Choice-Based Airline Schedule Design and Fleet Assignment: A Decomposition Approach

Chiwei Yan
Department of Industrial and Systems Engineering, University of Washington, Seattle WA, USA
Operations Research Center, Massachusetts Institute of Technology, Cambridge MA, USA

Cynthia Barnhart
Sloan School of Management and Operations Research Center
Massachusetts Institute of Technology, Cambridge MA, USA

Vikrant Vaze
Thayer School of Engineering, Dartmouth College, Hanover NH, USA

We study an integrated airline schedule design and fleet assignment model for constructing schedules by simultaneously selecting from a pool of optional flights and assigning fleet types to these scheduled flights. This is a crucial tactical decision which greatly influences airline profits. As passenger demand is often substitutable among available fare products (defined as a combination of an itinerary and a fare class) between the same origin-destination pair, we present an optimization approach that includes a passenger choice model for fare product selections. To tackle the formidable computational challenge of solving this large-scale network design problem, we propose a decomposition approach based on partitioning the flight network into smaller subnetworks by exploiting weak dependencies in network structure. The decomposition relies on a series of approximation analyses and a novel fare split problem to allocate optimally the fares of products that are shared by flights in different subnetworks. We present several reformulations that represent fleet assignment and schedule decisions and formally characterize their relative strengths. This gives rise to a new reformulation that is able to trade-off strength and size flexibly. We conduct detailed computational experiments using two realistically-sized airline instances to demonstrate the effectiveness of our approach. Under a simulated passenger booking environment with both perfect and imperfect forecasts, we show that the fleeting and scheduling decisions informed by our approach deliver significant and robust profit improvement over all benchmark implementations and previous models in the literature.

Key words: Airline, Network Design, Fleet Assignment, Schedule Design, Decomposition, Passenger Choice
History: First version, 01/2020; Revised on 02/2021, 11/2021, 02/2022.

---

1. Introduction

Airlines often operate on thin and volatile profit margins. Even before the ongoing COVID-19 crisis unfolded, Statista (2021) reported an industry average of only a 5.5% net profit margin in 2020 due to rising operating costs and an increasing level of competition. In the United States, numerous mergers have occurred during the past decade and half, suggesting an industry trend toward (1) consolidation and better utilization of capacity; and (2) reduction in operations and service frequencies in unprofitable markets. Indeed, the total number of departures in the US domestic markets reduced by around 15% from 9.8 million in 2006 to 8.5 million in 2019 (Bureau of Transportation Statistics 2021c), load factors surged upward from 78.8% to 83.8% during the same period (Bureau of Transportation Statistics 2021a), and average seating capacity increased substantially from 90 to 110 seats per flight (Bureau of Transportation Statistics 2021b). The recent COVID-19 crisis has also created a pressing need for airlines to redesign their existing schedules and reassign fleet capacity in response to significant changes in passenger demand (OAG 2020).

Airline schedule design and fleet assignment models are important types of network design problems that capture decisions related to flight frequencies, seating capacities and load factors. Airline schedule design (SD) concerns determining flight schedules, which includes tasks ranging from the tactical decisions of deciding the timings of individual flights within a narrow time window to the more strategic decisions about flight frequencies in markets that directly influence market shares. From a technical perspective, there are two key types of modeling approaches for schedule design: a direct timing approach decides individual flight departure and arrival times as continuous decision variables (e.g., Desaulniers et al. 1997), while an alternative approach selects flights from a pool of optional flights (e.g., Lohatepanont and Barnhart 2004). Our approach falls into the second category. From an application perspective, there are often two problem types: a complete timetabling problem which determines flight schedules from scratch; and an incremental schedule design problem where modifications are made to an existing schedule. While this paper mainly demonstrates the second problem, our approach is not restrictive and can be applied to the first problem as well.

The airline fleet assignment model (FAM) assigns a fleet type to each flight leg in the schedule in order to maximize profit contribution subject to various operational constraints. Profit contribution is calculated as the difference between expected revenue and operating cost. Airline operating cost is dependent on both fleet type characteristics and flight leg characteristics, and it includes crew costs, fuel costs, landing fees, and aircraft maintenance costs (Subramanian et al. 1994). Expected revenue is a complicated function of flight schedules (of the host airline as well as its competitors), flight seating capacities, passenger demand and the revenue management policy adopted by the airline. As schedule design and fleet assignment decisions are highly intertwined, they are often jointly considered in the planning process. We refer to this problem as the integrated schedule design and fleet assignment (SD-FAM) problem.

SD-FAM involves a modeling challenge including the *spill* and *recapture* behavior of passengers. In fleet assignment, when demand exceeds assigned capacity, spilled passengers (that is, excess demand) might be recaptured on other alternative fare products of the airline (a fare product is defined as a combination of an itinerary and a fare class), or might be lost to competitors. Similarly, in schedule design, when a certain itinerary is not available because a flight is removed from the schedule, demand increases for fare products on substitutable itineraries. Such spill and recapture behavior adds extra complexity to revenue modeling in SD-FAM.

This paper studies an integration of SD-FAM with choice-based demand. Passenger spill and recapture behaviors are modeled with a general attraction model (GAM) by Gallego et al. (2015), a model whose special case is the multinomial logit (MNL) choice model. In this approach, passengers choose to purchase available fare products based on their attractiveness values. The airline optimizes schedule design and fleet assignment decisions in reaction to such demand behaviors. The advantages of this are many. First, choice models can be estimated rigorously using the wealth of passenger transaction and shopping data (Vulcano et al. 2012, Abdallah and Vulcano 2021) that are now prevalent in the airline industry. Second, choice models systematically capture changes in passenger demand as fare product availability changes. Third, modeling and evaluating revenue using choice-based demand is consistent with airline practice in which choice-based network revenue management systems are increasingly being used in the industry (Vulcano et al. 2010, Dai et al. 2014). We denote the choice-based SD-FAM model by CSD-FAM.

The disadvantage, however, is that CSD-FAM is notoriously hard to solve (Wang et al. 2014, Di et al. 2016). Wang et al. (2014) attempted to solve CSD-FAM for a mid-sized South American airline. They reported significant difficulties and concluded that new methodologies and further computational analyses were necessary to solve such models successfully in practice. Specifically, they mentioned in their paper that "the significant increase in the number of nonzero coefficients in the constraint matrix makes even the root node relaxation much harder to solve than the one from the traditional FAM." Recently, Wei et al. (2020) aimed to solve a complete timetabling problem with choice-based demand using optional flights with one optional flight corresponding to each 15 minute time window. They developed heuristic rules to speed up the solution process. When solving such large-scale problems, their heuristics were able to demonstrate substantial gains with respect to schedule profitability and computational runtimes compared to the solutions obtained by commercial solvers. Yet, similar to prior works, they observed significant computational difficulties. First, their approach is not able to tractably handle multiple fare classes per itinerary. Moreover, their approach does not provide a performance guarantee with respect to the true optimum. In our paper, we provide new methodologies and analyses to partly address these gaps in the literature. Specifically, while demonstrating our approach in the context of the incremental, rather than the complete, schedule design problem with choice-based demand, we are able to provide tight bounds and provably near-optimal solutions in reasonable runtime budgets.

The major contribution of this work is to enhance the computational performance of CSD-FAM through effective *network decomposition*. From a solution methodology standpoint, our approach is related to an earlier work of Barnhart et al. (2009) who developed decomposition methods for pure fleet assignment problem with a particular demand assumption — revenue derived from a set of fare products does not depend on the capacities of flights that are not used by these products. The approach partitions the flight network into smaller subnetworks. Profit contribution is evaluated separately for each subnetwork. Fleeting and scheduling decisions within each subnetwork can then be segregated from the others and represented using composite variables. This results in a more tractable formulation whose profit function is an upper bound on the original one. The authors demonstrated the approach's superior computational performance.

However, the aforementioned assumption made in Barnhart et al. (2009) violates the choice-based nature of passenger demand. For example, demand for a particular fare product going from Boston to Seattle with a connection at Chicago can depend on the capacity of a direct flight from Boston to Seattle. Especially under schedule design, this assumption can be highly inaccurate — whether the direct flight from Boston to Seattle exists or not significantly impacts the demand on the Boston-Chicago-Seattle itinerary and fare classes therein. Capturing this dependence has important consequences from a practical viewpoint. In fact, as we will show later in the computational experiments, the performance of the approach by Barnhart et al. (2009) can greatly deteriorate under choice-based demand.

Applying network decomposition under a choice-based demand is nontrivial. Under the demand assumption in Barnhart et al. (2009), network dependency exists only when a fare product uses resources on multiple flight legs, e.g., fare products of a connecting itinerary. However, when passenger spill and recapture behaviors are modeled, they exhibit much stronger, and new types of, network dependencies between flights. Flights can be interdependent not only because they provide capacity for the same fare product, but also because the products for which they provide capacity are jointly considered (as substitutes) by passengers in an origin-destination market. An effective and efficient decomposition requires significant new developments, which we summarize as follows.

- *New decomposition algorithm and approximation analyses*: We construct a flight-product-market dependency graph, rather than Barnhart et al. (2009)'s flight-product graph, depicting strengthened network dependencies under choice-based demand. This necessitates conducting new approximation analyses, which we perform, to drive a new decomposition algorithm for creating flight subnetworks.
- *Optimization-based fare allocation*: To tighten the profit upper bound, we develop an original fare split linear program that optimally allocates connecting itinerary product fares over flights and across different subnetworks and effectively measures approximation quality of a given flight partition. This new method of creating fare proration is shown to be critical to the success

of our proposed decomposition framework — as we will see later, a widely used distance-based proration scheme performs quite poorly in comparison.

- *New subnetwork-based reformulation with partial representation*: Choice-based demand strengthens network dependencies across flights causing significantly looser profit upper bound. As a result, representing all fleet assignment and schedule design decisions within each subnetwork using composite variables, as proposed in Barnhart et al. (2009), becomes impractical. As we will show later in the computational experiments, it leads to either an enormously large formulation for good enough approximation quality, or poor approximation quality for a tractable formulation size. This motivates us to propose a new reformulation representing a *subset* of the subnetworks using composite variables. This reformulation allows us to flexibly trade-off formulation strength and size. In addition, we formally characterize the strengths of several variations of the subnetwork-based reformulations (including the one we propose).
- *Extensive numerical experiments*: Finally, we conduct comprehensive numerical experiments using two realistically-sized airline instances to demonstrate the effectiveness of our approach. Under a simulated passenger booking environment with both perfect and imperfect forecasts, we show that the fleeting and scheduling decisions produced by our proposed approach deliver significant and robust profit improvement compared to benchmark implementations and models in the existing literature.

The remainder of the paper is organized as follows. In Section 2 we discuss relevant literature. In Section 3, we introduce the basic formulation of CSD-FAM. In Section 4, we present a reformulation based on decomposition of the flight network and conduct approximation analyses. In Section 5, we discuss the solution approach for our formulation. In Section 6, we describe computational experiments using two full-scale instances from two major airlines to demonstrate the superior performance of our proposed approach. We conclude in Section 7. All proofs, auxiliary technical results and additional computational results are relegated to the appendix. We also provide a modeling and algorithmic extension in Appendix D in which passenger fare product choice depends on not only the product availability but also the fleet type(s) assigned to the flight(s) used by the fare product.

2. Related Literature

In this section, we discuss literature related to our work. Related literature is primarily concentrated in the general area of airline fleet assignment and schedule design, and various attempts to incorporate passenger spill and recapture behaviors into the network planning models.

Fleet Assignment and Schedule Design. Hane et al. (1995) proposed a basic fleet assignment model (BFAM) using a multi-commodity flow formulation on a time-space network. BFAM lays the foundation for later development of FAM. However, it assumes that revenue of the entire network can be perfectly prorated over individual flight legs. This assumption is problematic because airlines sell fare products that often contain multiple flight legs. Each fare product is characterized by a given itinerary and a given fare class (e.g., a Y-class ticket from Seattle (SEA) to Boston (BOS) through Dallas/Fort Worth (DFW) on given flight legs) and an itinerary can use capacities on multiple flight legs creating network dependencies.

These network dependencies motivated Barnhart et al. (2002) to propose an itinerary-based fleet assignment model (IFAM) which explicitly models itinerary demand and its network effects using a passenger mix model (PMM) to evaluate revenue. PMM is similar to, but a simplification of, the network revenue management (NRM) problem in which airlines optimally control passenger flows to maximize revenue for given flight seating capacities. In PMM, fare products are typically aggregated into itineraries, itinerary demand is assumed to be deterministic, and passengers are assigned to itineraries. On the other hand, NRM is a stochastic control problem where demand is random and controlled by dynamically accepting or rejecting booking requests for fare products (usually implemented by adjusting the fare product offer set over time). Jacobs et al. (2008) develop an approach which integrates BFAM with NRM. The approach relies on a decomposition similar to Benders' decomposition, where the NRM is solved iteratively as a subproblem using stochastic optimization approaches.

As we mentioned in the introduction, within the FAM literature, Barnhart et al. (2009) is the closest work to ours. They built a decomposition framework under a more restricted demand assumption and demonstrated its superior computational performance over other approaches which existed at that time. They assumed that the demand for a set of fare products does not depend on the capacities of flights not used by these products. This violates the spill and recapture behaviors observed in practice. In our paper, we develop a new decomposition framework for choice-based demand. This extension requires substantial new mathematical and algorithmic developments, enables improved computational performance of CSD-FAM, and achieves significant and robust profit improvements over all known benchmarks, including Barnhart et al. (2009).

Aside from FAM, a number of previous studies have also focused on the airline fleet assignment model integrated with schedule design (SD-FAM) given its potential for increasing profitability by simultaneously optimizing flight scheduling and fleet assignment decisions. Desaulniers et al. (1997) studied an integrated model allowing flight departure times to be altered within a time window and proposed a solution methodology based on branch-and-price. Lohatepanont and Barnhart (2004) presented a model that selects flights to be operated from a base schedule containing some mandatory and some optional flights, which is the setting of our paper as well. Their model is an extension of IFAM (Barnhart et al. 2002). Sherali et al. (2010) conducted polyhedral analyses of an SD-FAM with optional flight selection, and proposed several techniques to enhance its computational performance.

Incorporating Passenger Spill and Recapture. In fleet assignment, Barnhart et al. (2002) used spill variables and recapture rates defined for pairs of substitutable itineraries to model the substitution behavior when demand exceeds capacity. The recapture rate was estimated using the Quality of Service Index (QSI). In schedule design, Lohatepanont and Barnhart (2004) additionally introduced demand correction terms as demand adjustments for changes made to the schedule.

A more recent advancement in modeling passenger spill and recapture behavior has occurred in choice modeling and its application to (airline) revenue management. Discrete choice models capture such behavior by recognizing that demand is tied to markets and not to individual fare products. Passengers then choose fare products belonging to that market according to probabilities that depend on the set of fare products being offered. Talluri and van Ryzin (2004) utilized discrete choice models to describe passenger fare product substitution behavior and provided structural results related to the single-leg revenue management problem. Later, Gallego et al. (2004), Liu and van Ryzin (2008) and Bront et al. (2009) extended these results to an NRM setting, using a choice-based deterministic linear programming (CDLP) model to produce both an upper bound on the optimal revenue value and an effective heuristic for the stochastic control problem. Gallego et al. (2015) further show that the CDLP formulation can be reformulated as a compact linear program, referred to as the sales-based linear program (SBLP).

These advances were soon incorporated into airline network planning models. Wang et al. (2014) and Di et al. (2016) both focus on embedding SBLP as the revenue evaluator into fleet assignment models. The CSD-FAM formulation in our paper is based on their work. However, both these studies focus on modeling, and not on the solution algorithms for solving these models. In contrast, we focus on decomposition-based solution strategies to overcome the substantial computational difficulties reported in their studies. We also focus on the integrated schedule design and fleet assignment problem while these two previous studies mainly focus on fleet assignment only. The incorporation of schedule design decisions significantly increases the complexity of the problem.

More recently, Wei et al. (2020) studied an integrated timetabling and fleet assignment model with passenger choice using similar SBLP models. Their model focuses on a complete timetable development from a large set of optional flights. To maintain tractability, heuristics — such as imposing symmetry in fleet assignment, fare products aggregation for each itinerary, using a multi-phase method to narrow the optional flight selection time windows — have been proposed and implemented successfully by Wei et al. (2020) to make the solution process much more tractable. Our approach, although demonstrated in the context of incremental schedule design, explicitly considers substitution among different fare products and focuses on exploiting network dependencies to strike a balance between solution quality and computation time. Moreover, as we will see later in the computational experiments, the heuristic proposed in Wei et al. (2020) is often not effective in our setting and sometimes results in highly sub-optimal solutions. A common theme observed from previous studies is that, although the incorporation of passenger choice could potentially bring significant profit improvement, the associated computational challenges are formidable enough to prevent application to full-scale instances of large carriers. We develop an original solution approach and provide general insights on an effective solution process for such problems, one that potentially can be generalized to other types of network design problems where decomposition is beneficial.

Aside from the SD-FAM and revenue management literature, there are also some other studies which specifically focus on passenger flow models for airline networks (Soumis and Nagurney 1993, Dumas and Soumis 2008). These studies aim to assign passengers to itineraries, given seating capacities, using equilibrium models. These methods consider stochastic demand and also passenger spill-and-recapture behavior by assuming certain recapture rates. However, these approaches typically do not consider the impact of NRM on passenger flows.

3. Choice-Based Schedule Design and Fleet Assignment

In this section, we present the nominal model of CSD-FAM with choice-based demand, which was first introduced by Wang et al. (2014). The formulation is based on a multi-commodity flow problem defined on a set of time-space networks with each commodity/network corresponding to a specific type of fleet (Hane et al. 1995). For each fleet type, it has a separate time-space network consisting of all compatible flights that can be flown by this type of fleet. Flows in each network represent the assignment of a particular fleet type to flights and are coupled by flight cover constraints.

Let $\mathcal{L}$ denote the set of all flight legs, among which $\mathcal{L}_m$ is the set of mandatory flights that *must* be flown in the schedule and $\mathcal{L}_o$ is the set of optional flights which *can* be flown. Each flight in the set $\mathcal{L}_o$ could be either flown or not included in the schedule for profitability or operational reasons. Throughout the paper, we will use the terms "flight" and "flight leg" interchangeably. $\mathcal{F}$ is the set of all fleet types. $\mathcal{F}(l)$ is the set of compatible fleet types that can be used to fly flight leg l. Each fleet type $f \in \mathcal{F}$ has its own time-space network of all flights that are compatible with it. Figure 1 illustrates part of one such time-space network with two airports: LaGuardia Airport (LGA) in New York City and Boston Logan International Airport (BOS). The horizontal axis represents the progress of time from left to right. The vertical axis represents different airports, with one horizontal line representing each distinct airport.

Let N_f be the set of nodes in the time-space network of fleet type f. Each node $v \in N_f$ corresponds to a specific event: either an arrival or a departure of a flight. Note that the time of an arrival node is the scheduled arrival time plus the turnaround time of fleet type f. This network includes only the flights that are compatible with fleet type f. There are three types of directed arcs in the network: (1) mandatory flight arcs, represented by solid arrows linking a departure node and an arrival node in two different airports; (2) optional flight arcs, represented by dash-dotted arrows linking a departure node and an arrival node in two different airports; and (3) ground arcs, represented by dashed arrows linking two nodes, consecutive in time, at the same airport. Set of ground arcs also includes the wrap-around ground arcs connecting the last node to the first node at a specific airport to ensure that the fleet assignment of fleet type f can be operated in a repeated schedule.

Let v^- denote the predecessor node located at the tail of the ground arc entering node v. Denote by $I(v)$ the set of incoming flight arcs into node v and by $O(v)$ the set of outgoing flight arcs from node v. Note that $|I(v)| + |O(v)| = 1$, i.e., there is either one incoming flight arc or one outgoing flight arc for each node, depending on whether it is an arrival or a departure node. We define a count time (dashed vertical line in Figure 1) to count the total number of aircraft both in the air and on the ground at that time. This time epoch can be chosen arbitrarily because we ensure that there is conservation of flow throughout the time-space network. Let $T_F(f)$ denote the set of flight arcs (including both mandatory and optional flight arcs) in the time-space network of fleet type f that cross the count time. According to the construction of the time-space network, for each node $v \in N_f$ there is only one incoming ground arc as well as only one outgoing ground arc. Thus, any ground arc can be uniquely identified by the node it originates from. Let $T_N(f)$ denote the set of all nodes in the time-space network of fleet type f such that the ground arcs originating from them cross the count time. Let n_f be the number of available aircraft for fleet type f. We define binary decision variables $x_{l,f} = 1, l \in \mathcal{L}, f \in \mathcal{F}$ if fleet type f is assigned to flight leg l and zero otherwise. Observe that, setting $x_{l,f} = 0, \forall f \in \mathcal{F}$ for an optional flight leg l enforces the removal of the optional flight from the schedule. In other words, the $x_{l,f}$ variables represent both fleet assignment and schedule design decisions. We use $y_v \in \mathbb{Z}_{\geq 0}, v \in N_f, f \in \mathcal{F}$ to represent the number of aircraft on the ground arc originating from each node v.

Let $\mathcal{P}$ denote the set of fare products the airline offers. Each fare product is a unique combination of an itinerary and a fare class. h_p is the price of fare product p and $\mathbf{h} = \{h_p\}_{p \in \mathcal{P}}$ is the vector of prices of all fare products. With this notation, we now present the formulation of CSD-FAM.

$$\text{(CSD-FAM)} \quad V_{\text{CSD-FAM}} = \max \qquad r(\mathbf{x}; \mathbf{h}) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \tag{1}$$

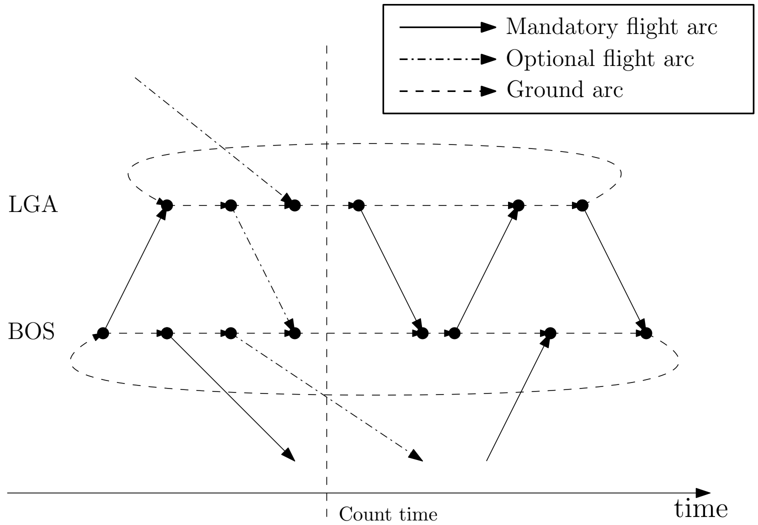

Figure 1 **A simple example of a part of a time-space network with two airports for a specific fleet type.**

$$\text{s.t.} \quad \sum_{f \in \mathcal{F}(l)} x_{l,f} = 1, \quad \forall l \in \mathcal{L}_m, \tag{2}$$

$$\sum_{f \in \mathcal{F}(l)} x_{l,f} \leq 1, \quad \forall l \in \mathcal{L}_o, \tag{3}$$

$$y_{v^-} + \sum_{l \in I(v)} x_{l,f} - y_v - \sum_{l \in O(v)} x_{l,f} = 0, \quad \forall f \in \mathcal{F},\ \forall v \in N_f, \tag{4}$$

$$\sum_{v \in T_N(f)} y_v + \sum_{l \in T_F(f)} x_{l,f} \leq n_f, \quad \forall f \in \mathcal{F}, \tag{5}$$

$$y_v \geq 0, \quad \forall f \in \mathcal{F},\ \forall v \in N_f, \tag{6}$$

$$x_{l,f} \in \{0,1\}, \quad \forall f \in \mathcal{F}(l),\ \forall l \in \mathcal{L}. \tag{7}$$

Let $V_{\textsf{CSD-FAM}}$ denote the optimal value of this optimization problem. We also define $\tilde{V}_{\textsf{CSD-FAM}}$ as the optimal value of the continuous relaxation of CSD-FAM where constraints (7) are replaced with the continuous unit interval as $x_{l,f} \in [0,1]$. The $\tilde{V}$ notation will be used throughout the paper to represent optimal values of the continuous relaxations of the formulations that we introduce. Constraints (2) ensure that each mandatory flight is covered by exactly one compatible fleet type. In contrast, constraints (3) state that optional flights can either remain uncovered, that is, be removed from the schedule, or be covered by one compatible fleet type. Constraints (4) are the flow balance constraints that ensure that the number of aircraft of a particular fleet type entering a node equals the number leaving. Constraints (5) limit the number of aircraft of each fleet type f (either in the air or on ground) to be less than or equal to the available number of aircraft of that fleet type, denoted as n_f. The integrality of y_v is relaxed here because it is implicitly satisfied through the inclusion of constraints (4), (5) and the integrality of $x_{l,f}$ and n_f. More specifically, for any integral values of $\mathbf{x}$, the coefficient matrix corresponding to $\mathbf{y}$ is totally unimodular. Note that another approach to model flight removal is to assign to such optional flights a null fleet type with zero capacity, zero operating cost and infinite availability, and then relax the model such that the null fleet type does not have to satisfy the flow balance constraints (4). Finally, the objective (1) is to maximize the total profit contribution which is the revenue from selling fare products minus the total operating cost, where $c_{l,f}$ is the operating cost associated with assigning fleet type f to flight l. The revenue function $r(\mathbf{x};\mathbf{h})$ depends on the fleet assignment and schedule design decisions $\mathbf{x}$, and fare vector $\mathbf{h}$.

