-SOR policy along all performance metrics. This is likely a result of the fact that when b_m is low, users can street-park low-battery vehicles that cannot serve many future rides without incurring a fine.

With regards to the sensitivity analysis, Table 3 summarizes the impact of varying each operational parameter on the performance of 1VMC-SOR policies. More specifically, this table presents the percentage change in each performance metric as the operational parameters c_m and b_m are changed from their baseline values. This table does does include the sensitivity analysis for $\mathcal{DM}$ and p_m since changing these parameters had only a mild impact on the many performance metrics that we consider. Of particular interest is how the repositioning cost c_m affects the number of manual repositioning events, which is reported in column four of Table 3. When the cost c_m decreases, the system realizes 16% more revenue, but the caveat is that the number of manual movements jumps by 238% because manual moves are cheaper. As expected, the system relies on manual repositioning much more to bring vehicles back to charging stations when c_m is low. When the cost c_m is higher, we see the opposite effect: accrued revenues drop by 9% and the system utilizes free rides instead of high-cost more manual movements to keep to prevent stranded low-battery vehicles. Interestingly, we also see dramatic swings in the levels of both types of unmet demand as the battery-move threshold b_m is changed. Naturally, as b_m is decreased from its baseline value of 0.2, we see more unmet demand due to insufficient battery levels, since low-battery vehicles will not always prompt a manual move. However, increases of over 100% and 800% are surprising and reflect the potential impact of the parameter b_m. These large percentage increases can be explained by noting that when b_m is set to its lowest value of 0.05, a vehicle with a battery level of 0.06, for example, will not be manually moved to a charging station and will result in lots of unmet demand due to the fact that this vehicle cannot serve many rides.

In addition to monitoring changes in performance metrics, we also study how the structure of the SOR policy changes as we vary key operational parameters. The first column of Table 4 reports the average length of the offer range across all regions of the SOR policy under each parameter configuration that we consider. The remaining three columns report the correlation of the length of the offer range with several region-specific features, such as the arrival rate λ_i and the expected battery consumption from i when the vehicle is fully charged (i.e., $w = 1$). Note that if the correlation is negative, the average SOR length decreases with the corresponding feature. This analysis helps illuminate additional drivers behind the frequency with which free-rides are offered. For instance, the last column indicates that offer ranges are larger at regions where users take rides that consume more battery, since, in this case, vehicles are more likely to end trips with low battery.

Perhaps the most striking insight from Table 4 is how the cost of manual moves c_m and customers' willingness to walk $\mathcal{DM}$ impact the length of the SOR. Our results indicate that the mean SOR length approximately doubles both when c_m increases from \$5 to \$50, and when the variability in willingness to walk increases from $\mathcal{DM} = 0.5$ to $\mathcal{DM} = 20$. This latter change corresponds to a decrease from 97% to 49% in the probability of accepting a free-ride offer, and hence we observe larger free-ride offer ranges as the system operator faces more uncertainty surrounding each customer's response to a free-ride offer.

Results of experiment Demand_Sensitivty. In this experiment, we fix the operational parameters at their baseline values and we study the impact of varying demand patterns on the performance of the Fine-Based and 1VMC-SOR policies. For the 1VMC-SOR policy, we also study how the structure of the SOR changes as we shift demand. To accomplish this task, we create nine different demand settings which are characterized by the generall proximity of the underlying demand to the scattered charging stations. The full set of results are presented in Table 5, where the performance of the 1VMC-SOR policy is reported relative to that of the Fine-Based policy on six different performance metrics (the top six charts) for each of the nine demand scenarios. The full set of results are available in Appendix E of the Online Appendix.

As is the case in the previous two experiments, the revenue generated under the Fine-Based policy is greater than that generated under the 1VMC-SOR policy over each demand scenario that we consider. However again, this realization should be taken with a grain of salt, since this lower revenue is driven by the fact that the 1VMC-SOR policy forgoes revenue in exchange for the opportunity to keep a charged fleet, and not because it fulfills fewer rides per day. As a result, the 1VMC-SOR policy is indeed able to preserve a higher charged fleet, which in turn results in less unmet demand as users are able to access sufficiently charge vehicles that allow them to take their desired rides. And like we see in the previous experiments, the free-ride policy provides a better customer experience with an average disutility that is 13-16% of the disutility under the Fine-Based policy.

We note that the metric most strongly affected by the shifting demand is the number of moves per day. As users' intended destinations shift to being farther from charging stations, the number of manual moves decreases. This occurs because free rides are offered more liberally in this setting in an attempt to lure vehicles back towards charging stations.

The bottom three charts in Table 5 show how the structure of the SOR changes as demand is varied. Generally, we find that if users are already planning on ending their trip near a charging station, then there is less of a need to offer discounts to charging stations. When the demand pattern flips in the opposite direction and users generally begin their rides near charging stations and end far away from charging stations, then the SORs are selected to be quite large in an effort to keep traffic in close proximity to where demand is expected to arise. The results in the bottom right of Table 5 shows that d_{iz_i}, the distance between each region and its closest charging station, can be useful in characterizing SOR length when destinations are far away from charging stations. In the left-most column (i.e., $p_{ij} = \mathrm{C}$), the correlation between distance and offer range is fairly non-existent, but as the destinations move farther away from charging stations, the correlation becomes larger. One explanation for this phenomena is that when regions near charging stations are not common destinations, free rides need to be offered liberally to maintain a charged fleet.

Table 2 Experiment True_EVSS: Performance of Policies on Historical Data.

Metric	1VMC-SOR	1VMC-50	NVMC-SALP	NO	FB
Revenue	\$2,174.39	\$2,184.89	\$2,336.79	\$2,391.83	\$2,386.87
Rides Completed	348.84	344.44	347.31	350.16	349.61
Moves	8.31	5.39	24.24	24.50	2.26
Offers	35.79	62.01	5.62	–	38.40
Accepts	32.05	49.96	5.06	–	36.11
Unmet Demand | Vehicle	79.65	84.97	79.82	77.16	77.50
Unmet Demand | Battery	3.33	2.42	4.70	4.50	4.71
Utility per Ride	-0.47	-2.06	-0.47	–	-4.75
Average Battery (All Vehicles)	71.2%	75.5%	68.0%	66.9%	68.3%
Average Battery (w.o. Vehicles at $\mathcal{Z}$)	55.4%	58.9%	51.8%	51.9%	53.1%
Proportion at $\mathcal{Z}$	41.1%	46.3%	39.0%	36.6%	38.1%
Rides Fulfilled at i	55.9%	55.5%	55.7%	55.9%	56.0%

Simulation based on historical ride demand and baseline parameters. Vehicles start at 50% battery. Metrics are averaged over 15 demand sample paths.

Table 3 Experiment Parameter_Sensitivity: Impact of Network Parameters on Performance Metrics.

Parameter Change	Revenue	Rides Completed	Moves	Offers	Unmet Demand Vehicle	Unmet Demand Battery	Average Battery	Proportion in $\mathcal{Z}$
Baseline	\$1,185	231.5	4.84	63.70	196.83	4.31	0.88	0.78
$c_m = \$5$	\$1370	238.5	18.53	40.21	184.78	9.32	0.84	0.74
	(16%)	(3%)	(283%)	(-37%)	(-6%)	(116%)	(-4%)	(-5%)
$c_m = \$50$	\$1074	217.5	3.09	65.82	212.49	3.14	0.90	0.81
	(-9%)	(-6%)	(-36%)	(3%)	(8%)	(-27%)	(2%	(4%)
$b_m = 0.05$	\$1262	248.9	2.99	68.72	141.78	38.98	0.79	0.67
	–	–	–	–	(-28%)	(804%)	(-10%)	(-14%)
$b_m = 0.10$	\$1240	244.0	3.00	68.26	175.27	12.11	0.85	0.73
	–	–	–	–	(-11%)	(181%)	(-3%)	(-5%)

Performance metrics computed as the average over 30 demand sample paths. For each configuration, we use the performance of the *1VMC-SOR* policy to report the percentage change in each metric relative to the baseline. Only if the difference in performance is significant at 5%, then we report the relative percent change below the metric.

