An Online Cost Allocation Model for Horizontal Supply Chains

Han Zou, Maged M. Dessouky, and Shichun Hu

Daniel J. Epstein Department of Industrial and Systems Engineering, University of Southern California,
3715 McClintock Avenue, GER240, Los Angeles, CA 90089-0193, United States
Han Zou hanzou@usc.edu, Maged M. Dessouky maged@usc.edu, Shichun Hu shichunh@usc.edu

Abstract

Emerging technologies in cargo shipping delivery have provided a way to facilitate horizontal cooperation in the transportation of goods to reduce the shipping cost of the cooperating firms, but an open question in this cooperation is how to allocate the costs fairly among the participants. In this paper, we focus on routing in real time a fleet of capacitated vehicles to satisfy requests submitted by a set of customers with some of the requests unknown while assigning the service cost fairly among the requested customers. We propose a Hybrid Proportional Online Cost-Sharing (HPOCS) mechanism to tackle the cost-sharing problem and analyze its performance using simulation instances. Although HPOCS does satisfy the desirable properties, namely online fairness, budget balance, immediate response, individual rationality and ex-post incentive compatibility, it has a few drawbacks in certain scenarios. Therefore, we make two extensions to HPOCS: 1) we introduce the idea of discounts to encourage customers to request in advance to facilitate efficient vehicle routing; 2) we incorporate periodical re-optimization within the dynamic vehicle routing framework. Experimental analysis are made in both extensions to see the tradeoff between the performance and the loss of certain desirable properties.

Keywords: *Online Cost-Sharing, Dynamic Vehicle Routing*

1 Introduction

The logistics sector as it is today functions in a way that is economically, environmentally, and socially unsustainable (Montreuil, 2011). In order to compete effectively against their peers, companies have relied on internal optimization to reduce operating costs, but have overlooked opportunities for external cooperation. As a result, the logistics sector has become highly fragmented, with each supplier developing and operating its own distribution network that sees low capacity usage, high energy consumption, and high greenhouse gas emission across the entire system (Montreuil, 2011).

As opportunities for internal optimization are becoming fully exploited, fierce competition drives companies to focus on reducing costs of non-value adding activities (Skjoett-Larsen, 2000), especially logistic activities. Emerging technologies in cargo shipping delivery have provided a way to facilitate horizontal co-operation in the transportation of goods to reduce the shipping cost of the cooperating firms. In the trucking industry, many Uber analogue services have come into existence after the success of Uber and Lyft in the U.S. and DiDi in China. Companies like GoShare and Traansmission are utilizing Internet and big data to help the industry realize a horizontal supply chain. The concept of horizontal cooperation sees potential benefits (Cruijssen et al., 2007) and is formally defined to be the cooperation between businesses operating at the same level(s) in the market. When applying to the logistics sector, horizontal cooperation could refer to the pooling of freight transportation networks and sharing of customers. External cooperation allows consolidation of vehicle capacity, delivery routes, and shipment orders among different suppliers or logistic service providers, thus creating a unified logistics network that sees increased capacity usage, reduced energy usage, pollution, and operating costs. For example, a case study of the Swedish forest industry has shown that potential savings of cooperation among several forest companies operating in the same region are large, often in the range of 5 to 15 percent (Frisk et al., 2010). A shared transportation network also reduces the total truck miles, which in turn reduces the usage of the road infrastructure that it shares with passenger traffic. Similarly, reduced freight traffic helps alleviate traffic congestion and the safety threat it poses on passenger traffic. Horizontal cooperation would not only generate savings for companies already in business, but also lower the potential barrier for new (and possibly small) businesses to enter the market.

Besides, operations in any real world transportation network contain a fairly high level of uncertainties including variable waiting and travel times due to traffic congestion, arrival of new service requests, cancellation of existing requests, unknown demand sizes, etc. Under changing and gradually revealed information, the problem of designing real-time collection and/or delivery routes from one or several depots to a set of geographically dispersed customers falls in the scope of the Dynamic Vehicle Routing Problem (DVRP). The DVRP derives from the Vehicle Routing Problem (VRP) when some element of the problem becomes non-deterministic; for instance, randomness exists in the probability of customer realization which is the source of randomness in our studied DVRP (Ghiani et al., 2012; Furuhata et al., 2013; Laporte, 2009; Pillac et al., 2013).

One crucial component of a shared transportation system is the method used to allocate costs and/or savings to each participant in the system. A cost-sharing mechanism serves as the basis for any economic analysis of horizontal cooperation. While the DVRP problem focuses on minimizing the total travel distance, a cost allocation problem's purpose is about how to fairly share the total cost among customers. However, the cost allocation problem in the vehicle routing context remains rarely studied in the literature, especially for the dynamic case discussed above. For a "static" cost-sharing problem in which the set of players and the cost function are both known and deterministic, Moulin mechanisms (Moulin, 1999) and acyclic mechanisms (Mehta et al., 2009) are among the most studied families of cost-sharing mechanisms. In the context of vehicle routing problems, a "static" cost-sharing problem means that the set of customers to be served is known and the optimal total cost can be calculated. Unfortunately, neither of these two assumptions holds in the dynamic vehicle routing problem we study.

Little work has been conducted on designing online cost-sharing mechanisms that work when the set of players are gradually revealed, instead of known beforehand. Even less work on cost allocation has been done in the vehicle routing context. The majority of this subset of work has assumed a static operating environment, in which the tasks of designing vehicle routes and allocating costs can be tackled separately and independently. Thus, there is a need for a unified solution approach that combines dynamic vehicle routing with online cost allocation for dynamic cost-sharing transportation systems. It is important to point out that the problem of dynamically routing vehicles and the problem of real-time cost allocation are highly interdependent and must be considered simultaneously. In particular, the vehicle routes depend on whether the new customers accept or decline the quote for service, and the quote (shared cost) in turn depends on how vehicle routes are designed and what is the expected total cost of such routes. As much as these two problems are intertwined with each other, the contribution of this paper is to develop a cost-allocation mechanism that satisfies certain desirable properties based on the literature. Since the contribution of the paper is in the cost mechanism, we use standard techniques in the literature to solve the routing algorithm. That is, we provide a heuristic for solving the DVRP such that our cost allocation method which distributes the total cost calculated by the DVRP algorithm satisfies well-established desirable properties.

The rest of the paper is organized as follows. In Section 2, a literature review of the relevant problems is presented and our contribution to the literature is summarized. Section 3 formally defines the problem. In Section 4, we introduce the Hybrid Proportional Online Cost-Sharing (HPOCS) mechanism. We introduce a DVRP algorithm that helps illustrate the performance of our proposed cost-sharing mechanism. We then prove that HPOCS satisfies all of the desirable properties we propose. Section 4.2 and Section 4.3 present two extensions of the HPOCS mechanism that improve the performance of the baseline model. We then analyze the base mechanism HPOCS and the two extension mechanisms via experiments in Section 5. We conclude in Section 6.

2 Literature Review

In this section, we review the literature relevant to our research. We first focus on previous work on cost-sharing methods, then review studies on cost allocation problems in the domain of transportation.

2.1 Cost-sharing Methods

A cost allocation problem specifies a set of players who request services that require a common and limited resource. Each player has a private, non-negative valuation for the service. This valuation is sometimes referred to as the willingness-to-pay value or the bid of the player. A cost function is defined on all subsets of players. The value of the function usually denotes the minimum total cost of serving the corresponding subset of players. The objective is to determine the cost allocated (or the price charged) to each player and the subset of players who are willing to participate in the contention given the prices. The final solution needs to not only specify the membership of the contention, but also provide exact ways to facilitate such a contention in the context of the problem (Mehta et al., 2009). For example, to solve the cost allocation problem corresponding to a vehicle routing problem, the final solution needs to specify the group of customers to participate in the cooperation, a routing schedule that accommodates the same group of customers, and the exact cost share for each customer in the group.

One approach for solving the cost allocation problem is to design a cost-sharing mechanism that incentivizes all players to participate in the cooperation. A cost-sharing mechanism needs to define an algorithm to calculate the shared cost for each player, and a process to determine the subset of players who end up participating in the cooperation. During this process, the algorithm compares the shared cost of each player with its willingness-to-pay value; only the players whose quotes are no larger than their willingness-to-pay values accept the quotes and receive service.

Researchers have focused on studying three desired properties of cost-sharing mechanisms, namely truthfulness (strategyproofness), budget balance, and economic efficiency (Moulin, 1999; Mehta et al., 2009). Truthfulness (strategyproofness) requires that no player can strictly increase its utility by misreporting its valuation for the service. The budget balance property requires that the sum of the prices charged to each participant equals to the total operating cost of facilitating the cooperation. Economically efficient mechanisms are those maximizing the welfare of all players in the problem, not only those who end up participating in the contention (Mehta et al., 2009). Unfortunately, no mechanism could simultaneously satisfy all of the above-mentioned constraints, as has been proved by Green et al. (1976) and Roberts (1979).

Thus researchers have focused on developing cost-sharing techniques that relax at least one of the constraints. Approximate measures have also been proposed on budget balance and economic efficiency in Roughgarden and Sundararajan (2009).

The only known general technique for designing truthful and approximately budget-balanced cost-sharing mechanisms is due to Moulin (1999) and Moulin and Shenker (2001). Despite the fact that designing such mechanisms is highly non-trivial and that the Moulin mechanisms have gained significant attention and seen applications in a wide range of cost-sharing problems (Bleischwitz and Monien, 2009; Gupta et al., 2008; Li et al., 2014), recent work in the literature have criticized their poor performance in terms of budget-balance and economic efficiency (Mehta et al., 2009; Immorlica et al., 2008). Thus, new families of cost-sharing mechanisms have been proposed, among which is the acyclic mechanism by Mehta et al. (2009).

2.2 Cost Allocation in Transportation

As transportation costs continue to increase due to increased competition, horizontal collaboration in the logistics sector has received increasing attention from both the research community and players in the industry. In the context of transportation, horizontal cooperation refers to the pooling of transportation capacity and customer demands among businesses operating at the same level(s) in the market (Cruijssen et al., 2007). A cost-sharing transportation system is formed as a result. One crucial component of such a system is the allocation of total operating costs and/or savings to each participant in the system.

The work by Anderson and Claus (1976) represents one of the earliest attempts to study the cost allocation problem in transportation collaboration . The authors studied and compared multiple basic cost allocation methods as applied to a minimum cost network problem. In particular, the authors showed that the average cost-sharing, unit (per mileage) cost-sharing and marginal cost-sharing all suffer from various inefficiencies when applied naively. For example, average cost-sharing cannot guarantee that each rational player will participate in the cooperation, while unit mileage pricing cannot prevent subgroups of users to form coalitions outside the grand coalition.

Cooperative Game Theory (CGT) appears to be one of the popular approaches for solving cost allocation problems in transportation research. Many CGT solution concepts have been studied, including the Shapley value (Shapley, 1953; Krajewska et al., 2008), the core and related concepts (Gillies, 1953; Drechsel and Kimms, 2010, 2011), the nucleolus (Schmeidler, 1995; Liu et al., 2010), and the $\tau-$value methods (Tijs and Driessen, 1986).

Other streams of research exist that study the cost allocation problem in transportation outside the scope of CGT. Sayarshad and Gao (2018) proposed a dynamic pricing scheme for a multi-server queue incorporating social welfare. The research focused on designing a competitive on-demand mobility model that employs a Markov decision process to increase social welfare, which can be applied to the pricing of flexible transit systems. Liu and Li (2017) studied the problem of the morning commute and transformed it into a pricing scheme design problem for ridesharing.

It can be easily shown that typical cost-sharing mechanisms such as proportional cost-sharing and marginal cost-sharing fail to possess desired properties when adapted naively to the dynamic setting. Indeed, the problem of allocating costs in a real-time cost-sharing transportation system is highly non-trivial and is ranked among the top impediments for successful horizontal cooperation (Cruijssen et al., 2007). The research on designing online and dynamic cost-sharing mechanisms for transportation systems have been very limited. A major line of research considering the competitive pricing problem in a dynamic transportation system is due to Figliozzi et al. (2003, 2007, 2004). The problem is framed as a sequential auction marketplace where new customer orders arrive stochastically and the logistics service provider must offer a competitive price bid to win the order from its competitors. New orders arrive at the same time when existing orders are being served. Each order served generates a reward. The objective is to maximize the profit as measured by the total rewards collected minus the total transportation cost. The authors developed a stochastic dynamic programming-based formulation that solves for the optimal price whenever a new order arrives.

The work by Furuhata et al. (2015) is concerned with a demand-responsive transport (DRT) system where new service requests are submitted sequentially over time, but all of them are still submitted before the vehicles start service. The authors developed a cost-sharing mechanism, namely the Proportional Online Cost Sharing (POCS), that handles sequential customer submissions. POCS draws upon features of proportional and marginal cost-sharing and has been proved to satisfy a list of desirable properties, including online fairness, immediate response, individual rationality, budget balance, and ex-post incentive compatibility. POCS is a flexible framework in the sense that no specific cost function is defined. All of the desired properties hold as long as the cost function of choice satisfies the following two properties: 1) the total cost is non-decreasing over time (over order submissions); 2) the total cost is independent of the submit order of customers who have already submitted their requests. Although POCS represents a step forward in the research on cost-sharing mechanism design because it relaxes the constraint that the entire set of players must be known at once, limitations remain. POCS assumes that all customers submit their service requests before vehicle operations start. In the dynamic vehicle routing environment we study, the two assumptions may or may not hold and this shall result in loss of desirable properties.

In this paper, we focus on developing an online cost-sharing mechanism that allocates the cost to each customer in a dynamic vehicle routing setting where only some of the customers are known in advance, and the remaining customers become known in real time. Without lack of generality, we use the total miles travelled as a surrogate measure of the total cost. Our approach combines two cost-sharing mechanisms originally designed for the static and the online environment, respectively. With specially designed cost functions and routing schedules, the hybrid mechanism is shown to possess all of the five properties originally proposed in

Furuhata et al. (2015), namely online fairness, immediate response, individual rationality, budget balance, and ex-post incentive compatibility. We extend our work by proposing several variations of the baseline mechanism which can be formulated by relaxing some of the model assumptions. We compare and contrast different variations of the mechanism through extensive numerical simulations.