We now discuss in detail the specification of the revenue function $r(\mathbf{x};\mathbf{h})$. We partition passengers into a set of markets $\mathcal{M}$. A market may be a specific origin-destination market, for example, all fare products from Boston to New York City. A market may also be more restrictive, for example, morning direct flights from New York City to San Francisco that have fares less than \$600. Let Λ_m be the total demand for market m. Passengers in a particular market $m \in \mathcal{M}$ are *only* interested in a specific subset of fare products $\mathcal{P}(m) \subset \mathcal{P}$ offered by the airline. We call $\mathcal{P}(m)$ the consideration set of market m. We assume that passengers purchase fare products according to a general attraction model (GAM) (Gallego et al. 2015). For each product p in consideration set $\mathcal{P}(m), m \in \mathcal{M}$, let w_p and $\tilde{w}_p \in [0, w_p]$ denote the measure of its attractiveness and shadow attractiveness respectively. Intuitively, shadow attractiveness $\tilde{w}_p$ represents the attractiveness of some latent alternative of product p when it is not available. For each market m, we also define w_m as the attractiveness of the outside option, that is, the sum of the attractiveness values of competing airlines' products as well as the no-fly alternative in the same market. If $\bar{\mathcal{P}}$ is the set of available products, then a passenger from market m will choose fare product p with probability $w_p/(\sum_{p' \in \bar{\mathcal{P}} \cap \mathcal{P}(m)} w_{p'} + \sum_{p' \in \mathcal{P}(m): p' \notin \bar{\mathcal{P}}} \tilde{w}_{p'} + w_m)$ if $p \in \bar{\mathcal{P}} \cap \mathcal{P}(m)$ and zero otherwise. It is not hard to see that when $\tilde{w}_p = 0, \forall p \in \mathcal{P}(m)$, the model reduces to the MNL model. On the other hand, when $\tilde{w}_p = w_p, \forall p \in \mathcal{P}(m), \forall m \in \mathcal{M}$, it reduces to the independent[1] demand model where each fare product has its own separate demand. Thus, GAM represents a spectrum of choice models between independent demand model and MNL. One potential advantage of using GAM (over MNL) is that it bypasses the independence of irrelevant alternatives (IIA) limitation or the famous red bus/blue bus paradox. This could be relevant in the context of schedule design and fleet assignment since different fare products might share some similar attributes. Indicator variable $\delta_{l,p} = 1$ if fare product p uses flight leg l and zero otherwise. We then define continuous decision variables s_p, $p \in \mathcal{P}$ as the

[1] We call such demand *independent* not following exactly the notion in probability theory as demand in our model is essentially deterministic. However, it does share some characteristics of independence in the sense that demand for one fare product does not depend on those of other products. This notion of independent demand also follows the standard terminology used in the revenue management literature (see e.g., Gallego and Topaloglu 2019).

tickets sale volume, that is, number of tickets sold, for product p; s_m, $m \in \mathcal{M}$ as the total tickets sale volume for the outside option in market m. The revenue function can then be defined as the optimal value of the following linear program.

$$
\begin{aligned}
(\mathsf{SBLP}) \quad r(\mathbf{x};\mathbf{h}) = \max_{\mathbf{s}} \quad & \sum_{p\in\mathcal{P}} h_p s_p && (8)\\
\text{s.t.} \quad & \sum_{p\in\mathcal{P}} \delta_{l,p} s_p \le \sum_{f\in\mathcal{F}(l)} \mathrm{CAP}_f x_{l,f},\ \forall l \in \mathcal{L}, && (9)\\
& \sum_{p\in\mathcal{P}(m)} \frac{w_p - \tilde{w}_p}{w_p} s_p + \frac{w_m + \sum_{p\in\mathcal{P}(m)} \tilde{w}_p}{w_m} s_m = \Lambda_m,\ \forall m \in \mathcal{M}, && (10)\\
& \frac{s_p}{w_p} - \frac{s_m}{w_m} \le 0,\ \forall m \in \mathcal{M},\ \forall p \in \mathcal{P}(m), && (11)\\
& s_m, s_p \ge 0, \quad \forall m \in \mathcal{M},\ \forall p \in \mathcal{P}. && (12)
\end{aligned}
$$

This linear program is the sales-based linear program (SBLP) first developed by Gallego et al. (2015). It is a fluid approximation of the stochastic dynamic programming formulation of the NRM problem with GAM demand. Its optimal value serves as a fairly tight upper bound on the optimal value of the exact stochastic control problem in practice (see a recent study by Kunnumkal and Talluri (2019) showing a gap of less than 5% using the data from a major carrier under MNL demand). Constraints (9) are the capacity constraints that limit the aggregated sales on a flight leg to its assigned seating capacity, where CAP_f is the seating capacity of fleet type f. Constraints (10) are the balance constraints requiring that when some fare products are closed for sale, the demands and sales for the remaining open fare products change accordingly. Finally, constraints (11) are the scaling constraints which ensure that product sales depend on the availability of each fare product in a manner that is consistent with their attractiveness values. It requires that the sale of a product $p \in \mathcal{P}(m)$ is upper bounded by $(w_p/w_m)s_m$. The equality is achieved when product p is always available to book during the selling horizon. If product p is ever closed for sale during the selling horizon — due to a seating capacity constraint or revenue management policy — a strict inequality $s_p < (w_p/w_m)s_m$ may result. Note that the outside option (that is, the aggregation of the competing airlines' products and the no-fly alternative) is assumed to be always available.

We now make two remarks regarding the formulation we use.

Consideration Set $\mathcal{P}(m)$. Note that one implicit assumption ensuring the validity of (8) - (12) is that all consideration sets $\mathcal{P}(m), m \in \mathcal{M}$ are mutually exclusive. This assumption is commonly used in schedule design and fleet assignment literature and practice where all products from the same origin-destination pair form a market segment. If the airline can manage different markets independently, this assumption can be readily relaxed to capture overlapping markets without changing the structure of the formulation by simply indexing sales variables with both market and product, $s_{p,m}$ (see Section 5.3 in Gallego et al. 2015). On the other hand, if a fare product cannot be offered in one market without offering it in other markets, Talluri (2014) shows that SBLP still provides a valid (but looser compared to CDLP) upper bound on the optimal value of the stochastic seat inventory control problem, and can be effectively tightened with additional cuts. Importantly, *none* of the technical results developed in our paper (results in Section 4 and Section 5) rely on this assumption. For ease of presentation and to be consistent with the schedule design and fleet assignment literature and practice, we presented the model with this assumption in place.

Choice Model. Although our models and upcoming solution approaches are developed based on the assumption that passenger demand is governed by a general attraction model (GAM), they can be extended to other choice models especially where a compact deterministic linear programming formulation for network revenue management (NRM) exists. For example, the Markov chain choice model (Blanchet et al. 2016) (which subsumes GAM) also has a compact deterministic sales-based linear programming formulation for NRM. These advanced choice models often overcome structural limitations and sometimes enjoy greater predictive power. In Appendix D, we also discuss a modeling extension with fleet-type dependent choice behaviors and show that they can be naturally incorporated into our proposed framework. This captures a setting where differences in attributes such as seating capacities, comfort levels, amenities, and meal options associated with different fleet types affect the choice of travelers (Clark 2017). This could become an important consideration in modeling travelers' choice in a post COVID-19 world.

4. A Decomposition-Based Reformulation

Although the CSD-FAM formulation is compact, it still poses a formidable computational challenge. Wang et al. (2014) and Di et al. (2016) both report substantial difficulties in solving it. This challenge is further amplified with the incorporation of schedule design decisions that additionally require the consideration of optional flight legs. In this section, we explore decomposition opportunities to improve the computational performance of the model.

4.1. Subnetwork Decomposition

We first define a *dependency graph* $\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M})$ which is constructed with flight set $\mathcal{L}$, product set $\mathcal{P}$ and market set $\mathcal{M}$. We illustrate one example in Figure 2. Nodes denoted by circles correspond to flights, squares correspond to fare products and hexagons correspond to markets. An edge is established between a flight node and a product node if the product uses capacity of the flight. Similarly, an edge is established between a product node and a market node if the product belongs to the market. Figure 2 shows a toy flight network with five flights, six fare products and four markets. Note that, in comparison to the flight-market dependency graph in Barnhart et al. (2009), the addition of market set in our dependency graph is a key distinction, enabling us to capture the stronger, and new types of, network dependencies induced by passenger spill and recapture behavior, as captured by the choice-based demand.

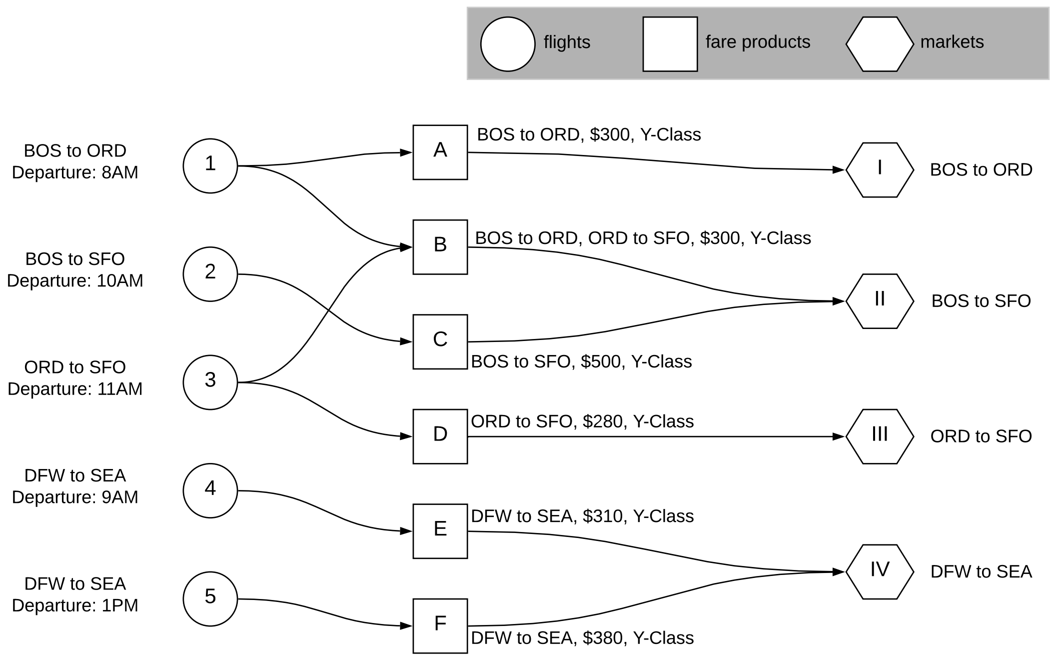

Figure 2 **An example of flight, fare product and market dependency graph.**

We now illustrate how to exploit independent structures inside such flight-product-market dependency graphs. For example, there are two separable components in the network in Figure 2 which correspond to two flight subnetworks — $\{1, 2, 3\}$ and $\{4, 5\}$ — such that any fleeting and scheduling decisions within one subnetwork will not impact the revenue of the other, and vice-versa. We formalize this intuition here. We start by partitioning the set of flight legs $\mathcal{L}$ into K subsets (called subnetworks): $\mathcal{L}^1, \cdots, \mathcal{L}^K$ such that $\cup_{k=1}^K \mathcal{L}^k = \mathcal{L}$ and $\mathcal{L}^i \cap \mathcal{L}^j = \emptyset$, $1 \le i < j \le K$. $|\mathcal{L}^k|$ denotes number of flights in subnetwork k and is referred to as the size of subnetwork k. We denote such a partition by $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$. For each $\mathcal{L}^k$, we define local product set $\mathcal{P}^k = \{p \in \mathcal{P} : \exists l \in \mathcal{L}^k \text{ s.t. } \delta_{l,p} = 1\}$, as the set of products such that each product uses at least one flight leg in $\mathcal{L}^k$. Similarly, we define local market set $\mathcal{M}^k = \{m \in \mathcal{M} : \mathcal{P}^k \cap \mathcal{P}(m) \neq \emptyset\}$, as the set of markets such that at least one product in $\mathcal{P}^k$ belongs to that market. Using the toy network in Figure 2 as an example, if we partition the flights into $\mathcal{L}^1 = \{1, 2, 3\}, \mathcal{L}^2 = \{4, 5\}$, then local product sets are $\mathcal{P}^1 = \{A, B, C, D\}, \mathcal{P}^2 = \{E, F\}$ and local market sets are $\mathcal{M}^1 = \{\mathrm{I}, \mathrm{II}, \mathrm{III}\}, \mathcal{M}^2 = \{\mathrm{IV}\}$. It is worth noting that, in this case, $\mathcal{P}^1 \cap \mathcal{P}^2 = \emptyset$ and $\mathcal{M}^1 \cap \mathcal{M}^2 = \emptyset$, which does not necessarily always hold for any given partition. We have the following result on the revenue calculations of subnetworks when such conditions hold.

Proposition 1. *Suppose* $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ *is such that its corresponding local product sets* $\{\mathcal{P}^1, \cdots, \mathcal{P}^K\}$ *and local market sets* $\{\mathcal{M}^1, \cdots, \mathcal{M}^K\}$ *satisfy* $\mathcal{P}^i \cap \mathcal{P}^j = \emptyset,\ \forall i \neq j$ *and* $\mathcal{M}^i \cap \mathcal{M}^j = \emptyset,\ \forall i \neq j$*. Define* $\mathbf{h}(\mathcal{P}^k) \in \mathbb{R}^{|\mathcal{P}|}_{\geq 0}$ *as the projection of a fare vector* $\mathbf{h}$ *onto a local product set* $\mathcal{P}^k$*, i.e.,* $[\mathbf{h}(\mathcal{P}^k)]_p = h_p,\ \forall p \in \mathcal{P}^k$ *and* $[\mathbf{h}(\mathcal{P}^k)]_p = 0,\ \forall p \notin \mathcal{P}^k$*. We have*

$$r\left(\mathbf{x}; \mathbf{h}(\mathcal{P}^k)\right) = r\left(\mathbf{x}'; \mathbf{h}(\mathcal{P}^k)\right),\ \forall \mathbf{x}, \mathbf{x}' \text{ such that } x'_{l,f} = x_{l,f},\ \forall l \in \mathcal{L}^k,\ \forall f \in \mathcal{F}(l);$$

and moreover,

$$r(\mathbf{x}; \mathbf{h}) = \sum_{k=1}^{K} r\left(\mathbf{x}; \mathbf{h}(\mathcal{P}^k)\right), \forall \mathbf{x}.$$

We call such a flight partition with mutually exclusive local product sets and local market sets an *independent partition*. The proposition states that the revenue of a subnetwork $\mathcal{L}^k$ within an independent partition does not depend on the capacity of flight legs $l \notin \mathcal{L}^k$; also, the total revenue of the network can be exactly represented as the sum of the revenues of individual subnetworks calculated using projections of the overall fare vector. Such independence motivates a decomposition approach where fleet assignment and schedule design is performed separately for each subnetwork, and only coupled by the flow balance and fleet availability constraints (4) and (5) across the entire network.

4.2. Subnetwork-based Reformulations

We now exploit Proposition 1 to develop several reformulations based on subnetwork decomposition. We first define *local SD-FAM decision* $\mathbf{x}^k$ and *local fare vector* $\mathbf{h}^k \in \mathbb{R}^{|\mathcal{P}|}_{\geq 0}$ for each subnetwork $\mathcal{L}^k$ as follows. Note that throughout the paper, we do *not* restrict to independent partitions when we refer to a flight partition Π and subnetworks $\mathcal{L}^k$.

Definition 1 (Local SD-FAM Decisions). Given a SD-FAM decision $\mathbf{x}$, the local SD-FAM decision $\mathbf{x}^k$ for subnetwork $\mathcal{L}^k$ is defined as:

$$x^k_{l,f} = \begin{cases} x_{l,f}, & \forall l \in \mathcal{L}^k, \forall f \in \mathcal{F}(l), \\ 1, & \forall l \in \mathcal{L} \setminus \mathcal{L}^k,\ f = \arg\max_{f' \in \mathcal{F}(l)} \mathrm{CAP}_{f'}, \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

In words, $\mathbf{x}^k$ is the SD-FAM decision $\mathbf{x}$ for flights in $\mathcal{L}^k$, and for each flight leg not in $\mathcal{L}^k$, the assigned fleet type is the one with the largest possible seating capacity among the fleet types that are compatible for with flight leg. Similarly, we define local fare vectors for each subnetwork.

Definition 2 (Local Fare Vectors). For any partition $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^k\}$, we define a set of local fare vectors $\mathbf{h}^k \in \mathbb{R}^{|\mathcal{P}|}_{\geq 0}, k \in \{1, \cdots, K\}$ as any set of $|\mathcal{P}|$-dimensional vectors that satisfies the following conditions:

$$\mathbf{h}^1 + \cdots + \mathbf{h}^K = \mathbf{h}, \tag{14}$$

$$h_p^k = 0 \text{ if } p \notin \mathcal{P}^k,\ \forall k \in \{1, \cdots, K\}. \tag{15}$$

Let $\mathcal{H} = \{\mathbf{h}^k\}_{k \in \{1,\cdots,K\}}$ denote any set of local fare vectors for partition Π. In words, local fare vectors are non-negative vectors that sum to the original fare vector $\mathbf{h}$, with zeros for products that do not belong to the local product set. This definition allows for multiple or even infinite sets of valid local fare vectors for a given fare vector $\mathbf{h}$ and a given partition Π. We will address the selection of $\mathbf{h}^1, \cdots, \mathbf{h}^K$ later in the paper.

Having $\mathbf{x}^k$ and $\mathbf{h}^k$ defined for all $k \in \{1, \cdots, K\}$, we now introduce formulation CSD-FAM(Π, $\mathcal{H}$), as a function of a partition $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^k\}$ and its corresponding set of local fare vectors $\mathcal{H}$, as below. This is a slight variant of the formulation that was first presented in Barnhart et al. (2009).

$$\begin{aligned}
&(\textsf{CSD-FAM}(\Pi,\mathcal{H})) \\
V_{\textsf{CSD-FAM}(\Pi,\mathcal{H})} = \max \quad & \sum_{k=1}^{K} r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \qquad (16) \\
\text{s.t.} \quad & (2) - (7), \\
& x_{l,f}^k \text{ satisfies } (13), \quad \forall l \in \mathcal{L},\ \forall f \in \mathcal{F}(l),\ \forall k \in \{1, \cdots, K\}.
\end{aligned}$$

Note that the product attractiveness values w_p and $\tilde{w}_p$ continue to be functions of the fare h_p and not functions of the local fare h_p^k, irrespective of whether they are used in the full revenue function formulation $r(\mathbf{x}; \mathbf{h})$, or in the subnetwork revenue function formulation $r(\mathbf{x}^k; \mathbf{h}^k)$. Since the fare h_p stays the same, the product attractiveness values w_p and $\tilde{w}_p$ also stay the same throughout the analysis. We define $V_{\textsf{CSD-FAM}(\Pi,\mathcal{H})}$ and $\tilde{V}_{\textsf{CSD-FAM}(\Pi,\mathcal{H})}$ as the optimal values of CSD-FAM($\Pi, \mathcal{H}$) and of its continuous relaxation, respectively, for a given partition Π and a given set of local fare vectors $\mathcal{H}$. One can verify that CSD-FAM($\Pi, \mathcal{H}$) is an equivalent formulation to CSD-FAM, for any valid set of local fare vectors $\mathcal{H}$, if Π is an independent partition. We formalize this equivalence in the following corollary.

Corollary 1. *Let $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ be an independent partition, then* CSD-FAM($\Pi, \mathcal{H}$) *is equivalent to* CSD-FAM *for any set of local fare vectors $\mathcal{H}$ satisfying* (14) *and* (15)*, in that (1) their optimal objective values are the same, i.e., $V_{\textsf{CSD-FAM}} = V_{\textsf{CSD-FAM}(\Pi,\mathcal{H})}$; and (2) their resultant sets of optimal SD-FAM decisions are also the same.*

Note that the only difference between CSD-FAM and CSD-FAM$(\Pi, \mathcal{H})$ is in the objective functions (1) and (16) (and the additional definitions (13) of $x_{l,f}^k$ variables needed to formulate the latter's objective function (16)). CSD-FAM$(\Pi, \mathcal{H})$ decomposes the revenue calculation by using a summation of revenue obtained from each subnetwork. This decomposition allows the calculation of local revenue function $r(\mathbf{x}^k; \mathbf{h}^k)$ to depend on the SD-FAM decisions only within that subnetwork $\mathcal{L}^k$.

This observation motivates a further reformulation of CSD-FAM$(\Pi, \mathcal{H})$. For a subnetwork $\mathcal{L}^k$, we define $\mathcal{X}^k$ as the set of all possible combinations of local SD-FAM decisions for flights in $\mathcal{L}^k$. To illustrate this point, suppose $\mathcal{L}^k = \{l_1, l_2\}$ and $\mathcal{F}(l_1) = \mathcal{F}(l_2) = \{f_1, f_2\}$. Moreover, $l_1 \in \mathcal{L}_o$ and $l_2 \in \mathcal{L}_m$. Then $\mathcal{X}^k = \{(f_1, f_1), (f_1, f_2), (f_2, f_1), (f_2, f_2), (\mathsf{X}, f_1), (\mathsf{X}, f_2)\}$, where X indicates that the flight is not included in the schedule. For each local SD-FAM decision $j \in \mathcal{X}^k$, a binary decision variable ω_j is defined such that it takes value 1 if the SD-FAM decision specified by j is adopted for all flights in $\mathcal{L}^k$ and zero otherwise.

Such a definition of the variables is called *composite* in the sense that the value of a variable indicates the SD-FAM decisions for multiple flights. For each local SD-FAM decision $j \in \mathcal{X}^k$, define $\kappa_{l,f}^j$ such that, $\kappa_{l,f}^j = 1$ if setting $\omega_j = 1$ assigns fleet type f to flight leg l, and $\kappa_{l,f}^j = 0$ otherwise. For all $j \in \mathcal{X}^k, l \notin \mathcal{L}^k$, we have $\kappa_{l,f}^j = 0,\ \forall f \in \mathcal{F}(l)$. For all $v \in N_f$, let $\text{in}_v^j \in \{0,1\}$ such that $\text{in}_v^j = \sum_{l \in I(v)} \kappa_{l,f}^j$ denote the number of incoming flights (flight arcs) to node v as specified by local SD-FAM decision j. Similarly, for all $v \in N_f$, $\text{out}_v^j \in \{0,1\}$ such that $\text{out}_v^j = \sum_{l \in O(v)} \kappa_{l,f}^j$ is the number of outgoing flights (flight arcs) from node v as specified by local SD-FAM decision j. Note that in_v^j and out_v^j do not have subscripts f because node $v \in N_f$ already ensures that it relates to fleet type f. Also, let $\text{count}_f^j = \sum_{l \in T_F(f)} \kappa_{l,f}^j$ be the corresponding number of flights that cross the count time under local SD-FAM decision j. Let $c_j = \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} \kappa_{l,f}^j$ be the operating cost of local SD-FAM decision j. Similarly, for each $j \in \mathcal{X}^k$, we can calculate r_j as the local revenue $r(\mathbf{x}^k; \mathbf{h}^k)$ where for $l \in \mathcal{L}^k$, $x_{l,f}^k = \kappa_{l,f}^j$ and for $l \notin \mathcal{L}^k$, $x_{l,f}^k = 1$ if $f = \arg\max_{f' \in \mathcal{F}(l)} \text{CAP}_{f'}$ and $x_{l,f}^k = 0$ otherwise. Using these new variable definitions, we present formulation S-CSD-FAM$(\Pi, \mathcal{H})$ below, which was first presented in Barnhart et al. (2009).