5. Conclusion and Directions for Future Work

In this paper, we study the use of discounted rides as a mechanism to directly incentivize users to park vehicles at charging stations in order to keep the fleet of vehicles adequately

Table 4 **Experiment Parameter_Sensitivity: Sensitivity of SOR Size for Various Network Configurations.**

			Correlation between SOR Length and:		
Instance	Parameter Changes	Mean Length	Arrival Probability	Distance to Closest Charging Station	Expected Battery Consumption w. $w = 1$
1	Baseline	37.97	-0.29	-0.27	0.62
2	$c_m = \$5$	22.32	-0.26	-0.26	0.49
3	$c_m = \$50$	42.59	-0.26	-0.26	0.53
4	$b_m = 0.05$	33.53	-0.32	-0.18	0.70
5	$b_m = 0.10$	33.53	-0.32	-0.18	0.70
6	$p_m = 5\%$	37.50	-0.29	-0.21	0.66
7	$p_m = 10\%$	37.79	-0.29	-0.26	0.64
8	$\mathcal{DM} = 0.5$	26.30	-0.07	-0.43	0.14
9	$\mathcal{DM} = 20$	53.40	-0.34	-0.09	0.80

The mean length of the SOR policy is computed as the average length over all the regions in the network.

Table 5 **Experiment Demand_Sensitivity: Impact of Varying Ride Demand (λ_i and p_{ij}).**

Revenue Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	74%	73%	70%
	U	76%	74%	69%
	F	84%	82%	75%

Moves Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	98%	83%	58%
	U	104%	85%	62%
	F	243%	182%	90%

Unmet Demand | Vehicle Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	110%	110%	116%
	U	106%	107%	114%
	F	101%	101%	102%

Unmet Demand | Battery Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	39%	33%	25%
	U	40%	36%	26%
	F	69%	58%	37%

Average Disutility per Offer Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	13.5%	13.5%	13.6%
	U	13.5%	13.4%	13.7%
	F	13.5%	13.5%	16.3%

Rides Fulfilled Relative to FB

		p_{ij}		
		C	U	F
λ_i	C	97%	98%	95%
	U	98%	97%	93%
	F	100%	100%	97%

Mean SOR Length

		p_{ij}		
		C	U	F
λ_i	C	38.40	40.70	44.67
	U	36.14	39.69	43.47
	F	28.38	31.46	39.52

Correlation: SOR Length & Arrival Probability (λ_i)

		p_{ij}		
		C	U	F
λ_i	C	-0.20	-0.10	0.05
	U	-0.35	-0.33	-0.23
	F	-0.28	-0.39	-0.49

Correlation: SOR Length & Distance to Closest Charging Station (d_{iz_i})

		p_{ij}		
		C	U	F
λ_i	C	-0.12	-0.22	-0.33
	U	0.04	-0.24	-0.36
	F	-0.16	-0.23	-0.34

The arrival probability (λ_i) and the transition probabilities (p_{ij}) vary from close (C) to charging stations, uniform (U) across the entire service area, and far (F) from charging stations. We use baseline parameters values. The optimal *1VMC-SOR* policy is obtained by solving *Single Threshold*.

charged. We focus on developing simple free-ride offer policies, which we refer to as single-offer range policies. These policies specify a critical battery levels for each region i, which serve as cut-off points for whether or a not a free ride should be offered. Not only do we provide a formulation to find the optimal single-offer range policy for certain settings, but we also demonstrate that such policies can be quite effective in terms of their ability to generate revenue, keep the fleet of vehicles charged and keep the user-base happy. We show this latter point through an extensive discrete event simulation that is seeded with historical ride data from a real EVSS. While offering price incentives to users to change their destination to charging stations seems like a good idea, our results show that Fine-Based policies are also effective, and the advantages between the two types of policies depend on the system's objectives and features of the network and user base.

There are many interesting directions for future work with regards to dockless EVSS systems. Since most mobility sharing systems experience travel demands varying by hours over the day, one potential extension of our work involves incorporating time-varying ride patterns. We attempted to extend our infinite horizon framework to a time-varying setting where we solved for an time-specific SOR policy for several time windows throughout the day. Depending on the time block of an arriving customer, the corresponding time-specific policy is used to determine if a free ride should be offered. Unfortunately, this time-block policy did not perform as well as the free-ride policy generated from using the ride data from the entire day. Another direction for future work within the multiple-vehicle framework could consider a setting in which the EVSS consists of multiple vehicle types. In our case, we assume that all vehicles are homogeneous, but many VSSs have several vehicle types (i.e. sedan and SUV or e-scooters and e-bikes) whose functionalities are all different. On the behavioral revenue management side, there are several fascinating directions. For instance, in a ride-sharing setting, Cohen et al. (2018) compare the effectiveness of immediate ride discounts versus future ride credit. In the same spirit, an interesting question is examining the long-term implications that discounts and fines have on customer retention and ridership in a dockless, EVSS.

References

Avci B, Girotra K, Netessine S (2014) Electric vehicles with a battery switching station: Adoption and environmental impact. *Management Science* 61(4):772–794.

Banerjee S, Freund D, Lykouris T (2016) Pricing and optimization in shared vehicle systems: An approximation framework. *CoRR abs/1608.06819* .

Bellos I, Ferguson M, Toktay LB (2017) The car sharing economy: Interaction of business model choice and product line design. *Manufacturing & Service Operations Management* 19(2):185–201.

Boyacı B, Zografos KG, Geroliminis N (2015) An optimization framework for the development of efficient one-way car-sharing systems. *European Journal of Operational Research* 240(3):718–733.

Brandstatter G, Leitner M, Ruthmair M (2017) Charging station placement in free-floating electric car sharing systems .

Browning RC, Baker EA, Herron JA, Kram R (2006) Effects of obesity and sex on the energetic cost and preferred speed of walking. *Journal of Applied Physiology* 100(2):390–398.

Bruglieri M, Colorni Ab, Luè A (2014) The relocation problem for the one-way electric vehicle sharing. *Networks* 64(4):292–305.

Chang J, Yu M, Shen S, Xu M (2017) Location design and relocation of a mixed car-sharing fleet with a co2 emission constraint. *Service Science* 9(3):205–218.

Chemla D, Meunier F, Pradeau T, Calvo RW, Yahiaoui H (2013) Self-service bike sharing systems: simulation, repositioning, pricing .

Cohen M, Fiszer MD, Kim BJ (2018) Frustration-based promotions: Field experiments in ride-sharing .

Desai VV, Farias VF, Moallemi CC (2012) Approximate dynamic programming via a smoothed linear program. *Operations Research* 60(3):655–674.

Fishman E, Washington S, Haworth N (2014) Bike share's impact on car use: evidence from the united states, great britain, and australia. *Transportation Research Part D: Transport and Environment* 31:13–20.

Freund D, Henderson SG, Shmoys DB (2017) Minimizing multimodular functions and allocating capacity in bike-sharing systems. *International Conference on Integer Programming and Combinatorial Optimization*, 186–198 (Springer).

Fricker C, Gast N (2014) Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity. *EURO Journal on Transportation and Logistics* 1–31.

He L, Mak HY, Rong Y, Shen ZJM (2017) Service region design for urban electric vehicle sharing systems. *Manufacturing & Service Operations Management* 19(2):309–327.

Jorge D, Correia G (2013) Carsharing systems demand estimation and defined operations: a literature review. *European Journal of Transport and Infrastructure Research* 13(3):201–220.

Kabra A, Belavina E, Girotra K (2018) Bike-share systems: Accessibility and availability. *Chicago Booth Research Paper* (15-04).

Lim MK, Mak HY, Rong Y (2014) Toward mass adoption of electric vehicles: impact of the range and resale anxieties. *Manufacturing & Service Operations Management* 17(1):101–119.

Lin JR, Yang TH (2011) Strategic design of public bicycle sharing systems with service level constraints. *Transportation research part E: logistics and transportation review* 47(2):284–294.

Lu M, Chen Z, Shen S (2017) Optimizing the profitability and quality of service in carshare systems under demand uncertainty. *Manufacturing & Service Operations Management* .

Mak HY, Rong Y, Shen ZJM (2013) Infrastructure planning for electric vehicles with battery swapping. *Management Science* 59(7):1557–1575.

Marecek J, Shorten R, Yu JY (2016) Pricing vehicle sharing with proximity information. *arXiv preprint arXiv:1601.06672* .

Nair R, Miller-Hooks E (2011) Fleet management for vehicle sharing operations. *Transportation Science* 45(4):524–540.

Navigant Research (2016a) Carsharing programs. URL `http://bit.ly/2JKpHD5`.

Navigant Research (2016b) Carsharing services will surpass 12 million members worldwide by 2020 navigant research. URL `http://bit.ly/2w3uZS9`.

O'Mahony E, Shmoys DB (2015) Data analysis and optimization for (citi) bike sharing. *AAAI*, 687–694.

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

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

Sayarshad H, Tavassoli S, Zhao F (2012) A multi-periodic optimization formulation for bike planning and bike utilization. *Applied Mathematical Modelling* 36(10):4944–4951.