3 The Online Cost Allocation Problem

To study a static cost allocation problem, one needs to define the set of players, the total cost function, and the calculation of the shared costs. In the online cost allocation problem we study, the key challenge lies in how to incorporate the time dimension into a cost-sharing mechanism. In particular, we need to specifically design how the set of players, the total cost function, and the calculation of shared costs evolve over time, as more problem information becomes available.

3.1 Problem Definition

Suppose that the operation consists of routing a fleet of capacitated vehicles to collect shipments from a set of customers and transport them to a central depot. The length of the planning horizon is T_{max}. There are $\mathcal{N}$ potential customers. Each customer has a fixed location, a known demand size, a known service time window and a service time of fixed length. The service time window specifies the earliest and latest times when service can be started at the corresponding customer and cannot be violated. Each customer requests service at most once during the planning horizon. The uncertainty lies in the fact that not all customers would request service. Some customers request service in advance (prior to the beginning of the planning horizon), and are called advance customers. The rest of the customers are called dynamic customers, who may or may not request service during the planning horizon. We assume that the probability a dynamic customer requests service can be estimated from historical information. The time when a dynamic customer requests service is called its request time. It is also the time when it becomes certain that the customer needs to be served. The following notations are used.

$\mathcal{N}$	total number of customers
$\mathbb{AC}$	set of advance customers
$\mathbb{DC}$	set of dynamic customers
$\mathcal{K}$	total number of vehicles
$\mathcal{C}$	capacity of each vehicle
e_i	the earliest time that service can begin at customer i
l_i	the latest time that service can begin at customer i
v_i	request deadline of customer i

u_i	actual request time of customer i
d_i	demand of customer i
s_i	service time of customer i

The request time u_i of dynamic customer i represents the time when it becomes certain that customer i needs to be serviced. u_i is modeled as a random variable taking values on the interval $[0, v_i]$. The request deadline v_i denotes the latest time that the customer must make the decision on whether it needs to be serviced or not. Generally speaking it is reasonable to set $0 < v_i \leq e_i$.

We assume that the passenger cannot request service prior to its truthful request time, but may choose to delay its request in anticipation to take advantage of a possibly lower shared cost. In such cases, we distinguish its truthful request time, which is its earliest possible request time, from its actual, perhaps delayed, request time. Once a dynamic customer requests service, it is called a realized dynamic customer.

The solution of a cost allocation problem usually comes in the form of a cost-sharing mechanism, which takes the set of customers as the input and generates the shared cost of each participant as the output. A cost-sharing mechanism should specify at least two cost functions: a total cost function that returns the total transportation cost of serving the set of customers, and a shared cost function that returns the shared cost of each individual participant. The total transportation costs can include both variable and fixed costs. In the online cost allocation setting, however, the shared cost of each participant usually changes over time, possibly due to realization of new customers, cancellation of existing customers, and changes in network conditions that affects the total operating cost. An online cost-sharing mechanism should instead re-calculate the total operating cost and the shared cost of each customer whenever any of these changes happen.

When a dynamic customer requests service, the total cost of serving all customers may change, so does the shared cost of each existing customer. The dynamic customer should be immediately considered in the cost allocation problem and be offered a shared cost. The shared cost that a customer receives at the time of its request serves as its initial quote. Each customer may have a willingness-to-pay value that aligns with its valuation of the service received. According to McFadden (1998), it is rather complicated to determine the relationship between the willingness-to-pay level and the travel costs. Therefore, it is usually the case to set the willingness-to-pay level to be proportional to the travel costs. In this paper, we constrain it to be no less than twice the direct travel cost because a factor of two serves as a good lower bound since it represents the total travel cost of a customer when it is not sharing rides with others (a back and forth trip). In Section 4 we show that the routing strategy we use in our proposed mechanisms is guaranteed to provide an initial quote below this bound. The initial quote is the price that the customer would have to compare with its willingness-to-pay value to make the decision of whether to accept or decline the service.

How the total transportation cost should be calculated and shared among both advance and realized dynamic customers over time is a non-trivial problem for the following reasons: First, advance customers become known at the beginning of the planning horizon and should be offered their initial quotes at the same time, without knowledge on how many and which dynamic customers would request service. The way cost is shared among advance customers should obey standards typically required in static cost allocation problems, including fairness, budget balance, etc. As the planning horizon rolls out, the shared costs for advance customers together with the shared costs for realized dynamic customers should obey the properties required in the online setting. Second, customers should be given incentives to request service as early as possible to allow more time for calculating routing schedules. Therefore, an ideal mechanism should ensure that the best strategy for each individual customer to achieve the lowest possible shared cost is to request service at its truthful request time. For the same reason, a good mechanism should be able to demonstrate that it is more advantageous for each customer to make its service request known early as an advance customer than to request late as a dynamic customer. Last but not least, the initial quote provided to each customer should serve as an upper bound on the final shared cost of the customer, which is the shared cost value for the customer at the end of the planning horizon.

3.2 Desirable Properties

Before we develop a new mechanism, we first discuss a list of properties for an ideal online cost-sharing mechanism. Some of the properties correspond to their counterparts for static problems, such as fairness and budget balance. The rest are derived specifically for the online environment. Consistent with the literature, instead of focusing on the initial quotes, the following five desirable properties refer to the final shared costs which are the actual values customers pay.

Online Fairness. At any time during the planning horizon, the shared cost per demand value of any customer is never lower than those of customers who have requested service prior to the customer. The property has two implications. First, since for advance customers, their request times are the same. There should not be any notion of early and late among advance customers. Thus, fairness for advance customers means that the shared cost per demand value of all advance customers should be the same. Second, since all advance customers request service before all realized dynamic customers, the shared cost per demand value of any advance customer should never be higher than that of any dynamic customer. However, the online fairness property does not require that the initial quote per demand value provided to any customer to be never higher than the one provided to a subsequent customer. In other words, it can happen that a customer who requests service late receives a lower initial quote per demand value than a prior customer. Nevertheless, in such a situation it is guaranteed that the current shared cost per demand value of the prior customer is never higher than the initial quote per demand value provided to a subsequent customer.

Budget Balance. At any time during the planning horizon, the sum of the shared costs of all customers equals to the total travel cost of the current routing schedule, including both traveled and untraveled portions of the schedule. Here we say a solution is "Budget Balanced" if the costs are fully recovered. Budget balance is practically applicable if the platform is developed by a non-profit entity such as a transportation agency. With that being said, it does not imply that the mechanism is not applicable to for-profit firms. A budget balanced mechanism can be adapted to a for-profit firm by adding a profit multiplier to the costs charged. In this way, the mechanism results in a net positive payment from the customers to the mechanism (platform) and is in this called weak budget-balance.

Immediate Response. Each customer should be provided with an upper bound on its final shared cost at the time of its service request. Since each customer has to make the decision of whether to accept or decline the service based on its willingness-to-pay level, this property guarantees that each customer only has to make that decision once at the time of its request, without having to worry about being charged against its will for a higher price than it previously agreed to.

Individual Rationality. At any time during the planning horizon, the shared cost of any customer who has accepted its initial quote never exceeds its willingness-to-pay level. Since a customer only remains in the cooperation as long as its shared cost does not exceed its willingness-to-pay level, individual rationality guarantees that no customer will drop out of the cooperation once it joins since the initial quote serves as an upper bound on any subsequent quote for that customer.

Ex-Post Incentive Compatibility. The best strategy of each customer is to request service truthfully at its earliest possible time, provided that all other customers do not change their request times and whether they accept or decline their initial quotes. This property has two implications. First, an advance customer cannot decrease its final shared cost by choosing to become a dynamic customer and not request service at the beginning of the planning horizon. Second, a dynamic customer cannot decrease its final shared cost by delaying its actual request time to be later than its truthful request time. For similar reasons as discussed under the online fairness property, this property is concerned with the final shared costs rather than initial price quotes. Thus it is possible for a customer, either an advance customer or a dynamic customer, to delay its actual request time and receive a lower initial quote than it would have received at its truthful request time. Even if it happens, the final shared cost of the same customer in the delayed request case is guaranteed to be no lower than in the truthful request case.

In addition to the above five desirable properties that refer to the final shared cost, we state an additional property that refers to the initial quote and is especially desirable in an online context since a customer most likely will compare their willingness-to-pay level to the initial quote especially for new customers. Repeat customers may make some adjustments in their acceptance of the initial quote if they have seen that in previous experiences their final shared cost can be significantly reduced from their initial quote.

Early Incentive. The initial quote per demand value provided to advance customers who request service before the operation starts should not be higher than those dynamic customers who request service after the operation starts. This is desirable because customers most likely make their decisions on whether to join the operation or not based on the initial quote they receive. This property then incentivizes customers to become advance customers since it helps the system to make better routing decisions which has a lower total cost.

In a static cost allocation problem, where the entire set of players is known and the total cost of serving each subset of players is well defined, the most intuitive and fair way to share the cost is proportional cost-sharing (Wang and Zhu, 2002; Sprumont, 1998), where the total cost is distributed among all customers proportionally to their demand of the common resource. Now consider the online cost allocation problem we study, the most intuitive way of sharing the cost is incremental cost-sharing (Moulin, 1999), where the shared cost of each new player equals to the marginal cost generated from including the new player. Under incremental cost-sharing, the shared cost of each customer will remain the same through the planning horizon, and thus the final shared cost always equals to the initial quote for each customer. Another strategy is to naively adapt proportional cost-sharing to the online setting by re-calculating shared costs each time a dynamic customer requests service. That is to say, the shared cost of each customer may change each time an additional customer enters the system, and there is no guarantee that the shared cost for any customer will not increase over time. In summary, it is easy to show that proportional cost-sharing will violate the immediate response property and incremental cost-sharing will violate the fairness property when applying in an online setting. Thus, in the next section, we propose a hybrid mechanism and its extensions that address the above desired properties.

4 Proposed Mechanisms

In this section, we propose three different mechanisms and analyze their advantages and disadvantages. The first mechanism is the base case of the other two.

4.1 Hybrid Proportional Online Cost-Sharing (HPOCS) Mechanism

For each mechanism we first explain how the shared costs are calculated and updated over time in the dynamic vehicle routing problem. Then we prove that HPOCS satisfies all of the desirable properties except for the early incentive property discussed in the previous section.

We develop the HPOCS mechanism as an online cost-sharing mechanism that combines proportional cost-sharing for solving static cost allocation problems and the Proportional Online Cost-Sharing (POCS) mechanism in Furuhata et al. (2015) for handling sequential customer requests. In particular, proportional cost-sharing is used to calculate the initial quotes for advance customers at the beginning of the planning horizon, while the POCS mechanism is used to handle dynamic customer requests. The idea behind POCS is that customers are partitioned into coalitions, where each coalition contains a sequence of customers who request service within given time intervals. At the time of its request, each customer first forms its own coalition. However, customers can choose to form coalitions with customers who request service directly after them to decrease their shared costs. The formation of a coalition is determined by comparing the pooled marginal costs shared over subsets of customers each time a new customer enters the system. A set of specially designed total and marginal cost values for advance customers is used to initialize the POCS process for dynamic customers. This setup ensures that the coalition can be formed across both advance and dynamic customers. A routing technique together with the corresponding cost functions serves as the core of HPOCS.

Let $\mathbb{C}$ represent the grand set of potential customers, which is the union of the set of advance customers $\mathbb{AC}$ and the set of dynamic customers $\mathbb{DC}$, $\mathbb{C} = \mathbb{AC} \cup \mathbb{DC}$, $|\mathbb{C}| = \mathcal{N}$. Let $C(t)$ represent the set of customers who have requested service by time t. By definition, $C(0) = \mathbb{AC}$ since none of the dynamic customers have requested service but all of the advance customers are already known at time $t = 0$. Let c_{ij} and t_{ij} represent the minimum travel cost and travel time between location i and j and it is assumed that the unit cost is the same as the unit distance traveled by any vehicle. Thus, $c_{ij} = t_{ij}$. Without loss of generality we assume the only components of the total cost to recover are the variable costs. However, a fixed cost term could be added to the total cost component without affecting any of the resulting Propositions.

We now formally define terminologies related to the HPOCS mechanism.

Definition 1. The alpha value α_i of customer i quantifies the utilization of all relevant resources serving customer i. It can also be interpreted as the measure of inconvenience caused by accommodating the customer. The alpha value is assumed to be positive and independent of the request time of the passenger. Similarly, it is also independent of whether the customer is an advance customer or dynamic customer. We use

$$\alpha_i = c_{0,i} * d_i, \tag{1}$$

where $c_{0,i}$ represents the minimum travel cost between customer i and the depot, and d_i represents the demand of customer i.

With this definition, we formally define a coalition as a set of consecutive customer requests that have the same shared cost per alpha value.

Definition 2. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, π_t denotes a request order of the customers in $C(t)$. For $n \in [1, |C(t)|]$, $\pi_t(n)$ represents the n^{th} customer to request service under request order π_t. For example, $\pi_t(n) = i$ means that customer i is the n^{th}

customer to request service under request order π_t.

Definition 3. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, $\bar{\pi}_t$ denotes the special request order based on the realization of the dynamic vehicle routing problem up to time t, where all realized dynamic customers are ordered after all advance customers. In particular, the first part of $\bar{\pi}_t$ consists of all of the advance customers. Any ordering of advance customers can be used to build the first half of $\bar{\pi}_t$ and the exact ordering does not affect the properties of HPOCS, which will be proved in later sections. The second part of $\bar{\pi}_t$ records the ordering of realized dynamic customers based on the ordering of their actual request times.

It is important to point out that π_t is a general symbol used to represent any request order, while $\bar{\pi}_t$ is the request order uniquely defined by the realization of the DVRP. Nevertheless, given time $t \in [0, T_{max}]$, π_t and $\bar{\pi}_t$ will always contain exactly the same set of customers, namely $C(t)$. Although they contain the same set of customers π_t is used to denote when a statement is true for any given ordering while $\bar{\pi}_t$ denotes one particular order (i.e., the one associated with the realization of the dynamic customers). Recall that $C(0) = \mathbb{AC}$, meaning that π_0 consists of all advance customers. The same is true for $\bar{\pi}_0$.