(S-CSD-FAM$(\Pi, \mathcal{H})$)

$$V_{\textsf{S-CSD-FAM}(\Pi,\mathcal{H})} = \max \quad \sum_{k=1}^{K} \sum_{j \in \mathcal{X}^k} (r_j - c_j)\omega_j \tag{17}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{X}^k} \omega_j = 1,\ \forall k \in \{1, \cdots, K\}, \tag{18}$$

$$y_{v^-} + \sum_{k=1}^{K} \sum_{j \in \mathcal{X}^k} \text{in}_v^j \omega_j - y_v - \sum_{k=1}^{K} \sum_{j \in \mathcal{X}^k} \text{out}_v^j \omega_j = 0, \quad \forall f \in \mathcal{F},\ \forall v \in N_f, \tag{19}$$

$$\sum_{v \in T_N(f)} y_v + \sum_{k=1}^{K} \sum_{j \in \mathcal{X}^k} \text{count}_f^j \omega_j \le n_f, \quad \forall f \in \mathcal{F}, \tag{20}$$

$$y_v \geq 0, \quad \forall f \in \mathcal{F},\ \forall v \in N_f, \tag{21}$$

$$\omega_j \in \{0,1\}, \quad \forall j \in \mathcal{X}^k,\ \forall k \in \{1,\cdots,K\}. \tag{22}$$

We define $V_{\textsf{S-CSD-FAM}(\Pi,\mathcal{H})}$ and $\tilde{V}_{\textsf{S-CSD-FAM}(\Pi,\mathcal{H})}$ as the optimal values of $\textsf{S-CSD-FAM}(\Pi,\mathcal{H})$ and of its continuous relaxation. We calculate r_j offline for each $j \in \mathcal{X}^k,\ k \in \{1,\cdots,K\}$ by running a series of linear programs given by (8) - (12). $\textsf{S-CSD-FAM}(\Pi,\ \mathcal{H})$ is essentially a change of variables of $\textsf{CSD-FAM}(\Pi,\ \mathcal{H})$. Constraints (18) are the reformulated version of the cover constraints requiring that for each subnetwork $\mathcal{L}^k$, exactly one local SD-FAM decision out of $\mathcal{X}^k$ needs to be selected. In contrast to constraints (2) and (3) where mandatory and optional flights are modeled separately in $\textsf{CSD-FAM}(\Pi,\ \mathcal{H})$, constraints (18) model them in a single equality by incorporating the "no assignment option" (that is, the option to not include an optional flight leg in the network) in the definition of the composite variables. Constraints (19) and (20) are reformulated versions of the flow balance and fleet availability constraints respectively.

$\textsf{S-CSD-FAM}(\Pi,\ \mathcal{H})$ has a tighter LP relaxation bound than $\textsf{CSD-FAM}(\Pi,\ \mathcal{H})$, which we will formally prove in Theorem 1. However, the cardinality of $\mathcal{X}^k$ is $\mathcal{O}(|\mathcal{F}|^{|\mathcal{L}^k|})$. In order to maintain the tractability of $\textsf{S-CSD-FAM}(\Pi,\ \mathcal{H})$, subnetwork sizes need to be limited. In other words, it only makes sense to deploy this reformulation for subnetworks whose size $|\mathcal{L}^k|$ is small. To achieve this, we present a more practical subnetwork-based *mixed* formulation whose strength, as measured by the tightness of its LP relaxation, lies between $\textsf{CSD-FAM}(\Pi,\ \mathcal{H})$ and $\textsf{S-CSD-FAM}(\Pi,\ \mathcal{H})$, and allows us to obtain a good trade-off between strength and size.

We divide the subnetworks determined by a partition $\Pi = (\Pi_c, \Pi_t)$ into two separate groups: Π_c and Π_t. Subnetworks in set Π_c are relatively smaller in size and modeled using composite variables (ω_j). Subnetworks in set Π_t are larger in size and modeled using traditional flight-fleet variables ($x_{l,f}$). Define $\mathcal{L}(\Pi_c) = \cup_{\mathcal{L}' \in \Pi_c} \mathcal{L}'$ as the set of flights represented using composite variables, and similarly $\mathcal{L}(\Pi_t) = \cup_{\mathcal{L}' \in \Pi_t} \mathcal{L}'$ as those represented using flight-fleet variables. This approach can enhance tractability by using composite variables to represent subnetworks with smaller sizes, and traditional flight-fleet variables to represent subnetworks with larger sizes. We now present our new subnetwork-based mixed formulation, denoted as $\textsf{S-CSD-FAM}(\Pi_c,\ \Pi_t,\ \mathcal{H})$ as follows.

($\textsf{S-CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})$)

$$V_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} = \max \sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{j \in \mathcal{X}^k} (r_j - c_j)\omega_j + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{k:\mathcal{L}^k \in \Pi_t} \sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \tag{23}$$

$$\text{s.t.} \sum_{j \in \mathcal{X}^k} \omega_j = 1,\ \forall k : \mathcal{L}^k \in \Pi_c, \tag{24}$$

$$\sum_{f \in \mathcal{F}(l)} x_{l,f} = 1,\ \forall l \in \mathcal{L}_m \cap \mathcal{L}(\Pi_t), \tag{25}$$

$$\sum_{f\in\mathcal{F}(l)} x_{l,f} \leq 1, \ \forall l \in \mathcal{L}_o \cap \mathcal{L}(\Pi_t), \tag{26}$$

$$y_{v^-} + \sum_{k:\mathcal{L}^k\in\Pi_c} \sum_{j\in\mathcal{X}^k} \mathrm{in}_v^j \omega_j + \sum_{l\in I(v)\cap\mathcal{L}(\Pi_t)} x_{l,f}$$
$$- y_v - \sum_{k:\mathcal{L}^k\in\Pi_c} \sum_{j\in\mathcal{X}^k} \mathrm{out}_v^j \omega_j - \sum_{l\in O(v)\cap\mathcal{L}(\Pi_t)} x_{l,f} = 0, \quad \forall f\in\mathcal{F}, \ \forall v\in N_f, \tag{27}$$

$$\sum_{v\in T_N(f)} y_v + \sum_{k:\mathcal{L}^k\in\Pi_c} \sum_{j\in\mathcal{X}^k} \mathrm{count}_f^j \omega_j + \sum_{l\in T_F(f)\cap\mathcal{L}(\Pi_t)} x_{l,f} \leq n_f, \quad \forall f\in\mathcal{F}, \tag{28}$$

$$y_v \geq 0, \quad \forall f\in\mathcal{F}, \ \forall v\in N_f, \tag{29}$$

$$\omega_j \in \{0,1\}, \quad \forall k:\mathcal{L}^k\in\Pi_c, \ \forall j\in\mathcal{X}^k, \tag{30}$$

$$x_{l,f} \in \{0,1\}, \quad \forall l\in\mathcal{L}(\Pi_t), \ \forall f\in\mathcal{F}(l), \tag{31}$$

$$x_{l,f}^k \text{ satisfies (13)}, \quad \forall l\in\mathcal{L}, \ \forall f\in\mathcal{F}(l), \ \forall k:\mathcal{L}^k\in\Pi_t. \tag{32}$$

Again, we define $V_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}$ and $\tilde{V}_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}$ as the optimal values of $\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})$ and its continuous relaxation. Objective function (23) is the summation of revenues minus operating costs for flights in Π_c and Π_t. Constraints (24) are the cover constraints for flights in Π_c and constraints (25) and (26) are the cover constraints for flights in Π_t. Constraints (27) and (28) are flow balance and availability constraints, respectively, with both variable representations included in them.

Given partition Π, the following theorem formally characterizes the relationship between the strengths of the aforementioned three subnetwork-based formulations — $\textsf{S-CSD-FAM}(\Pi,\mathcal{H})$ has the strongest continuous relaxation, $\textsf{CSD-FAM}(\Pi,\mathcal{H})$ has the weakest and $\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})$ lies in between the two. Moreover, as more subnetworks in Π are represented with composite variables, the strength of $\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})$ improves, at the expense of increasing the number of variables. Formulation $\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})$ offers the opportunity to explore the appropriate trade-off between the strength and size of the formulation. The proof of Theorem 1 involves constructing a (partial) convex-hull formulation to connect all the optimal values being considered. It also exploits the concavity of the revenue function $r(\mathbf{x};\mathbf{h})$ in seating capacity (and SD-FAM decisions $\mathbf{x}$). Note that this result holds under *any* flight partition Π, irrespective of whether it is independent or not.

THEOREM 1. *Given any partition* $\Pi = \{\mathcal{L}^1,\cdots,\mathcal{L}^k\}$, *any split* (Π_c,Π_t) *and any set of local fare vectors* $\mathcal{H}$,

$$V_{\textsf{CSD-FAM}(\Pi,\mathcal{H})} = V_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} = V_{\textsf{S-CSD-FAM}(\Pi,\mathcal{H})},$$
$$\tilde{V}_{\textsf{CSD-FAM}(\Pi,\mathcal{H})} \geq \tilde{V}_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} \geq \tilde{V}_{\textsf{S-CSD-FAM}(\Pi,\mathcal{H})}.$$

Moreover, for any two splits (Π_c, Π_t) *and* (Π'_c, Π'_t) *of the same partition* Π *with* $\Pi_c \supset \Pi'_c$ *and* $\Pi_t \subset \Pi'_t$,

$$\tilde{V}_{\textsf{S-CSD-FAM}(\Pi'_c,\Pi'_t,\mathcal{H})} \geq \tilde{V}_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}.$$

4.3. Approximation Analysis

The efficacy of the decomposition depends on the extent to which the flight-product-market dependency graph in Figure 2 is separable into multiple connected components. Real-world networks for large airline carriers often exhibit strong interdependencies which makes it difficult, if not impossible, to decompose them into many smaller subnetworks. In this subsection, we evaluate the accuracy of the revenue approximation given by the total subnetwork revenue $\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k)$ when the independence of partition $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ is violated. From a theoretical standpoint, these analyses shed light on the trade-off between tractability (which is a function of the extent to which the network can be decomposed) and accuracy of the subnetwork revenue approximations. From an algorithmic standpoint, it further offers insights on how partition Π should be constructed. These results demonstrate some parallels to the analysis conducted in Barnhart et al. (2009), but with the key distinction that we incorporate passenger choice behaviors.

First, Proposition 2 states that the sum of optimal revenues of the K subnetworks is an upper bound on the true optimal revenue, for any local SD-FAM decisions and local fare vectors.

PROPOSITION 2. *For any partition* $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$*, its local SD-FAM decisions* $\{\mathbf{x}^k\}_{k\in\{1,\cdots,K\}}$ *corresponding to any SD-FAM decisions* $\mathbf{x}$ *and any set of local fare vectors* $\mathcal{H} = \{\mathbf{h}^k\}_{k\in\{1,\cdots,K\}}$ *corresponding to the fare vector* $\mathbf{h}$*, the following holds:*

$$\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k) \geq r(\mathbf{x};\mathbf{h}).$$

An immediate corollary of Proposition 2 is that the optimal objective value of $\textsf{CSD-FAM}(\Pi,\mathcal{H})$ is always an upper bound of that of $\textsf{CSD-FAM}$ as $\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k) - \sum_{l\in\mathcal{L}}\sum_{f\in\mathcal{F}(l)} c_{l,f}x_{l,f} \geq r(\mathbf{x};\mathbf{h}) - \sum_{l\in\mathcal{L}}\sum_{f\in\mathcal{F}(l)} c_{l,f}x_{l,f}$ for all SD-FAM decisions $\mathbf{x}$. Similar relationship holds between their continuous relaxations as well. This is formally stated below.

COROLLARY 2. $V_{\textsf{CSD-FAM}(\Pi,\mathcal{H})} \geq V_{\textsf{CSD-FAM}}, \tilde{V}_{\textsf{CSD-FAM}(\Pi,\mathcal{H})} \geq \tilde{V}_{\textsf{CSD-FAM}}, \forall \Pi, \forall \mathcal{H}.$

Proposition 2 provides a lower bound of zero on the revenue difference $\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k) - r(\mathbf{x};\mathbf{h})$. Next, Proposition 3 establishes an upper bound on the revenue difference $\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k) - r(\mathbf{x};\mathbf{h})$. For a subset of the product set $\widehat{\mathcal{P}} \subset \mathcal{P}$, we denote by $\mathcal{M}(\widehat{\mathcal{P}}) = \{m \mid m \in \mathcal{M} \text{ s.t. } \mathcal{P}(m) \cap \widehat{P} \neq \emptyset\}$, the set of markets such that each market has at least one product in $\widehat{\mathcal{P}}$.

PROPOSITION 3. *For any partition* $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ *and its corresponding local product sets* $\{\mathcal{P}^1, \cdots, \mathcal{P}^K\}$, *if there exists a subset* $\widehat{P} \subset P$ *such that,*

1. $(\mathcal{P}^i \cap \mathcal{P}^j) \setminus \widehat{\mathcal{P}} = \emptyset, \ \forall i \neq j,$
2. $\mathcal{M}(\mathcal{P}^i \setminus \hat{\mathcal{P}}) \cap \mathcal{M}(\mathcal{P}^j \setminus \hat{\mathcal{P}}) = \emptyset, \ \forall i \neq j,$

then,

$$\sum_{k=1}^{K} r(\mathbf{x}^k; \mathbf{h}^k) - r(\mathbf{x}; \mathbf{h}) \leq \sum_{p \in \widehat{\mathcal{P}}} h_p \overline{s}_p,$$

for any $\mathbf{x}$, $\mathbf{h}$ *and* $\mathcal{H}$. $\overline{s}_p$ *is the maximum possible sale for product* p.

Intuitively, the bound is calculated based on the maximum revenue collected by the set of products without which the partition $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ would be an independent partition. The maximum possible sale $\overline{s}_p$ of a product p can be calculated in closed-form as $\overline{s}_p = \min\left\{\sum_{m \in \mathcal{M}(p)} \Lambda_m w_p / (w_p + \sum_{p' \in \mathcal{P}(m): p' \neq p} \tilde{w}_{p'} + w_m), \ \min_{l: \delta_{l,p}=1} \left\{\max_{f \in \mathcal{F}(l)} \mathrm{CAP}_f\right\}\right\}$ where $\mathcal{M}(p)$ denotes the set of markets to which product p belongs. The first term is the maximum possible demand for product p, obtained by closing all products other than p in market $m \in \mathcal{M}(p)$, and the second term is the maximum possible seating capacity offered on the flights whose capacity is used by product p.

4.3.1. Finding Local Fare Vectors $\mathbf{h}^k$ — The Fare Split Problem

Based on Corollary 2, we see that for any partition Π and any set of valid local fare vectors $\{\mathbf{h}^k\}_{k \in \{1, \cdots, K\}}$, the optimal objective value of $\mathsf{CSD\text{-}FAM}(\Pi, \mathcal{H})$, $V_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$, is an upper bound of the optimal objective value of $\mathsf{CSD\text{-}FAM}$, $V_{\mathsf{CSD\text{-}FAM}}$. Given any partition Π, the purpose of the fare split problem is to find a good set of local fare vectors $\{\mathbf{h}^k\}_{k \in \{1, \cdots, K\}}$ so that this upper bound $V_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$ is as tight as possible. This leads to the following formulation for selecting $\{\mathbf{h}^k\}_{k \in \{1, \cdots, K\}}$. We call it the fare split (FS) problem and denote its optimal value by $V_{\mathsf{FS}}(\Pi)$:

$$(\mathsf{FS}) \quad V_{\mathsf{FS}}(\Pi) = \min_{\{\mathbf{h}^k\}_{k \in \{1, \cdots, K\}} : \sum_{k=1}^{K} \mathbf{h}^k = \mathbf{h}} V_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}.$$

Although FS is a well-formulated problem, it is still intractable — the inner maximization problem of computing $V_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$ poses the same degree of complexity as $\mathsf{CSD\text{-}FAM}(\Pi, \mathcal{H})$. Therefore, we consider the following approximation of FS. We use the continuous relaxation of $\mathsf{CSD\text{-}FAM}(\Pi, \mathcal{H})$, which makes the optimal value of the inner problem become $\tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$. $\tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$ still produces a valid but possibly looser upper bound on the true optimal profit contribution, $V_{\mathsf{CSD\text{-}FAM}}$. We denote this relaxed problem as $\widetilde{\mathsf{FS}}$ and its optimal value as $\tilde{V}_{\mathsf{FS}}(\Pi)$.

$$(\widetilde{\mathsf{FS}}) \quad \tilde{V}_{\mathsf{FS}}(\Pi) = \min_{\{\mathbf{h}^k\}_{k \in \{1, \cdots, K\}} : \sum_{k=1}^{K} \mathbf{h}^k = \mathbf{h}} \tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}. \tag{33}$$

To reformulate $\widetilde{\mathsf{FS}}$ in a tractable form, we take the dual of the inner maximization problem that computes $\tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})}$. This transforms the inner problem into an equivalent minimization problem via strong duality. We then collapse the inner and outer problems into a single minimization problem, and cast it as a linear program. See Appendix A for complete derivation and formulation.

PROPOSITION 4. *Problem $\widetilde{\mathsf{FS}}$ can be reformulated as a linear program.*

4.3.2. Measure of Approximation Quality of a Partition Π

The optimal value of the relaxed fare-split problem $\widetilde{\mathsf{FS}}$, $\tilde{V}_{\mathsf{FS}}(\Pi)$, is a measure of the quality of profit approximation using partition Π. Observe that $\tilde{V}_{\mathsf{FS}}(\Pi)$ is an upper bound on the optimal value of the continuous relaxation of CSD-FAM, $\tilde{V}_{\mathsf{CSD\text{-}FAM}}$:

$$\tilde{V}_{\mathsf{FS}}(\Pi) = \min_{\{\mathbf{h}^k\}_{k\in\{1,\cdots,K\}}:\sum_{k=1}^{K}\mathbf{h}^k=\mathbf{h}} \tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})} \geq \tilde{V}_{\mathsf{CSD\text{-}FAM}}. \tag{34}$$

The inequality holds because $\tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})} \geq \tilde{V}_{\mathsf{CSD\text{-}FAM}}$ for all valid sets of local fare vectors $\mathcal{H}$ based on Corollary 2. This means that $\tilde{V}_{\mathsf{FS}}(\Pi)$ is the *tightest* upper bound on $\tilde{V}_{\mathsf{CSD\text{-}FAM}}$ among all the possible sets of local fare vectors $\mathcal{H}$ under partition Π. Thus, the ratio $(\tilde{V}_{\mathsf{FS}}(\Pi) - \tilde{V}_{\mathsf{CSD\text{-}FAM}})/\tilde{V}_{\mathsf{CSD\text{-}FAM}} \geq 0$ naturally serves as a measure of profit approximation quality using partition Π. Both terms in this measure are computationally tractable because both are linear programs. The closer $(\tilde{V}_{\mathsf{FS}}(\Pi) - \tilde{V}_{\mathsf{CSD\text{-}FAM}})/\tilde{V}_{\mathsf{CSD\text{-}FAM}}$ is to 0, the higher the quality of partition Π in terms of profit approximation. This measure, named the *profit approximation ratio*, will be used as a key element in the decomposition algorithm to construct Π, as discussed in Section 5.

5. Solution Approach for $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$

In this section, we discuss the details of the proposed solution approach for $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$. The approach is based upon the approximation analysis in subsection 4.3, with the goal of striking the right balance between tractability and solution quality. The solution approach contains two major parts: (1) first, subsection 5.2 presents an algorithm to create a partition $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ and assign each subnetwork to either Π_c (composite variable representation) or Π_t (flight-fleet representation); and (2) subsection 5.3 discusses a variable reduction technique to significantly reduce the number of variables for SD-FAM decisions for flights in $\mathcal{L}(\Pi_c)$.

5.1. Algorithm Overview

A typical run of the proposed solution approach for the $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$ formulation has four major steps, which are summarized below.

Solving $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$

Step 1 Run a partition algorithm by specifying the approximation ratio to construct a partition $\{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ of subnetworks and assign each subnetwork to either Π_c or Π_t.

Step 2 Solve the linear programming formulation of the relaxed fare split problem $\widetilde{\mathsf{FS}}$ stated in (33) to get local fare vectors $\mathbf{h}^k, \forall k \in \{1, \cdots, K\}$.

Step 3 Trim the set of all possible local SD-FAM decisions $\mathcal{X}^k$ using a variable reduction technique; calculate $r_j, \forall j \in \mathcal{X}^k, \forall k \in \{1, \cdots, K\} : \mathcal{L}^k \in \Pi_c$.

Step 4 Solve the $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$ formulation described in (23) - (32).

We want to point out that in order to calculate r_j in Step 3, it is not necessary to solve $\sum_{k:\mathcal{L}^k \in \Pi_c} |\mathcal{X}^k|$ separate linear programs. Instead, most of the computational effort is saved by utilizing tools from sensitivity analysis for linear programming. For $i, j \in \mathcal{X}^k : i \neq j$, the linear programs calculating r_i and r_j differ only in the right hand sides of some of their constraints, and thus the *100 percent rule* (Bradley et al. 1977) can be applied to determine whether the optimal basis matrices for previously solved linear programs are optimal for the new problem. From our computational experiments, we found that for approximately 80% of these linear programs in Step 3, we were able to calculate their optimal solutions directly based on previously solved optimal bases. Additionally, while the remaining 20% of the problems required solving new linear programs, we found that solving them can be greatly expedited with effective warm start techniques, e.g., using the dual simplex algorithm.

5.2. Partition Algorithm for Creating Subnetworks

Recall that $\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M})$ is the dependency graph with flight set $\mathcal{L}$, product set $\mathcal{P}$ and market set $\mathcal{M}$. $\Pi(\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M}))$ returns a partition of flights $\mathcal{L}$ implied by the connected components of $\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M})$. Proposition 3 states that the revenue difference can be upper bounded by the maximum possible revenue of the set of products which, when excluded, causes partition $\{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ to be an independent partition. Due to the NP-hardness of set partitioning problems such as this one, we develop a greedy heuristic which fragments the dependency graph $\mathcal{G}$ via gradual removal of products in *ascending* order of their maximum possible revenue contributions, $h_p \bar{s}_p$. The details of the algorithm are presented in Algorithm 1 below. It starts with partition $\Pi(\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M}))$. This initial partition can be computed in linear time in terms of the numbers of nodes and edges in $\mathcal{G}(\mathcal{L}, \mathcal{P}, \mathcal{M})$ (Tarjan 1972). It then iteratively removes product nodes in *ascending order* of their maximum revenue $h_p \bar{s}_p$ and finds the flight partition induced by the resulting dependency graph. At each iteration, the algorithm examines if the profit approximation ratio $(\tilde{\mathsf{V}}_{\mathsf{FS}}(\Pi) - \tilde{\mathsf{V}}_{\mathsf{CSD\text{-}FAM}})/\tilde{\mathsf{V}}_{\mathsf{CSD\text{-}FAM}}$

exceeds the threshold max_profit_ratio. If so, then the algorithm returns the partition in the previous iteration; else, it continues expanding the set of products being removed.

The key parameter max_profit_ratio determines the approximation quality of the partition. Additionally, the granularity of the product node removal process is controlled by the parameter revenue_step. As the network is being fragmented, subnetworks whose size can be represented by no more than max_vars (a third parameter) number of composite variables (i.e., $|\mathcal{X}^k| \leq \mathsf{max\_vars}$) are added to the set Π_c, and the rest of the subnetworks are placed in the set Π_t.

Out of the three parameters, max_profit_ratio is the most important one when using Algorithm 1 in practice. max_profit_ratio controls the size of the error introduced by the decomposition and requires some tuning in practice. Our computational experience reported in Section 6 and Appendix C offers insights on the selection of max_profit_ratio. A rule of thumb is that if the problem instance is hard and computation time is limited, a relatively large max_profit_ratio helps obtain a good solution faster; in contrast, if the problem instance is easier and computation time is more abundant, a smaller max_profit_ratio ensures a higher quality solution. revenue_step should be similar to the minimum value among the maximum product revenues, $\min_{p\in\mathcal{P}} h_p\bar{s}_p$, to ensure that product removal is not performed too aggressively. Our experience suggests that the outcome of Algorithm 1 is often quite robust to the value of revenue_step. In practice, one can choose revenue_step to be small enough that it results in a granular removal of product nodes and at the same time avoids an unnecessarily large number of iterations of the algorithm. Finally, max_vars controls the size and strength trade-off of the formulation as outlined in Theorem 1. This trade-off can often be made in a straightforward manner because the number of composite variables grows exponentially as the subnetwork size grows. According to our computational experience, max_vars should be set in the low thousands to take advantage of increased strength without exploding the size of the formulation.

Future research could further streamline the procedures in Algorithm 1. For example, one can attempt to directly tackle the problem of finding the partition that minimizes the profit upper bound: $\min_\Pi \tilde{V}_{\mathsf{FS}}(\Pi)$, subject to constraints that ensure a minimum degree of fragmentation. The problem is hard due to the combinatorial nature of the partitioning problem, and it would be interesting to try to develop efficient formulations or heuristics (possibly with performance guarantees) by exploiting problem structure.

5.3. Variable Reduction

The cardinality of $|\mathcal{X}^k|$ for any $\mathcal{L}^k \in \Pi_c$ equals $\left(\prod_{l\in\mathcal{L}^k\cap\mathcal{L}_o}(|\mathcal{F}(l)|+1)\right)\left(\prod_{l\in\mathcal{L}^k\cap\mathcal{L}_m}|\mathcal{F}(l)|\right)$ in the formulation $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c,\Pi_t,\mathcal{H})$. Here we develop and implement a modified version of the parsimonious enumeration method in Barnhart et al. (2009) to significantly reduce the number of variables necessary to represent $\mathcal{L}^k$, without sacrificing optimality.