Schuijbroek J, Hampshire RC, Van Hoeve WJ (2017) Inventory rebalancing and vehicle routing in bike sharing systems. *European Journal of Operational Research* 257(3):992–1004.

Singla A, Santoni M, Bartók G, Mukerji P, Meenen M, Krause A (2015) Incentivizing users for balancing bike sharing systems. *AAAI*, 723–729.

The San Diego Union-Tribune (2016) Car2go ceases san Diego operations. URL `http://bit.ly/2EditnB`.

US Department of Energy (2016) Alternative fuels data center: Emissions from hybrid and plug-in electric vehicles. URL `http://bit.ly/2Edjs7h`.

Waserhole A, Jost V (2016) Pricing in vehicle sharing systems: Optimization in queuing networks with product forms. *EURO Journal on Transportation and Logistics* 5(3):293–320.

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

Appendix A: Notation Table

Table A.1 Summary of notation.

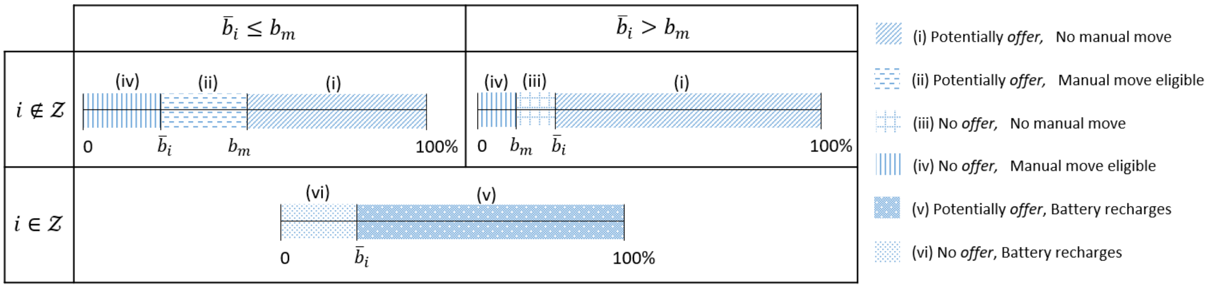
Parameter	Description
$\mathcal{R}$	Set of regions in the network, where i,j denote arbitrary locations in $\mathcal{R}$.
$\mathcal{Z}\subset\mathcal{R}$	Regions with charging stations, where z denotes an arbitrary charging station in $\mathcal{Z}$.
$\mathcal{R}(i,w)\subseteq\mathcal{R}$	Subset of regions that can be reached from location $i\in\mathcal{R}$ with battery $w\in\mathcal{W}$. Defined as $\mathcal{R}(i,w)=\{j\in\mathcal{R}:w\geq b_{ij}\}$.
$\mathcal{W}$	Set of feasible battery levels. Values are equally separated by δ, so $\mathcal{W}=[0,\delta,2\delta,...,1]$.
$\delta\in[0,1]$	Battery increment used in $\mathcal{W}$.
$\sigma\in[0,1]$	Percentage of the full fare that the firm decides to offer. $\sigma=1$ corresponds to no discount and $\sigma=0$ corresponds to a 100%, or free-ride, discount.
n	Number of vehicles in the network.
m	Number of charging stations in the network, so $\lvert\mathcal{Z}\rvert=m$.
$b_m\in[0,1]$	Battery threshold for manual move. If a parked vehicle has a battery level $w<b_m$, then this vehicle is eligible for manual repositioning.
$c_m\in[0,1]$	Cost of a manual repositioning move.
$p_m\in[0,1]$	Probability that a manual repositioning move occurs.
$t_m\in[0,1]$	Mean number of periods for a manual move to be completed.
$\gamma\in\mathcal{W}$	Battery recharge rate.
$\overline{b}_i\in\mathcal{W}$	The minimum battery required to reach the nearest charging station from i. Defined as $\overline{b}_i=\min_{z\in\mathcal{Z}}\{b_{iz}\}$.
$z_i\in\mathcal{Z}$	Closest charging station to region $i\in\mathcal{R}$.
$\lambda_i\in[0,1]$	Probability of seeing a request for a vehicle in region $i\in\mathcal{R}$.
$p_{ij}\in[0,1]$	Probability of a ride starting at region $i\in\mathcal{R}$ and ending at $j\in\mathcal{R}$.
$b_{ij}\in\mathcal{W}$	Battery consumption of a ride starting at region $i\in\mathcal{R}$ and ending at $j\in\mathcal{R}$.
$d_{ij}\in\mathbb{R}^{+}$	Distance between regions $i,j\in\mathcal{R}$. We note that $d_{ii}=0$ and the distance between regions does not depend on the direction, so $d_{ij}=d_{ji}$.
$f_{ij}\in\mathbb{R}^{++}$	Revenue or fare of a ride starting at region $i\in\mathcal{R}$ and ending at $i\in\mathcal{R}$.
$t_{ij}\in\mathbb{Z}^{++}$	Duration of a ride starting at region $i\in\mathcal{R}$ and ending at $j\in\mathcal{R}$.
$\beta\in(0,1)$	Discount factor used in the dynamic programs.
$\beta_{ij}\in(0,1)$	Adjusted discount factor of a ride starting at region $i\in\mathcal{R}$ and ending at $j\in\mathcal{R}$. Defined as $\beta_{ij}=\beta^{t_{ij}}$.
$u(d,f)$	The utility gained on a ride where d is the distance between the vehicle drop-off destination and desired destination, and f is the fare for the ride.
$\mathbb{P}(Accept_{ijz})$	The probability a customer accepts the free-ride from i to z instead of paying the full fare to go to j, the desired destination. Defined as $\mathbb{P}\big[u(d_{zj},0)\geq u(0,f_{ij})\big]$.
$\mathcal{DM}$	Dollar-to-Mile ratio captures the amount of money a user would take in exchange for walking 1 mile. The units are $\frac{\$}{Mile}$.

Appendix B: *1VMC* Dynamic Program for an Arbitrary Discount

The state of a vehicle, (i,w), can be categorized into 6 cases, which we outline in Figure A.1 and Table A.2. Based on each case, the value function takes a different form and in three of the cases the system operator needs to decide whether or not to offer a discounted ride. For the *1VMC* network with a general $1-\sigma$ discount for $\sigma\in[0,1]$, the value function for each case is provided in equations (i)-(vi).

$$V(i,w)=\max_{\sigma\in[0,1]}\Bigg\{\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\Big(\mathbb{P}(Decline^{\sigma}_{ijz_j})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)$$

Figure A.1 Description of Different Cases for State (i, w) in 1VMC.


$\bar{b}_i \leq b_m$
$\bar{b}_i > b_m$
$i \notin \mathcal{Z}$
(iv)
(ii)
(i)
0
$\bar{b}_i$
b_m
100%
(iv)
(iii)
(i)
0
b_m
$\bar{b}_i$
100%
$i \in \mathcal{Z}$
(vi)
(v)
0
$\bar{b}_i$
100%
(i) Potentially *offer*, No manual move
(ii) Potentially *offer*, Manual move eligible
(iii) No *offer*, No manual move
(iv) No *offer*, Manual move eligible
(v) Potentially *offer*, Battery recharges
(vi) No *offer*, Battery recharges

Note. For a vehicle at (i, w), there are six cases depending on if the vehicle is at a charging station (i.e., $i \notin \mathcal{Z}$), if the vehicle can reach a charging station (i.e., $w \geq \bar{b}_i$), and if the vehicle is eligible for a manual move (i.e., $w \geq b_m$).

Table A.2 Description of Six Cases for State (i, w) in 1VMC

Case	Region	Charging Station Accessible	Manual Move Eligible	Description of Case
(i)	$i \notin \mathcal{Z}$	$w \geq \bar{b}_i$	$w \geq b_m$	Do not move and potentially offer.
(ii)			$w < b_m$	Move and potentially offer.
(iii)		$w < \bar{b}_i$	$w \geq b_m$	Do not move and do not offer.
(iv)			$w < b_m$	Move and do not offer.
(v)	$i \in \mathcal{Z}$	$w \geq \bar{b}_i$	-	Battery replenishes and potentially offer.
(vi)		$w < \bar{b}_i$	-	Battery replenishes and do not offer.