Definition 4. The grand schedule $\bar{S}$ is a complete routing solution to the static vehicle routing problem corresponding to the grand set of customers $\mathbb{C}$. It uses the heuristic routing algorithm by Zou (2017) and satisfies the following requirement. For any dynamic customer i, the time when the assigned vehicle is scheduled to leave from its predecessor location is no earlier than the request deadline of the dynamic customer, v_i. That is, when a vehicle finishes service at its current customer and becomes idle, if the next customer on the schedule is a dynamic customer that has yet to request service, the vehicle should wait at its current location and only be allowed to travel either when the dynamic customer becomes realized or when its request deadline has been reached, whichever comes first. $\bar{S}$ takes the form of a set of vehicle routes each assigned to a single vehicle. $\bar{S} = \{r_k\}$ where $k = 1, \ldots, \mathcal{K}$. Each route r_k specifies the sequence of customer visits as well as the exact arrival and departure times at each customer, which satisfies the corresponding time window constraints and the additional requirement discussed above.

Definition 5. Let $\bar{S}$ be a grand schedule corresponding to the set of customers $\mathbb{C}$, and let $C \subset \mathbb{C}$ be a subset of customers. $\bar{S}(C)$ is called the partial schedule induced by the grand schedule $\bar{S}$ and the set C, which is constructed by removing all of the customers not in C from the grand solution $\bar{S}$. In particular, each customer that is not in C is removed from the route, and its predecessor and successor scheduled on the same vehicle are connected with a direct link. The related timings are also updated. That is, the time when the vehicle is scheduled to leave its predecessor is now the time when the vehicle is originally scheduled to leave from the removed customer to its successor. In other words, all of the extra slack time now present in the route due to the removal of a customer is added to the wait time at its predecessor location.

Given a feasible grand schedule $\bar{S}$ and any subset of customers $C \subset \mathbb{C}$, it can be easily shown that a feasible induced schedule $\bar{S}(C)$ is guaranteed to exist, based on the triangle inequality property of pairwise distances. It is also evident that such induced solutions are usually not unique. Besides, given the grand schedule $\bar{S}$, for any time $t \in [0, T_{max}]$, and any request order π_t, we use the notation $\bar{S}\left(\pi_t\left(n\right)\right)$ to represent the partial schedule induced by the set of first n customers on the request order π_t. More specifically, $\bar{S}\left(\pi_t\left(n\right)\right)$ is an equivalent notation used to denote the same induced solution as $\bar{S}(C)$, where $C = \{\pi_t\left(1\right), \ldots, \pi_t\left(n\right)\}$.

The following proposition states that given the set of customers who have requested service by time t, the induced partial schedule is independent from the request order among the customers within the set.

Proposition 1. *For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, and any two request orders π_t and π'_t, we have*

$$\bar{S}\left(\pi_t\right) = \bar{S}\left(\pi_t^{'}\right) = \bar{S}\left(C(t)\right) \tag{2}$$

Proof. By Definition 5, we have $\bar{S}\left(\pi_t\left(n\right)\right) = \bar{S}\left(\pi_t^{'}\left(n\right)\right)$ for any $n \in [1, |C(t)|]$. Setting $n = |C(t)|$, we have that

$$\bar{S}\left(\pi_t\left(n\right)\right) = \bar{S}\left(\pi_t\left(|C(t)|\right)\right) = \bar{S}\left(\pi_t^{'}\left(|C(t)|\right)\right) = \bar{S}\left(\pi_t^{'}\left(n\right)\right) \tag{3}$$

which proves the first equality. For the second equality, we note that by definition both schedules $\bar{S}\left(\pi_t^{'}\right)$ and $\bar{S}\left(C(t)\right)$ are induced by the same set of customers, namely those customers that have requested service by time t. In addition, both solutions are constructed in the same way by removing customers not in $C(t)$ from the grand schedule $\bar{S}$. The membership and ordering of each customer on each vehicle route is preserved. It follows that $\bar{S}\left(\pi_t^{'}\right)$ and $\bar{S}\left(C(t)\right)$ are exactly the same schedules. Thus we have completed the proof. □

We now define the cost functions used by HPOCS. Some cost functions are based on their counterparts in the POCS mechanism (Furuhata et al., 2015), such as coalition cost per alpha and shared cost. In the original POCS formulation, it is assumed that customers request service sequentially, and no two customers will request service at the same time. In the DVRP we study, all of the advance customers request service at the same time. Thus we extend the definitions in POCS to accommodate both advance and dynamic customers.

Definition 6. For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, and any request order π_t, the *totalcost* $\left(\bar{S}\left(C(t)\right)\right)$ is the total travel cost of the induced partial solution $\bar{S}\left(C(t)\right)$. Equivalently, *totalcost* $\left(\bar{S}\left(\pi_t\right)\right)$ can be used to represent the same total cost since the underlying partial schedules are practically the same, as stated by Proposition 1. We define *totalcost* $\left(\bar{S}\left(\emptyset\right)\right) \coloneqq 0$.

Definition 7. For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any integer $n \in [1, |C(t)|]$, $\mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n\right)\right)\right)$ is the total operating cost required to serve the first n customers on request order $\bar{\pi}_t$. For the case of advance customers, let $1 \leq n^* \leq |\mathbb{AC}|$, so that $\bar{\pi}_t(n^*)$ represents an advance customer. We define

$$\mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n^*\right)\right)\right) = \mathit{acpa} \sum_{n=1}^{n^*} \alpha_{\bar{\pi}_t(n)} \tag{4}$$

At $n^* = |\mathbb{AC}|$, $\bar{\pi}_t(n^*)$ represents the last advance customer on request order $\bar{\pi}_t$. We define

$$\mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(|\mathbb{AC}|\right)\right)\right) = \mathit{totalcost}\left(\bar{S}\left(\mathbb{AC}\right)\right) \tag{5}$$

which is a direct result of Proposition 1 and is consistent with Definition 6. For the case of dynamic customers, assume that $|\mathbb{AC}| < |C(t)|$. Let $|\mathbb{AC}| < n^* \leq |C(t)|$, so that $\bar{\pi}_t(n^*)$ represents a realized dynamic customer. Then $\mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n^*\right)\right)\right)$ is defined as the total travel cost of the induced partial solution corresponding to the first n^* customers on schedule $\bar{\pi}_t$. Similarly as in Definition 6, we define $\mathit{totalcost}\left(\bar{S}\left(\pi_t\left(0\right)\right)\right) \coloneqq 0$.

Definition 8. For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, any request order π_t, any customer $i \in C(t)$, let n be the index order of the customer on request order π_t. Equivalently, $\pi_t(n) = i$ for some $n \in [1, |C(t)|]$. $\mathit{mc}\left(\pi_t(n)\right)$ denotes the marginal cost of serving customer i under request order π_t and is defined as the increase in total cost due to its request. That is

$$\mathit{mc}\left(\pi_t(n)\right) \coloneqq \mathit{totalcost}\left(\bar{S}\left(\pi_t\left(n\right)\right)\right) - \mathit{totalcost}\left(\bar{S}\left(\pi_t\left(n-1\right)\right)\right) \tag{6}$$

Especially in the case of advance customers, let $1 \leq n^* \leq |\mathbb{AC}|$, so that $\bar{\pi}_t(n^*)$ represents an advance customer. Based on equations 4 and 6, we have

$$\begin{align} \mathit{mc}\left(\bar{\pi}_t\left(n^*\right)\right) &= \mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n^*\right)\right)\right) - \mathit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n^*-1\right)\right)\right) \tag{7}\\ &= \mathit{acpa} \sum_{n=1}^{n^*} \alpha_{\bar{\pi}_t(n)} - \mathit{acpa} \sum_{n=1}^{n^*-1} \alpha_{\bar{\pi}_t(n)} \tag{8}\\ &= \mathit{acpa} \times \alpha_{\bar{\pi}_t(n^*)} \tag{9} \end{align}$$

which states that the marginal cost of an advance customer equals to the product of the advance cost per alpha value and its alpha value.

We now define the coalition cost per alpha value, how HPOCS calculates the shared cost of each customer, and the concept of coalition.

Definition 9. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any two integers $n_1, n_2 \in [1, |C(t)|]$ with $n_1 \leq n_2$, the coalition cost per alpha value of customers $\{\bar{\pi}_t(n_1), \ldots, \bar{\pi}_t(n_2)\}$ at time t under submit order $\bar{\pi}_t$ is

$$ccpa_{\bar{\pi}_t(n_1,n_2)} := \frac{\sum_{n=n_1}^{n_2} mc\left(\bar{\pi}_t(n)\right)}{\sum_{n=n_1}^{n_2} \alpha_{\bar{\pi}_t(n)}} \tag{10}$$

Especially in the case of advance customers, we rename it into *acpa* since all advance customers share the same coalition per alpha value (proved in Proposition 2) and we have:

$$acpa = \frac{totalcost\left(\bar{S}\left(\mathbb{AC}\right)\right)}{\sum_{i\in\mathbb{AC}} \alpha_i} = \frac{totalcost\left(\bar{S}\left(C(0)\right)\right)}{\sum_{i\in\mathbb{AC}} \alpha_i} \tag{11}$$

Definition 10. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n be the index order of the customer on request order $\bar{\pi}_t$. Equivalently, $\bar{\pi}_t(n) = i$ for some $1 \leq n \leq |C(t)|$. Then the shared cost of customer i at time t under request order $\bar{\pi}_t$ is defined as

$$cost_t\left(\bar{\pi}_t(n)\right) := \alpha_{\bar{\pi}_t(n)} \min_{n\leq n'\leq |C(t)|} \max_{1\leq n''\leq n'} ccpa_{\bar{\pi}_t(n'',n')} \tag{12}$$

Definition 11. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any two integers $n_1, n_2 \in [1, |C(t)|]$ with $n_1 \leq n_2$, a coalition (n_1, n_2) at time t is a group of customers $\{\bar{\pi}_t(n_1), \ldots, \bar{\pi}_t(n_2)\}$ with

$$\frac{cost_t\left(\bar{\pi}_t(n)\right)}{\alpha_{\bar{\pi}_t(n)}} = \frac{cost_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} \tag{13}$$

for all order indices $n_1 \leq n \leq n_2$ and

$$\frac{cost_t\left(\bar{\pi}_t(n)\right)}{\alpha_{\bar{\pi}_t(n)}} \neq \frac{cost_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} \tag{14}$$

for both order indices with $n = n_1 - 1$ and $n = n_2 + 1$ and $1 \leq n \leq |C(t)|$.

Definition 11 suggests that the membership of a coalition is determined solely by the shared cost per alpha value of each customer. A sequence of customers who request service consecutively in time and have the same shared cost per alpha value are said to be in the same coalition. In terms of coalition formation, it is irrelevant whether a customer is an advance customer or a dynamic customer; a single coalition can consist of both advance and dynamic customers. Nor is it relevant whether the group of customers are assigned on the same vehicle or not.

The following statements are concerned with the way coalitions form and evolve over time under the special request order $\bar{\pi}_t$.

Proposition 2. *At any time $t \in [0, T_{max}]$, under the special request order $\bar{\pi}_t$, the coalition cost per alpha value of any coalition consisting solely of advance customers is a constant value. The value is fixed given the set of advance customers $\mathbb{AC}$ and is independent from the actual subset of advance customers in the coalition.*

Proof. For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any two integers $n_1, n_2 \in [1, |C(t)|]$ with $n_1 \leq n_2$, suppose that both n_1 and n_2 represent advance customers. That is, $n_1, n_2 \in [1, |\mathbb{AC}|]$. Then the coalition cost per alpha value of customers $\{\bar{\pi}_t(n_1), \ldots, \bar{\pi}_t(n_2)\}$ at time t under submit order $\bar{\pi}_t$ is

$$ccpa_{\bar{\pi}_t(n_1,n_2)} = \frac{\sum_{n=n_1}^{n_2} mc\left(\bar{\pi}_t(n)\right)}{\sum_{n=n_1}^{n_2} \alpha_{\bar{\pi}_t(n)}} \tag{15}$$

$$= \frac{\sum_{n=n_1}^{n_2} acpa \times \alpha_{\bar{\pi}_t(n)}}{\sum_{n=n_1}^{n_2} \alpha_{\bar{\pi}_t(n)}} \tag{16}$$

$$= acpa \tag{17}$$

The second equality follows from equation 9. Note that the coalition cost per alpha value equals to the advance cost per alpha value, which only depends on the set of advance customers $\mathbb{AC}$ and is independent of n_1, n_2, and even the request order $\bar{\pi}_t$. Equivalently speaking, given the set of advance customers, the coalition cost per alpha value of any coalition formed solely by advance customers is the same. Thus we have completed the proof. □

Proposition 3. *At time $t = 0$, under the special request order $\bar{\pi}_0$, all advance customers form a single coalition.*

Proof. At time $t = 0$, for any customer $i \in \mathbb{AC}$, let n be the index order of the customer on the special request order $\bar{\pi}_0$. Equivalently, $\bar{\pi}_0(n) = i$ for some $1 \leq n \leq |\mathbb{AC}|$. By Definition 10, the shared cost of customer i at time $t = 0$ under request order $\bar{\pi}_0$ is

$$cost_0\left(\bar{\pi}_0(n)\right) = \alpha_{\bar{\pi}_t(n)} \min_{n \leq n' \leq |\mathbb{AC}|} \max_{1 \leq n'' \leq n'} ccpa_{\bar{\pi}_t(n", n')} \tag{18}$$

$$= \alpha_{\bar{\pi}_t(n)} \min_{n \leq n' \leq |\mathbb{AC}|} \max_{1 \leq n'' \leq n'} acpa \tag{19}$$

$$= \alpha_{\bar{\pi}_t(n)} \times acpa \tag{20}$$

The second equality follows from the fact that both $\bar{\pi}_0(n')$ and $\bar{\pi}_0(n'')$ represent advance customers and that the coalition cost per alpha value of any coalition consisting solely of advance customers is always equal to *acpa* (Proposition 2). The third equality follows since the term inside the minimization and maximization operator is a constant and independent from both operators. Equation 20 shows that the shared costs among advance customers at time $t = 0$ under the special request order $\bar{\pi}_0$ obey the proportional cost-sharing rule. It then follows that the shared cost per alpha values of any two advance customers $\bar{\pi}_0(n_1)$ and $\bar{\pi}_0(n_2)$ with $n_1, n_2 \in [1, |\mathbb{AC}|]$ must be the same, which in turn proves that all advance customers form a single coalition at time $t = 0$ under the special request order $\bar{\pi}_0$. □

Corollary 3.1. *For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n be the index order of the customer on request order $\bar{\pi}_t$. Equivalently, $\bar{\pi}_t(n) = i$ for some $1 \leq n \leq |C(t)|$. Then*

$$\frac{cost_t\left(\bar{\pi}_t(n)\right)}{\alpha_{\bar{\pi}_t(n)}} = \min_{n \leq n' \leq |C(t)|} \frac{cost_{u_{\bar{\pi}_t(n')}}\left(\bar{\pi}_t(n')\right)}{\alpha_{\bar{\pi}_t(n')}} \tag{21}$$

where $u_{\bar{\pi}_t(n')}$ is the request time of customer $\bar{\pi}_t(n')$ and $cost_{u_{\bar{\pi}_t(n')}}\left(\bar{\pi}_t(n')\right)$ represents the initial quote this customer receives at the time of its request.