Algorithm 1 Construct Partition $\Pi = (\Pi_c, \Pi_t)$

initialize

$\Pi_c \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| \leq \mathsf{max\_vars}\}$

$\Pi_t \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| > \mathsf{max\_vars}\}$

$\Pi \leftarrow (\Pi_c, \Pi_t)$

$\Pi_{\text{prev}} \leftarrow \Pi$

iter $\leftarrow 1$

while $|\mathcal{P}| > 0$ **do**

if $(\tilde{\mathsf{V}}_{\mathsf{FS}}(\Pi) - \tilde{\mathsf{V}}_{\mathsf{CSD\text{-}FAM}})/\tilde{\mathsf{V}}_{\mathsf{CSD\text{-}FAM}} \geq \mathsf{max\_profit\_ratio}$ **then**

return Π_{prev}

else

$\Pi_{\text{prev}} \leftarrow (\Pi_c, \Pi_t)$

$\mathcal{P} \leftarrow \{p \in \mathcal{P} : h_p \overline{s}_p \geq \mathsf{revenue\_step} \times \text{iter}\}$

$\Pi_c \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| \leq \mathsf{max\_vars}\}$

$\Pi_t \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| > \mathsf{max\_vars}\}$

$\Pi \leftarrow (\Pi_c, \Pi_t)$

end if

iter $\leftarrow$ iter $+ 1$

end while

return Π

For each flight leg $l \in \mathcal{L}$, define $\mathcal{M}(l) = \{m \in \mathcal{M} : \exists p \in \mathcal{P}(m) \text{ s.t. } \delta_{l,p} = 1\}$ as the set of markets that contain at least one product that uses the capacity of flight l. Define $\overline{d}_l$ as the maximum demand that a flight leg can capture if there is no seating capacity limit. It is calculated as,

$$\overline{d}_l = \sum_{m \in \mathcal{M}(l)} \left(\frac{\sum_{p \in \mathcal{P}(m)} \delta_{l,p} w_p}{w_m + \sum_{p \in \mathcal{P}(m)} (\delta_{l,p} w_p + (1 - \delta_{l,p}) \tilde{w}_p)} \right) \Lambda_m.$$

The intuition behind this formula is that in order to maximize demand captured on a particular flight, it is optimal to keep all the products related to that flight leg open all the time and keep all other products closed, so that demand can be maximally attracted to that specific flight.

Based on this concept of $\overline{d}_l$, we define sets of *constrained* and *unconstrained* fleet types, $\mathcal{F}_c(l)$ and $\mathcal{F}_{uc}(l)$ respectively, for each flight leg l. $\mathcal{F}_c(l) = \{f \in \mathcal{F}(l) : \text{CAP}_f < \overline{d}_l\}$ is the set of compatible fleet types for flight l whose capacity is smaller than the maximum possible demand $\overline{d}_l$. Similarly $\mathcal{F}_{uc}(l) = \{f \in \mathcal{F}(l) : \text{CAP}_f \geq \overline{d}_l\}$ is the set of fleet types whose capacity is at least equal to $\overline{d}_l$. To illustrate how $\mathcal{F}_c(l)$ and $\mathcal{F}_{uc}(l)$ are used to reduce the number of required variables, consider a subnetwork, $\mathcal{L}^k = \{l_1, l_2\}$, consisting of only two flights. Also, consider the case where both the flights are optional, i.e., $l_1, l_2 \in \mathcal{L}_o$. Let there be three fleet types f_1, f_2, f_3, and $\mathcal{F}(l_1) = \mathcal{F}(l_2) =$

$\{f_1, f_2, f_3\}$, $\mathcal{F}_c(l_1) = \mathcal{F}_c(l_2) = \{f_1\}$ and $\mathcal{F}_{uc}(l_1) = \mathcal{F}_{uc}(l_2) = \{f_2, f_3\}$. Parsimonious representation of $\mathcal{L}^k$ requires two sets of variables: *primary variables* and *secondary variables*, illustrated in Table 1.

Primary Variables	Secondary Variables
1. $(l_1, l_2) \leftarrow (f_1, f_1)$	10. $l_1 \leftarrow f_2$
2. $(l_1, l_2) \leftarrow (f_1, \mathsf{X})$	11. $l_1 \leftarrow f_3$
3. $(l_1, l_2) \leftarrow (f_1, \emptyset)$	12. $l_2 \leftarrow f_2$
4. $(l_1, l_2) \leftarrow (\mathsf{X}, f_1)$	13. $l_2 \leftarrow f_3$
5. $(l_1, l_2) \leftarrow (\mathsf{X}, \mathsf{X})$	
6. $(l_1, l_2) \leftarrow (\mathsf{X}, \emptyset)$	
7. $(l_1, l_2) \leftarrow (\emptyset, f_1)$	
8. $(l_1, l_2) \leftarrow (\emptyset, \mathsf{X})$	
9. $(l_1, l_2) \leftarrow (\emptyset, \emptyset)$	

Table 1 Parsimonious Representation

As before, X denotes the case where the flight is not included in the schedule. $\emptyset$ represents the case where the fleet assignment is not specified in primary variables, but instead described in the secondary variables. More specifically, it means that a collection of unconstrained fleet types, $\mathcal{F}_{uc}(l)$, is assigned to flight l, but which specific fleet type to assign is determined by the secondary variables. Note that if $\mathcal{F}_{uc}(l)$ is empty, then there are no secondary variables, and also no $\emptyset$ values in the primary variables. The reason why the unconstrained fleet types can be collectively represented as a single primary variable per flight leg is that assigning different unconstrained fleet types f to flight l does not change its revenue contribution. In this example, assignments $(l_1, l_2) \leftarrow (f_1, f_2)$ and $(l_1, l_2) \leftarrow (f_1, f_3)$ have the same revenue. All 16 possible fleet assignments, $\{f_1, f_2, f_3, \mathsf{X}\}^2$, thus can be represented by this list of primary and secondary variables, which contains 13 variables in total. For example, $(l_1, l_2) \leftarrow (\mathsf{X}, f_3)$ can be represented by primary variable $(l_1, l_2) \leftarrow (\mathsf{X}, \emptyset)$ and secondary variable $l_2 \leftarrow f_3$. The magnitude of savings in terms of the reduction in the number of variables can grow significantly as the size of the subnetwork and the number of fleet types grow.

We define $\widehat{\mathcal{X}}_1^k$ as the set of primary variables and $\widehat{\mathcal{X}}_2^k$ as the set of secondary variables for $\mathcal{L}^k$. We then replace constraints (24) with two sets of constraints, (35) and (36), as below.

$$\sum_{j \in \widehat{\mathcal{X}}_1^k} \omega_j = 1, \;\forall k : \mathcal{L}^k \in \Pi_c, \tag{35}$$

$$\sum_{j \in \widehat{\mathcal{X}}_1^k \cup \widehat{\mathcal{X}}_2^k} \kappa_{l,f}^j \omega_j + \sum_{j \in \widehat{\mathcal{X}}_1^k} \sigma_l^j \omega_j = 1, \;\forall l \in \mathcal{L}^k, \;\forall k : \mathcal{L}^k \in \Pi_c. \tag{36}$$

Recall the definition of $\kappa_{l,f}^j$ as $\kappa_{l,f}^j = 1$ if flight l is assigned fleet type f in variable ω_j, and zero otherwise. We also define $\sigma_l^j = 1$ if flight l is deleted from the schedule (assigned X) in variable ω_j, and zero otherwise. Constraints (35) require that exactly one primary variable is selected. Constraints (36) state that a flight is either assigned to a specific fleet type or not included in the schedule. For consistent accounting of revenue and cost, secondary variables are assigned zero revenue, i.e., $r_j = 0,\ \forall j \in \widehat{\mathcal{X}}_2^k,\ \forall k : \mathcal{L}^k \in \Pi_c$. Primary variables use the same revenue calculation as defined previously in Section 4, i.e., $r_j = r(\mathbf{x}^k; \mathbf{h}^k)$. Note that for flight leg l assigned to $\emptyset$ in variable ω_j, we assign any fleet type in the set of unconstrained fleet types $\mathcal{F}_{uc}(l)$ for revenue calculation. Cost calculation remains the same for both variables as $c_j = \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} \kappa_{l,f}^j$. This effectively means that flights assigned with X or $\emptyset$ do not contribute to cost c_j for all primary variables $j \in \widehat{\mathcal{X}}_1^k$.

6. Computational Experiments

The goal of the experiments is to demonstrate the effectiveness of our proposed S-CSD-FAM$(\Pi_c, \Pi_t, \mathcal{H})$ model and our solution approach by comparing it to several baseline approaches and implementations. In Section 6.1, we report the computational performance of solving various formulations and demonstrate the advantage of the proposed S-CSD-FAM$(\Pi_c, \Pi_t, \mathcal{H})$ model and its solution approaches. In Section 6.2, we provide an objective evaluation of profit contributions of the solutions obtained from solving the various formulations, by means of a passenger booking simulation. In the simulation, passengers randomly arrive over a booking horizon, a simple seat inventory control policy is implemented and revenue is accumulated as booking decisions are made.

We use problem instances from two major airlines whose aggregate characteristics are summarized in Table 2. These two instances are provided by our industry partner. Instance D1 is an airline serving mainly international markets, and instance D2 is a US domestic airline. Both instances represent daily schedules created from actual schedules in 2016. We label all flights as optional, i.e., we consider a full schedule reduction and fleet assignment problem providing challenging instances for a maximum difficulty level. Airlines often solve such full schedule reduction and fleet assignment problems periodically to examine and enhance the profitability of their schedules. In Appendix C.3, we also present a set of experiments where all flights are mandatory which leads to a pure fleet assignment problem.

For each instance, we report the number of flights, itineraries, fare products (itinerary-fare class combinations), markets (origin-destination pairs), fleet types, number of individual aircraft, and market shares. All fare products sharing the same origin and destination will be jointly considered by passengers from that market. Note that this implicitly ensures that the sets of fare products in different markets are mutually exclusive, i.e., non-overlapping. The estimation of product attractiveness values was conducted by our industry partner using an MNL model (i.e., while setting

$\tilde{w}_p = 0, \forall p \in \mathcal{P}$). The product attractiveness values w_p, attractiveness values w_m of the outside option, and market demands Λ_m are estimated using passenger shopping, airline booking, and product availability data as well as market share estimates, with methods described in Vulcano et al. (2012). The estimation is performed for each origin-destination market which includes all fare products with that origin and destination.

Instance D1 has a flight network with fewer flights, but covers more origin-destination markets than instance D2. D1 also has an overall smaller unconstrained market share (20.7%) compared to instance D2 (45.9%). Unconstrained market share, in this context, is computed as the ratio of the sum of the attractiveness values of all products of the host airline to the sum of the attractiveness values of all products of the host airline and the attractiveness value of the outside option, in all markets, $(\sum_{m\in\mathcal{M}}\sum_{p\in\mathcal{P}(m)} w_p)/(\sum_{m\in\mathcal{M}}\sum_{p\in\mathcal{P}(m)} w_p + \sum_{m\in\mathcal{M}} w_m)$. Note that demand estimates are based on recent transaction data or historical data from similar time periods. In practice, since SD-FAM decisions are typically applied to future seasons, these estimates may need to be adjusted to account for expected changes over time. For simplicity, we do not make such adjustments in this subsection because we do not have access to the information that airlines use in making such adjustments. However, in the next subsection, we will evaluate the robustness of profit contributions when the demand estimates are subject to errors.

Instance	**Flights**	**Itineraries**	**Fare Products**	**Markets**	**Fleet Types**	**Aircraft**	**Market Share**
D1	180	1,195	10,645	1,648	9	71	20.7%
D2	815	4,290	47,190	819	7	186	45.9%

Table 2 Problem Instances

We apply Algorithm 1 to both instances D1 and D2 to create flight partitions (Π_c, Π_t). For each instance, we create three different partitions $(\Pi_c^0, \Pi_t^0), (\Pi_c^1, \Pi_t^1), (\Pi_c^2, \Pi_t^2)$ corresponding to three different values of the maximum profit approximation ratio, max_profit_ratio $\in \{0.5\%, 1\%, 2\%\}$. The maximum number of composite variables in each subnetwork, max_vars, is set to 1,000. revenue_step for removing product nodes from the dependency graph $\mathcal{G}$ is set to \$500. Table 3 illustrates the characteristics of the resulting partitions.

Column 3 in Table 3 specifies the max_profit_ratio we use for creating each partition, and column 4 specifies the actual maximum profit approximation ratio achieved. Column 5 ("FS Time") lists the CPU time (in seconds) used in solving the relaxed linear programming version of the fare split problem ($\widetilde{\mathsf{FS}}$) to obtain local fare vectors. It was found to be just a few seconds. Columns 6 to 10 present the number of subnetworks within Π_c with different numbers of flights (columns 6 to 9), and the total number of flights in Π_c (column 10). Similarly, columns 11 to 13 provide details about the subnetworks in Π_t. Columns 11 and 12, respectively, list the minimum and maximum number of flights across all subnetworks in Π_t, and column 13 lists the total number of subnetworks in Π_t. We observe that under similar profit approximation ratios, we are able to partition D1 into much smaller subnetworks than what is possible for D2. The average sizes of subnetworks under the three approximation ratios 0.5%, 1% and 2% are 3.33, 1.94 and 1.25 for instance D1, and 8.40, 6.74 and 5.03 for instance D2, respectively. This is partly because D2 exhibits much stronger dependencies among its flights, fare products and markets. This is consistent with D2's high market share, which suggests that passengers tend to substitute more often within the host airline's offerings when the preferred fare product is not available.

Inst.	Partition	Approx. Quality & Fare Split			# of subnetworks in Π_c of size					Subnetworks in Π_t		
		max_profit_ratio	$\frac{\tilde{V}_{\text{FS}}(\Pi)-\tilde{V}_{\text{CSD-FAM}}}{\tilde{V}_{\text{CSD-FAM}}}$	FS time	1	2	3	4	# of flights in Π_c	min size	max size	$\|\Pi_t\|$
D1	(Π_c^0, Π_t^0)	0.5%	0.5%	1.46	49	2	2	0	59	121	121	1
	(Π_c^1, Π_t^1)	1.0%	0.9%	3.47	85	2	3	0	98	20	41	3
	(Π_c^2, Π_t^2)	2.0%	1.8%	2.14	127	9	5	1	164	5	11	2
D2	(Π_c^0, Π_t^0)	0.5%	0.3%	24.35	65	15	10	2	133	4	661	5
	(Π_c^1, Π_t^1)	1.0%	0.8%	25.67	79	19	14	1	163	4	611	8
	(Π_c^2, Π_t^2)	2.0%	1.5%	24.85	104	25	19	1	212	4	532	13

Table 3 Subnetwork Partitions

6.1. Computational Performance of the Formulations

We first test the computational performance of our solution approach to S-CSD-FAM$(\Pi_c, \Pi_t, \mathcal{H})$ model with three partitions corresponding to three different levels of the maximum profit approximation ratios, as illustrated in Table 3. We compare it with the following six baselines:

- ISD-FAM. This is an SD-FAM model with independent demands as used in Sherali et al. (2010) – see Appendix B.1 for the formulation details. Demand is associated with each fare product and if a particular fare product is not available, then its demand does not get recaptured by another product. We use the unconstrained demand of a product as its independent demand.
- ISD-FAM-SR-ITIN. This is an extension of ISD-FAM with spill and recapture. It was first introduced in Lohatepanont and Barnhart (2004). The model uses the Quality of Service Index (QSI) to model spill-and-recapture among fare products. However, the formulation requires defining a decision variable for each pair of substitutable fare products, which leads to significantly increased computational complexity. Thus the authors proposed a method to aggregate different fare products of the same itinerary to improve computational tractability. They use a weighted average to aggregate fare products where weights are the unconstrained demands of the individual products. Appendix B.2 provides the details of this formulation. In Appendix C, we also test the performance of another version of their model where fare products are not aggregated, and are instead individually modeled explicitly.

- CSD-FAM. This is the formulation we introduced in (1) - (7), which is the schedule design and fleet assignment model with GAM demand *without* any subnetwork-based decomposition.
- CSD-FAM-S. This is the same as the CSD-FAM model but with added symmetry-inducing constraints used in the heuristic proposed by Wei et al. (2020). They require that the number of flight legs flown by each aircraft type from airport A to airport B directly is the same as those from B to A. These constraints sometimes speed up the solution procedure by reducing the solution space of the SD-FAM decisions, but they might introduce sub-optimality.
- S-CSD-FAM-BFM. This is the subnetwork-based decomposition framework developed by Barnhart et al. (2009). BFM stands for the initials of the three authors. They require all subnetworks to be represented using composite variables. For fare proration, we use proration by distance for simplicity and consistency with industry practice. Subnetworks are constructed using the partition algorithm provided in Barnhart et al. (2009) whose details are reported in Appendix B.3.
- CSD-FAM$(\Pi, \mathcal{H})$. This is the formulation we introduced in (16), which is the subnetwork-based decomposition under partition Π with *all* subnetworks represented by fleet-flight variables.

Table 4 summarizes the computational performance of the various models and solution approaches for instances D1 and D2. The experiments were conducted over a Microsoft Azure cluster with 16 cores and 64GB memory. All mathematical programs were solved using Gurobi 9.0 with default parameters. We set a 5-hour computation time limit. This computation budget is consistent with prior studies (e.g., Barnhart et al. 2009). In practice, airlines often want to re-run the model with different demand scenarios, and smaller computation budgets for each run can be desirable despite the strategic nature of the SD-FAM problem.

Column 3 in Table 4 lists two fare decomposition methods: (1) "dist" denotes the fare split calculated using the proration by distance heuristic, (2) "opt" denotes the fare split calculated by solving the linear programming formulation $(\widetilde{\mathsf{FS}})$ as in Section 4.3.1. proration by distance is a commonly used heuristic to calculate flight level revenue contribution, both in FAM literature (see Williamson (1992) and Barnhart et al. (2002) for detailed discussions) and industry practice (Cirium 2019). In our case, we set $h_p^k = h_p \cdot \left(\sum_{l\in\mathcal{L}^k} \delta_{l,p}\mathrm{dist}_l\right) / \left(\sum_{l\in\mathcal{L}} \delta_{l,p}\mathrm{dist}_l\right)$, where dist_l is the flying distance of flight leg l. Column 4 reports the best objective values the optimizer obtains (in million \$ per day) when the solution process terminates either because an optimal solution is found or the CPU time limit is reached. Column 5 reports the optimal values (in million \$ per day) of the continuous relaxations (LP relaxations) of the corresponding models. Column 6 lists the CPU time (in seconds) to find a certifiably optimal solution. Note that for various S-CSD-FAM$(\Pi_c, \Pi_t, \mathcal{H})$ formulations, this CPU time also includes the time spent in constructing partitions Π (Algorithm

Inst.	Model	Fare Split	Obj.	LP Relax.	CPU Time	MIP Gap	B&B Nodes	#Flights	ΔProfit
D1	ISD-FAM (indep. demand)	-	2.87	3.36	304	0.00%	2,186	122	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	2.26	2.57	10,503	0.00%	194,785	118	(30.90)
	CSD-FAM (choice based demand)	-	3.66	4.36	LIMIT	0.98%	131,202	126	3.74
	CSD-FAM-S (w/ symmetry constr.)	-	3.66	4.36	LIMIT	0.95%	118,932	126	4.28
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	3.93	3.94	1,324	0.00%	0	128	(10.14)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	3.75	4.44	LIMIT	0.90%	51,934	124	1.31
		opt	3.73	4.38	LIMIT	0.76%	59,566	126	5.22
	S-CSD-FAM($\Pi_c^0, \Pi_t^0, \mathcal{H}$) (w/ composite vars)	dist	3.75	4.31	LIMIT	0.60%	72,898	124	1.31
		opt	3.73	4.28	LIMIT	0.44%	73,837	126	5.22
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	4.00	4.64	17,952	0.00%	90,044	126	(4.52)
		opt	3.82	4.40	17,292	0.00%	170,836	126	5.62
	S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) (w/ composite vars)	dist	4.00	4.44	1,837	0.00%	43,280	126	(4.52)
		opt	3.82	4.26	1,807	0.00%	39,768	126	**5.62**
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	4.17	4.83	8,111	0.00%	47,411	126	(30.33)
		opt	3.93	4.45	3,193	0.00%	82,840	128	1.39
	S-CSD-FAM($\Pi_c^2, \Pi_t^2, \mathcal{H}$) (w/ composite vars)	dist	4.17	4.23	25	0.00%	0	126	(30.33)
		opt	3.93	3.97	25	0.00%	0	128	1.39
D2	ISD-FAM (indep. demand)	-	4.25	4.50	LIMIT	0.03%	27,727	743	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	3.94	4.15	LIMIT	2.21%	7,992	728	(41.55)
	CSD-FAM (choice based demand)	-	6.65	7.25	LIMIT	6.48%	7,109	656	11.98
	CSD-FAM-S (w/ symmetry constr.)	-	6.41	7.13	LIMIT	8.33%	6,917	622	(79.36)
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	9.48	9.48	6,705	0.00%	299	775	(182.92)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	6.76	7.31	LIMIT	5.85%	6,922	671	20.02
		opt	6.77	7.27	LIMIT	5.06%	7,160	651	35.41
	S-CSD-FAM($\Pi_c^0, \Pi_t^0, \mathcal{H}$) (w/ composite vars)	dist	6.79	7.26	LIMIT	5.13%	7,022	674	25.51
		opt	6.72	7.22	LIMIT	5.68%	6,214	653	42.25
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	6.83	7.37	LIMIT	5.60%	6,683	651	11.68
		opt	6.85	7.30	LIMIT	4.13%	7,509	669	38.60
	S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) (w/ composite vars)	dist	6.90	7.31	LIMIT	3.67%	9,522	678	42.99
		opt	6.86	7.24	LIMIT	3.96%	6,634	683	**47.15**
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	6.88	7.47	LIMIT	6.32%	7,209	676	(11.52)
		opt	6.86	7.35	LIMIT	5.20%	7,504	679	14.04
	S-CSD-FAM($\Pi_c^2, \Pi_t^2, \mathcal{H}$) (w/ composite vars)	dist	7.01	7.39	LIMIT	3.56%	7,432	692	26.68
		opt	6.96	7.28	LIMIT	2.79%	10,046	691	45.52

Table 4 Computational Results with Maximum 5hr CPU Time Limit. Profit changes relative to ISD-FAM are reported in million \$/365 days. Negative numbers are reported in parentheses. Solution times are reported in CPU seconds. Baseline profit for ISD-FAM is \$1.329 billion for D1 and \$2.419 billion for D2 per 365 days.

1), computing the set of local fare vectors $\mathcal{H}$, and calculating r_j values. "LIMIT" means that a certifiably optimal solution was not found within the time limit, and in such cases, a percentage gap of the solution value with respect to the best bound is reported in Column 7. Column 8 reports the number of branch-and-bound nodes explored until termination. Column 9 reports how many flights are retained in the resulting solution. Finally, we report separately in Column 10 the *annual* profit change (in million \$ per 365 days) of the resulting solution compared to that of ISD-FAM, under the SBLP revenue evaluation (8) - (12). Next, we discuss the performance of the baselines.

As expected, CSD-FAM faces substantially greater computational difficulty compared to ISD-FAM. In instance D1, the optimizer fails to find a certifiably optimal solution for CSD-FAM within 5 hours, while it solves ISD-FAM in around 5 minutes. In instance D2, ISD-FAM can be solved within an optimality gap of 0.03% within 5 hours, while there is still a significantly larger gap (6.48%) for CSD-FAM at the end of 5 hours. Despite the increased computational complexity, CSD-FAM achieves an improved profit contribution compared to ISD-FAM when evaluated under SBLP. This is especially true for instance D2 where the host airline has a strong market presence and passengers tend to substitute more often within the host airline's own offerings compared to that in instance D1. The tractability of ISD-FAM-SR-ITIN lies between those for ISD-FAM and CSD-FAM. However, because of the aggregation of fare products into itineraries and possibly due to its different way of modeling passenger spill-and-recapture, ISD-FAM-SR-ITIN delivers lower profit compared to ISD-FAM under the SBLP revenue evaluation. We also tested the version of ISD-FAM-SR-ITIN without aggregating fare products – see results in Table 9 in Appendix C.

CSD-FAM-S exhibits drastically different performance under the two instances. For instance D1, imposing the symmetry-inducing constraints seems not to impact optimality much; it helps to speed up CSD-FAM and produce a slightly better solution than CSD-FAM (though not as good as that produced by S-CSD-FAM$(\Pi_c^1, \Pi_t^1, \mathcal{H})$). On the other hand, its performance deteriorates dramatically in instance D2 where imposing symmetry takes a heavy toll on profitability. In addition, in the pure fleet assignment task reported in Appendix C.3, CSD-FAM-S is infeasible for instance D2. These observations suggest that the value of the symmetry-inducing constraints greatly depends on the underlying network structure and might not be that effective in the context of incremental schedule design – recall that Wei et al. (2020) proposed them for the complete timetabling problem. Finally, S-CSD-FAM-BFM consistently underperforms in both instances. This pure subnetwork-based formulation gives very loose profit bounds (column 4), and serves as evidence of the need for our mixed formulation and our original optimization-based fare-splitting method.

Now that we have discussed the performance of the baselines, we summarize key observations from the computational performance of our new subnetwork-based formulations.

Balance between tractability and quality. Out of the three subnetwork partitions, the partition with a 1% maximum profit approximation ratio – (Π_c^1, Π_t^1) – generally produces solutions with the highest profit contributions in both instances within 5 hours of CPU time. Allowing for only up to 1% profit approximation error seems to balance solution quality and tractability, and yield the best performance in each of these two instances under this computational budget. Partition (Π_c^0, Π_t^0) seems to focus too much on improving the profit approximation at the expense of tractability, resulting in subnetworks that are not fragmented enough. In contrast, partition (Π_c^2, Π_t^2) increases tractability by additional segmenting of flights into smaller subnetworks, but this additional tractability comes at the cost of a looser profit approximation.