This table maps a vehicle's state (i, w) to one of six cases in the *1VMC* dynamic program. The first four rows are for vehicles not in charging stations stations and the last two rows are for vehicles in charging stations. The column "Charging Station Accesible" contains $w \geq \bar{b}_i$ if a vehicle at (i, w) can reach a charging station. The column "Manual Move Eligible" contains $w < b_m$ if a vehicle at (i, w) is eligible to be manually repositioned.

$$+\mathbb{P}(Accept^{\sigma}_{ijz_j})\cdot\big(\sigma f_{ij}+\beta_{iz_j}V(z_j,w-b_{iz_j})\big)\Big)\Bigg\}+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,w) \tag{i}$$

$$\begin{aligned} V(i,w) =& p_m\cdot\big(-c_m+\beta_m V(z_i,w)\big)+(1-p_m)\\ &\cdot\Bigg(\max_{\sigma\in[0,1]}\Bigg\{\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\Big(\mathbb{P}(Decline^{\sigma}_{ijz_j})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)\\ &+\mathbb{P}(Accept^{\sigma}_{ijz_j})\cdot\big(\sigma f_{ij}+\beta_{iz_j}V(z_j,w-b_{iz_j})\big)\Big)\Bigg\}+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,w)\Bigg) \end{aligned} \tag{ii}$$

$$V(i,w) =\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,w) \tag{iii}$$

$$\begin{aligned} V(i,w) =& p_m\cdot(-c_m+\beta_m V(z_i,w))+(1-p_m)\cdot\\ &\Bigg(\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,w)\Bigg) \end{aligned} \tag{iv}$$

$$\begin{aligned} V(i,w) =& \max_{\sigma\in[0,1]}\Bigg\{\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\Big(\mathbb{P}(Decline^{\sigma}_{ijz_j})\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)\\ &+\mathbb{P}(Accept^{\sigma}_{ijz_j})\cdot\big(\sigma f_{ij}+\beta_{iz_j}V(z_j,w-b_{iz_j})\big)\Big)\Bigg\}+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,\min\{w+\gamma,1\}) \end{aligned} \tag{v}$$

$$V(i,w) =\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\cdot\big(f_{ij}+\beta_{ij}V(j,w-b_{ij})\big)+\Big(1-\lambda_i\sum_{j\in\mathcal{R}(i,w)}p_{ij}\Big)\cdot\beta V(i,\min\{w+\gamma,1\}) \tag{vi}$$

Free Rides in Dockless, Electric Vehicle Sharing Systems – Online Appendix

Bobby S. Nyotta, Fernanda Bravo
UCLA Anderson School of Management, bnyotta@anderson.ucla.edu, fernanda.bravo@anderson.ucla.edu

Jacob Feldman
Olin School of Business, jbfeldman@wustl.edu

Appendix A: Example More Battery Being Less Lucrative

The following example illustrates that more battery can be less lucrative. Let us consider a network with three regions and one charging station, $\mathcal{R} = \{1,2,3\}$ and $\mathcal{Z} = \{2\}$, respectively. The arrival and transition probabilities, revenues, battery consumption, and manual repositioning parameters are defined in Table OA.1. Furthermore, we assume that users accept a free ride half of the time. We plot the optimal value function at region #1 in Figure OA.1 for two values of b_m, the battery threshold required for a vehicle to be manually repositioned. When $b_m = 50\%$, more battery is always lucrative, however when $b_m = 25\%$, more battery can be less lucrative ($w \leq 25\%$).

Table OA.1 Parameters for Example where More Battery is Less Lucrative

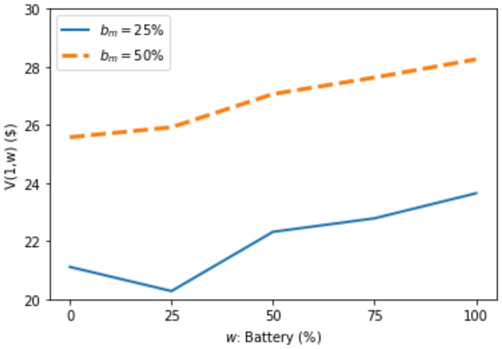
p_{ij}	1	2	3
1	1/3	1/3	1/3
2	1/3	1/3	1/3
3	1/3	1/3	1/3

f_{ij}	1	2	3
1	\$1	\$2	\$2
2	\$2	\$1	\$2
3	\$2	\$2	\$1

b_{ij}	1	2	3
1	25%	50%	75%
2	50%	25%	50%
3	75%	50%	25%

λ_1	1/3
λ_2	1/3
λ_3	1/3
p_m	0.5
b_m	25%
c_m	\$1

Figure OA.1 More Battery is Not Always Lucrative.



Appendix B: Proofs

Proof of Proposition 1. For an arbitrary region i, we show that there exist threshold battery levels w_1^i and w_2^i such that $\tilde{x}^o(i,w)=1$ if $w_1^i \le w \le w_2^i$ and $\tilde{x}^o(i,w)=0$ otherwise. Based on our construction of $\tilde{\pi}$, this is equivalent to showing that $\tilde{\pi}\in\Pi_{SOR}$. We note that if $\sum_{w\in\mathcal{W}}\tilde{x}^o(i,w)=0$, then the result holds trivially. Hence for the remainder of the proof, we assume that $\sum_{w\in\mathcal{W}}\tilde{x}^o(i,w)\ge 1$. First, we consider the case in which $\tilde{x}^o(i,\bar{b}_i)=1$ and so we know that $\sum_{w\in\mathcal{W}}\tilde{y}(i,w)=0$ by constraint (5). In this case, we claim that $\tilde{\pi}$ is the single-offer range policy where $w_1^i=\bar{b}_i$ and $w_2^i=\max\{w\in\mathcal{W}:\tilde{x}^o(i,w)=1\}$. To see this, note that constraint (4) together with the definition of w_2^i ensure that both $\tilde{x}^o(i,w)=1$ if $w_1^i\le w\le w_2^i$ and $\tilde{x}^o(i,w)=0$ if $w>w_2^i$.

Next, we consider the case in which $\tilde{x}^o(i,\bar{b}_i)=0$. Here, we see that constraint (5) ensures that there must exists $w'\in\mathcal{W}$ such that $\tilde{y}(i,w')=1$. We claim that in this case, $\tilde{\pi}$ is the single-offer range policy with $w_1^i=w'$ and $w_2^i=\max\{w\in\mathcal{W}:\tilde{x}^o(i,w)=1\}$. To show this claim, we first note that we must have $\tilde{x}^o(i,w)=0$ for $\bar{b}_i\le w<w'$ due to constraint (4) and the assumption that $\tilde{x}^o(i,\bar{b}_i)=0$ and the fact that $\tilde{y}(i,w)=0$ for a fixed region i and any $w\ne w'$. Finally, using the same argument as above, we can show that $\tilde{x}^o(i,w)=1$ if $w_1^i\le w\le w_2^i$ and $\tilde{x}^o(i,w)=0$ if $w>w_2^i$. $\square$

Proof of Theorem 1. We focus on the case where $w\ge\max\{\bar{b}_i,b_m\}$ for the definition of the value functions, which is the most common case, and note that the proof goes through in the case when $\bar{b}_i\le w<b_m$. We assume that $M\ge\max\{\tilde{Z},\max_{\pi\in\Pi_{SOR}}Z(\pi)\}$. First, we show that $\tilde{Z}\le\max_{\pi\in\Pi_{SOR}}Z(\pi)$. Let $\tilde{V}(i,w)$ be the optimal $V(\cdot)$ decision variables in *Single Threshold*, and abusing notation slightly, we let $\tilde{V}^o(i,w)$ and $\tilde{V}^{no}(i,w)$ denote the value of $V^o(i,w)$ and $V^{no}(i,w)$ at optimality. We establish that $\tilde{Z}\le\max_{\pi\in\Pi_{SOR}}Z(\pi)$ by showing that $\tilde{V}(i,w)$ is a feasible solution to *LP Policy* when $\pi=\tilde{\pi}$. First, we note that at optimality $\tilde{V}(i,w)=\min\{\tilde{V}^o(i,w)+M\tilde{x}^{no}(i,w),\tilde{V}^{no}(i,w)+M\tilde{x}^o(i,w)\}$ for $i\in\mathcal{R}\setminus\mathcal{Z}, w\ge\max\{b_m,\bar{b}_i\}$. Without loss of generality, we assume $\tilde{x}^o(i,w)=1$ and $\tilde{x}^{no}(i,w)=0$, since the other case in symmetric. Then, we have

$$\tilde{V}(i,w)=\min\{\tilde{V}^o(i,w),\tilde{V}^{no}(i,w)+M\}=\tilde{V}^o(i,w)=\mathbb{1}_{(i,w)\in S_{\tilde{\pi}}}\tilde{V}^o(i,w)+\mathbb{1}_{(i,w)\notin S_{\tilde{\pi}}}\tilde{V}^{no}(i,w).$$

The second equality follows because $M\ge\tilde{Z}\ge\tilde{V}(i,w)=\tilde{V}^o(i,w)$ and third equality follows by definition of $S_{\tilde{\pi}}$. The other constraints are trivially satisfied and hence we get that $\tilde{V}(i,w)$ is feasible to *LP Policy* when $\pi=\tilde{\pi}$ and so $\tilde{Z}=\sum_{i\in\mathcal{R}}\sum_{w\in\mathcal{W}}\tilde{V}(i,w)\le\max_{\pi\in\Pi_{SOR}}Z(\pi)$ since $\tilde{\pi}\in\Pi_{SOR}$ by Proposition 1.