Proof. Consider any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n be the index order of the customer on request order $\bar{\pi}_t$. Equivalently, $\bar{\pi}_t(n) = i$ for some $1 \leq n \leq |C(t)|$. Then we have

$$\frac{cost_t\left(\bar{\pi}_t(n)\right)}{\alpha_{\bar{\pi}_t(n)}} = \min_{n \leq n' \leq |C(t)|} \max_{1 \leq n'' \leq n'} ccpa_{\bar{\pi}_t(n",n')} \tag{22}$$

$$= \min_{n \leq n' \leq |C(t)|} \min_{n' \leq m \leq n'} \max_{1 \leq n'' \leq m} ccpa_{\bar{\pi}_t(n",m)} \tag{23}$$

$$= \min_{n \leq n' \leq |C(t)|} \frac{cost_{u_{\bar{\pi}_t(n')}}\left(\bar{\pi}_t(n')\right)}{\alpha_{\bar{\pi}_t(n')}} \tag{24}$$

where the first and third equalities both follow from Definition 10. □

Lemma 1. *Under the special request order $\bar{\pi}_t$, once a group of customers forms a coalition at time t, they will remain in the same coalition until the end of the planning horizon. More customers may join the same coalition over time, but the original group of customers will never depart the coalition.*

Proof. For any time $t_1 \in [0, T_{max})$ and the corresponding set of customers who have requested service $C(t_1)$, let (n_1, n_2) be a coalition at time t_1 under the special request order $\bar{\pi}_{t_1}$, where $1 \leq n_1 \leq n_2 \leq |C(t_1)|$. Let $t_2 \in (t_1, T_{max}]$ be any later point of time in the planning horizon. Now consider any customer with the order index $n_1 \leq n \leq n_2$ under the special request order $\bar{\pi}_{t_1}$. Then

$$\min_{n \leq n' \leq |C(t_1)|} \frac{cost_{u_{\bar{\pi}_{t_1}(n')}}\left(\bar{\pi}_{t_1}(n')\right)}{\alpha_{\bar{\pi}_{t_1}(n')}} = \frac{cost_{t_1}\left(\bar{\pi}_{t_1}(n)\right)}{\alpha_{\bar{\pi}_{t_1}(n)}} \tag{25}$$

$$= \frac{cost_{t_1}\left(\bar{\pi}_{t_1}(n_1)\right)}{\alpha_{\bar{\pi}_{t_1}(n_1)}} \tag{26}$$

$$= \min_{n_1 \leq n' \leq |C(t_1)|} \frac{cost_{u_{\bar{\pi}_{t_1}(n')}}\left(\bar{\pi}_{t_1}(n')\right)}{\alpha_{\bar{\pi}_{t_1}(n')}} \tag{27}$$

where the first and third equalities both follow from Corollary 3.1 and the second equality follows from Definition 11. In addition, since $t_1 \leq t_2 \leq T_{max}$, request order $\bar{\pi}_{t_2}$ is an extension of the order $\bar{\pi}_{t_1}$. Thus $\bar{\pi}_{t_2}(m) = \bar{\pi}_{t_1}(m)$ for all $1 \leq m \leq |C(t_1)|$ by definition. Equation 27 can be rewritten as follows

$$\min_{n \leq n' \leq |C(t_1)|} \frac{cost_{u_{\bar{\pi}_{t_2}(n')}}\left(\bar{\pi}_{t_2}(n')\right)}{\alpha_{\bar{\pi}_{t_2}(n')}} = \min_{n_1 \leq n' \leq |C(t_1)|} \frac{cost_{u_{\bar{\pi}_{t_2}(n')}}\left(\bar{\pi}_{t_2}(n')\right)}{\alpha_{\bar{\pi}_{t_2}(n')}} \tag{28}$$

Now consider adding the following set of terms to the minimization operators on both sides of equation 27.

$$\left\{ \frac{cost_{u_{\bar{\pi}_{t_2}(j)}}\left(\bar{\pi}_{t_2}(j)\right)}{\alpha_{\bar{\pi}_{t_2}(j)}} \right\}_{|C(t_1)| < j \leq |C(t_2)|} \tag{29}$$

Since the same set of terms are added to both minimization operators, the equality is preserved. Equation 27 can be rewritten as follows

$$\min_{n \leq n' \leq |C(t_2)|} \frac{cost_{u_{\bar{\pi}_{t_2}(n')}}\left(\bar{\pi}_{t_2}(n')\right)}{\alpha_{\bar{\pi}_{t_2}(n')}} = \min_{n_1 \leq n' \leq |C(t_2)|} \frac{cost_{u_{\bar{\pi}_{t_2}(n')}}\left(\bar{\pi}_{t_2}(n')\right)}{\alpha_{\bar{\pi}_{t_2}(n')}} \tag{30}$$

which by Corollary 3.1 is equivalent to

$$\frac{cost_{t_2}\left(\bar{\pi}_{t_2}(n)\right)}{\alpha_{\bar{\pi}_{t_2}(n)}} = \frac{cost_{t_2}\left(\bar{\pi}_{t_2}(n_1)\right)}{\alpha_{\bar{\pi}_{t_2}(n_1)}} \tag{31}$$

We have established that all of the customers in the original coalition at time t_1 have the same shared cost per alpha value at any future time t_2. By the definition of coalition, all of these customers must be in the same coalition at time t_2. Thus we have completed the proof. □

Remark. *Directly from Proposition 3 and Lemma 1, we have that the set of advance customers will remain in the same coalition throughout the planning horizon.*

We now present the HPOCS mechanism. For a realization of the dynamic vehicle routing problem, the shared costs are calculated as follows.

Initialization. $t = 0$. Formulate a static vehicle routing problem corresponding to the set of customers $\mathbb{C} = \mathbb{AC} \cup \mathbb{DC}$ and construct the grand solution $\bar{S}$. Construct the special request order $\bar{\pi}_0$ consisting of all advance customers. Any ordering among advance customers can be used.

Quoting advance customers. All advance customers receive their initial quotes at time $t = 0$. This step calculates the advance cost per alpha value *acpa* based on Definition 9 and then calculates the total cost, marginal cost, coalition cost per alpha, and the shared cost of each advance customer under the special request order $\bar{\pi}_0$ by Definition 7, equation 9, Definition 9, and equation 20. And for each advance customer $i \in \mathbb{AC}$, suppose that n is its order index on request order $\bar{\pi}_0$. Provide $cost_0\left(\bar{\pi}_0(n)\right)$ as the initial quote for customer i.

Quoting dynamic customers. A dynamic customer i receives its initial quote when it requests service at time $t = u_i$. It first appends customer i to the end of the special request order $\bar{\pi}_{u_i - 1}$ to form the new special request order $\bar{\pi}_{u_i}$. Recall that $|C(t)|$ represents the total number of customers who have requested service. By definition, $\bar{\pi}_{u_i}\left(|C(u_i)|\right) = i$. Then it constructs the partial schedule induced by $C(u_i)$ and the grand schedule $\bar{S}$. After that, it calculates and updates the total costs, marginal costs, coalition cost per alpha values, and the shared costs of all existing customers on request order $\bar{\pi}_{u_i}$ by Definition 7, equation 6, Definition 9, and Definition 10. Lastly, it provides $cost_{u_i}\left(\bar{\pi}_{u_i}(|C(u_i)|)\right)$ as the initial quote for customer i.

Final shared costs. At time $t = T_{max}$, all of the randomness in the system has been realized. The special request order $\bar{\pi}_{T_{max}}$ consists of all advance and realized dynamic customers, namely the set $C(T_{max})$. For $1 \leq n \leq |C(T_{max})|$, the shared cost of customer $\bar{\pi}_{T_{max}}(n)$ at time T_{max} under the special request order $\bar{\pi}_{T_{max}}$ is $cost_{T_{max}}\left(\bar{\pi}_{T_{max}}(n)\right)$. This is also the final cost of service for customer $\bar{\pi}_{T_{max}}(n)$.

4.1.1 Analysis of Properties

The HPOCS mechanism defines a way to allocate the total travel cost to each customer in the dynamic vehicle routing problem. By definition, this mechanism follows the same framework as the POCS mechanism, with the exception that the total cost function is defined differently. Given that the original POCS mechanism satisfies the first five desirable properties discussed in Section 3.2, it follows that the HPOCS mechanism also possess these properties, if it can be shown that the new total cost function satisfies the same assumptions as made by the POCS framework.

The POCS framework makes two assumptions of the total cost function. First, the total cost is non-decreasing over time. Second, the total cost at any time is independent of the request order among the group of customers that have requested service. These assumptions are, for example, satisfied for the minimal operating cost, which is the cost function used in the original POCS paper. However, optimality is not required in order for all of the desirable properties to be satisfied, as long as the cost function follows the two assumptions.

Proposition 4. *For any grand schedule $\bar{S}$, any time $t \in [0, T_{max}]$ and the corresponding set of customers* *who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any integer $n \in [1, |C(t)|]$, the HPOCS total cost function totalcost $\left(\bar{S}\left(\bar{\pi}_t\left(n\right)\right)\right)$ is nondecreasing in n and is independent of the request order of customers $\{\bar{\pi}_t\left(1\right), \ldots, \bar{\pi}_t\left(n\right)\}$. That is, for any request order π_t satisfying $\{\bar{\pi}_t\left(1\right), \ldots, \bar{\pi}_t\left(n\right)\} = \{\pi_t\left(1\right), \ldots, \pi_t\left(n\right)\}$, totalcost $\left(\bar{S}\left(\bar{\pi}_t\left(n\right)\right)\right) =$ totalcost $\left(\bar{S}\left(\pi_t\left(n\right)\right)\right)$.*

Proof. We first prove that *totalcost* $\left(\bar{S}\left(\bar{\pi}_t\left(n\right)\right)\right)$ is nondecreasing in n. Without loss of generality, let n_1 be any order index satisfying $1 \leq n_1 < |C(t)|$ and let $n_2 = n_1 + 1$. By definition, the partial schedule $\bar{S}\left(\bar{\pi}_t\left(n_1\right)\right)$ is constructed by removing customer $\bar{\pi}_t\left(n_2\right)$ from the schedule $\bar{S}\left(\bar{\pi}_t\left(n_2\right)\right)$. Let i^- and i^+ represent the predecessor and successor locations of customer $\bar{\pi}_t\left(n_2\right)$ in the schedule $\bar{S}\left(\bar{\pi}_t\left(n_2\right)\right)$. Then we have

$$\textit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n_1\right)\right)\right) = \textit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n_2\right)\right)\right) - c_{i^-\bar{\pi}_t(n_2)} - c_{\bar{\pi}_t(n_2)i^+} + c_{i^-i^+} \tag{32}$$

Based on the triangle inequality property of pairwise distances, we have

$$c_{i^-\bar{\pi}_t(n_2)} + c_{\bar{\pi}_t(n_2)i^+} - c_{i^-i^+} \geq 0 \tag{33}$$

Thus equation 32 implies that

$$\textit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n_1\right)\right)\right) \leq \textit{totalcost}\left(\bar{S}\left(\bar{\pi}_t\left(n_2\right)\right)\right) \tag{34}$$

We next prove the total cost is independent of the request order of customers $\{\bar{\pi}_t\left(1\right), \ldots, \bar{\pi}_t\left(n\right)\}$. Let π_t be any request order satisfying $\{\bar{\pi}_t\left(1\right), \ldots, \bar{\pi}_t\left(n\right)\} = \{\pi_t\left(1\right), \ldots, \pi_t\left(n\right)\}$. That is to say, the first n positions of π_t and of $\bar{\pi}_t$ consist of the same group of customers. By Definition 5, $\bar{S}\left(\bar{\pi}_t\left(n\right)\right)$ and $\bar{S}\left(\pi_t\left(n\right)\right)$ represent the same induced partial schedule. It then follows that *totalcost* $\left(\bar{S}\left(\bar{\pi}_t\left(n\right)\right)\right) =$ *totalcost* $\left(\bar{S}\left(\pi_t\left(n\right)\right)\right)$ □

When implementing the HPOCS mechanism to solve the cost allocation problem associated with a DVRP, one must specify the way vehicles are routed in real time. Thus the objective of this paper is to minimize the total cost used in the mechanism such that it satisfies the proposed desirable properties. Without lack of generality, we use the total miles travelled as a surrogate measure of the total cost. In order to satisfy all of the desirable properties, we need to define a dynamic vehicle routing strategy that can guarantee that the actual total travel cost incurred by the vehicles equals to the total cost calculated by the HPOCS mechanism. Since the VRP is a NP-hard problem heuristics are commonly used to find a routing solution, especially for the dynamic VRP. Thus, we next present a dynamic routing heuristic that satisfies this requirement.

1. Vehicles are routed based on the grand schedule $\bar{S}$.

2. No re-optimization is done during the planning horizon.

3. At the time when a vehicle is scheduled to depart from its current location and travel to a dynamic customer, if the customer has yet to request service, it is skipped and the vehicle travels directly from the predecessor location to the successor location of the dynamic customer.