Efficiency due to composite variables. Within each partition, we observe a clear computational advantage of our mixed formulation $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$ over $\mathsf{CSD\text{-}FAM}(\Pi, \mathcal{H})$, which does not use any composite variables to represent the subnetwork SD-FAM decisions. This is reflected in tightened continuous relaxation bounds (Column 5) for both instances, all three partitions, both fare split methods; significantly reduced computation times (Column 6) with reductions for instance D1 on the order of 10 (for the second partition) to 1,000 (for the third partition) times; decreased optimality gaps (Column 7) in almost all combinations in the table; and most importantly, improved solution quality within the same computational budget (as clearly demonstrated by instance D2 for all three partitions and both fare split methods). These observations are consistent with our theoretical results in Theorem 1 comparing the strengths of these two reformulations.

Advantage of optimization-based fare allocation. When comparing the two fare splitting methods, we clearly and consistently observe that a significant profit boost comes from the optimization-based fare split (problem $\widetilde{\mathsf{FS}}$) over the commonly used proration-by-distance heuristic. This was found to be true for all three partitions in both instances for mixed as well as pure subnetwork reformulations, and was often critical to the subnetwork-based reformulations performing better than CSD-FAM. The profit improvement is due to the tighter profit upper bounds (Column 5) obtained with the opt fare split than those with dist fare split within the same partition.

Comparing subnetwork-based formulations $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^0, \Pi_t^0, \mathcal{H})$, $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^1, \Pi_t^1, \mathcal{H})$ and $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^2, \Pi_t^2, \mathcal{H})$ under the optimization-based fare split with CSD-FAM, one could observe that both $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^0, \Pi_t^0, \mathcal{H})$ and $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^1, \Pi_t^1, \mathcal{H})$ are able to find better solutions for instance D1 under the same computational budget, and additionally $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^1, \Pi_t^1, \mathcal{H})$ is able to accomplish it with significantly (almost tenfold) lower computation time. For instance D2, our results are even more impressive – all three mixed formulations $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^0, \Pi_t^0, \mathcal{H})$, $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^1, \Pi_t^1, \mathcal{H})$ and $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^2, \Pi_t^2, \mathcal{H})$ with either fare split method are able to significantly improve profit values over CSD-FAM, and again $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c^1, \Pi_t^1, \mathcal{H})$ has the best performance.

Based on the SBLP revenue evaluation, the best subnetwork-based formulation (S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$)) achieves \$5.62 million (around 0.5%) annual profit improvement in instance D1 and \$47.15 million (around 2.0%) in instance D2, over ISD-FAM. The greater profit improvement in instance D2 results from a higher market share causing more substitutions within the host airline's own fare products.

In Table 8 in Appendix C, we provide additional computational results with the same evaluation procedure, but with an extended 10 hours of maximum computation time. We observe similar insights as in Table 4. For instance D1, the quality of the CSD-FAM solution after 10 hours of computation time is still moderately inferior to that of S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) after 30 minutes of computation time. For instance D2, S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) still provides a clear and substantial edge over CSD-FAM even after running both for 10 hours of maximum computation budget.

For the purpose of ensuring reproducibility, we made public a synthetic dataset with realistic characteristics. This dataset and its detailed computational results can be accessed at `https://github.com/yanchiwei/Choice_FAM_datasets`.

6.2. Booking Simulation

One drawback of the profit evaluation reported in Table 4 is that it is based on a deterministic upper bound of the actual expected profit collected from randomly arriving demand over the booking horizon. In this subsection, we provide a more accurate and objective evaluation of the profit corresponding to different SD-FAM decisions using a booking simulation.

In the simulation, the entire booking horizon is equally divided into $T = 100$ time periods, where each time period can be considered to be approximately two to three days. This mimics industry practice where the booking horizon corresponds to the several months ahead of the schedule period and the entire booking horizon is divided into a set of time periods. It is a simplification to some extent, as time periods are often set to be wider in the early stage of the booking horizon and narrower toward the end of the booking horizon to account for the increasing intensity of passenger arrivals closer to the flight departure time. We simplify this aspect by assuming a homogeneous Poisson arrival process.

At each time period, the number of passengers arriving in origin-destination market $m \in \mathcal{M}$ is distributed according to a Poisson random variable with rate Λ_m/T. Each arrived passenger chooses a fare product $p \in \mathcal{P}(m)$ to purchase according to probability $w_p/(\sum_{p' \in \bar{\mathcal{P}} \cap \mathcal{P}(m)} w_{p'} + w_m)$ where $\bar{\mathcal{P}}$ is the set of available fare products at the time of arrival. At the beginning of each time period, the airline reviews the currently remaining seating capacity, and based on the forecasts of the remaining demand until departure, it decides which fare products to offer. We implement a standard bid-price control heuristic based on the dual formulation of the SBLP formulation (8) to (12). Let α_l denote the dual variable for the capacity constraint (9) of flight leg $l \in \mathcal{L}$, and β_m denote the dual variable for the market demand constraint (10) of market $m \in \mathcal{M}$. A product $p \in \mathcal{P}$ is offered if and only if $h_p - \sum_{l \in \mathcal{L}} \delta_{l,p}\alpha_l - \sum_{m \in \mathcal{M}(p)} \beta_m \geq 0$. The dual variable values are updated after every time period by resolving the SBLP formulation with updated demand and remaining capacity information (see Chapter 7.3.3 of Gallego and Topaloglu (2019) for more details and a formal analysis of this method). At the end of each simulation run, operating profit is calculated as the total collected revenue minus the operating cost.

In practice, estimates/forecasts of product attractiveness w_p, outside option attractiveness w_m and market demand Λ_m are subject to errors. This motivates us to evaluate profit contribution when these estimates are imperfect. Specifically, we assume that in each simulation, the true values of w_p and w_m (and consequently those for market demand Λ_m too) can deviate from their estimated values by $x\%$ where $x\%$ is sampled from a uniform distribution on $[-e, e]$, with $e \in \{0\%, 10\%, 25\%, 50\%\}$. To isolate the effects of SD-FAM decisions from the seat inventory control policy, we assume that the SD-FAM decisions are computed based on such imperfect (nominal) demand model estimates at the beginning of the booking horizon, while the bid-price control heuristic uses the accurate demand information and is unaffected by these forecasting imperfections.

Note that $e = 0\%$ represents the perfect estimation case. For each $e \in \{10\%, 25\%, 50\%\}$, we simulate five demand scenarios $\{(w_p, w_m, \Lambda_m)\}_{p \in \mathcal{P}(m), m \in \mathcal{M}}$. Importantly, each row in Table 5 corresponds to *one single realization* of demand scenario in terms of parameters $\{(w_p, w_m, \Lambda_m)\}_{p \in \mathcal{P}(m), m \in \mathcal{M}}$, and it reports the average and the 95% confidence interval of annual profit change (in million \$ per 365 days) compared to the ISD-FAM solution, based on 100 simulation runs with same random numbers. This is reported in the ΔProfit(Sim) column. We also report the profit change (ΔProfit(SBLP)) evaluated using the SBLP formulation, as we did in Table 4.

Our first observation is that the revenue evaluation under the SBLP formulation is quite reliable. A solution with higher profit under the SBLP formulation often has higher profit (with only one exception in Table 5) in the booking simulation with similar levels of profit improvement. This empirically justifies using a deterministic linear program to approximate expected revenue in our SD-FAM model.

When demand estimates/forecasts are perfectly accurate ($e = 0\%$), compared to ISD-FAM, S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) produces on average \$5.93 million profit improvement per 365 days for instance D1 and \$40.11 million for instance D2. CSD-FAM delivers on average \$5.65 million profit improvement per 365 days for instance D1 and \$3.37 million for instance D2. As expected, when demand model estimates become less accurate, the relative advantage of CSD-FAM and S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) over ISD-FAM weakens. Nevertheless, even under the most inaccurate demand estimates ($e = 50\%$), S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) is still able to outperform ISD-FAM in most of the demand scenarios for both instances (on average \$0.78 million additional profit per 365 days for instance D1, \$31.75 million for instance D2). In contrast, the performance of CSD-FAM often is worse than that of ISD-FAM (on average \$0.61 million less profit per 365 days for instance D1, and \$1.90 million for instance D2). This demonstrates the fact that S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) is able to produce consistently better SD-FAM solutions under limited computation time and with both accurate and inaccurate demand forecasts, an appealing property for practical implementation.

In Table 7 in Appendix C, we also provide simulation results for solutions obtained after 10 hours of computation time. They mirror the qualitative insights gleaned from the results in Table 5. After 10 hours of computation time, the quality of the CSD-FAM and S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) solutions for instance D1 are comparable. However, for instance D2, S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$) still considerably outperforms CSD-FAM in the majority of demand scenarios.

Inst.	Perturb.	Scenario	ISD-FAM-SR-ITIN		CSD-FAM		S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$)	
			ΔProfit(SBLP)	ΔProfit(Sim)	ΔProfit(SBLP)	ΔProfit(Sim)	ΔProfit(SBLP)	ΔProfit(Sim)
D1	±0%	1	(30.90)	(25.75) ± 6.22	3.74	5.65 ± 6.18	5.62	**5.93 ± 6.16**
	±10%	1	(29.92)	(27.44) ± 6.79	3.43	**6.47 ± 6.45**	6.65	5.72 ± 6.47
		2	(31.08)	(27.44) ± 6.51	3.60	5.19 ± 6.95	4.60	**6.62 ± 6.96**
		3	(30.76)	(24.69) ± 6.13	3.03	0.54 ± 6.59	5.34	**5.23 ± 6.04**
		4	(29.90)	(26.97) ± 6.20	2.66	2.16 ± 6.24	5.82	**7.11 ± 6.22**
		5	(31.13)	(29.32) ± 6.60	2.65	0.59 ± 6.30	4.34	**5.01 ± 6.23**
	±25%	1	(28.14)	(23.49) ± 6.61	3.02	4.64 ± 6.97	8.19	**4.80 ± 6.66**
		2	(31.16)	(30.45) ± 5.97	3.26	**6.30 ± 5.91**	2.95	4.00 ± 5.90
		3	(30.86)	(24.18) ± 7.06	1.24	2.42 ± 6.77	4.25	**6.52 ± 6.68**
		4	(27.40)	(26.29) ± 7.53	0.99	(0.20) ± 7.14	6.69	**5.53 ± 7.03**
		5	(30.74)	(29.11) ± 6.46	1.07	(0.48) ± 6.16	3.01	**6.16 ± 6.11**
	±50%	1	(25.18)	(21.28) ± 6.78	2.86	2.84 ± 7.28	3.60	**3.70 ± 6.80**
		2	(30.08)	(27.67) ± 7.46	2.54	**5.01 ± 6.58**	0.05	(3.12) ± 6.81
		3	(29.68)	(28.15) ± 6.28	(1.86)	(2.76) ± 6.05	2.56	**0.41 ± 6.51**
		4	(22.19)	(19.39) ± 7.03	(1.63)	(6.14) ± 7.21	8.59	**2.57 ± 6.71**
		5	(29.50)	(26.77) ± 6.59	(1.24)	(2.01) ± 6.61	0.85	**0.34 ± 6.49**
D2	±0%	1	(41.55)	(41.42) ± 4.48	11.98	3.37 ± 4.63	47.15	**40.11 ± 4.64**
	±10%	1	(41.86)	(42.54) ± 4.83	10.87	0.28 ± 4.28	46.67	**38.35 ± 4.67**
		2	(41.64)	(42.89) ± 5.01	9.21	2.46 ± 4.65	44.96	**37.10 ± 5.11**
		3	(40.74)	(39.57) ± 4.48	11.95	2.35 ± 4.37	46.78	**36.83 ± 4.41**
		4	(40.53)	(39.33) ± 5.30	14.25	6.57 ± 4.73	49.85	**41.71 ± 4.71**
		5	(41.13)	(43.51) ± 4.69	11.35	4.49 ± 4.42	46.96	**36.26 ± 4.86**
	±25%	1	(42.38)	(38.89) ± 4.53	8.61	5.22 ± 4.47	43.29	**36.82 ± 4.64**
		2	(42.19)	(41.64) ± 4.74	3.81	(5.31) ± 4.37	40.52	**30.70 ± 4.81**
		3	(39.51)	(35.40) ± 4.64	11.26	6.05 ± 4.09	44.83	**36.51 ± 4.54**
		4	(38.22)	(41.91) ± 4.56	16.84	5.13 ± 4.36	51.57	**38.38 ± 4.63**
		5	(40.54)	(40.67) ± 4.70	8.60	3.14 ± 4.71	45.52	**39.64 ± 4.67**
	±50%	1	(41.45)	(42.16) ± 4.44	4.57	(0.30) ± 4.47	43.15	**35.68 ± 4.64**
		2	(44.00)	(41.86) ± 4.67	(7.38)	(11.45) ± 4.56	28.08	**24.38 ± 4.28**
		3	(36.06)	(37.97) ± 4.62	6.57	(2.31) ± 4.17	39.04	**27.10 ± 4.94**
		4	(34.68)	(36.15) ± 4.47	13.88	5.35 ± 4.44	46.45	**37.15 ± 4.41**
		5	(37.18)	(36.32) ± 4.37	2.76	(0.80) ± 4.46	41.71	**34.45 ± 4.38**

Table 5 Profit Evaluation, with Booking Simulations, of SD-FAM Solutions Obtained with a 5hr CPU Time Limit. Profit changes are reported in million \$ per 365 days relative to ISD-FAM. Negative numbers are reported in parentheses. The baseline expected profit for ISD-FAM is \$1.294 billion for D1 and \$2.360 billion for D2 per 365 days.

7. Concluding Remarks and Future Directions

In this paper, we extend the work by Barnhart et al. (2009) along multiple theoretical, computational and practical dimensions to provide a tractable and effective decomposition-based solution approach for the choice-based integrated airline schedule design and fleet assignment problem. Our method relies on a partition of the flight network into smaller subnetworks, guided by new approximation analyses and a novel fare split linear program to reliably measure the approximation quality and optimally allocate fares of shared products to different subnetworks. The formulation is strengthened by selective use of composite variables via a new algorithmic approach to represent SD-FAM decisions of small subnetworks. We conduct comprehensive numerical experiments using two realistically-sized airline instances to demonstrate the effectiveness of our approach. Under a simulated booking environment, we show that the SD-FAM decisions generated by our proposed approach deliver significant and robust profit improvements over all benchmark implementations and models in the literature.

One possible extension of this work is to integrate it with other airline operations such as maintenance routing and crew scheduling. This will create additional computational complexity — aircraft routing and crew scheduling often exhibit different types of network effects than schedule design and fleet assignment — and will therefore require new modeling approaches and analyses.

With enhanced tractability, further integration of choice-based passenger demand models with other airline planning processes (such as clean-slate timetable development including frequency and departure time selection) will become computationally possible. This enhanced tractability also facilitates the use of stochastic planning models that consider demand fluctuations (Sherali and Zhu 2008). With the ever-growing collection of passenger shopping datasets, booking behavior can be estimated increasing accurately using choice models. We believe that choice-based airline planning models represent a promising future research direction that can enable airlines to leverage a better understanding of customer demand for smarter data-driven planning decisions.

Acknowledgments

The authors want to thank Xiaodong Luo, Richard Ratliff, Sergey Shebalov and Xiaoqing Sun from Sabre Research and Sabre Airline Solutions for generously sharing their expertise, engaging in helpful discussions, and providing the testing data sets. The authors would also like to thank the reviewers and the associate editor for their detailed comments and suggestions that improved the quality of the paper. The first author was supported by the UPS Doctoral Fellowship from the MIT Center for Transportation and Logistics.

References

Abdallah, Tarek, Gustavo Vulcano. 2021. Demand estimation under the multinomial logit model from sales transaction data. *Manufacturing & Service Operations Management* **23**(5) 1196–1216.

Barnhart, Cynthia, Amr Farahat, Manoj Lohatepanont. 2009. Airline fleet assignment with enhanced revenue modeling. *Operations Research* **57**(1) 231–244.

Barnhart, Cynthia, Timothy S Kniker, Manoj Lohatepanont. 2002. Itinerary-based airline fleet assignment. *Transportation Science* **36**(2) 199–217.

Bertsimas, Dimitris, John N Tsitsiklis. 1997. *Introduction to linear optimization*, vol. 6. Athena Scientific Belmont, MA.

Blanchet, Jose, Guillermo Gallego, Vineet Goyal. 2016. A markov chain approximation to choice modeling. *Operations Research* **64**(4) 886–905.

Bradley, Stephen, Arnoldo Hax, Thomas Magnanti. 1977. Applied mathematical programming .

Bront, Juan José Miranda, Isabel Méndez-Díaz, Gustavo Vulcano. 2009. A column generation algorithm for choice-based network revenue management. *Operations Research* **57**(3) 769–784.

Bureau of Transportation Statistics. 2021a. Transtats. `https://www.transtats.bts.gov/Data_Elements.aspx?Data=5`. Accessed: 2021-08-23.

Bureau of Transportation Statistics. 2021b. Transtats. `https://www.transtats.bts.gov/tables.asp?table_id=311`. Accessed: 2021-08-23.

Bureau of Transportation Statistics. 2021c. U.S. air carrier traffic statistics. `https://www.transtats.bts.gov/TRAFFIC/`. Accessed:2021-08-23.

Cirium. 2019. FM fare proration methodology. `https://helpdesk.cirium.com/hc/en-us/articles/360025305652-FM-Fare-Proration-Methodology`.

Clark, Paul. 2017. *Buying the big jets: fleet planning for airlines*. Taylor & Francis.

Dai, JG, Weijun Ding, Anton Kleywegt, Xinchang Wang, Yi Zhang. 2014. Choice based revenue management for parallel flights. Working paper.

Desaulniers, Guy, Jacques Desrosiers, Yvan Dumas, Marius M Solomon, François Soumis. 1997. Daily aircraft routing and scheduling. *Management Science* **43**(6) 841–855.

Di, Mingyang, Diego Klabjan, Sergey Shebalov. 2016. Solving attractiveness-based stochastic fleeting by mapreduce. Working paper.

Dumas, Jonathan, François Soumis. 2008. Passenger flow model for airline networks. *Transportation Science* **42**(2) 197–207.

Gallego, G., G. Iyengar, R. Phillips, A. Dubey. 2004. Managing flexible products on a network. Technical Report TR-2004-01. Dept. of Industrial Engineering and Operations Research, Columbia University.

Gallego, Guillermo, Richard Ratliff, Sergey Shebalov. 2015. A general attraction model and sales-based linear program for network revenue management under customer choice. *Operations Research* **63**(1) 212–232.

Gallego, Guillermo, Huseyin Topaloglu. 2019. *Revenue management and pricing analytics*, vol. 209. Springer.

Hane, Christopher A, Cynthia Barnhart, Ellis L Johnson, Roy E Marsten, George L Nemhauser, Gabriele Sigismondi. 1995. The fleet assignment problem: Solving a large-scale integer program. *Mathematical Programming* **70**(1) 211–232.

Jacobs, Timothy L, Barry C Smith, Ellis L Johnson. 2008. Incorporating network flow effects into the airline fleet assignment process. *Transportation Science* **42**(4) 514–529.

Kunnumkal, Sumit, Kalyan Talluri. 2019. Choice network revenue management based on new tractable approximations. *Transportation Science* **53**(6) 1591–1608.

Liu, Qian, Garrett van Ryzin. 2008. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management* **10**(2) 288–310.

Lohatepanont, Manoj, Cynthia Barnhart. 2004. Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science* **38**(1) 19–32.

OAG. 2020. Coronavirus: understand the airline schedule changes and manage the impact. `https://www.oag.com/coronavirus-airline-schedules-data`.

Sherali, Hanif D, Ki-Hwan Bae, Mohamed Haouari. 2010. Integrated airline schedule design and fleet assignment: Polyhedral analysis and benders' decomposition approach. *INFORMS Journal on Computing* **22**(4) 500–513.

Sherali, Hanif D, Xiaomei Zhu. 2008. Two-stage fleet assignment model considering stochastic passenger demands. *Operations Research* **56**(2) 383–399.

Soumis, François, Anna Nagurney. 1993. A stochastic, multiclass airline network equilibrium model. *Operations Research* **41**(4) 721–730.

Statista. 2021. Ebit margin of airlines worldwide 2010-2021. URL `https://www.statista.com/statistics/225856/ebit-margin-of-commercial-airlines-worldwide/`.

Subramanian, Radhika, Richard P Scheff, John D Quillinan, D Steve Wiper, Roy E Marsten. 1994. Coldstart: fleet assignment at delta air lines. *Interfaces* **24**(1) 104–120.

Talluri, Kalyan. 2014. New formulations for choice network revenue management. *INFORMS Journal on Computing* **26**(2) 401–413.

Talluri, Kalyan, Garrett van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* **50**(1) 15–33.

Tarjan, Robert. 1972. Depth-first search and linear graph algorithms. *SIAM Journal On Computing* **1**(2) 146–160.

Vulcano, Gustavo, Garrett van Ryzin, Wassim Chaar. 2010. Choice-based revenue management: An empirical study of estimation and optimization. *Manufacturing & Service Operations Management* **12**(3) 371–392.

Vulcano, Gustavo, Garrett van Ryzin, Richard Ratliff. 2012. Estimating primary demand for substitutable products from sales transaction data. *Operations Research* **60**(2) 313–334.

Wang, Di, Diego Klabjan, Sergey Shebalov. 2014. Attractiveness-based airline network models with embedded spill and recapture. *Journal of Airline and Airport Management* **4**(1) 1–25.

Wei, Keji, Vikrant Vaze, Alexandre Jacquillat. 2020. Airline timetable development and fleet assignment incorporating passenger choice. *Transportation Science* **54**(1) 139–163.

Williamson, Elizabeth Louise. 1992. Airline network seat inventory control: methodologies and revenue impacts. Ph.D. thesis, Massachusetts Institute of Technology.

Appendix of Choice-Based Airline Schedule Design and Fleet Assignment: A Decomposition Approach

Appendix A: Proofs of Technical Results

Proof of Proposition 1. The first statement of the proposition can be seen from the following equivalent formulation of $r(\mathbf{x};\mathbf{h}(\mathcal{P}^k))$ where redundant variables and constraints have been removed.

$$r(\mathbf{x};\mathbf{h}(\mathcal{P}^k)) = \max_{\mathbf{s}} \quad \sum_{p\in\mathcal{P}^k} h_p s_p \tag{37}$$

$$\text{s.t.} \quad \sum_{p\in\mathcal{P}^k} \delta_{l,p} s_p \le \sum_{f\in\mathcal{F}(l)} \mathrm{CAP}_f x_{l,f},\ \forall l\in\mathcal{L}^k, \tag{38}$$

$$\sum_{p\in\mathcal{P}(m)} \frac{w_p-\tilde{w}_p}{w_p} s_p + \frac{w_m+\sum_{p\in\mathcal{P}(m)}\tilde{w}_p}{w_m} s_m = \Lambda_m,\ \forall m\in\mathcal{M}^k, \tag{39}$$

$$\frac{s_p}{w_p} - \frac{s_m}{w_m} \le 0, \quad \forall m\in\mathcal{M}^k,\ \forall p\in\mathcal{P}^k, \tag{40}$$

$$s_m, s_p \ge 0, \quad \forall m\in\mathcal{M}^k,\ \forall p\in\mathcal{P}^k. \tag{41}$$

It is clear that for any x' such that $x'_{l,f} = x_{l,f}, \forall l\in\mathcal{L}^k, f\in\mathcal{F}(l)$, we have $r(\mathbf{x};\mathbf{h}(\mathcal{P}^k)) = r(\mathbf{x}';\mathbf{h}(\mathcal{P}^k))$ since all $l\notin\mathcal{L}^k$ does not appear in constraints (38).

