The Anheuser Busch InBev Trailer Problem: An Application of Online Resource Allocation and Inventory Selection

Xingxing Chen
Olin Business School, Washington University, Saint Louis, MO 63130, xingxing.chen@wustl.edu

Jacob Feldman
Olin Business School, Washington University, Saint Louis, MO 63130, jbfeldman@wustl.edu

Seung Hwan Jung
College of Business Administration, Texas A&M University-Kingsville, Kingsville, TX 78363, Seung-Hwan.Jung@tamuk.edu

Panos Kouvelis
Olin Business School, Washington University, Saint Louis, MO 63130, kouvelis@wustl.edu

In this paper, we introduce and study the Anheuser Busch InBev trailer problem, which considers how Anheuser Busch InBev ships its beer to wholesalers via third party delivery trucks. In this problem, Anheuser Busch InBev must select the weights and inventory levels of preloaded trailers of beer, which are then matched in an online fashion to arriving third party delivery trucks. What ultimately arises is an interesting online resource allocation problem, which also must incorporate an initial inventory decision as a precursor. We develop two distinct approaches for this problem under two different settings, which are distinguished by whether or not Anheuser Busch InBev would like to consider altering the weights of trailers that have been preloaded as the delivery trucks arrive. In both settings, we characterize optimal policies for matching trailers to arriving delivery trucks and develop efficient approaches to select the set of trailer weights to preload as well as their corresponding inventory levels. Using historical truck arrival data from Anheuser Busch InBev, we simulate our proposed approaches and find that their performances are within 1% of optimality on average.

Key words: dynamic matching, inventory selection, piecewise linear approximation, constraint generation.

1. Introduction

Two problems that have received increased attention in the operations and revenue management literatures are those of online resource allocation/matching and optimal inventory selection. The former refers to the problem of matching customers to a scarce set of products in an online fashion so as to maximize the revenue accrued over a finite selling horizon. There are many widely studied problems that fall within this framework. Examples include the classic network revenue management problem, where the goal is to dynamically adjust the set of offered products over a selling horizon to maximize expected revenue when the sale of each product consumes a combination of resources. More directed versions of this problem are studied by Rusmevichietong et al. (2014)

and Ma and Levi (2017), who consider online assortment problems in which a retailer is allowed to customize her offer decisions to each arriving customer based on just revealed features of the given customer and current inventory levels for each product. Additionally, the Display Ads problem, which is the edge weighted and capacitated generalization of the online bipartite matching problem, is another widely studied problem in the computer science literature that falls within the framework of online resource allocation.

On the other hand, the inventory selection problem considers how to choose initial inventory levels for a collection of products, which are then subsequently consumed over a finite selling horizon according to some known demand process. Examples of works that have studied this problem include Honhon et al. (2010), Goyal et al. (2016), and Aouad et al. (2018), who consider versions of this problem when demand is governed by a particular customer choice model. Interestingly, even though the inventory selection problem is a natural precursor to the online resource allocation problem in most settings, there is little work that tackles the two problems simultaneously. For example, the many solution approaches developed for the network revenue management problem all assume that initial inventory levels are given. In a similar vein, the online assortment problem studied by Rusmevichietong et al. (2014) assumes the initial inventory decision is exogenous and hence only considers how to vary the offered assortment to each arriving customer. In reality, however, retailers generally have control over the initial inventory decision. For example, Lufthansa Airlines has a movable curtain that allows them to dynamically adjust the number of Economy seats on each of their flights. Consequently, it is intriguing to wonder if it is possible to incorporate an initial inventory decision within solution approaches for online resource allocation problems.

In what follows, we introduce the Anheuser Busch Inbev trailer problem, which considers how Anheuser Busch InBev should preload trailers of beer that then must be matched to arriving third party delivery trucks. We ultimately arrive at a problem setting where a manager must develop and link solution approaches to online resource matching and inventory selection problems. In this way, to the best of our knowledge, we present one of the first approaches which simultaneously accounts for both of these critical operational decisions.

The Anheuser Busch InBev trailer problem and its significance. Anheuser Busch InBev (abbreviated as ABI for the remainder of the paper) brews and packages its beer in multiple locations throughout the United States. After packaging, the finished product is transported to beer wholesalers via third party trucks. Beer is transported from each brewery via drop trailers, which are preloaded trailers of beer, whose weights have been chosen in advance of the arrival of the third party trucks. Typically, drop trailers are preloaded 4hrs - 48hrs in advance of the truck arrivals. For each third party truck that arrives to their brewery warehouse, ABI must match this truck with a trailer of beer so as to maximize their revenue, which is proportional to the total amount of beer shipped out. If, however, the gross weight of the trailer and truck exceeds 80,000 lbs., then the truck must return to the loading dock for adjustment due to federally mandated weight limits. This action is termed a "scaleback", and causes additional costs to be incurred by ABI since workers must be paid to remove beer from the preloaded trailer. The problem is further complicated due to the fact that the weights of the arriving trucks are unpredictable, since each third party delivery service has a variety of different trucks in their fleet and ABI is not able to plan and coordinate the exact sequence of truck arrivals on each day.