Recall that by Definition 4, the grand schedule $\bar{S}$ requires that the time when a vehicle starts to travel to a dynamic customer is no earlier than the request deadline of the customer. If the customer has yet to request service by this time, it is certain that the customer will not request service at all. Thus if the customer is removed from the current schedule, it will not request service at a later time. The only diversion of vehicles happen when an unrealized dynamic customer is skipped, and no traveling is wasted due to the absence of dynamic customers. As a result, the total travel cost incurred by the vehicles is always equal to the total cost of the induced partial solution as calculated in Definition 7. Thus, we can conclude that under the dynamic routing strategy defined above, the HPOCS mechanism satisfies the first five desirable properties discussed in Section 3.2. The proofs follow directly from the proofs presented in Furuhata et al. (2015)[1]. Additionally, under this routing strategy, we can see that the upper bound of any customer's shared cost is twice its direct travel cost. Due to the immediate response property, a customer's highest shared cost throughout the timeline occurs at the initial quote. For any initial quote, it will be lower than twice that customer's direct travel cost because if that customer is the first customer in the coalition, its initial quote is equal to twice its direct travel cost, and if the customer is not the first customer in the coalition, its shared cost will be less than or equal to twice its direct travel cost due to the triangular inequality.

As for the early incentive property, HPOCS fails to satisfy this property. A simple counter-example would be when there is only one advance customer request, the initial quote per alpha value for this advance customer is the back and forth travel cost divided by its alpha value. When a dynamic customer makes its request and suppose the two customers form into a coalition, then their shared cost per alpha value will be the same which is the initial quote per alpha value for the dynamic customer. This value is no larger than that of the first customer because HPOCS satisfies the immediate response property. With this in mind, we device another mechanism based on HPOCS to see if we can solve this problem. In Section 4.2, we use non-decreasing discount functions to provide an extra incentive for customers to request early at the expense of losing the budget balance property.

We note it is possible that the solution to the static VRP is infeasible while the dynamic VRP is feasible. But HPOCS is designed to only consider dynamic VRP solutions from feasible static VRP solutions in order to ensure that the first five properties hold. However, we derive an extended mechanism, HPOCSrO, which relaxes this assumption but at the expense of the loss of the ex-post incentive compatibility property. As shown in Section 4.3, we incorporate a re-optimization method in generating total costs for partial schedules

[1] The proof of the individual rationality property is a general proof such that the customer can have any willingness-to-pay level and the property still holds.

to improve the performance of the HPOCS mechanism by reducing the overall shared cost. Compared with the DVRP solution (grand schedule and induced partial schedule) used in HPOCS, this extension fails to satisfy one of the major assumptions for the total cost which leads to the loss of the ex-post incentive compatibility property.

4.2 Hybrid Proportional Online Cost-Sharing with Discount (HPOCSD)

In this section, we introduce a modification of the HPOCS mechanism that aims to incentivize customers to request service early. Generally speaking, this can be achieved by offering discounts to advance customers and applying overcharge to dynamic customers. The same discount factor should be used for all advance customers in order to maintain the online fairness property. However, the overcharge factor can be different for different dynamic customers, and may be dependent on their actual request times. We design and study the exponential overchage heuristic method for calculating the suitable overcharge factor for realized dynamic customers, based on their request orders and the discount factor for advance customers. In the following sections, we formally define the Hybrid Proportional Online Cost-Sharing with Discount (HPOCSD) mechanism and study its properties.

4.2.1 Mechanism Design

The idea behind HPOCSD is to use the modified charges to substitute for the HPOCS shared costs and offer the modified charges to the customers. All of the calculations of the total costs, marginal costs, coalition cost per alpha values, shared costs, and the definition of coalitions remain the same as defined by the HPOCS mechanism. Additional notations and definitions are as follows.

δ	the discount factor
λ_i	the cost modifier of customer i
$g(n, \delta)$	the overcharge function

We require that $0 < \delta \leq 1$ and that $g(n, \delta) \geq 1, \forall n, \delta$.

Definition 12. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n be the index order of the customer on request order $\bar{\pi}_t$. Equivalently, $\bar{\pi}_t(n) = i$ for some $1 \leq n \leq |C(t)|$. Then the cost modifier of customer i under request order $\bar{\pi}_t$ is defined as

$$\lambda_{\bar{\pi}_t(n)} = \begin{cases} (1-\delta) & \text{for } 1 \leq n \leq |\mathbb{AC}| \\ (1+g(n,\delta)) & \text{for } |\mathbb{AC}| < n \leq |C(t)| \end{cases} \tag{35}$$

The cost modifier for all advance customers is the same, and is equal to $1 - \delta$. The cost modifier for a dynamic customer depends on the value of the function $g(n, \delta)$, which returns the overcharge factor based on the request index of the customer and the discount factor used for advance customers.

Definition 13. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n be the index order of the customer on request order $\bar{\pi}_t$. Equivalently, $\bar{\pi}_t(n) = i$ for some $1 \leq n \leq |C(t)|$. Then the charge of customer i at time t under request order $\bar{\pi}_t$ is defined as

$$charge_t\left(\bar{\pi}_t(n)\right) = cost_t\left(\bar{\pi}_t(n)\right) \lambda_{\bar{\pi}_t(n)} \tag{36}$$

where $cost_t\left(\bar{\pi}_t(n)\right)$ denotes the HPOCS shared cost as defined in Definition 10. $charge_t\left(\bar{\pi}_t(n)\right)$ is the value that is provided to the customer.

We define the HPOCSD mechanism by using the same structure as the HPOCS mechanism presented in Section 4.1, except that all $cost_t\left(\bar{\pi}_t(n)\right)$ values are replaced with $charge_t\left(\bar{\pi}_t(n)\right)$ values. The same dynamic routing strategy presented in Section 4.1.1 is used for scheduling and routing vehicles.

4.2.2 Analysis of Properties

We now discuss the properties of the HPOCSD mechanism.

Proposition 5. *The HPOCSD mechanism satisfies the online fairness, immediate response, individual rationality, and ex-post incentive compatibility properties, provided that the* overcharge function $g(n, \delta)$ is nondecreasing in n.

Proof. We first prove the online fairness property. For any time $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any customer $i \in C(t)$, let n_1 and n_2 be two indices representing advance customers, $1 \leq n_1 \leq n_2 \leq |\mathbb{AC}|$. Since the HPOCS mechanism satisfies the online fairness property, we have

$$\frac{cost_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} = \frac{cost_t\left(\bar{\pi}_t(n_2)\right)}{\alpha_{\bar{\pi}_t(n_2)}} \tag{37}$$

Since both n_1 and n_2 are advance customers, their cost modifiers are the same and are equal to δ. The equation above then implies that

$$\frac{charge_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} = \frac{cost_t\left(\bar{\pi}_t(n_1)\right)(1-\delta)}{\alpha_{\bar{\pi}_t(n_1)}} = \frac{cost_t\left(\bar{\pi}_t(n_2)\right)(1-\delta)}{\alpha_{\bar{\pi}_t(n_2)}} = \frac{charge_t\left(\bar{\pi}_t(n_2)\right)}{\alpha_{\bar{\pi}_t(n_2)}} \tag{38}$$

which proves the online fairness property for advance customers. Now suppose n_1 and n_2 be two indices representing dynamic customers, $|\mathbb{AC}| < n_1 \le n_2 \le |C(t)|$. Since the HPOCS mechanism satisfies the online fairness property, we have

$$\frac{cost_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} \le \frac{cost_t\left(\bar{\pi}_t(n_2)\right)}{\alpha_{\bar{\pi}_t(n_2)}} \tag{39}$$

Given that both n_1 and n_2 are dynamic customers and that function $g(n, \delta)$ is nondecreasing in n, we have $1 \le g(n_1, \delta) \le g(n_2, \delta)$. It then follows that

$$\frac{charge_t\left(\bar{\pi}_t(n_1)\right)}{\alpha_{\bar{\pi}_t(n_1)}} = \frac{cost_t\left(\bar{\pi}_t(n_1)\right)\left(1 + g(n_1, \delta)\right)}{\alpha_{\bar{\pi}_t(n_1)}} \le \frac{cost_t\left(\bar{\pi}_t(n_2)\right)\left(1 + g(n_2, \delta)\right)}{\alpha_{\bar{\pi}_t(n_2)}} = \frac{charge_t\left(\bar{\pi}_t(n_2)\right)}{\alpha_{\bar{\pi}_t(n_2)}} \tag{40}$$

We have now proved that the online fairness property is satisfied for both advance and realized dynamic customers.

Similarly, given that for each customer i, the cost modifier λ_i is fixed and independent of time, and that the overcharge function $g(n, \delta)$ is nondecreasing in n, it can be proved that the HPOCSD mechanism inherits the immediate response, individual rationality, and ex-post incentive compatibility properties from the HPOCS mechanism. □

Proposition 6. *The HPOCSD mechanism is $\delta-$budget balanced. That is to say, at any time during the planning horizon, the sum of the charges for all customers that have become realized recovers at least* $100 \times (1 - \delta)$ *percent of the total travel cost of the corresponding induced partial schedule.*

Proof. For any grand schedule $\bar{S}$, at time $t = 0$, $C(0) = \mathbb{AC}$. We have

$$\sum_{n=1}^{|\mathbb{AC}|} charge_0\left(\bar{\pi}_0(n)\right) = \sum_{n=1}^{|\mathbb{AC}|} cost_0\left(\bar{\pi}_0(n)\right)(1 - \delta) \tag{41}$$

$$= (1 - \delta) \times totalcost\left(\bar{S}\left(\bar{\pi}_0\left(|\mathbb{AC}|\right)\right)\right) \tag{42}$$

$$= (1 - \delta) \times totalcost\left(\bar{S}\left(C(0)\right)\right) \tag{43}$$

which means that at time $t = 0$, the sum of the charges for advance customers using the HPOCSD mechanism recovers exactly $100 \times (1 - \delta)$ percent of the total travel cost of the partial solution induced by $\bar{S}$ and the set $\mathbb{AC}$. Now consider any time during the planning horizon, $1 < t \leq T_{max}$. We have

$$\sum_{n=1}^{|C(t)|} charge_t\left(\bar{\pi}_t(n)\right) = \sum_{n=1}^{|\mathbb{AC}|} cost_t\left(\bar{\pi}_t(n)\right)(1-\delta) + \sum_{n=|\mathbb{AC}|+1}^{|C(t)|} cost_t\left(\bar{\pi}_t(n)\right)(1+g(n,\delta)) \tag{44}$$

$$\geq \sum_{n=1}^{|\mathbb{AC}|} cost_t\left(\bar{\pi}_t(n)\right)(1-\delta) + \sum_{n=|\mathbb{AC}|+1}^{|C(t)|} cost_t\left(\bar{\pi}_t(n)\right)(1-\delta) \tag{45}$$

$$= (1-\delta) \times \sum_{n=1}^{|C(t)|} cost_t\left(\bar{\pi}_t(n)\right) \tag{46}$$

$$= (1-\delta) \times totalcost\left(\bar{S}\left(C(t)\right)\right) \tag{47}$$

where the inequality follows from the fact that the cost modifier of any dynamic customer is always greater than or equal to the cost modifier of any advance customer, $g(n,\delta) \geq (1-\delta), \forall n, \delta$. Equation 47 implies that the sum of the charges for all customers who have requested service recovers at least $100 \times (1-\delta)$ percent of the total travel cost of the corresponding induced partial solution. Thus we can conclude that the HPOCSD mechanism is $\delta-$budget balanced. □

In addition, we note that the equality in equation 45 is achieved if and only if $1 + g(n,\delta) = 1-\delta, \forall n, \delta$. This can only be true if $g(n,\delta) = \delta = 0$. Without the discounts and overcharges, the HPOCSD mechanism reduces to the HPOCS mechanism. In the HPOCSD setup with strictly positive discounts and overcharges, equation 45 will always imply an inequality relationship. It then follows that

$$\sum_{n=1}^{|C(t)|} charge_t\left(\bar{\pi}_t(n)\right) > (1-\delta) \times totalcost\left(\bar{S}\left(C(t)\right)\right) \tag{48}$$

at any time $1 < t \leq T_{max}$. This means that the worst-case budget deficit scenario always happens at time $t = 0$, when there is no realized dynamic customer and the sum of the HPOCSD charges recover exactly $100 \times (1-\delta)$ percent of the total travel cost.

We have shown that the HPOCSD mechanism is approximately budget balanced. The loss of the budget balance property is the sacrifice that is made to encourage customers to request early. Proposition 6 provides an upper bound on the worst-case budget deficit, which is dependent on the discount factor provided to the advance customers. Intuitively speaking, the larger the discount, the more incentive it provides to encourage customers to request early, and the bigger the risk of not being able to recover the total operating cost. On the other hand, Proposition 6 does not state that the HPOCSD mechanism will always incur a budget deficit. It could happen that the overcharge on dynamic customers recovers fully the discounts provided to advance customers and a budget balance is achieved. It could also happen that the overcharge over compensates for the discounts, such that a budget surplus is generated.

As for the early incentive property, HPOCSD does not guarantee that the property holds for all cost modifiers. Referring to Definition 12, whether this property is satisfied depends on the choice of the discount factor δ and the overcharge function $g(n, \delta)$. In fact, we can select the parameters such that the early incentive property holds. For example, by setting $acpa(1-\delta) \leq \frac{cost_t(\bar{\pi}_t(n_i))(1+g(n_i,\delta))}{\alpha_{\bar{\pi}_t(n_i)}}$ for all $|\mathbb{AC}| < n_i \leq |C(t)|$, the early incentive property holds. In Section 5.2, we show in the experiments that the early incentive property is satisfied given a proper choice of these two parameters.

4.3 Hybrid Proportional Online Cost-Sharing with Re-optimization (HPOCSrO)

In this section, we propose to incorporate re-optimization to tackle the problem in HPOCS that the grand solution used to calculate total cost may perform poorly when the request probability is low and the number of realized customers is small since the operation cost of the grand schedule is less representative of the actual total cost. This problem will not only cause the early incentive property to fail, but also drive the final total cost far away from optimal, making all the customers' final cost less than ideal. In general, we address the above problem by replacing the grand solution in HPOCS with repeated re-optimization to compute the schedule that can reduce the total cost and therefore boost the overall performance of the HPOCS mechanism. However, this modification itself has a major issue of violating one of the desired properties of a well-designed cost-sharing scheme, the ex-post incentive compatibility property. We first introduce the mechanism design for HPOCSrO and then we analyze the properties of this mechanism.

4.3.1 Mechanism Design

The HPOCSrO mechanism shares the same framework as the HPOCS mechanism, with the exception that the total cost function is defined differently. Recall in Section 4.1 that the grand schedule $\bar{S}$ is calculated based on solving a deterministic VRP problem. Differently, the HPOCSrO mechanism calculates a partial schedule initially as well as throughout the whole time horizon. The general framework of the proposed mechanism can be summarized as follows.