To prove the second statement, since $\cup_{k=1}^K\mathcal{L}^k=\mathcal{L}, \cup_{k=1}^K\mathcal{P}^k=\mathcal{P}, \cup_{k=1}^K\mathcal{M}^k=\mathcal{M}$ and $\mathcal{L}^i\cap\mathcal{L}^j=\mathcal{P}^i\cap\mathcal{P}^j=\mathcal{M}^i\cap\mathcal{M}^j=\emptyset,\ \forall i,j\in\{1,\cdots,K\}, i\ne j$, we observe that $r(\mathbf{x};\mathbf{h}(\mathcal{P}^k)), \forall k\in\{1,\cdots,K\}$ is a perfect decomposition of $r(\mathbf{x};\mathbf{h})$ by splitting independent constraints, variables, and terms in the objective function into separate problems. This proves that $\sum_{k=1}^K r(\mathbf{x};\mathbf{h}(\mathcal{P}^k)) = r(\mathbf{x};\mathbf{h})$. □

Proof of Corollary 1. For any feasible solution $(\mathbf{x},\{\mathbf{x}^k\}_{k\in\{1,\cdots,K\}},\mathbf{y})$ of $\mathsf{CSD\text{-}FAM}(\Pi,\ \mathcal{H})$, it is clear that $(\mathbf{x},\mathbf{y})$ is a feasible solution of $\mathsf{CSD\text{-}FAM}$ with the same objective value because $r(\mathbf{x};\mathbf{h}) = \sum_{k=1} r(\mathbf{x}^k;\mathbf{h}^k)$ by Proposition 1. Moreover, for any feasible solution $(\mathbf{x},\mathbf{y})$ of $\mathsf{CSD\text{-}FAM}$, consider $\mathbf{x}^k$ constructed according to (13) as follows:

$$x^k_{l,f} = \begin{cases} x_{l,f}, & \forall l\in\mathcal{L}^k, \forall f\in\mathcal{F}(l),\\ 1, & \forall l\in\mathcal{L}\setminus\mathcal{L}^k,\ f=\arg\max_{f'\in\mathcal{F}(l)}\mathrm{CAP}_{f'},\\ 0, & \text{otherwise.}\end{cases}$$

It can be seen that $(\mathbf{x},\{\mathbf{x}^k\}_{k\in\{1,\cdots,K\}},\mathbf{y})$ is a feasible solution of $\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})$, and again, by Proposition 1, they have the same objective values. All in all, this establishes a one-to-one mapping from any feasible solution of $\mathsf{CSD\text{-}FAM}$ to a feasible solution of $\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})$ with the same objective value, and thus proves the statement of the equivalence of the two formulations. □

Proof of Theorem 1. Given any partition, $V_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})} = V_{\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c,\Pi_t,\mathcal{H})} = V_{\mathsf{S\text{-}CSD\text{-}FAM}(\Pi,\mathcal{H})}$ is straightforward to see as the set of feasible fleet assignments and their corresponding profit contributions are the same across these three subnetwork-based formulations. To prove $\tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\mathcal{H})} \ge \tilde{V}_{\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c,\Pi_t,\mathcal{H})} \ge \tilde{V}_{\mathsf{S\text{-}CSD\text{-}FAM}(\Pi,\mathcal{H})}$, we first introduce an intermediate problem, $\mathsf{CSD\text{-}FAM}(\Pi_c,\Pi_t,\mathcal{H})\text{-}\mathsf{CONV}$, to facilitate the proof. The optimal value of the continuous relaxation of this intermediate problem bridges together the optimal values of the continuous relaxations of the formulations we are interested in comparing. We define the set of feasible (binary) integer solutions corresponding to fleet assignment and schedule design decisions for flights in $\mathcal{L}^k$, $\mathcal{X}_{\mathcal{L}^k} \triangleq \left\{ \{x_{l,f}\}_{l\in\mathcal{L}^k, f\in\mathcal{F}(l)} \in \{0,1\} \mid \sum_{f\in\mathcal{F}(l)} x_{l,f} = 1, \forall l \in \mathcal{L}^k \cap \mathcal{L}_m, \sum_{f\in\mathcal{F}(l)} x_{l,f} \le 1, \forall l \in \mathcal{L}^k \cap \mathcal{L}_o \right\}$. Define $\text{CONV}(\mathcal{X}_{\mathcal{L}^k})$ as the convex hull generated by all the (binary) integer solutions in $\mathcal{X}_{\mathcal{L}^k}$. We now define formulation $\textsf{CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})\textsf{-CONV}$ below. Note that this formulation is nothing but the formulation of $\textsf{CSD-FAM}(\Pi, \mathcal{H})$ except for an added constraint (43) whose purpose is to tighten its continuous relaxation.

$(\textsf{CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})\textsf{-CONV})$

$$V_{\textsf{CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})\textsf{-CONV}} = \max \quad \sum_{k=1}^{K} r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{l\in\mathcal{L}} \sum_{f\in\mathcal{F}(l)} c_{l,f} x_{l,f} \tag{42}$$

$$\text{s.t.} \quad \{x_{l,f}\}_{l\in\mathcal{L}^k, f\in\mathcal{F}(l)} \in \text{CONV}(\mathcal{X}_{\mathcal{L}^k}),\ \forall k : \mathcal{L}^k \in \Pi_c, \tag{43}$$

$$\mathbf{x}, \mathbf{y} \text{ satisfies (2) - (7)},$$

$$\mathbf{x}^k \text{ satisfies (13)},\ \forall k \in \{1, \cdots, K\}.$$

Define $V_{\textsf{CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})\textsf{-CONV}}$ and $\tilde{V}_{\textsf{CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})\textsf{-CONV}}$ as the optimal values of $\textsf{CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})$-CONV and its continuous relaxation, respectively. We first claim that $\tilde{V}_{\textsf{CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})\textsf{-CONV}} \le \tilde{V}_{\textsf{CSD-FAM}(\Pi,\mathcal{H})}$ for any split (Π_c, Π_t) of the same partition Π. This is obvious since the formulations of these two continuous relaxations are identical except for the extra constraint (43) in the continuous relaxation of CSD-$\textsf{FAM}(\Pi_c, \Pi_t, \mathcal{H})$-CONV.

We then claim that $\tilde{V}_{\textsf{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} \le \tilde{V}_{\textsf{CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})\textsf{-CONV}}$. Recall the definition of the parameters $\kappa^j_{l,f}$: $\kappa^j_{l,f} = 1$ if flight l is assigned with fleet type f in variable ω_j, and is zero otherwise. For any feasible solution $(\omega, \mathbf{x}, \{\mathbf{x}^k\}_{k:\mathcal{L}^k\in\Pi_t}, \mathbf{y})$ of the continuous relaxation of $\textsf{S-CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})$, construct fleet assignment variables $\mathbf{x}' = \{x'_{l,f}\}_{l\in\mathcal{L}, f\in\mathcal{F}(l)}$ such that:

$$x'_{l,f} = \begin{cases} \sum_{k:\mathcal{L}^k\in\Pi_c} \sum_{j\in\mathcal{X}^k : l\in\mathcal{L}^k} \omega_j \kappa^j_{l,f},\ \forall l \in \mathcal{L}(\Pi_c), \forall f \in \mathcal{F}(l), \\ x_{l,f},\ \forall l \in \mathcal{L}(\Pi_t), \forall f \in \mathcal{F}(l). \end{cases} \tag{44}$$

In addition, for any $k \in \{1, \cdots, K\}$, construct local fleet assignment variables $\mathbf{x}'^k = \{x'^k_{l,f}\}_{l\in\mathcal{L}, f\in\mathcal{F}(l)}$ according to (13),

$$x'^k_{l,f} = \begin{cases} x'_{l,f}, & \forall l \in \mathcal{L}^k, \forall f \in \mathcal{F}(l), \\ 1, & \forall l \in \mathcal{L} \setminus \mathcal{L}^k,\ f = \arg\max_{f'\in\mathcal{F}(l)} \text{CAP}_{f'}, \\ 0, & \text{otherwise.} \end{cases} \tag{45}$$

We show that $(\mathbf{x}', \{\mathbf{x}'^k\}_{k\in\{1,\cdots,K\}}, \mathbf{y})$ is a feasible solution of the continuous relaxation of CSD-$\textsf{FAM}(\Pi_c, \Pi_t, \mathcal{H})$-CONV. It is clear that $(\mathbf{x}', \{\mathbf{x}'^k\}_{k\in\{1,\cdots,K\}}, \mathbf{y})$ satisfies (2) - (6) and (13), and $x'_{l,f} \in [0,1]$. To see that $(\mathbf{x}', \{\mathbf{x}'^k\}_{k\in\{1,\cdots,K\}}, \mathbf{y})$ satisfies (43), it can be seen from (44) and (24) that $\{\omega_j\}_{j\in\mathcal{X}^k}$ can be simply interpreted as the weights of the convex combination of all the binary assignment solutions contained in $\mathcal{X}_{\mathcal{L}^k}$ for all k such that $\mathcal{L}^k \in \Pi_c$.

Next we show that the objective value of the continuous relaxation of $\textsf{CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})$-CONV evaluated at $(\mathbf{x}', \{\mathbf{x}'^k\}_{k\in\{1,\cdots,K\}}, \mathbf{y})$ is greater than or equal to the objective value of the continuous relaxation of $\textsf{S-CSD-FAM}(\Pi_c, \Pi_t, \mathcal{H})$ evaluated at $(\omega, \mathbf{x}, \{\mathbf{x}^k\}_{k:\mathcal{L}^k\in\Pi_t}, \mathbf{y})$. To do that, for each subnetwork k such that $\mathcal{L}^k \in \Pi_c$, we derive an equivalent construction for $x'^k_{l,f}$. For any $j \in \mathcal{X}^k$, let $\overline{\kappa}^j = \{\overline{\kappa}^j_{l,f}\}_{l\in\mathcal{L}, f\in\mathcal{F}(l)}$ with $\overline{\kappa}^j_{l,f} =$

$\kappa^j_{l,f}$ if $l \in \mathcal{L}^k$; for $l \neq \mathcal{L}^k$, $\overline{\kappa}^j_{l,f} = 1$ if $f = \arg\max_{f' \in \mathcal{F}(l)} \mathrm{CAP}_{f'}$ and $\overline{\kappa}^j_{l,f} = 0$ otherwise. It can be seen that $x'^k_{l,f} = \sum_{j \in \mathcal{X}^k} \omega_j \overline{\kappa}^j_{l,f}$ also satisfies (45). This yields,

$$
\begin{aligned}
&\sum_{k=1}^{K} r(\mathbf{x}'^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} \\
= &\sum_{k:\mathcal{L}^k \in \Pi_c} r(\mathbf{x}'^k; \mathbf{h}^k) + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}'^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} \\
= &\sum_{k:\mathcal{L}^k \in \Pi_c} r\left(\sum_{j \in \mathcal{X}^k} \omega_j \overline{\kappa}^j_{l,f}; \mathbf{h}^k\right) + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}'^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} \\
\geq &\sum_{k:\mathcal{L}^k \in \Pi_c} \left(\sum_{j \in \mathcal{X}^k} \omega_j \cdot r\left(\overline{\kappa}^j; \mathbf{h}^k\right)\right) + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}'^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} \\
= &\sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{j \in \mathcal{X}^k} \omega_j r_j - \sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}'^k; \mathbf{h}^k) - \sum_{k:\mathcal{L}^k \in \Pi_t} \sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} x'_{l,f} \\
= &\sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{j \in \mathcal{X}^k} \omega_j r_j - \sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{j \in \mathcal{X}^k} \omega_j \left(\sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} \overline{\kappa}^j_{l,f}\right) + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{k:\mathcal{L}^k \in \Pi_t} \sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \\
= &\sum_{k:\mathcal{L}^k \in \Pi_c} \sum_{j \in \mathcal{X}^k} (r_j - c_j)\omega_j + \sum_{k:\mathcal{L}^k \in \Pi_t} r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{k:\mathcal{L}^k \in \Pi_t} \sum_{l \in \mathcal{L}^k} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f}.
\end{aligned}
$$

The first equality trivially holds by observing that each subnetwork is either in Π_c or in Π_t. The second equality holds by applying the alternative definition of $x'^k_{l,f}$ for subnetwork k such that $\mathcal{L}^k \in \Pi_c$. The first inequality holds by concavity of the revenue function $r(\mathbf{x}; \mathbf{h})$ in $\mathbf{x}$ by standard results in linear programming sensitivity analysis (see e.g., Bertsimas and Tsitsiklis 1997). The fourth equality holds by applying (44) and (45). This proves that for any feasible solution of the continuous relaxation of $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$, we can construct a feasible solution of the continuous relaxation of $\mathsf{CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})\mathsf{\text{-}CONV}$ with equal or larger objective values. This proves that $\tilde{V}_{\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c,\,\Pi_t,\,\mathcal{H})} \leq \tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi_c,\,\Pi_t,\,\mathcal{H})\mathsf{\text{-}CONV}} \leq \tilde{V}_{\mathsf{CSD\text{-}FAM}(\Pi,\,\mathcal{H})}$.

For any two splits $(\Pi_c, \Pi_t), (\Pi'_c, \Pi'_t)$ for the same flight partition Π with $\Pi_c \supset \Pi'_c$ and $\Pi_t \subset \Pi'_t$, and for any feasible solution $(\omega, \mathbf{x}, \{\mathbf{x}^k\}_{k:\mathcal{L}^k \in \Pi_t}, \mathbf{y})$ of the continuous relaxation of $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$, construct the following fleet assignment solution $\mathbf{x}'$ for all flights in $\mathcal{L}(\Pi'_t)$.

$$
x'_{l,f} = \begin{cases} \sum_{k:\mathcal{L}^k \in \Pi'_t \setminus \Pi_t} \sum_{j \in \mathcal{X}^k : l \in \mathcal{L}^k} \omega_j \kappa^j_{l,f}, \;\; \forall l \in \mathcal{L}(\Pi'_t \setminus \Pi_t), \forall f \in \mathcal{F}(l), \\ x_{l,f}, \;\; \forall l \in \mathcal{L}(\Pi_t), \forall f \in \mathcal{F}(l). \end{cases} \tag{46}
$$

In addition, for any subnetwork k such that $\mathcal{L}^k \in \Pi'_t$, construct local fleet assignment variables $\mathbf{x}'^k = \{x'^k_{l,f}\}_{l \in \mathcal{L}, f \in \mathcal{F}(l)}$ according to (13),

$$
x'^k_{l,f} = \begin{cases} x'_{l,f}, & \forall l \in \mathcal{L}^k, \forall f \in \mathcal{F}(l), \\ 1, & \forall l \in \mathcal{L} \setminus \mathcal{L}^k,\ f = \arg\max_{f' \in \mathcal{F}(l)} \mathrm{CAP}_{f'}, \\ 0, & \text{otherwise}. \end{cases} \tag{47}
$$

It can be seen that solution $(\{\omega_j\}_{j \in \mathcal{X}^k, k:\mathcal{L}^k \in \Pi'_c}, \mathbf{x}', \{\mathbf{x}'^k\}_{k:\mathcal{L}^k \in \Pi'_t}, \mathbf{y})$ is feasible to the continuous relaxation of $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi'_c, \Pi'_t, \mathcal{H})$. Next we show that the objective value of the continuous relaxation of $\mathsf{S\text{-}CSD\text{-}FAM}(\Pi_c, \Pi_t, \mathcal{H})$ evaluated at $(\omega, \mathbf{x}, \{\mathbf{x}^k\}_{k:\mathcal{L}^k \in \Pi_t}, \mathbf{y})$ is less than or equal to the objective value of the continuous relaxation of S-CSD-FAM$(\Pi'_c,\Pi'_t,\mathcal{H})$ evaluated at $(\{\omega_j\}_{j\in\mathcal{X}^k,k:\mathcal{L}^k\in\Pi'_c},\mathbf{x}',\{\mathbf{x}'^k\}_{k:\mathcal{L}^k\in\Pi'_t},\mathbf{y})$. To do that, for each subnetwork k such that $\mathcal{L}^k\in\Pi'_t\setminus\Pi_t$, we derive an equivalent construction of $x'^k_{l,f}$. For any $j\in\mathcal{X}^k$, again let $\overline{\kappa}^j=\{\overline{\kappa}^j_{l,f}\}_{l\in\mathcal{L},f\in\mathcal{F}(l)}$ with $\overline{\kappa}^j_{l,f}=\kappa^j_{l,f}$ if $l\in\mathcal{L}^k$; for $l\notin\mathcal{L}^k$, $\overline{\kappa}^j_{l,f}=1$ if $f=\arg\max_{f'\in\mathcal{F}(l)}\mathrm{CAP}_{f'}$ and $\overline{\kappa}^j_{l,f}=0$ otherwise. It can be seen that $x'^k_{l,f}=\sum_{j\in\mathcal{X}^k}\omega_j\overline{\kappa}^j_{l,f}$ also satisfies (47). We have,

$$
\begin{aligned}
&\sum_{k:\mathcal{L}^k\in\Pi_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x_{l,f}\\
&=\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi_c\setminus\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x_{l,f}\\
&=\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi'_t\setminus\Pi_t}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x'_{l,f}\\
&=\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi'_t\setminus\Pi_t}\left(\sum_{j\in\mathcal{X}^k}\omega_j\cdot r\left(\overline{\kappa}_j;\mathbf{h}^k\right)\right)+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t\cup(\Pi'_t\setminus\Pi_t)}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x'_{l,f}\\
&\le\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi'_t\setminus\Pi_t}r\left(\sum_{j\in\mathcal{X}^k}\omega_j\cdot\overline{\kappa}_j;\mathbf{h}^k\right)+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t\cup(\Pi'_t\setminus\Pi_t)}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x'_{l,f}\\
&=\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi'_t\setminus\Pi_t}r(\mathbf{x}'^k;\mathbf{h}^k)+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}'^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi'_t}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x'_{l,f}\\
&=\sum_{k:\mathcal{L}^k\in\Pi'_c}\sum_{j\in\mathcal{X}^k}(r_j-c_j)\omega_j+\sum_{k:\mathcal{L}^k\in\Pi_t}r(\mathbf{x}'^k;\mathbf{h}^k)-\sum_{k:\mathcal{L}^k\in\Pi_t}\sum_{l\in\mathcal{L}^k}\sum_{f\in\mathcal{F}(l)}c_{l,f}x'_{l,f}.
\end{aligned}
$$

Again, the first equality trivially holds by observing that each subnetwork in Π_c is either in Π'_c or not. The second equality holds because of (46) and because (Π_c,Π_t) and (Π'_c,Π'_t) form the same partition Π implying that $\Pi'_t\setminus\Pi_t=\Pi_c\setminus\Pi'_c$. The third equality holds because of (46) as well. The first inequality holds because of the concavity of the revenue function $r(\mathbf{x};\mathbf{h})$ in $\mathbf{x}$ by standard results in linear programming sensitivity analysis (see e.g., Bertsimas and Tsitsiklis 1997). The fourth equality holds because of the alternative construction for $\mathbf{x}'^k$ and because of (47).

This shows that for any feasible solution of the continuous relaxation of S-CSD-FAM$(\Pi_c,\Pi_t,\mathcal{H})$, one can construct a feasible solution of the continuous relaxation of S-CSD-FAM$(\Pi'_c,\Pi'_t,\mathcal{H})$ with an equal or higher objective value. This proves that $\tilde{V}_{\text{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}\le\tilde{V}_{\text{S-CSD-FAM}(\Pi'_c,\Pi'_t,\mathcal{H})}$ for any $\Pi_c\supset\Pi'_c$ and $\Pi_t\subset\Pi'_t$ for the same partition Π. Note that, since the problem S-CSD-FAM$(\Pi,\emptyset,\mathcal{H})$ is the same as problem S-CSD-FAM$(\Pi,\mathcal{H})$, we also have $\tilde{V}_{\text{S-CSD-FAM}(\Pi,\mathcal{H})}=\tilde{V}_{\text{S-CSD-FAM}(\Pi,\emptyset,\mathcal{H})}\le\tilde{V}_{\text{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}$ for any split (Π_c,Π_t) of Π. This completes the proof. □

Proof of Proposition 2. For any fleet assignment $\mathbf{x}$ and fare vector $\mathbf{h}$, we define $\mathbf{s}^*$ as the optimal solution of problem $r(\mathbf{x};\mathbf{h})$. $\mathbf{s}^*$ is also a feasible solution to all problems $r(\mathbf{x}^1;\mathbf{h}^1),\cdots,r(\mathbf{x}^K;\mathbf{h}^K)$ for any set of local fare vectors $\mathcal{H}=\{\mathbf{h}^k\}_{k\in\{1,\cdots,K\}}$. This is so because, going from $r(\mathbf{x};\mathbf{h})$ to $r(\mathbf{x}^k;\mathbf{h}^k)$, the only change that happens is in the right hand sides of constraints (9): they become larger or equal for $l\notin\mathcal{L}^k$ and stay the same for $l\in\mathcal{L}^k$. Thus we have,

$$\sum_{k=1}^{K}r(\mathbf{x}^k;\mathbf{h}^k)\ge\sum_{k=1}^{K}\left(\sum_{p\in\mathcal{P}}h^k_p s^*_p\right)=\sum_{p\in\mathcal{P}}\left(\sum_{k=1}^{K}h^k_p\right)s^*_p=\sum_{p\in\mathcal{P}}h_p s^*_p=r(\mathbf{x};\mathbf{h}).$$

The first inequality holds because $\mathbf{s}^*$ is a feasible solution for these K problems but it may not be the optimal solution, and hence it provides a lower or equal total revenue. The first equality is a re-arrangement of terms. The second equality holds because of Definition 2. The last equality holds due to the fact that s^* is the optimal solution of problem $r(\mathbf{x};\mathbf{h})$. $\square$

Proof of Proposition 3. For each local fare vector $\mathbf{h}^k$, we define a *restricted local fare vector* $\widehat{\mathbf{h}}^k$ and its complement $\widetilde{\mathbf{h}}^k$ as follows:

$$\widehat{h}_p^k = \begin{cases} h_p^k, & p \in \widehat{\mathcal{P}} \\ 0, & \text{otherwise} \end{cases}, \qquad \widetilde{h}_p^k = \begin{cases} 0, & p \in \widehat{\mathcal{P}} \\ h_p^k, & \text{otherwise.} \end{cases}$$

We have,

$$\begin{aligned}
&\sum_{k=1}^{K} r(\mathbf{x}^k;\mathbf{h}^k) - r(\mathbf{x};\mathbf{h}) \\
&\leq \sum_{k=1}^{K} r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) + \sum_{k=1}^{K} r(\mathbf{x}^k;\widetilde{\mathbf{h}}^k) - r(\mathbf{x};\mathbf{h}) && (48)\\
&= \sum_{k=1}^{K} r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) + r(\mathbf{x};\sum_{k=1}^{K}\widetilde{\mathbf{h}}^k) - r(\mathbf{x};\mathbf{h}) && (49)\\
&\leq \sum_{k=1}^{K} r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) && (50)\\
&\leq \sum_{k=1}^{K} \sum_{p\in\widehat{P}} \widehat{h}_p^k \overline{s}_p && (51)\\
&= \sum_{p\in\widehat{P}} \left(\sum_{k=1}^{K} h_p^k\right) \overline{s}_p && (52)\\
&= \sum_{p\in\widehat{P}} h_p \overline{s}_p. && (53)
\end{aligned}$$

One can verify that inequality (48) holds because $r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) + r(\mathbf{x}^k;\widetilde{\mathbf{h}}^k) \geq r(\mathbf{x}^k;\mathbf{h}^k)$. To see this, note that the optimal solution s^* of problem $r(\mathbf{x}^k;\mathbf{h}^k)$ is feasible for both problems $r(\mathbf{x}^k;\widehat{\mathbf{h}}^k)$ and $r(\mathbf{x}^k;\widetilde{\mathbf{h}}^k)$ (since all three of them share the same set of constraints). Since $\widehat{\mathbf{h}}^k + \widetilde{\mathbf{h}}^k = \mathbf{h}^k$, we have $r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) + r(\mathbf{x}^k;\widetilde{\mathbf{h}}^k) \geq \sum_{p\in\mathcal{P}^k}(\widehat{h}_p^k + \widetilde{h}_p^k)s_p^* = \sum_{p\in\mathcal{P}^k} h_p^k s_p^* = r(\mathbf{x}^k;\mathbf{h}^k)$. Equality (49) follows directly from Proposition 1 by observing that $\mathbf{h}(\mathcal{P}^k \setminus \widehat{\mathcal{P}}) = \widetilde{\mathbf{h}}^k$ where $\mathbf{h}(\cdot)$ is the projection of a fare vector onto a product set as defined in Proposition 1; and $\Pi = \{\mathcal{L}^1, \cdots, \mathcal{L}^K\}$ is an independent partition for a dependency graph with product set $\mathcal{P} \setminus \widehat{\mathcal{P}}$ and market set $\mathcal{M} \setminus \mathcal{M}(\widehat{\mathcal{P}})$. Inequality (50) holds because $\sum_{k=1}^{K} \widetilde{\mathbf{h}}^k \leq \mathbf{h}$ and $r(\mathbf{x};\mathbf{h})$ is non-decreasing in $\mathbf{h}$. Inequality (51) is correct because $r(\mathbf{x}^k;\widehat{\mathbf{h}}^k) = \sum_{p\in\widehat{P}} \widehat{h}_p^k s_p^{k*} \leq \sum_{p\in\widehat{P}} \widehat{h}_p^k \overline{s}_p$ where $\mathbf{s}^{k*}$ is the optimal solution of $r(\mathbf{x}^k;\widehat{\mathbf{h}}^k)$ and hence necessarily does not exceed the maximum possible sale values of the individual products. Equality (52) is a slight re-arrangement of terms combined with a direct application of the definition of $\widehat{h}_p^k$. Finally, the last equality (53) is true due to Definition 2 which states that $\sum_{k=1}^{K} \mathbf{h}^k = \mathbf{h}$ for any fare vector $\mathbf{h}$. $\square$

Proof of Proposition 4. We first give a simplified linear programming formulation of $r(\mathbf{x}^k;\mathbf{h}^k)$ below by removing redundant variables and constraints. $\mathcal{L}_+^k$ is defined as $\{l \notin L^k : \exists p \in \mathcal{P}^k \text{ s.t. } \delta_{l,p} = 1\}$, which is the set of flight legs not in $\mathcal{L}^k$ but related to products in $\mathcal{P}^k$. Note that here we define a new set of variables $\mathbf{s}^k$

representing product sales for each subnetwork $k \in \{1, \cdots, K\}$, as sales for the same product across different subnetworks have to be independently optimized to maximize total revenue within each subnetwork.