Next, we show that $\tilde{Z}\ge\max_{\pi\in\Pi_{SOR}}Z(\pi)$. Let $\pi^*_{SOR}=\arg\max_{\pi\in\Pi_{SOR}}Z(\pi)$ be the optimal single-offer range free-ride policy and let $V_\pi(i,w)$ be the optimal decision variables to *LP Policy* when $\pi=\pi^*_{SOR}$. Further, for each region i, assume that w_1^{i*} and w_2^{i*} give the critical battery level threshold under π^*_{SOR}. To show the desired result, we construct a feasible solution $\hat{V}(i,w),\hat{x}^o(i,w),\hat{x}^{no}(i,w),\hat{y}(i,w)$ for each $i\in\mathcal{R}$ and $w\in\mathcal{W}$ to *Single Threshold* that achieves an objective of $Z(\pi^*_{SOR})$. First, we set $\hat{x}^o(i,w)=1$ if $(i,w)\in S_{\pi^*_{SOR}}$, $\hat{x}^o(i,w)=0$ if $(i,w)\notin S_{\pi^*_{SOR}}$. We then set $\hat{y}(i,w)=1$ if $w=\max\{w>\bar{b}_i:(i,w)\in S_{\pi^*_{SOR}}\}$ and $\hat{y}(i,w)=0$ if $w\ne\max\{w>\bar{b}_i:(i,w)\in S_{\pi^*_{SOR}}\}$. Finally, we set $\hat{V}(i,w)=V_\pi(i,w)$ and let $\hat{V}^o(i,w)$ and $\hat{V}^{no}(i,w)$ be the resulting values of $V^o(i,w)$ and $V^{no}(i,w)$. First, we show that this solution is feasible in *Single Threshold*. We trivially get that $\hat{x}^o(i,w)+\hat{x}^{no}(i,w)=1$ and that constraint (5) is satisfied by construction. To show that constraint (4) is satisfied note that we have $\hat{x}^o(i,w-\delta)=\hat{x}^o(i,w)$ if $w\notin\{w_1^{i*},w_2^{i*}-\delta\}$. If $w=w_1^{i*}$, then we get that

$\hat{x}^o(i,w) - y(i,w) = 0 \le \hat{x}^o(i,w-\delta)$ and if $w = w_2^{i*} - \delta$ then we get that $x^o(i,w-\delta) = 1$ and so the constraint must be satisfied. Hence it remains to show that $\hat{V}(i,w) \le \min\{\hat{V}^o(i,w) + M\hat{x}^{no}(i,w), \hat{V}^{no}(i,w) + M\hat{x}^o(i,w)\}$ when $w \ge \max\{b_m, \bar{b}_i\}$. We again assume without loss of generality that $\hat{x}^o(i,w) = 1$ and $\hat{x}^{no}(i,w) = 0$. We have that

$$\begin{aligned} \hat{V}(i,w) = V_\pi(i,w) = \hat{V}^o(i,w) &\le \min\{\hat{V}^o(i,w), \hat{V}^{no}(i,w) + M\} \\ &= \min\{\hat{V}^o(i,w) + M\hat{x}^{no}(i,w), \hat{V}^{no}(i,w) + M\hat{x}^o(i,w)\}, \end{aligned}$$

where the second equality follows by definition of π^*_{SOR} and the fact that we assumed that $\hat{x}^o(i,w) = 1$. The third equality follows because $M \ge Z(\pi^*_{SOR}) \ge V_\pi(i,w) = \hat{V}^o(i,w)$. Finally note that this solution attains the desired objective value of $\sum_{i\in\mathcal{R}}\sum_{w\in\mathcal{W}} \hat{V}(i,w) = Z(\pi^*_{SOR})$ and so we have established that $Z(\tilde{\pi}) = \tilde{Z} = Z(\pi^*_{SOR})$ as desired. □

Appendix C: Description of Discrete Event Simulation

In this section we provide additional details of our simulation and experiments to complement the description provided in Section 4 of the paper. First we describe how we use the historical data to estimate trip parameters, such as duration, battery consumption, and fare, between regions in the network. Then we describe how to solve for the optimal discount policies. And finally, we describe the dynamics of the simulation, which includes how the system is seeded, how trips are generated, and how manual moves take place.

C.1. Estimating Parameters

Trip Features. Using all trips in the data, we run three linear regressions to estimate a trip's mileage (m), duration (t), and battery consumption (b). We index trips from $k = 1, ..., K$ and in all regressions we use the geocoordinates of the trip's origin and destination, respectively $(i_k^{\text{lat}}, i_k^{\text{long}})$ and $(j_k^{\text{lat}}, j_k^{\text{long}})$, and the distance between the two locations, $d_{i_k j_k}$. The output to the regression in Eqs. (1)-(3) is available in Table OA.2.

$$m_k = \beta_0 + \beta_{i^{\text{lat}}} \cdot i_k^{\text{lat}} + \beta_{i^{\text{long}}} \cdot i_k^{\text{long}} + \beta_{j^{\text{lat}}} \cdot j_k^{\text{lat}} + \beta_{j^{\text{long}}} \cdot j_k^{\text{long}} + \beta_d \cdot d_{i_k j_k} + \epsilon_k \qquad \text{where } \epsilon_k \sim \mathcal{N}(0,\sigma_m) \tag{1}$$

$$t_k = \beta_0 + \beta_{i^{\text{lat}}} \cdot i_k^{\text{lat}} + \beta_{i^{\text{long}}} \cdot i_k^{\text{long}} + \beta_{j^{\text{lat}}} \cdot j_k^{\text{lat}} + \beta_{j^{\text{long}}} \cdot j_k^{\text{long}} + \beta_d \cdot d_{i_k j_k} + m_k + \epsilon_k \qquad \text{where } \epsilon_k \sim \mathcal{N}(0,\sigma_t) \tag{2}$$

$$b_k = \beta_0 + \beta_{i^{\text{lat}}} \cdot i_k^{\text{lat}} + \beta_{i^{\text{long}}} \cdot i_k^{\text{long}} + \beta_{j^{\text{lat}}} \cdot j_k^{\text{lat}} + \beta_{j^{\text{long}}} \cdot j_k^{\text{long}} + \beta_d \cdot d_{i_k j_k} + m_k + t_k + \epsilon_k \qquad \text{where } \epsilon_k \sim \mathcal{N}(0,\sigma_b) \tag{3}$$

Table OA.2 Regression Output.

Dependent Variable	β_0	i^{lat}	i^{long}	j^{lat}	j^{long}	d_{ij}	True Distance (Miles)	True Duration (Minutes)	R^2	σ: Standard Error of Regression
m	-1089.98	-0.79	-3.77	-1.81	-5.96	0.48	-	-	0.08	2.07
t	11309.20	-9.76	30.07	16.60	64.31	-8.76	11.21	-	0.55	21.30
b	-2529.54	15.31	23.38	5.49	-37.63	-0.23	4.39	0.01	0.63	7.53

We use the above regressions to generate the specific features of a ride from i to j in the simulation. Thus, using Eqs. (1)-(3) and the coefficients detailed in Table OA.2, we compute each trip's predicted mileage ($\hat{m}_{ij}$), predicted duration ($\hat{t}_{ij}$), and predicted battery consumption ($\hat{b}_{ij}$). Using $\hat{t}_{ij}$, we compute the predicted fare $f_{ij} = \$1 + \$0.15 \cdot \hat{t}_{ij}$. Since the units of $\hat{t}_{ij}$ are in minutes, the value must be converted to periods (recall that 1 period equals 2.66 min) to compute the adjusted discount rate β_{ij}.