Initialization. $t = 0$. Formulate a static vehicle routing problem corresponding to the set of customers $\mathbb{AC}$ and construct the partial solution $\bar{S}(\mathbb{AC})$ using the same heuristics as the grand solution $\bar{S}$.

Quoting advance customers. All advance customers receive their initial quotes at time $t = 0$. This step calculates the advance cost per alpha value *acpa* and the shared cost of each advance customer using the same method as in HPOCS (see Section 4.1).

Quoting dynamic customers. A dynamic customer i receives its initial quote when it requests service at time $t = u_i$. Customer i is added into the existing partial schedule using the cheapest insertion method (Zou and Dessouky, 2018). Then the mechanism updates the total cost and calculates the shared cost accordingly.

Re-optimizing and updating the costs. At each decision epoch, the same heuristics in Zou and Dessouky (2018) are used to optimize the current partial schedule resulting in a reduction in the total cost and the shared cost of all customers who have requested service by this decision epoch are updated.

Final shared costs. At time $t = T_{max}$, all of the randomness in the system has been realized. The solution schedule consisting of all advance and realized dynamic customers is produced and the shared cost of these customers at time T_{max} is outputted as the final cost of service for them.

4.3.2 Analysis of Property

Given that the HPOCS mechanism is proven to possess the first five desirable properties discussed in Section 4.1.1, it follows that the HPOCSrO mechanism also possesses these properties except for the ex-post incentive compatibility property.

Recall in Section 4.1.1, we explain that for a proportional online cost-sharing mechanism to satisfy all five desirable properties, its total cost function should be non-decreasing over time and be independent of the request order at any time. It is trivial to show that the HPOCSrO mechanism does not satisfy the first assumption. The total cost function over time is not an optimal solution to the current customer group but rather a good solution obtained by local search heuristics. In other words, adding a customer into the dynamic vehicle route after a re-optimization is executed may have less total cost than before. Removing this assumption will lead to the loss of ex-post incentive compatibility property which implies that if we can prove that the total cost function in the HPOCSrO mechanism satisfies the independence assumption, the first four desirable properties are maintained (Furuhata et al., 2015).

Proposition 7. *For any partial solution S_t, $t \in [0, T_{max}]$ and the corresponding set of customers who have requested service $C(t)$, the special request order $\bar{\pi}_t$, and any integer $n \in [1, |C(t)|]$, the HPOCSrO total cost function totalcost$(S_t(\bar{\pi}_t(n)))$ is independent of the request order of customers $\{\pi_t(1), \ldots, \pi_t(n)\}$.*

Proof. The partial solution $S_t(\pi_t(n))$ is constructed by inserting a new dynamic customer using the cheapest insertion method. As a result, $S_t(\pi_t(n))$ is only concerned with the set of customers that have requested service, but not about the ordering of the requests. Therefore, for any two different orderings π_t and $\pi_t^{'}$ containing the same n customers, we have $S_t(\pi_t(n)) = S_t(\pi_t^{'}(n))$. □

Given Proposition 7, and following the same framework as in HPOCS, we can conclude that the HPOCSrO mechanism satisfies the online fairness, immediate response, individual rationality and budget balance properties. Regarding the early incentive property, there is no guarantee that the HPOCSrO satisfies this property. However, using re-optimization to achieve a lower total cost for all the customers results in lower initial quotes for all the customers and the experiments in Section 5.3 show that the initial quotes for the advance customers are generally lower than those for the dynamic customers and thus providing an incentive for customers to request early.

5 Experimental Analysis

In this section, we present the simulation results to study the effectiveness of the proposed mechanisms in terms of maintaining the desirable properties and providing desirable final costs to customers. We first present the results for the HPOCS mechanism and then the results for the two extension mechanisms, HPOCSD and HPOCSrO.

All simulations are performed on a modified Solomon RC201 instance for the vehicle routing problem with time windows (VRPTW). The instance is representative of the benchmark cases in the literature for VRP (Solomon, 1987). The instance specifies all of the deterministic information on customer locations, demands, service time windows, and fleet capacity. There are 100 customers, $\mathcal{N} = 100$. The length of the planning horizon is 960 time steps, $T_{max} = 960$. A dynamic vehicle routing instance is constructed by specifying two parameters, namely the percentage of advance customers - $ACPercent$, and the probability that a dynamic customer requests service - *RequestProb*. These two parameters jointly determine the mixture between the number of advance customers and the expected number of realized dynamic customers in the problem. We assume that all dynamic customers have the same probability of requesting service. $q_i = RequestProb, \forall i$. We use a triangular distribution function to model $f_i(t)$, the conditional probability density function of request time u_i. In particular, the minimum value of the distribution is set to 0, and the maximum value of the distribution is set to be equal to the request deadline, v_i. The mode of the distribution is set to $\frac{3}{4}v_i$. Within this time frame, the dynamic customers are more likely to make the request close to the time they need service. A realization of the problem specifies the actual set of advance customers, a group of dynamic customers who are to make requests, and the precise request times of these customers. For each dynamic instance, we simulate 50 realizations and report the average results. The grand schedule of each realization is calculated based on the assumption that all customers (both advance and dynamic) are known at the beginning of the planning horizon and must be served. According to our problem definition in Section 3.1, we assume the willing-to-pay level of customer i, W_i, has the following form: $W_i = \beta_i * \alpha_i$ where $\beta_i \geq 2$. In the next set of experiments, we set all β_i to 2. We note that if β_i is less than 2 then it is possible that some customers may decline the service. However, this does not fundamentally change the proposed methodology (mechanism and routing strategy) since these customers will be treated like potential dynamic customers who do not eventually request service on that particular day.

The shared cost gets updated each time when a new dynamic customer requests service because existing customers can choose to form a coalition with the new customer if it can lower their shared costs. It is worth exploring how the sequence of the shared costs changes over time and how the overall pattern may be different for different customers.

5.1 Analysis for HPOCS

To test the performance of the HPOCS mechanism, we set the demand scenario to be $ACPercent = 0.25$ and $RequestProb = 0.75$. This setup reflects an operating environment with a relatively high proportion of dynamic customers. The number of advance customers is $100 * 0.25 = 25$ and the expected number of realized dynamic customers is $100 * (1 - 0.25) * 0.75 \approx 57$.

Figure 1 illustrates a graph of a series of HPOCS shared costs of selected customers in the demand scenario, where The horizontal axis represents the request order. In this scenario, the first 25 positions of the request order correspond to advance customers. The vertical axis represents the shared cost per alpha value. Each data point on the graph represents the shared cost per alpha value of a selected customer at the time when the dynamic customer whose order index corresponds to the horizontal axis value requests service. Each trajectory on the graph represents the series of shared cost per alpha values of a selected customer. The first data point on each trajectory shows the initial quote per alpha value of the customer.

For example, the first trajectory shows the series of shared cost per alpha values of the first advance customer on the special request order. Since all advance customers have the same shared cost per alpha value at any time throughout the planning horizon, it is sufficient to use the first advance customer to represent the entire set. The following four series correspond to four dynamic customers. “Dynamic 1” corresponds to the first dynamic customer to request service. “Dynamic 2” represents the dynamic customer whose request position falls around the first 3-quantiles of the total expected number of realized dynamic customers. Similarly, “Dynamic 3” represents the dynamic customer whose request position falls around the second 3-quantiles of the total expected number of realized dynamic customers. The last series represents a dynamic customer positioned near the end of the request order. It is worth pointing out that the request order shown by the horizontal axis is not equivalent to time.

It is evident from the graph that the shared cost of any customer is nonincreasing over the request order, which is a direct outcome of the way shared costs are calculated in the HPOCS mechanism. In particular, each time when a new customer requests service, existing customers will have the opportunity to form a coalition with the new customer. They will choose to form a new coalition if and only if their shared cost per alpha values can be lowered. Otherwise, existing customers will choose to stay in their current coalitions.

Figure 2 illustrates a graph of the HPOCS initial quotes and the final shared costs of all customers in the base case demand scenario. Recall that the initial quote is the first shared cost value a customer receives and is the value that the customer has to use to make the decision of whether to accept the service

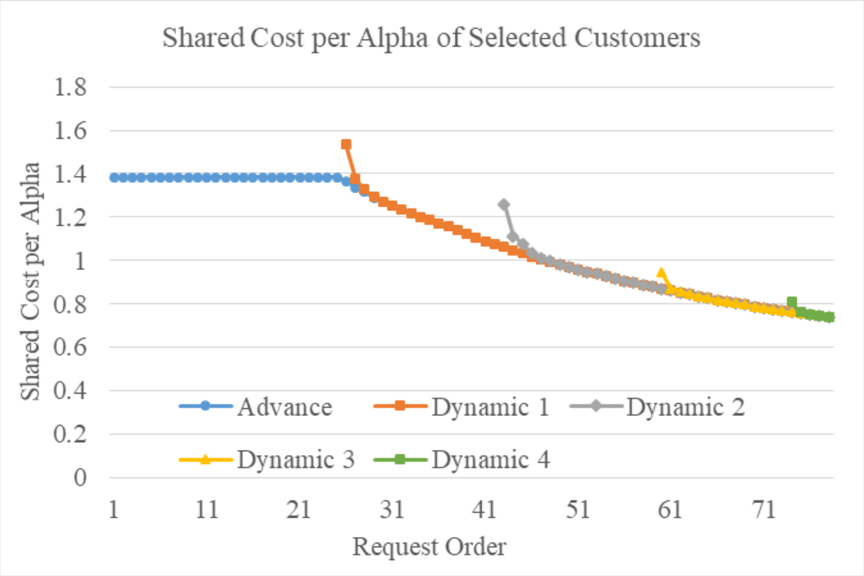

Figure 1: Trajectories of the HPOCS shared cost per alpha values in base case

or not. The final shared cost is the price that the customer actually pays for the service. These two values are the two most important shared cost values. All of the values shown on the graph are on the per-alpha basis. Similar to Figure 1, the horizontal axis represents the request order and the vertical axis represents the shared cost per alpha value. The upper series contains the initial quotes of all customers and the lower series contains the corresponding final shared costs. For each customer, its initial quote is always greater than or equal to its final shared cost, as guaranteed by the immediate response and individual rationality properties.

1. We first study the initial quotes provided to all customers. By Proposition 2, the initial quote per alpha value at time $t = 0$ of all advance customers are the same, and are equal to the advance cost per alpha *acpa* value. This is reflected by the level segment on the initial quote curve. For the realized dynamic customers, their initial quotes start higher than that of the advance customers, but drop very quickly as more dynamic customers become realized. Recall that the HPOCS mechanism calculates the total costs based on the total travel costs of induced partial solutions. All of these partial solutions are induced by a single grand solution that is constructed at time $t = 0$ and is fixed throughout the planning horizon. As more customers request service, the grand schedule is gradually recovered and the synergy among the group of customers who have requested service increases. The marginal cost decreases, which makes it more attractive and likely for existing customers to form a new coalition with the customer who just requested. This in turn causes the initial quote offered to the dynamic customer that just became realized to decrease over time. This phenomenon can be undesirable since higher initial quotes offered to early request dynamic customers may turn them away if a lower β_i is implemented. If those early request dynamic customers decline service, the similar high level initial

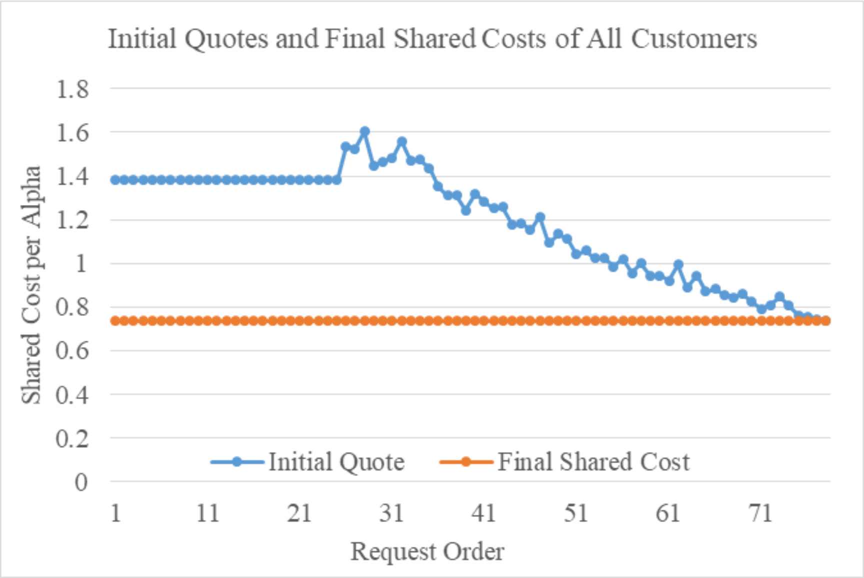

Figure 2: The HPOCS initial quotes and final shared cost values in base case

quotes will be offered to subsequent dynamic customers who request service, and the same problem remains. For the same reason, it fails to satisfy the early incentive property since the initial quotes offer to many realized dynamic customers drop below the initial quote of advance customers.

2. We then study the final shared costs of all customers. It can be clearly seen from the graph that the final shared cost curve nearly represents a flat line. The final shared cost per alpha values across all advance and realized dynamic customers tend to be the same, which suggests that all of the customers tend to form a single coalition. The synergy among customers becomes so high that existing customers almost always can lower their shared costs by forming a new coalition with the dynamic customer that just became realized. This may be undesirable since customers that request early do not have any advantage over customers that request late. The lack of differentiation in the final shared costs fails to encourage customers to request service early.

To improve the performance of the HPOCS mechanism, we proposed HPOCSD and HPOCSrO, their experimental results are shown in the following subsections.

5.2 Analysis for HPOCSD

We use the same experimental setup as introduced in the beginning of Section 5. For each realization of the dynamic vehicle routing problem, we solve the corresponding cost allocation problem using the HPOCSD mechanism paired with the exponential overcharge heuristic method for calculating the overcharge factors.