$$
\begin{aligned}
r(\mathbf{x}^k; \mathbf{h}^k) = \max_{\mathbf{s}^k} \quad & \sum_{p \in \mathcal{P}^k} h_p^k s_p^k \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}^k} \delta_{l,p} s_p^k \le \sum_{f \in \mathcal{F}(l)} \text{CAP}_f x_{l,f}^k, \ \forall l \in \mathcal{L}^k \cup \mathcal{L}_+^k, \\
& \sum_{p \in \mathcal{P}(m) \cap \mathcal{P}^k} \frac{w_p - \tilde{w}_p}{w_p} s_p^k + \frac{w_m + \sum_{p \in \mathcal{P}(m) \cap \mathcal{P}^k} \tilde{w}_p}{w_m} s_m^k \le \Lambda_m, \ \forall m \in \mathcal{M}^k, \\
& \frac{s_p^k}{w_p} - \frac{s_m^k}{w_m} \le 0, \ \forall m \in \mathcal{M}^k, \ \forall p \in \mathcal{P}(m) \cap \mathcal{P}^k, \\
& s_m^k, s_p^k \ge 0, \ \forall m \in \mathcal{M}^k, \ \forall p \in \mathcal{P}^k.
\end{aligned}
$$

Problem $\tilde{V}_{\textsf{CSD-FAM}}(\Pi, \mathcal{H}) = \max_{0 \le \mathbf{x} \le 1, \{\mathbf{x}^k\}_{k \in \{1,\cdots,K\}} \text{ satisfy (2)–(6),(13)}} \left(\sum_{k=1}^K r(\mathbf{x}^k; \mathbf{h}^k) - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \right)$ can then be written as the following linear program.

$$
\begin{aligned}
\max_{\mathbf{x}, \mathbf{y}, \{\mathbf{s}^k\}_{k \in \{1,\cdots,K\}}} \quad & \sum_{k=1}^K \sum_{p \in \mathcal{P}^k} h_p^k s_p^k - \sum_{l \in \mathcal{L}} \sum_{f \in \mathcal{F}(l)} c_{l,f} x_{l,f} \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}^k} \delta_{l,p} s_p^k \le \sum_{f \in \mathcal{F}(l)} \text{CAP}_f x_{l,f}, \ \forall l \in \mathcal{L}^k, \ k \in \{1, \cdots, K\}, && (u_l^k) \\
& \sum_{p \in \mathcal{P}^k} \delta_{l,p} s_p^k \le \max_{f \in \mathcal{F}(l)} \text{CAP}_f, \ \forall l \in \mathcal{L}_+^k, \ k \in \{1, \cdots, K\}, && (u_l'^k) \\
& \sum_{p \in \mathcal{P}(m) \cap \mathcal{P}^k} \frac{w_p - \tilde{w}_p}{w_p} s_p^k + \frac{w_m + \sum_{p \in \mathcal{P}(m) \cap \mathcal{P}^k} \tilde{w}_p}{w_m} s_m^k \le \Lambda_m, \ \forall m \in \mathcal{M}^k, \ k \in \{1, \cdots, K\}, && (t_m^k) \\
& \frac{s_p^k}{w_p} - \frac{s_m^k}{w_m} \le 0, \ \forall m \in \mathcal{M}^k, \ \forall p \in \mathcal{P}(m) \cap \mathcal{P}^k, \ k \in \{1, \cdots, K\}, && (a_p^k) \\
& \sum_{f \in \mathcal{F}(l)} x_{l,f} = 1, \quad \forall l \in \mathcal{L}_m, && (r_l) \\
& \sum_{f \in \mathcal{F}(l)} x_{l,f} \le 1, \quad \forall l \in \mathcal{L}_o, && (r_l') \\
& y_{v^-} + \sum_{l \in I(v)} x_{l,f} - y_v - \sum_{l \in O(v)} x_{l,f} = 0, \quad \forall v \in N_f, \ \forall f \in \mathcal{F}, && (q_{v,f}) \\
& \sum_{v \in T_N(f)} y_v + \sum_{l \in T_F(f)} x_{l,f} \le n_f, \quad \forall f \in \mathcal{F}, && (z_f) \\
& x_{l,f} \ge 0, \quad \forall f \in \mathcal{F}(l), \ \forall l \in \mathcal{L}, \\
& y_v \ge 0, \quad \forall v \in N_f, \ \forall f \in \mathcal{F}, \\
& s_m^k, s_p^k \ge 0, \ \forall m \in \mathcal{M}^k, \ \forall p \in \mathcal{P}^k, \ k \in \{1, \cdots, K\}.
\end{aligned}
$$

We take the dual of the above linear program. The corresponding dual variables are listed on the right hand sides of the constraints. This gives us the following formulation for problem $\min_{\sum_{k=1}^K \mathbf{h}^k = \mathbf{h}} \tilde{V}_{\textsf{CSD-FAM}}(\Pi, \mathcal{H})$. Let v^+ denote the successor node located at the head of the ground arc exiting node v.

$$
\min_{\substack{\mathbf{h}, \mathbf{u}, \mathbf{u}', \mathbf{t}, \\ \mathbf{w}, \mathbf{r}, \mathbf{r}', \mathbf{q}, \mathbf{z}}} \sum_{k=1}^K \left(\sum_{l \in \mathcal{L}_+^k} \left(\max_{f \in \mathcal{F}(l)} \text{CAP}_f \right) u_l'^k + \sum_{m \in \mathcal{M}^k} \Lambda_m t_m^k \right) + \sum_{l \in \mathcal{L}_m} r_l + \sum_{l \in \mathcal{L}_o} r_l' + \sum_{f \in \mathcal{F}} n_f z_f
$$

$$
\begin{aligned}
\text{s.t. } & \sum_{l \in \mathcal{L}^k} \delta_{l,p} u_l^k + \sum_{l \in \mathcal{L}_+^k} \delta_{l,p} u_l'^k + \sum_{m \in \mathcal{M}^k} \mathbf{1}(p \in \mathcal{P}(m)) \frac{w_p - \tilde{w}_p}{w_p} t_m^k + (1/w_p) a_p^k \geq h_p^k, \ \forall p \in \mathcal{P}^k, \forall k \in \{1, \cdots, K\}, \\
& \frac{w_m + \sum_{p \in \mathcal{P}(m) \cap \mathcal{P}^k} \tilde{w}_p}{w_m} t_m^k - \sum_{p \in \mathcal{P}^k} \mathbf{1}(p \in \mathcal{P}(m))(1/w_m) a_p^k \geq 0, \ \forall m \in \mathcal{M}^k, \forall k \in \{1, \cdots, K\}, \\
& -\mathrm{CAP}_f u_l^k + \mathbf{1}(l \in \mathcal{L}_m) r_l + \mathbf{1}(l \in \mathcal{L}_o) r_l' + \sum_{v \in N_f} \left(\mathbf{1}(l \in I(v)) - \mathbf{1}(l \in O(v)) \right) q_{v,f} + \mathbf{1}(l \in T_F(f)) z_f \geq -c_{l,f}, \\
& \qquad \forall l \in \mathcal{L}^k, \forall f \in \mathcal{F}(l), \forall k \in \{1, \cdots, K\}, \\
& q_{v^+,f} - q_{v,f} + \mathbf{1}(v \in T_N(f)) z_f \geq 0, \ \forall v \in N_f, \forall f \in \mathcal{F}, \\
& \sum_{k : p \in \mathcal{P}^k} h_p^k = h_p, \quad \forall p \in \mathcal{P}, \\
& h_p^k \geq 0, \quad \forall p \in \mathcal{P}^k, \forall k \in \{1, \cdots, K\}, \\
& \mathbf{u}^k, \mathbf{u}'^k, \mathbf{t}^k, \mathbf{a}^k, \mathbf{r}', \mathbf{z} \geq 0, \quad \forall k \in \{1, \cdots, K\}. \qquad \square
\end{aligned}
$$

Appendix B: Benchmark Formulations in Section 6.1

B.1. ISD-FAM Formulation

We present the formulation of one of the baseline approaches: ISD-FAM. All notation follows that in Section 3 except that we introduce d_p as the unconstrained demand of product p. Demand in ISD-FAM is tied to individual products, not markets. Moreover, demand for different products is independent. In our implementation, we set $d_p = \Big(w_p/(w_m + \sum_{p' \in \mathcal{P}(m)} w_{p'})\Big)\Lambda_m,\ p \in \mathcal{P}(m)$.

$$
\begin{aligned}
\text{(ISD-FAM)} \qquad \max \quad & \sum_{p\in\mathcal{P}} h_p s_p - \sum_{l\in\mathcal{L}} \sum_{f\in\mathcal{F}(l)} c_{l,f} x_{l,f} \\
\text{s.t.} \quad & \sum_{f\in\mathcal{F}(l)} x_{l,f} = 1, \quad \forall l \in \mathcal{L}_m, \\
& \sum_{f\in\mathcal{F}(l)} x_{l,f} \le 1, \quad \forall l \in \mathcal{L}_o, \\
& y_{v^-} + \sum_{l\in I(v)} x_{l,f} - y_v - \sum_{l\in O(v)} x_{l,f} = 0, \quad \forall f\in\mathcal{F},\ \forall v \in N_f, \\
& \sum_{v\in T_N(f)} y_v + \sum_{l\in T_F(f)} x_{l,f} \le n_f, \quad \forall f \in \mathcal{F}, \\
& \sum_{p\in\mathcal{P}} \delta_{l,p} s_p \le \sum_{f\in\mathcal{F}(l)} \mathrm{CAP}_f x_{l,f},\ \forall l \in \mathcal{L}, \\
& 0 \le s_p \le d_p, \quad \forall p \in \mathcal{P}, \\
& x_{l,f} \in \{0,1\}, \quad \forall l \in \mathcal{L},\ \forall f \in \mathcal{F}(l), \\
& y_v \ge 0, \quad \forall v \in N_f,\ \forall f \in \mathcal{F}.
\end{aligned}
$$

B.2. ISD-FAM-SR-ITIN Formulation

We present the formulation of one of the baseline approaches: ISD-FAM-SR-ITIN, i.e., ISD-FAM with spill and recapture, which is adapted from Lohatepanont and Barnhart (2004). In this formulation, fare products are aggregated into itineraries for tractability reasons (Lohatepanont and Barnhart 2004). We also denote by ISD-FAM-SR an alternative formulation where fare products are modeled explicitly and are not aggregated into itineraries. We solve ISD-FAM-SR separately and report its performance in Appendix C. ISD-FAM-SR is tremendously hard to solve due significantly increased number of variables. With a slight abuse of notation, let $\mathcal{P}$ now denote the set of itineraries. The unconstrained demand d_p of itinerary $p \in \mathcal{P}$ is the sum of unconstrained demand of all fare products belonging to that itinerary. The fare h_p of itinerary $p \in \mathcal{P}$ is the weighted average of fares of all products belonging to itinerary p, where the weight is the unconstrained demand of the product. Passengers' spill and recapture behaviors are modeled using a recapture rate b_p^r, which is a measure of the probability that a passenger spilled from itinerary p will accept itinerary r as an alternative. In Lohatepanont and Barnhart (2004), it is assumed to be calculated as follows,

$$
b_p^r = \begin{cases} 1, & \text{if } p = r; \\ q_r/(1 - Q_m + q_r), & \text{if } p \in \mathcal{P}(m) : p \ne r; \\ 0, & \text{otherwise.} \end{cases}
$$

Q_m represents the airline's unconstrained market share of the origin-destination market m to which itinerary p belongs. It is measured by the ratio of the sum of unconstrained demands of all itineraries of that airline belonging to the origin-destination market over the total demand of that market. q_r is the unconstrained share of itinerary r which is measured by the ratio of unconstrained demand of itinerary r over the total demand of the origin-destination market to which it belongs. Note that recapture rate is zero if r does not belong to $\mathcal{P}(m)$, the itineraries of origin-destination market m containing itinerary p.

Decision variables $t_p^r \geq 0$ denote the number of passengers redirected from itinerary p to r. For flight leg $l \in \mathcal{L}$, $\sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} t_p^r - \sum_{p\in\mathcal{P}} \delta_{l,p} t_p^p$ is the number of passengers who are spilled from their desired itinerary; $\sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} b_r^p t_r^p - \sum_{p\in\mathcal{P}} \delta_{l,p} b_p^p t_p^p$ is the number of passengers who are recaptured by the airline. Therefore, $\sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} t_p^r - \sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} b_r^p t_r^p$ is the number of passengers spilled minus those recaptured on flight leg l. We now formally present the formulation of ISD-FAM-SR-ITIN:

$$
\begin{aligned}
\text{(ISD-FAM-SR-ITIN)} \quad \max \quad & \sum_{p\in\mathcal{P}} h_p d_p - \sum_{p\in\mathcal{P}}\sum_{r\in\mathcal{P}} (h_p - b_p^r h_r) t_p^r - \sum_{l\in\mathcal{L}}\sum_{f\in\mathcal{F}(l)} c_{l,f} x_{l,f} \\
\text{s.t.} \quad & \sum_{f\in\mathcal{F}(l)} x_{l,f} = 1, \quad \forall l \in \mathcal{L}_m, \\
& \sum_{f\in\mathcal{F}(l)} x_{l,f} \leq 1, \quad \forall l \in \mathcal{L}_o, \\
& y_{v^-} + \sum_{l\in I(v)} x_{l,f} - y_v - \sum_{l\in O(v)} x_{l,f} = 0, \quad \forall f \in \mathcal{F},\ \forall v \in N_f, \\
& \sum_{v\in T_N(f)} y_v + \sum_{l\in T_F(f)} x_{l,f} \leq n_f, \quad \forall f \in \mathcal{F}, \\
& \sum_{p\in\mathcal{P}} \delta_{l,p} d_p - \sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} t_p^r + \sum_{r\in\mathcal{P}}\sum_{p\in\mathcal{P}} \delta_{l,p} b_r^p t_r^p \leq \sum_{f\in\mathcal{F}(l)} \mathrm{CAP}_f x_{l,f}, \ \forall l \in \mathcal{L}, \quad (54) \\
& \sum_{r\in\mathcal{P}} t_p^r \leq d_p, \quad \forall p \in \mathcal{P}, \quad (55) \\
& x_{l,f} \in \{0,1\}, \quad \forall l \in \mathcal{L},\ \forall f \in \mathcal{F}(l), \\
& y_v \geq 0, \quad \forall v \in N_f,\ \forall f \in \mathcal{F}, \\
& t_p^r \geq 0, \quad \forall p, r \in \mathcal{P}.
\end{aligned}
$$

The objective function maximizes the total profit, calculated by subtracting the sum of the lost revenue (spilled revenue less recaptured revenue) and the fleet operating cost from the maximum possible revenue. Most constraints are the standard fleet assignment constraints. Constraints (54) are the capacity constraints. The total sale on a flight leg is calculated by the maximum possible demand less the difference of spilled demand and recaptured demand. Constraints (55) restrict the total number of passengers spilled from itinerary p to the unconstrained demand for itinerary p. In the original paper by Lohatepanont and Barnhart (2004), a column and row generation method is proposed to solve ISD-FAM-SR-ITIN. In our numerical experiments, we found that the model can be solved well (it is solved to full optimality for instance D1 and around 2% optimality gap for instance D2 within 5 hours) without such a decomposition method due to much improved computational capabilities and solver performance.

B.3. S-CSD-FAM-BFM Formulation

We first outline the partition algorithm in Barnhart et al. (2009), which we used for one of the baseline formulations, S-CSD-FAM-BFM. The key difference between Algorithm 2 and Algorithm 1 is that Algorithm

Algorithm 2 Construct Partition Π_c (Barnhart et al. 2009)

initialize

$\Pi_c \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| \leq \mathsf{max\_vars}\}$

$\overline{\Pi}_c \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \Pi(\mathcal{G}(\mathcal{L},\mathcal{P},\mathcal{M})), |\mathcal{X}^k| > \mathsf{max\_vars}\}$

iter $\leftarrow 1$

while $\overline{\Pi}_c \neq \emptyset$ **do**

$\mathcal{P} \leftarrow \{p \in \mathcal{P} : h_p \overline{s}_p \geq \mathsf{revenue\_step} \times \text{iter}\}$

for $\mathcal{L}^k \in \overline{\Pi}_c$ **do**

$\overline{\Pi}_c \leftarrow \left(\overline{\Pi}_c \setminus \mathcal{L}^k\right) \cup \Pi\left(\mathcal{G}(\mathcal{L}^k,\mathcal{P},\mathcal{M})\right)$

end for

$\Pi_c \leftarrow \Pi_c \cup \{\mathcal{L}^k : \mathcal{L}^k \in \overline{\Pi}_c, |\mathcal{X}^k| \leq \mathsf{max\_vars}\}$

$\overline{\Pi}_c \leftarrow \{\mathcal{L}^k : \mathcal{L}^k \in \overline{\Pi}_c, |\mathcal{X}^k| > \mathsf{max\_vars}\}$

iter $\leftarrow$ iter $+ 1$

end while

return Π_c

1 is guided by a measure of approximation quality enabled by the fare split problem, which is entirely absent in Algorithm 2.

We select revenue_step to be \$1,000 and max_var to be $100,000$ (same as the one used in the SFAM4 run in Barnhart et al. 2009). Table 6 lists the resulting partitions used in S-CSD-FAM-BFM for the two instances.

Instance	# of subnetworks of size							
	1	2	3	4	5	6	7	Total
D1	96	17	6	0	4	2	0	125
D2	335	44	53	19	30	0	1	482

Table 6 **Subnetwork Partitions for S-CSD-FAM-BFM**

Appendix C: Additional Computational Experiments

C.1. 10 Hour Computation Time

We report the performance of approaches tested in Sections 6.1 and 6.2 with a computation time of 10 hours.

Inst.	Perturb.	Scenario	ISD-FAM-SR-ITIN		CSD-FAM		S-CSD-FAM($\Pi_c^1, \Pi_t^1, \mathcal{H}$)	
			ΔProfit(SBLP)	ΔProfit(Sim)	ΔProfit(SBLP)	ΔProfit(Sim)	ΔProfit(SBLP)	ΔProfit(Sim)
D1	±0%	1	(30.90)	(25.75) ± 6.22	5.22	**6.07 ± 6.36**	5.62	5.93 ± 6.16
	±10%	1	(29.92)	(27.44) ± 6.79	6.56	4.99 ± 6.39	6.65	**5.72 ± 6.47**
		2	(31.08)	(27.44) ± 6.51	4.60	**6.71 ± 6.77**	4.60	6.62 ± 6.96
		3	(30.76)	(24.69) ± 6.13	5.12	**5.96 ± 6.09**	5.34	5.23 ± 6.04
		4	(29.90)	(26.97) ± 6.20	5.58	5.16 ± 6.10	5.82	**7.11 ± 6.22**
		5	(31.13)	(29.32) ± 6.60	4.35	(1.44) ± 6.73	4.34	**5.01 ± 6.23**
	±25%	1	(28.14)	(23.49) ± 6.61	8.63	**8.58 ± 6.19**	8.19	4.80 ± 6.66
		2	(31.16)	(30.45) ± 5.97	3.73	**4.31 ± 6.20**	2.95	4.00 ± 5.90
		3	(30.86)	(24.18) ± 7.06	4.27	**7.66 ± 6.26**	4.25	6.52 ± 6.68
		4	(27.40)	(26.29) ± 7.53	6.61	4.98 ± 7.16	6.69	**5.53 ± 7.03**
		5	(30.74)	(29.11) ± 6.46	3.68	1.09 ± 6.58	3.01	**6.16 ± 6.11**
	±50%	1	(25.18)	(21.28) ± 6.78	3.27	3.12 ± 6.33	3.60	**3.70 ± 6.80**
		2	(30.08)	(27.67) ± 7.46	1.94	**0.16 ± 7.05**	0.05	(3.12) ± 6.81
		3	(29.68)	(28.15) ± 6.28	(1.23)	(0.89) ± 6.16	2.56	**0.41 ± 6.51**
		4	(22.19)	(19.39) ± 7.03	9.38	**6.31 ± 6.64**	8.59	2.57 ± 6.71
		5	(29.50)	(26.77) ± 6.59	3.36	**1.87 ± 6.71**	0.85	0.34 ± 6.49
D2	±0%	1	(15.70)	(18.25) ± 5.06	48.39	40.17 ± 4.84	53.44	**45.27 ± 4.51**
	±10%	1	(15.79)	(18.70) ± 4.52	48.34	36.64 ± 4.60	52.39	**41.47 ± 4.54**
		2	(15.80)	(17.40) ± 5.24	45.89	36.91 ± 4.78	51.15	**41.72 ± 5.38**
		3	(15.35)	(15.40) ± 4.48	48.33	37.90 ± 4.49	52.76	**44.93 ± 4.49**
		4	(14.57)	(11.39) ± 5.12	50.35	42.57 ± 5.04	55.59	**47.51 ± 4.90**
		5	(15.50)	(17.08) ± 5.00	48.51	38.98 ± 4.68	52.19	**40.09 ± 4.52**
	±25%	1	(16.00)	(12.54) ± 4.34	48.18	41.64 ± 4.46	49.90	**43.93 ± 4.59**
		2	(16.34)	(15.76) ± 4.93	41.20	33.09 ± 4.59	47.13	**39.44 ± 4.59**
		3	(14.92)	(12.64) ± 4.54	47.06	**42.29 ± 4.26**	50.45	42.17 ± 4.53
		4	(11.97)	(14.80) ± 4.55	52.58	39.25 ± 4.46	57.69	**44.28 ± 4.39**
		5	(15.02)	(14.20) ± 4.65	46.64	41.70 ± 4.81	49.04	**42.69 ± 4.61**
	±50%	1	(15.69)	(16.47) ± 4.75	46.88	**41.01 ± 4.30**	46.19	38.44 ± 4.56
		2	(18.93)	(17.40) ± 4.95	30.91	24.78 ± 4.71	37.07	**30.20 ± 4.78**
		3	(12.57)	(16.99) ± 4.78	42.28	**32.97 ± 4.59**	42.14	30.85 ± 4.40
		4	(8.95)	(9.15) ± 4.91	49.04	37.42 ± 4.65	53.23	**43.77 ± 4.66**
		5	(12.02)	(12.93) ± 4.86	41.60	35.90 ± 4.58	42.43	**36.90 ± 4.58**

Table 7 Profit Evaluation, with Booking Simulations, of SD-FAM Solutions Obtained with a 10hr CPU Time Limit. Profit changes relative to ISD-FAM are reported in million \$ per 365 days. Negative numbers are reported in parentheses. Baseline profit for ISD-FAM is \$1.294 billion for D1 and \$2.360 billion for D2 per 365 days.

Inst.	Model	Fare Split	Obj.	LP Relax.	CPU Time	MIP Gap	B&B Nodes	#Flights	ΔProfit
D1	ISD-FAM (indep. demand)	-	2.87	3.36	304	0.00%	2,186	122	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	2.26	2.57	10,503	0.00%	194,785	118	(30.90)
	CSD-FAM (choice based demand)	-	3.67	4.36	LIMIT	0.48%	293,764	126	5.22
	CSD-FAM-S (w/ symmetry constr.)	-	3.67	4.36	LIMIT	0.37%	266,934	126	5.53
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	3.93	3.94	1,324	0.00%	0	128	(10.14)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	3.75	4.44	LIMIT	0.50%	161,644	124	1.31
		opt	3.73	4.38	LIMIT	0.42%	175,345	126	5.22
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	3.75	4.31	LIMIT	0.21%	274,603	124	1.31
		opt	3.73	4.28	27,667	0.00%	213,118	126	5.22
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	4.00	4.64	17,952	0.00%	90,044	126	(4.52)
		opt	3.82	4.40	17,292	0.00%	170,836	126	5.62
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	4.00	4.44	1,837	0.00%	43,280	126	(4.52)
		opt	3.82	4.26	1,807	0.00%	39,768	126	**5.62**
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	4.17	4.83	8,111	0.00%	47,411	126	(30.33)
		opt	3.93	4.45	3,193	0.00%	82,840	128	1.39
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	4.17	4.23	25	0.00%	0	126	(30.33)
		opt	3.93	3.97	25	0.00%	0	128	1.39
D2	ISD-FAM (indep. demand)	-	4.25	4.50	LIMIT	0.02%	55,868	743	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	3.96	4.15	LIMIT	1.56%	15,887	729	15.70
	CSD-FAM (choice based demand)	-	6.74	7.25	LIMIT	4.50%	10,348	671	44.37
	CSD-FAM-S (w/ symmetry constr.)	-	6.62	7.13	LIMIT	4.87%	10,684	656	(2.04)
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	9.48	9.48	6,705	0.00%	299	775	(182.92)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	6.82	7.31	LIMIT	4.38%	9,173	681	39.11
		opt	6.81	7.27	LIMIT	4.04%	8,279	675	47.47
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	6.85	7.26	LIMIT	3.73%	9,324	687	42.25
		opt	6.79	7.22	LIMIT	4.05%	8,412	680	48.60
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	6.90	7.37	LIMIT	3.96%	9,180	681	49.55
		opt	6.87	7.30	LIMIT	3.61%	12,554	677	49.46
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	6.93	7.31	LIMIT	3.29%	16,380	677	47.51
		opt	6.89	7.24	LIMIT	3.25%	12,000	688	**53.44**
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	6.99	7.47	LIMIT	4.17%	9,996	688	24.77
		opt	6.96	7.35	LIMIT	3.24%	12,948	699	45.86
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	7.03	7.39	LIMIT	3.20%	10,021	689	37.00
		opt	6.97	7.28	LIMIT	2.64%	18,025	685	49.35

Table 8 Computational Results with 10hr CPU Time Limit. Profit changes relative to ISD-FAM are reported in million \$/365 days. Negative numbers are reported in parentheses. Solution times are reported in CPU seconds. Baseline profit for ISD-FAM is \$1.329 billion for D1 and \$2.419 billion for D2 per 365 days.