Users Utility. To compute the utility gained from a trip, we assume the utility function $u(d,f)$ takes the form $u(d,f) = -\alpha_d \cdot d - \alpha_f \cdot f$. With this structure, the utility is decreasing in both distance and price and the maximum utility is 0. We assume $\alpha_d \sim U(0,\mathcal{DM})$ and $\alpha_f \sim U(0,1)$, and use $\mathcal{DM} = \$5$/mile as the baseline value. This value is equivalent to \$15 per hour, which will be in the minimum wage in 2023 in California, where the EVSS collaborator is based, when we assume people walk at 3 miles per hour. Under this utility model, the probability of accepting a free ride is given in Eq. (4). To see how $\mathbb{P}(Accept_{ijz_j})$ fluctuates relative to the willingness to pay and walk, in Table OA.3 we show the average value over all feasible origin-destination routes in historical ride data.

$$\mathbb{P}(Accept_{ijz_j}) = \begin{cases} \frac{f_{ij}}{d_{z_j j}} \cdot \frac{1}{2 \cdot \mathcal{DM}} & , \frac{f_{ij}}{d_{z_j j}} \leq \mathcal{DM} \\ 1 - \frac{\mathcal{DM}}{2} \cdot \frac{d_{z_j j}}{f_{ij}} & , \text{otherwise} \end{cases} \tag{4}$$

Table OA.3 Average Probability of Accepting a Free-Ride Offer for Various Dollar-to-Mile ($\mathcal{DM}$) Values.

$\mathcal{DM}$ (\$/Mile)	0.5	1	2	5	10	20
Avg. $\mathbb{P}(Accept_{ijz_j})$	94.6%	91.9%	86.6%	72.5%	58.9%	49.3%

C.2. Computing Policies

Using the sets $\mathcal{R}, \mathcal{Z}$, and $\mathcal{W}$, and the parameters $\lambda_i, p_{ij}, \hat{t}_{ij}, \hat{b}_{ij}, f_{ij}, \beta_{ij}, \beta, \mathbb{P}(Accept_{ijz}), c_m, b_m, p_m, t_m$, and γ, we solve for the optimal 1VMC-SOR policy, by solving the MIP described in *Single Threshold*, and we solve for the optimal 1VMC-50 policy via linear programming. We found that value iteration and policy iteration did not scale well to large, realistic instances. Note that policies need to be recomputed anytime any of these parameters change. We provide a description of how we compute the NVMC-SALP policy in Appendix F.

C.3. Generating Trips in Simulation

A new trip is generated as follows,

- Sample inter-arrival time τ from the Beta Prime($\alpha = 0.92, \beta = 4.07$, location $= 0.01$, and scale $= 8.86$) distribution. We set the ride request's arrival time to $T + \tau$, where T is the simulation's current time.
- Sample i, the trip's origin region, from the discrete distribution of λ_i.
- Given the trip's origin region i, sample the destination region j from the discrete distribution of p_{ij}.
- Given i and j, we randomly generate the ride features (we use the tilde above m, t, and b to denote the trip specific features) by using the regression output in Table OA.2 as follows:

—We sample a random error $\epsilon \sim \mathcal{N}(0, \sigma_m)$. Using the predicted mileage value $\hat{m}_{ij}$ computed using the regression in Eq. (1) and the corresponding coefficients in Table OA.2, we set $\tilde{m}_{ij} = \hat{m}_{ij} + \epsilon$.

—Taking $\tilde{m}_{ij}$ to be the trip's true mileage, we use the regression in Eq. (2) and the corresponding coefficients in Table OA.2 to compute the trip's duration $\tilde{t}_{ij}$. In order to ensure that the trip's duration and mileage are related, we do not sample an error term when generating the trip's duration.

—Taking $\tilde{m}_{ij}$ and $\tilde{t}_{ij}$ to be respectively the trip's true mileage and duration, we use the regression in Eq. (3) and the coefficients in Table OA.2 to compute the trip's battery consumption $\tilde{b}_{ij}$. Again, we do not sample an error term when generating $\tilde{b}_{ij}$ to ensure that the trip's mileage, duration, and battery consumption are correlated.

—Finally, using $\tilde{t}_{ij}$, the duration of the trip in minutes, we compute the fare for the trip $f_{ij} = \$1 + \$0.15 \cdot \tilde{t}_{ij}$. This corresponds to a base fare of \$1 per trip plus a per-minute charge of \$0.15/minute.

- To generate the customer's willingness to walk and pay, we sample $\alpha_d \sim U(0, \mathcal{DM}), \alpha_f \sim U(0,1)$.

Therefore, a new trip request is characterized by the following information: $(T+\tau, i, j, \tilde{t}_{ij}, \tilde{b}_{ij}, f_{ij}, \alpha_d, \alpha_f)$. We note that we generate ride requests dynamically in the simulation, that is, at any time T, we can generate a stream of future demand requests. This approach is in contrast to generating all demand requests in advance, which turned out to be much more computationally expensive.

C.4. Simulation Dynamics

Initializing System. We track time in seconds and the simulation starts at $T = 0$. We randomly assign $n = 300$, fully-charged vehicles, to regions according to λ_i. We store the location and battery of each vehicle in a System State (SS) table. We also track in SS whether or not a vehicle is busy on a ride or being manually repositioned. Thus, vehicles status is either *idle* or *busy*. We seed the system with a single ride based on the procedure described in Appendix C.3, and store the request arrival time and the corresponding ride information in a Pending Events (PE) list. In PE we also keep track of in-progress rides and manual moves. We denote $Time\_Event$ as the time when the pending event is scheduled to occur in the simulation, that is, the request arrival time for a new trip, the completion time for an in-progress ride, and the completion time for an in-progress manual move. While each policy we test runs in its own simulation and has its own SS table and PE list, they all start under the same configuration and see the same stream of demand.

Run Simulation. While $T \leq 100$ days, we sequentially perform the following six steps:

Step_1 Find "next event" (NE). The NE is the event in the PE list with the minimum $Time\_Event$. Thus, we sort the PE list in ascending order of $Time\_Event$ and update $T' \leftarrow T$ and $T \leftarrow \min\{Time\_Event\}$, where T' is the time of the previous event. If NE occurs on the next day, then we record the status of the network and the corresponding performance measures that we track. Before we update the system based on the NE type (Steps 4-6), we update the battery of idle vehicles at charging stations (Step_2) and potentially schedule a manual move for a vehicle with remaining battery below b_m (Step_3).

Step_2 Update battery. For idle vehicles in table SS that are at charging stations, we update the battery level at the rate of corresponding to 5 hours for a full charge for a time interval corresponding to $T - T'$.

Step_3 Schedule manual move of vehicles. We check SS for idle vehicles not parked at charging stations that have remaining battery below the move threshold b_m. If there are any move-eligible vehicles, then a crew member will arrive with probability p_m. If there is more than one move-eligible vehicle, then we select one to move at random. The move is scheduled as an in-progress move on the PE list with $Time\_Event$ set to $T + t'_m$, where t'_m is drawn from a truncated normal distribution with a mean of 4 hours and standard deviation of 30 minutes. The selected vehicle will be moved to the nearest charging station at $Time\_Event$. The SS table is updated so the selected vehicle is shown as *busy* until the repositioning is completed.

Step_4 The NE is a new request for a ride.

(a) Check for available vehicles. We search the SS table for idle vehicles at the arrival region and in the regions immediately around the arrival region. If there are no available vehicles, we track the unmet demand due to not having a vehicle present. If there are vehicles, but they do not have enough battery to fulfill the ride, we track the unmet demand due to not having enough battery. Then we go to step (c).

If there are available, idle vehicles at the arrival region, or in the neighborhood of the arrival region, that have enough battery to complete the ride, then we assume users select the highest charged vehicle. We update the status of the vehicle in the SS table to be *busy*.

(b) Users' choice of drop-off location. Regardless of the policy in place, at the time of renting the vehicle, users weigh the utility of ending the ride at their intended destination with the utility of ending the ride at the charging station closest to their intended destination. Based on this decision, we schedule an in-progress ride on the PE list with a completion time and end destination of either $Time\_Event = T + \tilde{t}_{ij}$ and j or $T + \tilde{t}_{iz_j}$ and z_j. We make several distinctions in how customers choose their end destination depending on the policy being tested in the simulation:

Fine-Based policy: We note that the user only has to decide on their end destination if (i) their end destination is not at a charging station and (ii) the remaining battery at the end of the ride will be less than b_m, the manual move threshold. To decide on the end destination, the user computes her utilities $u(0, f_{ij} + fine)$ and $u(d_{z_j j}, f_{ij})$ using α_d, α_f, the customer's willingness to walk and pay parameters. If $u(0, f_{ij} + fine) \leq u(d_{z_j j}, f_{ij})$, the user chooses z_j as the end destination and the system accrues f_{ij} in revenue. Otherwise, the end destination is the user's intended destination, and the system collects f_{ij} in revenue plus the fine. However, we do not count the fine as revenue since this money will be used to subsidize the cost of a manual move. Regardless of what the user ultimately decides, we track that the customer had a decision to make, whether the customer paid the fine or not, and the utility gained in the chosen action.