Exponential overcharge. The overcharge factor is designed to be exponentially increasing over the request order, which provides smaller penalties for early request dynamic customers and larger penalties for late request dynamic customers as compared to a constant or linear overcharge heuristic.

$$g_{exp}(n,\delta) = \frac{\delta \mathcal{N}_{AC}}{\mathcal{N}_{ERDC}} \times \frac{(\exp(\gamma_{exp}(n+1-\mathcal{N}_{AC})) - 1)}{(\exp(\gamma_{exp}\mathcal{N}_{ERDC}) - 1)} \times \gamma_{exp}^{'} \qquad (49)$$

The above definition states that the exponential overcharge factor is calculated based on and in proportion to the discount factor, and is exponentially increasing over the request index n. Two parameters γ_{exp} and $\gamma_{exp}^{'}$ are needed to adjust the actual overcharge level to avoid bias.

Intuitively speaking, the larger the discount, the more significant the effect of incentivizing customers to request early. At the same time, the mechanism may be subject to bigger risks of not being able to recover the total operating cost. Thus, it is worth examining the performance of the exponential overcharge heuristic using different discount factor levels. We perform simulations using four discount factors, namely $\delta = 0.1, 0.2, 0.3$ and 0.4. We use the same base case demand scenario as used in Section 5.1, where $ACPercent = 0.25$ and $RequestProb = 0.75$.

Figure 3 shows graphs of the initial quote per alpha and the final charge per alpha values of all customers under the HPOCSD mechanism, when paired with the exponential overcharge heuristic. The figure contains four panels, and each panel contains the graph of the initial quotes and the final charges corresponding to one of the four discount factors that we have tested. All of the values shown on the graph are on the per-alpha basis. The legends and axis in each graph are arranged in the same manner as in Figure 2.

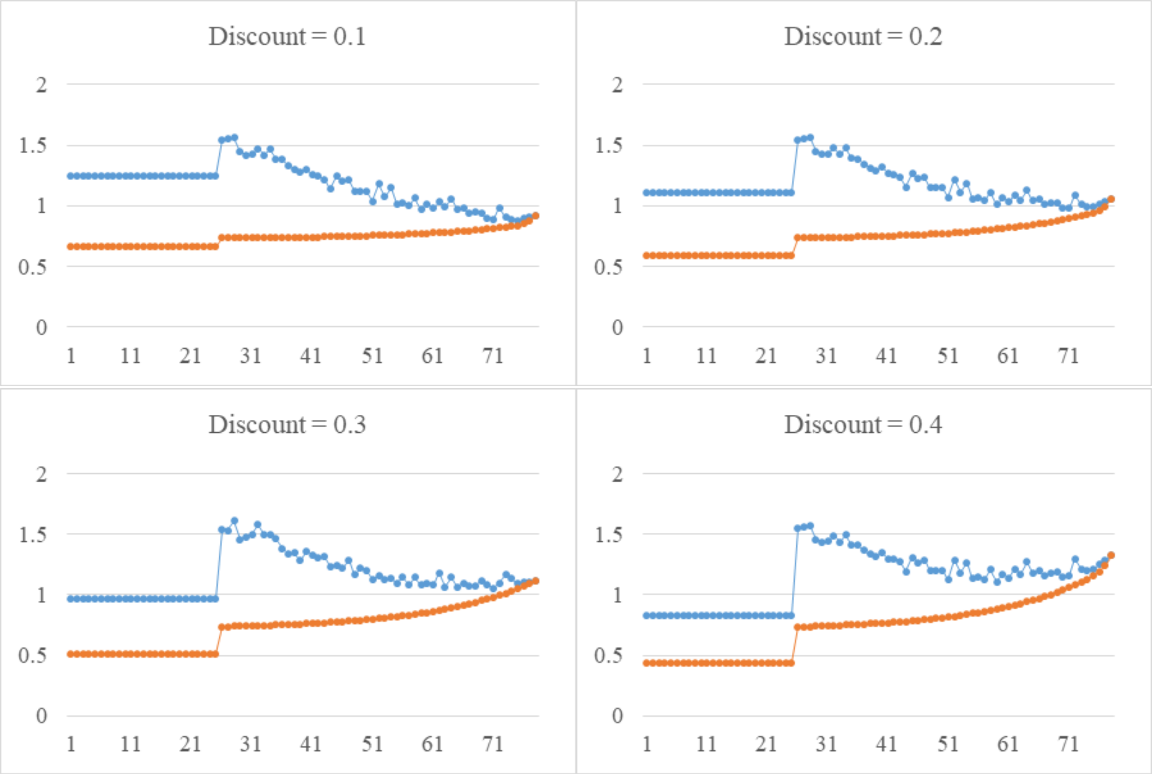

Figure 3: Initial quotes and final charges under HPOCSD with exponential overcharge

1. We start our analysis by focusing on the initial quote curve. When comparing the shape of the initial quote curve to that of the HPOCS model, it is evident that the flat segment corresponding to advance customers is lowered and the part corresponding to the realized dynamic customers is raised. As a result, the probability that an advance customer accepts its initial quote is increased. Meanwhile, dynamic customers are effectively penalized and the probability that they accept their initial quotes may decrease. This phenomenon can be observed for the exponential overcharge heuristic using any of the discount factors we have tested. Similarly, it is shown to be more effective when a larger discount factor is used. We also notice that when the discount factor is larger than 0.3, the early incentive property is satisfied.

2. As discussed in Section 3.2, the online fairness property is only concerned with the final charges of customers, rather than the initial quotes. Thus it is possible for a mechanism that satisfies the online fairness property to have undesirable behavior associated with the initial quotes as indicated in Section 5.1. In order to correct this issue, an effective overcharge heuristic should raise the initial quotes for dynamic customers high enough such that all of them are at least as high as that offered to advance customers. Based on Figure 3, a discount of 30% is sufficient for the exponential heuristic to be effective.

3. We now focus on the segment of the initial quote curve that corresponds to realized dynamic customers. The exponential heuristic tends to flatten the segment of the initial quote curve corresponding to realized dynamic customers, since it assigns increasingly larger overcharge factors to customers who request late. In particular, it can be observed that the decreasing trend can even be reversed at the tail of the initial quote curve when using a discount factor that is large enough.

4. We then analyze the effect of discounts and overcharges on the final charges. Recall that under the HPOCS mechanism, the final shared costs of all the customers tend to be the same as many dynamic customers become realized, as the synergy among customers becomes too high. Figure 3 shows that the exponential overcharge heuristic can prevent the advance and realized dynamic customers to have the same final charge per alpha value, even when a small discount factor is used. In particular, a jump in the final charge value can be observed for the first dynamic customer that becomes realized. In addition, it also causes the final charges for dynamic customers to resemble an exponential pattern respectively. Both effects are more significant when a larger discount factor is used.

The simulation results discussed above suggest that larger discount factors are generally more effective in terms of promoting customers to request early. Meanwhile, based on Proposition 6, a larger discount factor could also lead to a bigger budget deficit in the worst case. Thus it is worth examining the performance of the HPOCSD mechanism paired with the exponential overcharge heuristic on budget balance when using different discount factors. We use the percentage of the cost recovered as the performance measure. For each realization of the problem, and each discount factor, we calculate the percentage of the total travel cost that can be recovered by the sum of the final HPOCSD charges for all customers that become realized. In particular, the percentage of the cost recovered *pcr* is calculated as

$$pcr = \frac{\sum_{n=1}^{|C(T_{max})|} charge_{T_{max}}\left(\bar{\pi}_{T_{max}}(n)\right)}{totalcost(C\left(T_{max}\right))} \tag{50}$$

Table 1 summarizes the percentage of the cost recovered values under the exponential overcharge heuristic using different discount factors ranging from 0.1 to 0.4. We simulate the heuristic paired with each discount level on 50 realizations of the DVRP. The same set of realizations are used for all of the discount level. For each discount level, we report the average percentage of the cost recovered, the minimum percentage of the cost recovered among all realizations, and the maximum percentage of the cost recovered among all realizations.

	Percentage of the Cost Recovered		
	Avg.	Min.	Max.
Discount	Exponential		
0.1	100.0%	99.4%	100.8%
0.2	100.1%	98.8%	101.5%
0.3	100.1%	98.2%	102.3%
0.4	100.2%	97.6%	103.0%

Table 1: Budget balance analysis of HPOCSD for the base case

1. The average percentage of the cost recovered of the exponential overcharge heuristic at all discount levels are close to 100%, which is the target value we use when fine tuning the model parameters. Besides, for each discount level, the minimum and maximum percentage values of the cost recovered are generally positioned symmetrically around the corresponding mean value. Equivalently speaking, the maximum deficit and the maximum surplus incurred among all realizations are generally the same. This implies that the parameter settings that we use are not biased towards budget balance or surplus, and lead to budget balanced cost allocations in general.

2. There is bigger variation in the performance measure when a larger discount factor is used. For example, when using a discount of 40%, even though the HPOCSD mechanism is generally budget balanced on average, it could incur either a 2.4% budget deficit or a 3.0% budget surplus in the worst case. If a 10% discount is used, the worst-case deviations are both less than 1%.

Based on the above analysis, it can be concluded that the HPOCSD mechanism can indeed resolve the problems observed for the HPOCS mechanism, at the cost of losing the budget balance property of the original formulation. Nevertheless, the HPOCSD mechanism remains approximately budget balanced. And it is shown that the exponential heuristic we have tested is effective in providing an incentive to make an advance or early dynamic request.

5.3 Analysis for HPOCSrO

We now present simulation results to show the effectiveness of the HPOCSrO mechanism in improving the overall performance of the HPOCS mechanism. For HPOCSrO, the number of decision epochs which we use to re-optimize the partial solution is set at 20 which is shown to be a nice balance between identifying improvements in the solution quality and computation time (Zou and Dessouky, 2018).

The HPOCS mechanism holds all the desirable properties of a cost-sharing mechanism but could perform poorly in terms of the final shared cost when the number of dynamic customers is small, and this effect is magnified when the number of customers requesting service is small. We use the scenarios where $RequestProb = 0.25, 0.5$ and the number of advance customers is 10 which is a small value to compare the differences between the two above routing strategies.

Figures 4 and 5 show graphs of the initial quote per alpha value (with legend "Initial quote") and the final shared cost per alpha value (with legend "Final price") of all customers under the two strategies in each scenario. Each graph represents a scenario and the 2 panels within the graph are the routing performances corresponding to the two strategies: HPOCS and HPOCSrO.

Scenario 1: # Advance Customers = 10; $RequestProb = 0.25$

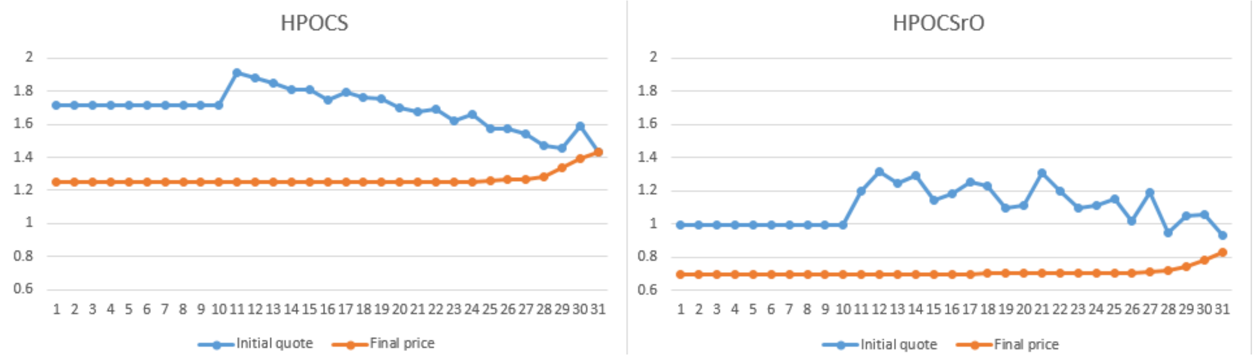

Figure 4: Initial quote and final shared cost of the two methods in scenario 1

Based on the simulation results, we can make the following observations:

1. We first examine the initial quotes. We find that the HPOCS results exhibit a downward trend with customers who call in later having a lower initial quote than the earlier customers, favoring those who request later than advance customers as described in Section 5.1. The HPOCSrO results have a smaller slope which implies dynamic customers benefit less by delaying. This is favorable because even though the early incentive property does not strictly hold, fewer dynamic customers have a lower initial quote than that of the advance customers.

Scenario 2: # Advance Customers = 10; $RequestProb = 0.5$

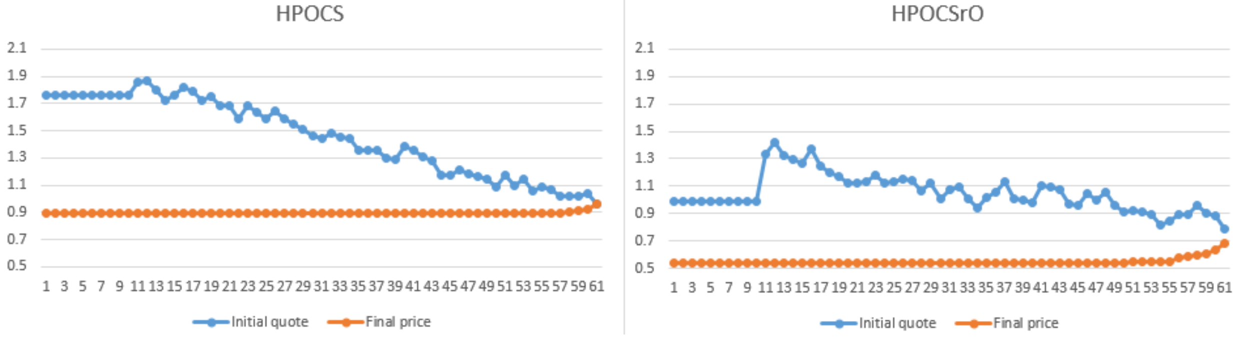

Figure 5: Initial quote and final shared cost of the two methods in scenario 2

2. We then examine the final shared cost. We find that HPOCSrO has a smaller final shared cost indicating the efficiency of the re-optimization approach in reducing the final shared cost of each customer. Additionally, the gap of the final costs between the two methods is getting smaller as $RequestProb$ gets higher which supports our assumption that the final cost performance of HPOCS is acceptable when the number of realized customers is large.

3. Next, when we fix the number of advance customers, as the probability of a dynamic customer calling in ($RequestProb$) gets higher, both methods encounter a lower final price.

Given the above analysis, we can conclude that the HPOCSrO mechanism does help improve the overall performance of the proportional cost-sharing design. However, we need to keep in mind that it suffers from the consequences of losing the ex-post incentive compatibility property which we will investigate in the next section.