C.2. Performance of ISD-FAM-SR Formulation

Table 9 reports the computational performance of ISD-FAM-SR, an expanded version of the ISD-FAM-SR-ITIN formulation, where individual fare products on the same itinerary are modeled explicitly. ISD-FAM-SR turns out to be significantly harder to solve even compared to CSD-FAM. We test ISD-FAM-SR with CPU time limits of 5 hours, 10 hours and 1 week. The quality of the solutions does improve as we increase computation time and is better than that obtained from the aggregated formulation ISD-FAM-SR-ITIN in instance D1. However, because of the significantly increased computational complexity, even with 1 week of CPU time, it cannot outperform the solution of ISD-FAM in both instances.

Inst.	Model	Obj. Val.	LP Relax.	CPU Time	Solver Gap	B&B Nodes	ΔProfit
D1	ISD-FAM-SR	3.08	3.75	18,000	13.40%	30,082	(39.21)
		3.13	3.75	36,000	9.82%	37,297	(27.15)
		3.14	3.75	604,800	6.74%	1,212,948	(11.87)
D2	ISD-FAM-SR	0.64	5.68	18,000	780%	2,004	(2035.93)
		4.66	5.68	36,000	21.00%	5,863	(170.82)
		4.95	5.68	604,800	9.28%	30,245	(95.39)

Table 9 Computational Results with ISD-FAM-SR with 5hr, 10hr and 1 Week CPU Time Limits. Profit changes relative to ISD-FAM are reported in million \$/365 days. Negative numbers are reported in parentheses. Solution times are reported in CPU seconds.

C.3. Pure Fleet Assignment $\mathcal{L}_m = \mathcal{L}$

We report the performance of an additional set of experiments when $\mathcal{L}_m = \mathcal{L}$, i.e., all flights are mandatory and must be assigned a fleet type. Tables 10 and 11 are results corresponding to 5hr and 10hr CPU times.

First of all, we observe that despite the fact that the pure fleet assignment problem is easier to solve, the S-CSD-FAM formulation still provides an edge over CSD-FAM after 5 or 10 hours of computation. This demonstrates the high approximation quality our approach can provide — in fact, Table 11 shows that, for instance D1, S-CSD-FAM$(\Pi_c^0, \Pi_t^0, \mathcal{H})$ gives the *exact* optimal solution of CSD-FAM with only around half of the computation time. Secondly, as the problem becomes easier to solve, a smaller max_profit_ratio focusing a little bit more on optimality, rather than tractability, will generally result in a better performance compared to CSD-FAM. Lastly, the symmetry-inducing constraints used in Wei et al. (2020) can result in infeasibility of the model (instance D2) for the pure fleet assignment task.

Inst.	Model	Fare Split	Obj.	LP Relax.	CPU Time	MIP Gap	B&B Nodes	#Flights	ΔProfit
D1	ISD-FAM (indep. demand)	-	1.71	1.78	17	0.00%	68	180	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	1.07	1.14	242	0.00%	6,231	180	(13.47)
	CSD-FAM (choice based demand)	-	2.52	2.66	LIMIT	0.15%	65,021	180	1.57
	CSD-FAM-S (w/ symmetry constr.)	-	2.52	2.66	LIMIT	0.11%	63,041	180	1.57
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	2.86	2.86	1,105	0.00%	0	180	(15.16)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	2.61	2.75	LIMIT	0.33%	60,145	180	0.48
		opt	2.57	2.70	17,416	0.00%	78,591	180	**1.84**
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	2.61	2.74	LIMIT	0.15%	77,995	180	0.48
		opt	2.57	2.69	12,187	0.00%	51,884	180	**1.84**
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	2.86	2.99	10,020	0.00%	37,245	180	(7.83)
		opt	2.67	2.82	10,871	0.00%	54,685	180	(6.34)
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	2.86	2.94	724	0.00%	9,291	180	(7.83)
		opt	2.67	2.77	2,281	0.00%	26,341	180	(6.34)
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	3.03	3.17	5,648	0.00%	35,880	180	(35.33)
		opt	2.89	3.01	600	0.00%	15,544	180	(23.67)
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	3.03	3.04	16	0.00%	0	180	(35.33)
		opt	2.89	2.90	15	0.00%	0	180	(23.67)
D2	ISD-FAM (indep. demand)	-	4.16	4.13	361	0.00%	1,453	815	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	3.85	3.89	LIMIT	0.01%	43,010	815	(13.97)
	CSD-FAM (choice based demand)	-	6.61	6.69	LIMIT	0.72%	21,869	815	42.18
	CSD-FAM-S (w/ symmetry constr.)	-	INFEAS	INFEAS	INFEAS	INFEAS	INFEAS	-	-
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	9.39	9.39	2,156	0.00%	0	815	(147.24)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	6.71	6.78	LIMIT	0.61%	23,037	815	42.78
		opt	6.68	6.75	LIMIT	0.67%	20,976	815	44.24
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	6.71	6.77	LIMIT	0.66%	20,717	815	40.43
		opt	6.68	6.73	LIMIT	0.54%	24,856	815	**46.00**
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	6.79	6.86	LIMIT	0.62%	15,301	815	40.20
		opt	6.73	6.80	LIMIT	0.78%	21,993	815	41.10
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	6.78	6.84	LIMIT	0.60%	24,546	815	37.69
		opt	6.73	6.79	LIMIT	0.60%	22,618	815	41.58
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	6.92	7.00	LIMIT	0.60%	25,929	815	31.43
		opt	6.85	6.91	LIMIT	0.55%	25,730	815	35.04
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	6.93	6.98	LIMIT	0.49%	25,984	815	29.77
		opt	6.85	6.89	LIMIT	0.53%	25,557	815	36.84

Table 10 Computational Results of Pure Fleet Assignment with 5hr CPU Time Limit. Profit changes relative to ISD-FAM are reported in million \$/365 days. Negative numbers are reported in parentheses. Solution times are reported in CPU seconds. Baseline profit for ISD-FAM is \$0.917 billion for D1 and \$2.370 billion for D2 per 365 days.

Inst.	Model	Fare Split	Obj.	LP Relax.	CPU Time	MIP Gap	B&B Nodes	#Flights	ΔProfit
D1	ISD-FAM (indep. demand)	-	1.71	1.78	17	0.00%	68	180	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	1.07	1.14	242	0.00%	6,231	180	(13.47)
	CSD-FAM (choice based demand)	-	2.52	2.66	23,196	0.00%	87,314	180	1.84
	CSD-FAM-S (w/ symmetry constr.)	-	2.52	2.66	18,947	0.00%	71,846	180	1.84
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	2.86	2.86	1,105	0.00%	0	180	(15.16)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	2.61	2.75	22,119	0.00%	73,456	180	0.64
		opt	2.57	2.70	17,416	0.00%	78,591	180	**1.84**
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	2.61	2.74	19,473	0.00%	97,796	180	0.64
		opt	2.57	2.69	12,187	0.00%	51,884	180	**1.84**
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	2.86	2.99	10,020	0.00%	37,245	180	(7.83)
		opt	2.67	2.82	10,871	0.00%	54,685	180	(6.34)
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	2.86	2.94	724	0.00%	9,291	180	(7.83)
		opt	2.67	2.77	2,281	0.00%	26,341	180	(6.34)
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	3.03	3.17	5,648	0.00%	35,880	180	(35.33)
		opt	2.89	3.01	600	0.00%	15,544	180	(23.67)
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	3.03	3.04	16	0.00%	0	180	(35.33)
		opt	2.89	2.90	15	0.00%	0	180	(23.67)
D2	ISD-FAM (indep. demand)	-	4.16	4.13	361	0.00%	1,453	815	0.00
	ISD-FAM-SR-ITIN (w/ spill and recap.)	-	3.85	3.89	18,527	0.00%	50,047	815	(13.97)
	CSD-FAM (choice based demand)	-	6.62	6.69	LIMIT	0.53%	45,951	815	44.29
	CSD-FAM-S (w/ symmetry constr.)	-	INFEAS	INFEAS	INFEAS	INFEAS	INFEAS	-	-
	S-CSD-FAM-BFM (Barnhart et al. 2009)	dist	9.39	9.39	2,156	0.00%	0	815	(147.24)
	CSD-FAM($\Pi^0, \mathcal{H}$) (w/o composite vars)	dist	6.72	6.78	LIMIT	0.47%	96,187	815	43.27
		opt	6.69	6.75	LIMIT	0.39%	104,205	815	45.82
	S-CSD-FAM($\Pi^0_c, \Pi^0_t, \mathcal{H}$) (w/ composite vars)	dist	6.72	6.77	LIMIT	0.49%	106,061	815	45.05
		opt	6.69	6.73	LIMIT	0.43%	90,144	815	**46.78**
	CSD-FAM($\Pi^1, \mathcal{H}$) (w/o composite vars)	dist	6.79	6.86	LIMIT	0.46%	88,192	815	40.20
		opt	6.74	6.80	LIMIT	0.47%	76,784	815	43.11
	S-CSD-FAM($\Pi^1_c, \Pi^1_t, \mathcal{H}$) (w/ composite vars)	dist	6.79	6.84	LIMIT	0.47%	115,168	815	38.33
		opt	6.74	6.79	LIMIT	0.43%	104,898	815	43.64
	CSD-FAM($\Pi^2, \mathcal{H}$) (w/o composite vars)	dist	6.93	7.00	LIMIT	0.49%	58,185	815	31.86
		opt	6.85	6.91	LIMIT	0.42%	84,237	815	37.02
	S-CSD-FAM($\Pi^2_c, \Pi^2_t, \mathcal{H}$) (w/ composite vars)	dist	6.93	6.98	LIMIT	0.39%	77,234	815	30.84
		opt	6.85	6.89	LIMIT	0.40%	112,255	815	37.54

Table 11 Computational Results of Pure Fleet Assignment with 10hr CPU Time Limit. Profit changes relative to ISD-FAM are reported in million $/365 days. Negative numbers are reported in parentheses. Solution times are reported in CPU seconds. Baseline profit for ISD-FAM is $0.917 billion for D1 and $2.370 billion for D2 per 365 days.

Appendix D: An Extension with Fleet-Type Dependent Product Attractiveness

An assumption we made in the paper is that the attractiveness value for each product $p \in \mathcal{P}$ only depends on the characteristics of the product itself, e.g., its price, departure/arrival times, connections, restrictions, etc. In practice, it could also depend on the fleet types assigned to the flight legs traversed by fare product p. This happens when different fleet types have different seating capacities, seating comfort, amenities, and meal options that affect the choice of travelers (Clark 2017). This could also become one of the important considerations in modeling travelers' choice in a post COVID-19 world. In this section, we briefly discuss a modeling extension with this feature of fleet-type dependent choice behaviors. We show that the proposed methods can be naturally extended to this setting. For the ease of presentation, we assume that the demand is MNL. The general case with GAM can be developed by extending the same treatment of fleet-type dependent product attractiveness values to fleet-type dependent shadow attractiveness values as well.

In this extension, we model attractiveness of product $p \in \mathcal{P}$ as a function of fleet assignment $\mathbf{x}$, $w_p(\mathbf{x})$. Specifically, the attractiveness of product p is a function of the fleet assignment of flights used by p, i.e., $\{l \in \mathcal{L} \mid \delta_{l,p} = 1\}$. We have the following updated revenue function, $r_{\text{fleet}}(\mathbf{x}; \mathbf{h})$, with fleet-type dependent choice behaviors:

$$
\begin{aligned}
r_{\text{fleet}}(\mathbf{x}; \mathbf{h}) = \max_{\mathbf{s}} \quad & \sum_{p \in \mathcal{P}} h_p s_p \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}} \delta_{l,p} s_p \le \sum_{f \in \mathcal{F}(l)} \text{CAP}_f x_{l,f}, \ \forall l \in \mathcal{L}, \\
& \sum_{p \in \mathcal{P}(m)} s_p + s_m = \Lambda_m, \ \forall m \in \mathcal{M}, \\
& \frac{s_p}{w_p(\mathbf{x})} - \frac{s_m}{w_m} \le 0, \ \forall m \in \mathcal{M}, \ \forall p \in \mathcal{P}(m), \qquad (56) \\
& s_m, s_p \ge 0, \quad \forall m \in \mathcal{M}, \ \forall p \in \mathcal{P}.
\end{aligned}
$$

Under a general form of fleet-type dependent attractiveness $w_p(\mathbf{x})$, constraints (56) potentially introduce non-linearity. Here we provide a reformulation to linearize it. With a slight abuse of notation, denote $\mathcal{X}^p$ as the set of all (local) fleet assignments for flights used by product p, $\{l \in \mathcal{L} \mid \delta_{l,p} = 1\}$. For each fleet assignment $j \in \mathcal{X}^p$, let $w_{p,j}$ be the attractiveness of product p under assignment j. For each $j \in \mathcal{X}^p$, also define $\kappa^j_{l,f} = 1$ if flight l is assigned with fleet type f and zero otherwise. Introducing non-negative continuous variables $\mu_{p,j}$, we have the following reformulation for $r_{\text{fleet}}(\mathbf{x}; \mathbf{h})$:

$$
r_{\text{fleet}}(\mathbf{x}; \mathbf{h}) = \max_{\mathbf{s}, \mu} \sum_{p \in \mathcal{P}} h_p s_p \qquad (57)
$$

$$
\text{s.t.} \sum_{p \in \mathcal{P}} \delta_{l,p} s_p \le \sum_{f \in \mathcal{F}(l)} \text{CAP}_f x_{l,f}, \ \forall l \in \mathcal{L}, \qquad (58)
$$

$$
\sum_{p \in \mathcal{P}(m)} s_p + s_m = \Lambda_m, \ \forall m \in \mathcal{M}, \qquad (59)
$$

$$
s_p - \frac{\sum_{j \in \mathcal{X}^p} w_{p,j} \mu_{p,j}}{w_m} \le 0, \ \forall m \in \mathcal{M}, \forall p \in \mathcal{P}(m), \forall j \in \mathcal{X}^p, \qquad (60)
$$

$$
\mu_{p,j} \le \Lambda_m \kappa^j_{l,f} x_{l,f}, \ \forall m \in \mathcal{M}, \forall p \in \mathcal{P}(m), \forall j \in \mathcal{X}^p, \forall l \in \mathcal{L} : \delta_{l,p} = 1, \forall f \in \mathcal{F}(l), \qquad (61)
$$

$$
\mu_{p,j} \le s_m, \ \forall m \in \mathcal{M}, \forall p \in \mathcal{P}(m), \forall j \in \mathcal{X}^p, \qquad (62)
$$

$$\mu_{p,j} \geq s_m - \Lambda_m(1-\kappa^j_{l,f}x_{l,f}), \forall m \in \mathcal{M}, \forall p \in \mathcal{P}(m), \forall j \in \mathcal{X}^p, \forall l \in \mathcal{L}: \delta_{l,p} = 1, \forall f \in \mathcal{F}(l), \quad (63)$$

$$s_m, s_p \geq 0, \ \forall m \in \mathcal{M}, \forall p \in \mathcal{P}. \quad (64)$$

The three sets of constraints (61), (62) and (63) jointly restrict that $\sum_{j\in\mathcal{X}^p} w_{p,j}\mu_{p,j}$ takes value $w_{p,j}s_m$ for the selected $j \in \mathcal{X}^p$ specified by any feasible fleet assignment and schedule design solution $\mathbf{x}$. CSD-FAM with revenue function (57) - (64) remains a mixed integer linear program. However, compared to the original revenue function formulation (8) - (12), additional $\sum_{p\in\mathcal{P}} |\mathcal{X}^p|$ number of continuous variables (and constraints of a similar order) are introduced. For the data instances in Section 6, the number of added variables and constraints is of the order of 10^5 to 10^6.

In contrast, formulation S-CSD-FAM(Π, $\mathcal{H}$) with fleet-type dependent attractiveness stays exactly the same as that without fleet-type dependent attractiveness. Moreover, the calculation process for subnetwork revenues $r_j, j \in \mathcal{X}^k$ remains the same — it requires solving the same linear programs, as fleet assignments $\mathbf{x}$ are fixed when calculating r_j and $\{w_p(\mathbf{x})\}_{p\in\mathcal{P}}$ are constants. This suggests that the decomposition-based solution framework that we have proposed naturally applies to this context and is likely to bring more computational benefits and profit contributions.

We now go over each of our major technical results and solution framework. It can be shown that all technical results developed in the paper can be applied to the case of fleet-type dependent attractiveness with minimal modifications under the following (mild) assumption on the revenue function $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$:

Assumption 1. $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$ *is concave in relaxed SD-FAM decisions* $\mathbf{x} \in [0,1]^{|\mathcal{L}|\times|\mathcal{F}|}$.

This assumption is expected to be mild because of three reasons: (1) under the case that product attractiveness w_p does not depend on fleet assignment, Assumption 1 is readily satisfied by standard sensitivity analysis of linear programs; (2) when attractiveness does depend on fleet assignment, this assumption requires that the marginal increment in revenue due to the increase in seating capacity (and possible resultant increase in attractiveness) is decreasing; in other words, one would expect this assumption to hold when the increase in seating capacity will not drastically increase the product attractiveness, in turn ensuring a decreasing marginal revenue; (3) finally, it is easy to see that this assumption always holds when the increase in seating capacity decreases the product attractiveness. A more precise characterization on conditions under which Assumption 1 holds might warrant further research.

For each fare product $p \in \mathcal{P}$, define $w^k_p(\mathbf{x}^k)$ as

$$w^k_p(\mathbf{x}^k) = \max_{\mathbf{x}': x'_{l,f} = x^k_{l,f}, \forall l \in \mathcal{L}^k, \forall f \in \mathcal{F}(l)} w_p(\mathbf{x}').$$

In words, $w^k_p(\mathbf{x}^k)$ is the maximum possible attractiveness for product p when the fleet assignment for flights in subnetwork k, $\mathcal{L}^k$, is set according to $\mathbf{x}^k$. We are now going to define a modified local revenue function $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ for each subnetwork $k \in \{1,\cdots,K\}$ whose summation over k will again serve as an upper bound on the revenue calculation. The difference between $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ and $r_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ is in constraint (65) where $w_p(\mathbf{x}^k)$ is replaced with $w^k_p(\mathbf{x}^k)$. In all subnetwork-based formulations CSD-FAM($\Pi,\mathcal{H}$), S-CSD-FAM($\Pi,\mathcal{H}$) and S-CSD-FAM($\Pi_c,\Pi_t,\mathcal{H}$), $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ will now replace $r(\mathbf{x}^k;\mathbf{h}^k)$. With this modified local revenue function, we show that all the major results and procedures developed in the paper still remain valid. Results that are not covered below remain unchanged.

$$\begin{aligned}
r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)=\max_{\mathbf{s}} \quad & \sum_{p\in\mathcal{P}} h^k_p s_p \\
\text{s.t.} \quad & \sum_{p\in\mathcal{P}} \delta_{l,p} s_p \le \sum_{f\in\mathcal{F}(l)} \text{CAP}_f x^k_{l,f},\ \forall l\in\mathcal{L}, \\
& \sum_{p\in\mathcal{P}(m)} s_p + s_m = \Lambda_m,\ \forall m\in\mathcal{M}, \\
& \frac{s_p}{w^k_p(\mathbf{x}^k)} - \frac{s_m}{w_m} \le 0,\ \forall m\in\mathcal{M},\ \forall p\in\mathcal{P}(m), \\
& s_m, s_p \ge 0,\quad \forall m\in\mathcal{M},\ \forall p\in\mathcal{P}.
\end{aligned} \tag{65}$$

Revisiting Proposition 1 and Corollary 1. Proposition 1 and Corollary 1 still hold because the attractiveness values, $w_p(\mathbf{x})$ and $w^k_p(\mathbf{x})$, for product p *only* depend on the fleet assignment of flights used by p, i.e., $\{l\in\mathcal{L}\mid \delta_{l,p}=1\}$. This implies that for any local fleet assignment of subnetwork k, $w^k_p(\mathbf{x}^k)=w_p(\mathbf{x})$ for all $p\in\mathcal{P}^k$. The original proof thus follows.

Revisiting Theorem 1. $V_{\text{CSD-FAM}(\Pi,\mathcal{H})} = V_{\text{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} = V_{\text{S-CSD-FAM}(\Pi,\mathcal{H})}$ still holds as the revenue objective values are the same across these three formulations. Following the original proof, $\tilde{V}_{\text{CSD-FAM}(\Pi,\mathcal{H})} \ge \tilde{V}_{\text{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})} \ge \tilde{V}_{\text{S-CSD-FAM}(\Pi,\mathcal{H})}$ and $\tilde{V}_{\text{S-CSD-FAM}(\Pi'_c,\Pi'_t,\mathcal{H})} \ge \tilde{V}_{\text{S-CSD-FAM}(\Pi_c,\Pi_t,\mathcal{H})}$ for any two splits (Π_c,Π_t) and (Π'_c,Π'_t) of the same partition Π with $\Pi_c\supset\Pi'_c$, because $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ is concave in $\mathbf{x}^k$ under Assumption 1.

Revisiting Proposition 2 and Corollary 2. For any fleet assignment $\mathbf{x}$ and fare vector $\mathbf{h}$, we define $\mathbf{s}^*$ as the optimal solution of problem $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$. $\mathbf{s}^*$ is also a feasible solution to all problems $r^k_{\text{fleet}}(\mathbf{x}^1;\mathbf{h}^1),\cdots,r^K_{\text{fleet}}(\mathbf{x}^K;\mathbf{h}^K)$ for any set of local fare vectors $\mathcal{H}=\{\mathbf{h}^k\}_{k\in\{1,\cdots,K\}}$. This is so because, going from $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$ to $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$, the changes are: (1) the right hand sides of the capacity constraints become larger or equal for $l\notin\mathcal{L}^k$ and stay the same for $l\in\mathcal{L}^k$; (2) the left hand sides become smaller or equal when going from constraints (56) to constraints (65), because for any subnetwork k, $w^k_p(\mathbf{x}^k)\ge w_p(\mathbf{x})$. Thus we have,

$$\sum_{k=1}^K r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k) \ge \sum_{k=1}^K \left(\sum_{p\in\mathcal{P}} h^k_p s^*_p\right) = \sum_{p\in\mathcal{P}}\left(\sum_{k=1}^K h^k_p\right) s^*_p = \sum_{p\in\mathcal{P}} h_p s^*_p = r_{\text{fleet}}(\mathbf{x};\mathbf{h}).$$

The first inequality holds because $\mathbf{s}^*$ is a feasible solution for these K problems but it may not be the optimal solution, and hence it provides a lower or equal total revenue. The first equality is a re-arrangement of terms. The second equality holds because of Definition 2. The last equality holds due to the fact that s^* is the optimal solution of problem $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$. The proof of Corollary 2 follows naturally from this as well.

Revisiting Proposition 3. The only needed change is to revise the calculation of the maximum possible sale of product p, $\bar{s}_p$, as

$$\bar{s}_p = \min\left\{\sum_{m\in\mathcal{M}(p)} \Lambda_m(\max_{\mathbf{x}} w_p(\mathbf{x}))/((\max_{\mathbf{x}} w_p(\mathbf{x}))+v_m),\ \ \min_{l:\delta_{l,p}=1}\left\{\max_{f\in\mathcal{F}(l)} \text{CAP}_f\right\}\right\},$$

where $\mathcal{M}(p)$ denotes the set of markets to which product p belongs. Observe that the only change here is that we replace w_p with $\max_{\mathbf{x}} w_p(\mathbf{x})$ in the definition of $\bar{s}_p$. The rest of the proof follows naturally from this.

Revisiting Proposition 4. The same dualization can be applied to $r^k_{\text{fleet}}(\mathbf{x}^k;\mathbf{h}^k)$ after reformulating it to a linear program using the same linearization approach for $r_{\text{fleet}}(\mathbf{x};\mathbf{h})$. To avoid repetition, we do not derive the full linear program here. The rest of the proof follows as before.

Revisiting Variable Reduction in Section 5.3. The only needed change is to the calculation of $\bar{d}_l$, the maximum demand that a flight leg can capture if there is no seating capacity limit. As with the revised definition of $\bar{s}_p$, the only change in this revised definition of $\bar{d}_l$ is that we replace w_p with $\max_{\mathbf{x}} w_p(\mathbf{x})$ in it.

$$\bar{d}_l = \sum_{m\in\mathcal{M}(l)} \left(\frac{\sum_{p\in\mathcal{P}(m)} \delta_{l,p} \max_{\mathbf{x}} w_p(\mathbf{x})}{w_m + \sum_{p\in\mathcal{P}(m)} \delta_{l,p} \max_{\mathbf{x}} w_p(\mathbf{x})} \right) \Lambda_m.$$

The rest of the variable reduction process follows as before.