Free-ride policy: First we note that the user only has a decision to make if the system actually offers a free ride to a charging station. The system does this by checking if, given the vehicle's location and the vehicle's remaining battery, offering is the optimal action. In other words, the operator offers a free ride if the state describing the location and remaining battery of the vehicles to be rented $(i, w) \in S_{\tilde{\pi}}$. If a free ride is offered, then the user computes her utilities $u(f_{ij}, 0)$ and $u(0, d_{z_j j})$ using α_d, α_f, the customer's willingness to walk and pay parameters. If $u(f_{ij}, 0) \leq u(0, d_{z_j j})$, the customer chooses to accept the free ride to the charging station z_j and we update the ride's end destination to z_j *and* the fare of the ride to \$0, otherwise we leave the ride information the same. We track that a free ride was offered, whether the customer accepted, and the utility gained in the chosen action. We note that under the 1VMC-50 policy, users also consider the utility of a 50% discounted-ride option by computing $u(0.5 \cdot f_{ij}, d_{z_j j})$. If this option results in the highest utility, then we update the ride's end destination to z_j and the fare paid to $0.5 \cdot f_{ij}$.

(c) Update PE list. First, we remove this arrival request event from the PE list. Next, we add a new arrival request onto the PE list per the process described in Appendix C.3.

Step_5 The NE is the completion of an in-progress ride:

(a) Update the SS table with the vehicle's region, remaining battery, and availability (to *idle*).

(b) Update PE list. We remove this in-progress ride event from the PE list. We track that a ride was completed and the fare collected by the system.

Step_6 The NE is the completion of an in-progress manual move. We update the SS table with the vehicle's new destination, which is a charging station, and allow the vehicle to be eligible for rides. We track that a manual move occurred in the metrics.

Appendix D: Experiment Parameter_Sensitivity Description and Results

In our second numerical experiment, Parameter_Sensitivity, we use a synthetic demand scenario where $\lambda_i = p_{ij} = \frac{1}{|\mathcal{R}|}$ and generate 8 parameter instances of an EVSS network by varying four parameters: c_m, b_m, p_m, and $\mathcal{DM}$. Table OA.4 describes each instance and the parameters values we vary from the baseline. For each instance, we report the performance of four policies: the optimal free-ride, the single-offer range (1VMC-SOR), the policy where the system operator can choose between offering a free ride, a 50% discounted ride, and not offering at all (1VMC-50), the Fine-Based policy (FB), and the "never offer" policy (NO).

Tables OA.5-OA.8 show how each policy performed across different metrics. The values presented have been averaged over the last 30 days of the 100-day simulation. Since the network we consider is fairly large, we use the first 70 days as a warm-up period. The results provide a comparative statics analysis of each policy that shows the impact that increasing each of the 4 parameters has on several performance metrics.

Table OA.4 Description of Each Instance

Instance	Varying Parameter	c_m	b_m	p_m	$\mathcal{DM}$
1	Base	\$25	0.2	20%	5
2	c_m	\$5	0.2	20%	5
3		\$50			
4	b_m	\$25	0.05	20%	5
5			0.10		
6	p_m	\$25	0.2	5%	5
7				10%	
8	$\mathcal{DM}$	\$25	0.2	20%	0.5
9					20

Appendix E: Experiment Demand_Sensitivity Description and Results

In our third numerical experiment, Demand_Sensitivity, we vary the demand parameters, λ_i and p_{ij} and fix the operational parameters to their baseline values.We specifically allow the arrival and transition probabilities to be either clustered close (C) to charging stations, spread uniformly (U) across the entire service area, or spread far (F) from charging stations. In total, this gives us 9 synthetic demand scenarios.

Tables OA.9-OA.11 show how each policy performed in each of the nine demand scenarios across several metrics. The values presented have been averaged over the last 30 days of the 100-day simulation. Since the

Table OA.5 Parameter_Sensitivity Results: Daily Revenue and Rides Fulfilled

		Average Daily Revenue (\$)				Rides Fulfilled per Day			
#	Vary	1VMC-SOR	1VMC-50	FB	NO	1VMC-SOR	1VMC-50	FB	NO
1	Base	1,185.43	1,210.28	1,527.35	1,570.66	231.55	217.60	232.12	238.52
2	c_m	1,369.88	1,393.94	1,560.42	1,589.95	238.47	235.90	236.82	241.35
3		1,074.36	1,107.87	1,539.40	1,556.21	217.48	202.70	233.74	236.28
4	b_m	1,261.70	1,273.01	966.02	984.79	248.89	230.11	160.77	163.49
5		1,240.02	1,262.29	1,766.93	1,777.30	244.04	227.79	277.56	278.65
6	p_m	1,236.15	1,259.59	1,544.10	1,024.16	233.86	221.90	234.54	159.63
7		1,243.13	1,280.04	1,531.49	1,577.63	235.47	225.65	232.89	239.95
8	$\mathcal{DM}$	1,267.01	1,358.02	1,519.21	1,538.23	232.54	227.37	230.60	233.23
9		1,194.91	1,236.19	1,548.15	1,558.30	237.87	235.14	234.81	236.33

Table OA.6 **Parameter_Sensitivity Results: Unmet Demand per Day, due to Vehicle Availability and Insufficient Battery**

		Unmet Demand per Day (Vehicle Availability)				Unmet Demand per Day (Insufficient Battery)				Moves per Day			
#	Vary	1VMC -SOR	1VMC -50	FB	NO	1VMC -SOR	1VMC -50	FB	NO	1VMC -SOR	1VMC -50	FB	NO
1	Base	196.8	212.8	189.4	183.2	4.3	2.3	11.1	11.0	4.84	2.95	4.90	47.79
2	c_m	184.8	191.6	185.1	180.3	9.3	5.1	10.7	11.0	18.53	8.72	4.96	48.12
3		212.5	228.6	188.1	185.8	3.1	1.8	11.3	11.0	3.09	2.04	4.86	47.47
4	b_m	141.8	175.8	39.1	39.6	39.0	23.7	229.8	226.6	2.99	1.70	13.16	27.85
5		175.3	196.9	84.3	84.3	12.1	6.8	69.6	68.5	3.00	1.64	11.99	48.21
6	p_m	192.3	207.4	186.7	118.4	6.0	2.8	10.8	154.0	7.05	3.89	4.82	31.32
7		191.7	204.5	189.6	180.0	5.9	2.9	10.6	13.1	7.02	3.82	4.72	47.53
8	$\mathcal{DM}$	191.2	197.7	189.6	187.2	7.5	6.2	11.1	10.9	8.50	5.59	3.58	47.18
9		189.3	192.2	186.2	184.6	4.7	4.4	10.8	10.9	12.05	11.11	9.67	47.40

Table OA.7 **Parameter_Sensitivity Results: Offers per Day, Accepts per Day, and Average Utility per Offer**

		Offers per Day				Accepts Per Day				Average Utilty per Offer			
#	Vary	1VMC -SOR	1VMC -50	FB	NO	1VMC -SOR	1VMC -50	FB	NO	1VMC -SOR	1VMC -50	FB	NO
1	Base	63.70	85.78	51.10	0.00	53.88	62.75	46.20	0.00	-0.69	-2.09	-5.09	0.00
2	c_m	40.21	72.50	51.83	0.00	33.51	51.35	46.87	0.00	-0.67	-2.19	-5.08	0.00
3		65.82	87.28	51.58	0.00	55.99	64.09	46.71	0.00	-0.69	-2.10	-5.06	0.00
4	b_m	68.72	90.43	30.14	0.00	58.03	66.18	16.98	0.00	-0.69	-2.07	-7.84	0.00
5		68.26	89.82	52.41	0.00	57.49	65.95	40.42	0.00	-0.69	-2.07	-6.01	0.00
6	p_m	57.73	81.61	50.84	0.00	48.41	59.08	45.99	0.00	-0.69	-2.12	-5.07	0.00
7		58.34	83.09	50.64	0.00	49.09	60.36	45.91	0.00	-0.68	-2.13	-5.05	0.00
8	$\mathcal{DM}$	45.58	50.50	51.19	0.00	44.83	48.85	47.62	0.00	-0.08	-1.64	-4.49	0.00
9		102.01	154.05	51.58	0.00	54.00	63.36	41.88	0.00	-1.78	-2.49	-6.98	0.00

Table OA.8 **Parameter_Sensitivity Results: Average Charge of Fleet and Proportion of Fleet at Charging Stations.**

		Average Battery (w.o. vehicles at Charging Stations)				Proportion of Fleet at Charging Stations			
#	Vary	1VMC-SOR	1VMC-50	FB	NO	1VMC-SOR	1VMC-50	FB	NO
1	Base	0.55	0.61	0.49	0.44	0.78	0.81	0.76	0.71
2	c_m	0.48	0.54	0.49	0.44	0.74	0.76	0.75	0.71
3		0.58	0.64	0.48	0.44	0.81	0.84	0.75	0.72
4	b_m	0.43	0.52	0.11	0.11	0.67	0.75	0.06	0.06
5		0.52	0.59	0.31	0.30	0.73	0.79	0.42	0.40
6	p_m	0.53	0.60	0.49	0.09	0.76	0.80	0.75	0.11
7		0.53	0.60	0.49	0.44	0.76	0.79	0.75	0.70
8	$\mathcal{DM}$	0.51	0.52	0.49	0.44	0.76	0.77	0.76	0.72
9		0.55	0.57	0.48	0.44	0.76	0.78	0.75	0.72

network we consider is fairly large, we use the first 70 days as a warm-up period. In this experiment we only test the 1VMC-SOR policy since the previous experiment indicated the 1VMC-SOR and 1VMC-50 policy perform similarly. We present the results to Demand_Sensitivity in Tables OA.9-OA.11.