To test the impact of losing the ex-post incentive compatibility property, we look into scenarios where there are 21 dynamic customers and the number of advance customers is 0, 10, and 20 respectively. Notice that each scenario has 100 instances that share the same generating method as the previous simulations. We then introduce the concept of **Delay Slot** which is a slot where the first dynamic customer is delayed to. For example, delay slot 6 means the previous 1st dynamic customer is now the 6th dynamic customer in the ex-post instance. For each scenario, all 100 instances are evaluated, and for each instance, we select 5 slots that are evenly distributed, namely the 2nd, 6th, 11th, 16th and 21st slots. This results in altogether $100 \times 5 = 500$ samples for each scenario. And if we aggregate all scenarios into one, the 1500 samples with 300 instances for each delay slot give us the general impact of losing ex-post incentive compatibility property regardless of scenario settings.

All scenarios are compared based on the final shared cost per alpha value. The average results of the 500 samples for each scenario are displayed in Table 2. The 2nd, 3rd and 4th columns of the table display the percentage of getting a lower or higher or the same final shared cost when a dynamic customer delays its request submission. The 5th column depicts the percentage increase in the final shared cost for a delayed customer. Table 2 shows that as the number of advance customers increases, both the chances of resulting in a higher final shared cost and a lower final shared cost increase. And in total, 32.1% of the time, a customer who delays its request submission shall end up with lower final shared cost while 55% of the time the cost ends up higher. We note that the results in the table show the grand average across all the scenarios. For brevity, we do not show the detailed results for the different delay slots but note there is little impact with delaying to slot 2 but a delay to slot 21 can increase the final shared cost on average by 54.4%.

Scenarios	% Better off	% Worse	% Same	AVG Price Change
0AC_21DC	27.2%	50.6%	22.2%	8.871%
10AC_21DC	32.4%	56.8%	10.8%	8.623%
20AC_21DC	36.6%	57.6%	5.8%	5.921%
Total	32.1%	55.0%	12.9%	7.805%

Table 2: Average gap results of 500 samples in each scenario

5.4 Comparison of the Different Routing Strategies

We next want to compare the effectiveness of the two dynamic routing heuristics proposed in this paper, the ones used in HPOCS and HPOCSrO. Note that the same routing heuristic is used in HPOCSD as in HPOCS. We perform this analysis by comparing the total cost of the dynamic routing solutions against the static solution. Since it is difficult to identify optimal dynamic solutions, we compare them against a static solution where all the information is known at the time of scheduling. We choose the same experimental setting as in Section 5.1 with scenarios summarized in Table 3. We run 50 instances. For each instance, the static solution is determined by using the same algorithm as the grand schedule while assuming all the realized customers' information are known in advance. That is, both the advanced and the generated dynamic customers are known in advance and now the problem can be treated as a static routing problem. In this case the grand schedule only includes the advanced and the "realized" dynamic customers so there is no need for rescheduling. Therefore, this static solution serves as a good benchmark to the dynamic routing heuristics where the full information of realized customers is not known when generating the grand schedule and at each schedule update. The results are presented in Table 4. Each column represents the different routing strategy while each row represents the different scenarios. The "static" column refers to the solution when all the "realized customers" are known in advance. The "base" column refers to the solution from the routing strategies used in HPOCS and HPOCSD. The "Re-Optimization" refers to the solution from the routing strategy in HPOCSrO. The gap in Table 4 is the average percentage increase in the total cost of the dynamic routing heuristic from the static solution. The routing algorithm used in HPOCS has at most an average gap of 21% of the static solution and a minimum average gap of 1.09% attained in the last scenario. We can see that when we fix the number of advance customers to be 25, as the request probability of dynamic customers increases, the solution of the base routing strategy improves significantly from a gap of 20.8% to 5.5%. When we fix the request probability to be 50%, we observe that as the number of advance customers increases, the solution quality improves from a gap of 12.6% to 2.7%. Thus we can conclude that for the routing strategy that is used in HPOCS and HPOCSD, the more "realized" total customers we have in the system, the better the solution quality. The re-optimization routing strategy, on the other hand is within 13% across all the scenarios. It significantly outperforms the base routing strategy especially when the dynamic request probability is low (e.g., .25). For the other cases, there is not much difference between the two routing strategies since the base performs reasonably well in these scenarios.

Scenarios	**ACPercent**	**RequestProb**
AC25C25	0.25	0.25
AC25C50	0.25	0.50
AC25C75	0.25	0.75
AC50C25	0.50	0.25
AC50C50	0.50	0.50
AC50C75	0.50	0.75
AC75C25	0.75	0.25
AC75C25	0.75	0.50
AC75C25	0.75	0.75

Table 3: Experiment Scenarios

Scenarios	**Static**	**Base**	**Gap**	**Re-Optimization**	**Gap**
AC25C25	703.8	850.14	20.79%	764.5	8.62%
AC25C50	904.66	1018.3	12.56%	995.44	10.03%
AC25C75	1090	1150.4	5.54%	1169.7	7.31%
AC50C25	822.24	993.99	20.89%	898.42	9.26%
AC50C50	1023.68	1100.84	7.54%	1126.56	10.05%
AC50C75	1157.16	1195.24	3.29%	1240.8	7.23%
AC75C25	977.72	1134.46	16.03%	1098.84	12.39%
AC75C50	1154.2	1184.87	2.66%	1181.4	2.36%
AC75C75	1204.28	1217.46	1.09%	1235.74	2.61%

Table 4: Comparisons of Different Routing Strategies

6 Conclusions

In this paper, we study the problem of building a real-time cost-sharing transportation system, which results from horizontal cooperation among multiple suppliers. In this problem, part of the customer requests are known at the beginning of the planning horizon, while the rest of the requests become realized dynamically over time. There are two major research issues closely related to the problem we study, namely the dynamic vehicle routing problem (DVRP) and cost-sharing mechanism design, and this paper focuses on designing the cost-sharing mechanism in a dynamic vehicle routing setting.

We develop the Hybrid Proportional Online Cost-Sharing (HPOCS) mechanism as an online cost-sharing mechanism that combines proportional cost-sharing for calculating the initial quotes for advance customers and the Proportional Online Cost-Sharing (POCS) mechanism (Furuhata et al., 2015) for handling dynamic customer requests. The idea behind HPOCS is that customers can choose to form coalitions with customers who request service directly after them to decrease their shared costs. It is proved that the HPOCS mechanism satisfies five out of six the desirable properties we propose, including online fairness, budget balance, immediate response, individual rationality, and ex-post incentive compatibility.

The baseline HPOCS model is extended in two directions. One extension is to incorporate discounts for advance customers and overcharges for dynamic customers, which both help to incentivize customers to request early. The new HPOCSD mechanism is proved to be approximately budget balanced. All of the other properties of HPOCS are preserved. We propose the exponential overcharge heuristic method for calculating the overcharge factors. Simulation results show that it appears to be quite effective. The other extension is to incorporate periodic re-optimization to improve the performance on the final shared cost for the customers. In experiments across multiple scenarios, though losing the ex-post incentive compatibility property, HPOCSrO is shown to be a good mechanism design alternative to HPOCS when the $RequestProb$ is low and the number of all realized customers is small since the grand schedule in HPOCS assumes all customers request service before operating service and is therefore less representative of the actual total cost. Experimentally it is shown that the HPOCS mechanism performs poorly for the early incentive property. Only the HPOCSD mechanism can ensure that the early incentive property holds given a proper selection of the discounting parameters. Furthermore, experiments show that in general both HPOCSD and HPOCSrO perform much better than HPOCS in lowering the initial quotes of the advance customers.

More work can be done along the lines of improving the HPOCS mechanism. For example, while incorporating the dynamic vehicle framework to calculate the shared costs, we can add customer forecasting to see if it can further reduce the final shared costs. Additionally, we can target reducing the initial quote for the dynamic customers while maintaining all the desirable properties. There may also exist other approaches to improve the HPOCS mechanism, possibly at the cost of sacrificing one or more of the desirable properties.

Acknowledgement

The research reported in this paper was partially supported by Metrans and the National Science Foundation under grant CNS-1932615.

References

R. C. Anderson and A. Claus. Cost allocation in transportation systems. *Southern Economic Journal*, 43(1):793–803, 1976.

Y. Bleischwitz and B. Monien. Fair cost-sharing methods for scheduling jobs on parallel machines. *Journal of Discrete Algorithms*, 7(3):280–290, 2009.

F. Cruijssen, M. Cools, and W. Dullaert. Horizontal cooperation in logistics: Opportunities and impediments. *Transportation Research Part E: Logistics and Transportation Review*, 43(2):129–142, 2007.

J. Drechsel and A. Kimms. Computing core allocations in cooperative games with an application to cooperative procurement. *International Journal of Production Economics*, 128(1):310–321, 2010.

J. Drechsel and A. Kimms. Cooperative lot sizing with transshipments and scarce capacities: Solutions and fair cost allocations. *International Journal of Production Research*, 49(9):2643–2668, 2011.

M. A. Figliozzi, H. Mahmassani, and P. Jaillet. Framework for study of carrier strategies in auction-based transportation marketplace. *Transportation Research Record*, (1854):162 – 170, 2003.

M. A. Figliozzi, H. Mahmassani, and P. Jaillet. Competitive performance assessment of dynamic vehicle routing technologies using sequential auctions. *Transportation Research Record*, (1882):10 – 18, 2004.

M. A. Figliozzi, H. S. Mahmassani, and P. Jaillet. Pricing in dynamic vehicle routing problems. *Transportation Science*, 41(3): 302–318, 2007.

M. Frisk, M. Göthe-Lundgren, K. Jörnsten, and M. Rönnqvist. Cost allocation in collaborative forest transportation. *European Journal of Operational Research*, 205(2):448–458, 2010.

M. Furuhata, M. Dessouky, F. Ordóñez, M.-E. Brunet, X. Wang, and S. Koenig. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57:28–46, 2013.

M. Furuhata, K. Daniel, S. Koenig, F. Ordóñez, M. Dessouky, M. E. Brunet, L. Cohen, and X. Wang. Online cost-sharing mechanism design for demand-responsive transport systems. *IEEE Transactions on Intelligent Transportation Systems*, 16 (2):692–707, 2015.

G. Ghiani, E. Manni, and B. W. Thomas. A comparison of anticipatory algorithms for the dynamic and stochastic traveling salesman problem. *Transportation Science*, 46(3):374–387, 2012.

D. B. Gillies. *Some theorems on n-person games*. PhD thesis, 1953.

J. Green, E. Kohlberg, and J. J. Laffont. Partial equilibrium approach to the free-rider problem. *Journal of Public Economics*, 6(4):375–394, 1976.

A. Gupta, A. Srinivasan, and É. Tardos. Cost-sharing mechanisms for network design. *Algorithmica*, 50(1):98–119, 2008.

N. Immorlica, M. Mahdian, and V. S. Mirrokni. Limitations of cross-monotonic cost-sharing schemes. *ACM Transactions on Algorithms*, 4(2):1–25, 2008.

M. A. Krajewska, H. Kopfer, G. Laporte, S. Röpke, and G. Zaccour. Horizontal cooperation among freight carriers: Request allocation and profit sharing. *Journal of the Operational Research Society*, 59(11):1483–1491, 2008.

G. Laporte. Fifty years of vehicle routing. *Transportation Science*, 43(4):408–416, 2009.

G. D. Li, D. L. Du, D. C. Xu, and R. Y. Zhang. A cost-sharing method for the multi-level economic lot-sizing game. *Science China Information Sciences*, 57(1):1–9, 2014.

P. Liu, Y. Wu, and N. Xu. Allocating collaborative profit in less-than-truckload carrier alliance. *Journal of Service Science and Management*, 03(01):143–149, 2010.

Y. Liu and Y. Li. Pricing scheme design of ridesharing program in morning commute problem. *Transportation Research Part C: Emerging Technologies*, 79:156–177, 2017.

D. McFadden. Measuring willingness-to-pay for transportation improvements. 1998.

A. Mehta, T. Roughgarden, and M. Sundararajan. Beyond Moulin mechanisms. *Games and Economic Behavior*, 67(1):125–155, 2009.

B. Montreuil. Toward a physical internet: Meeting the global logistics sustainability grand challenge. *Logistics Research*, 3 (2-3):71–87, 2011.

H. Moulin. Incremental cost sharing: Characterization by group strategyproofness. *Social Choice and Welfare*, 16(2):279–320, 1999.

H. Moulin and S. Shenker. Strategyproof sharing of submodular costs: Budget balance versus efficiency. *Economic Theory*, 18 (3):511–533, 2001.

V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2013.

K. Roberts. The characterization of implementable choice rules. In *Aggregation and revelation of preferences*, volume 12, pages 321–349. 1979.

T. Roughgarden and M. Sundararajan. Quantifying inefficiency in cost-sharing mechanisms. *Journal of the ACM*, 56(4):1–33, 2009.

H. R. Sayarshad and H. O. Gao. A scalable non-myopic dynamic dial-a-ride and pricing problem for competitive on-demand mobility systems. *Transportation Research Part C: Emerging Technologies*, 91:192–208, 2018.

D. Schmeidler. The nucleolus of a characteristic function game. In *Game and economic theory: selected contributions in honor of Robert J. Aumann*, chapter The Nucleo, pages 231 – 238. 1995.

L. S. Shapley. A value for n-person games. *Annals of Mathematics Studies*, 28(2):307–317, 1953.

T. Skjoett-Larsen. European logistics beyond 2000. *International Journal of Physical Distribution & Logistics Management*, 30(5):377–387, 2000.

M. M. Solomon. Algorithms for the vehicle-routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265, 1987.

Y. Sprumont. Ordinal cost sharing. *Journal of Economic Theory*, 81(1):126–162, 1998.

S. H. Tijs and T. S. H. Driessen. Game theory and cost allocation problems. *Management Science*, 32(8):1015–1028, 1986.

Y.-T. Wang and D. Zhu. Ordinal proportional cost sharing. *Journal of Mathematical Economics*, 37(3):215–230, 2002.

H. Zou. *An Online Cost Allocation Model for Horizontal Supply Chains*. dissertation, University of Southern California, 2017.

H. Zou and M. M. Dessouky. A look-ahead partial routing framework for the stochastic and dynamic vehicle routing problem. *Journal on Vehicle Routing Algorithms*, 1:73–78, 2018.