Table OA.9 Demand_Sensitivity Results: Daily Revenue and Rides Fulfilled

		Average Daily Revenue ($)			Rides Fulfilled per Day			Moves per Day		
λ_i	p_{ij}	1VMC-SOR	FB	NO	1VMC-SOR	FB	NO	1VMC-SOR	FB	NO
C	C	1,333.34	1,791.93	1,802.19	264.69	271.90	273.08	5.68	5.82	56.96
U		1,168.79	1,529.69	1,552.73	227.61	232.16	235.48	5.07	4.88	47.24
F		721.29	855.68	859.14	129.39	129.73	130.18	5.92	2.43	24.28
C	U	1,286.68	1,753.53	1,760.54	259.93	266.63	267.64	4.75	5.74	56.05
U		1,142.77	1,535.49	1,563.68	226.35	233.10	237.25	4.17	4.88	47.70
F		696.30	845.88	857.82	127.98	128.33	130.17	4.20	2.31	24.18
C	F	1,229.33	1,767.13	1,797.49	254.52	268.56	273.03	3.40	5.91	56.94
U		1,081.99	1,560.62	1,552.91	219.61	237.22	235.93	3.00	4.86	47.58
F		398.11	534.27	550.09	79.53	82.05	84.32	2.18	2.43	19.28

Table OA.10 Demand_Sensitivity Results: Unmet Demand per Day (due to Vehicle Availability and Insufficient Battery) and Proportion of Fleet at Charging Stations

		Unmet Demand per Day (Vehicle Availability)			Unmet Demand per Day (Insufficient Battery)			Proportion of Fleet at $\mathcal{Z}$		
λ_i	p_{ij}	1VMC-SOR	FB	NO	1VMC-SOR	FB	NO	1VMC-SOR	FB	NO
C	C	162.90	147.84	146.84	4.95	12.81	12.62	0.56	0.52	0.47
U		199.83	188.76	185.40	4.41	10.93	10.98	0.78	0.75	0.72
F		297.92	295.82	295.36	3.90	5.67	5.69	0.85	0.84	0.83
C	U	166.88	151.60	150.58	4.21	12.80	12.81	0.58	0.52	0.47
U		200.99	187.42	183.11	3.89	10.71	10.88	0.79	0.75	0.71
F		299.76	297.13	294.81	3.14	5.43	5.90	0.85	0.85	0.83
C	F	174.56	150.85	146.46	3.16	12.83	12.75	0.63	0.54	0.48
U		209.63	183.81	185.03	2.88	11.08	11.15	0.80	0.75	0.72
F		349.54	343.91	341.09	1.79	4.89	5.44	0.94	0.94	0.92

Table OA.11 Demand_Sensitivity Results: Offers per Day, Accepts per Day, and Average Utility per Offer

		Offers per Day			Accepts Per Day			Average Utilty per Offer		
λ_i	p_{ij}	1VMC -SOR	FB	NO	1VMC -SOR	FB	NO	1VMC -SOR	FB	NO
C	C	76.66	62.54	0.00	64.81	56.73	0.00	-0.69	-5.06	0.00
U		62.18	51.16	0.00	52.38	46.29	0.00	-0.69	-5.08	0.00
F		25.80	26.27	0.00	21.59	23.83	0.00	-0.69	-5.09	0.00
C	U	78.68	61.41	0.00	66.37	55.66	0.00	-0.69	-5.07	0.00
U		64.43	51.37	0.00	54.52	46.49	0.00	-0.69	-5.11	0.00
F		28.28	25.95	0.00	23.80	23.63	0.00	-0.68	-5.05	0.00
C	F	82.14	61.51	0.00	69.38	55.61	0.00	-0.69	-5.09	0.00
U		67.21	52.33	0.00	56.92	47.46	0.00	-0.69	-5.04	0.00
F		23.99	19.25	0.00	19.06	16.80	0.00	-0.90	-5.50	0.00

Appendix F: NVMC-SALP: Description of Basis Functions and Weight Vector

In Table OA.12, we provide a description of the 10 basis functions we use for solving NVMC-SALP. Each basis function captures valuable information about the system state and relates to the percentage of the fleet that is available, the geographic dispersion of vehicles, and the average charge of the fleet.

Next, we specify how we sample states and choose the constraint violation budget when solving the NVMC-SALP. Solving the resulting linear program yields a vector of basis function weights, which we provide in Table OA.12. It is these weights that we ultimately use to approximate the value function for a multi-vehicle network. Under the NVMC-SALP policy in the simulation, we use the approximate value functions to compute the expected discounted revenue of the $NoOffer$ and $Offer$ actions and choose the revenue-maximizing decision. In this sense, we say that the policy is greedy with respect to the value function approximations.

To generate the NVMC-SALP policy, we sample 500 states. In sampling a single state, we assume that each of the n vehicles are randomly chosen to be either busy or idle. If a vehicle is idle, then it is located in region i with probability λ_i. We assume that each vehicle has a remaining battery that is randomly drawn from $\mathcal{W}$. If a vehicle is busy, we sample the remaining time busy from a triangular distribution with mean 53 minutes and support [1 minutes, 180 minutes], the minimum and maximum trip duration in the data.

The violation budget restricts the expected constraint violation over all 500 states. Per the guidance of Desai et al. (2012), we test several budget values ranging from 0 to 0.05 and then test the performance out of sample and in the simulation. For extreme values, the resulting policy would offer too liberally, resulting in a policy that forgoes too much revenue by offering free rides, and poor dispersion of vehicles since there would be a glut of vehicles at charging stations.

Table OA.12 Description of Basis Functions used in NVMC-SALP and the Corresponding Weight Vectors.

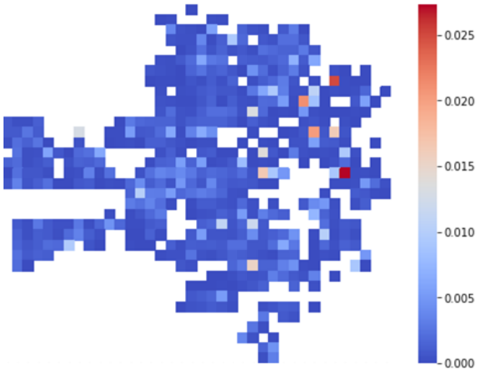
Basis Function	Description	**r**
1	% of fleet on trips busy on rides	-1846.83
2	% of busy vehicles available in 1 period	403.41
3	Mean fleet charge (available and non-available vehicles)	5219.46
4	Mean fleet charge (available and non-available vehicles) without vehicles located in $\mathcal{Z}$	-4679.12
5	% of available vehicles with $\lvert\mathcal{R}(i,w)\rvert = 0$	-132.896
6	% of available vehicles with $\lvert\mathcal{R}(i,w)\rvert > 0, w < \bar{b}_i$	408.34
7	% of available vehicles with $\lvert\mathcal{R}(i,w)\rvert > 0, w \geq \bar{b}_i$	639.047
8	% of regions with available vehicles	1729.18
9	% of available vehicles located in $\mathcal{Z}$	188.071
10	% of available vehicles eligible for a move (with battery $< b_m$)	2863.42

Appendix G: Heatmap of Historical Arrival Probability.

References

Desai V, Farias V, Moallemi C (2012) Approximate dynamic programming via a smoothed linear program. *Operations Research* 60(3):655–674.

Figure OA.2 Heatmap of Historical Arrival Probability.


0.025
0.020
0.015
0.010
0.005
0.000