The focus of this paper is on optimizing the drop trailer portion of ABI's shipping system. We consider a time horizon over which there are τ scheduled arrivals of third party trucks. We discretized this time horizon into τ time periods so that there is exactly one truck arrival in each time period. While we assume that ABI knows the number of truck arrivals, we assume that the weight of each truck is only revealed once the truck arrives to the warehouse and is hence ready to be assigned a trailer of beer. The ABI trailer problem can be summarized as a sequence of three sets of decisions that are made with the intention of maximizing the revenue earned over the time horizon. First, ABI must choose the weights of the preloaded trailers. We refer to each set of trailers that is loaded at a unique weight as a trailer type. In other words, the set of trailer types gives the set of weights at which ABI has preloaded trailers. Second, they must choose the number of trailers to preload at each of these weights. Finally, they must choose a trailer to match to each arriving truck by considering the immediate revenue and the expected revenue that could be derived from future truck arrivals. We refer to these three problems as the trailer type selection, inventory selection and trailer matching problems respectively.

The current practice at ABI is to preload only a single trailer type, which is then matched to each arriving truck. Consequently, within this current approach, there is no need to solve the aforementioned inventory selection and trailer matching problems since only a single trailer type is selected. We ultimately show that by allowing for multiple trailer types to be selected, ABI has the potential to improve its revenue by as much as 1.5%. Using daily logistics cost data provided to us by ABI, we have calculated the opportunity cost from suboptimal approaches to the ABI trailer problem to be approximately $9,420,000 per year for a single brewery. Considering that ABI has 21 breweries all over the U.S., this slight percentage improvement in performance can improve revenue by millions of dollars.

Relationship to existing problems. Before we thoroughly review the past literature, we briefly distinguish our problem setting from other similar settings. For a fixed collection of trailer types and inventories levels for each of these types, the trailer matching problem considers how to optimally match trailers to arriving third party trucks so as to maximize the expected revenue over the given time horizon. Variations of this problem by itself have been studied in the revenue management and online matching literature. Since both the trailer matching and network revenue management problems require managing a collection of perishable inventory units over a finite selling horizon, the two problems are seemingly quite similar. However, in the network revenue management problem, arriving customers choose among the offered products according to some customer choice model, while in our trailer matching problem, each arriving truck can be deterministically assigned to any trailer with available inventory. We ultimately show that despite this difference, we are able to apply classic approximate dynamic programming techniques from the network revenue management literature not only to the trailer matching problem, but also to the inventory selection problem. In this way, we extend the efficacy of such techniques to a new class of inventory problems, which require both selecting initial inventory levels and managing the consumption process of each product over a finite selling horizon.

The Display Ads problem, which again is the edge weighted and capacitated generalization of the online bipartite matching problem, can easily be seen to be a more general form of the trailer matching problem. In the online matching literature, the efficacy of an algorithm is generally measured by its competitive ratio, which compares the performance of the proposed online approach against an optimal offline algorithm that is given access to the entire input graph. In the adversarial setting, the theoretical performance of the proposed online algorithm is measured against the single worst case input graph that could ever arise. In the more structured IID setting, node arrivals follow some known stochastic process and hence the algorithm is benchmarked against the expected performance of an offline algorithm. With regards to the Display Ads problem, it is easy to construct simple instances (see Chapter 7 of Mehta (2013)) for which it is not possible to obtain a non-trivial competitive ratios in the adversarial setting . As is such, simplifications such as the free disposal model, which relax the capacity restrictions, have led to algorithms that yield a competitive ration of $1-\frac{1}{e}$ (see Feldman et al. (2009)). A simplification of this type is not amenable to our setting, as it would require allowing a trailer type with zero inventory to be assigned to an arriving truck. In the IID setting, which mirrors the setting that we study since the truck arrivals follow a known distribution, Stein et al. (2016) develop a 0.321-approximation scheme. While this approach could be used to solve the trailer matching problem, we show that by exploiting the simple structure of the matching reward function that is specific to the ABI trailer problem, we can uncover additional structure on the optimal policy that leads to simple approximation schemes and interesting managerial insights regarding the performance of greedy policies.

At this point, we reiterate that the approaches alluded to above only relate to possible algorithms for the trailer matching problem. Extending and combining ideas from the revenue management and online matching literature to sequentially account for each of the three problems that make up the ABI trailer problem forms the basis of our contributions. A central difficulty in doing such, is the fact that the three problems are not independent. For example, one of the algorithms that we develop for the inventory selection problem critically relies on the structure of the optimal policy for the trailer matching problem. Consequently, even if tractable approaches exist for the problems individually, there is still difficulty in stitching the three approaches together. In what follows, we provide a more detailed description of our main findings and of the related literature.

Contributions. We consider two settings for the ABI trailer problem, which are distinguished by whether or not ABI would like to consider scaleback events. In the first setting, we assume that once the trailers have been preloaded, they cannot be altered by removing beer. In this case, each arriving truck can only be matched to trailers whose combined weight does not exceed 80,000 pounds. If no such trailer exists, then the truck must leave the warehouse empty-handed. In the second setting, we relax the restriction on the set of feasible trailer-truck assignments by allowing ABI to remove beer from the preloaded trailers at a non-negligible operational cost. In both settings, initial formulations of the trailer matching problem as a dynamic program suffer from the so-called curse of dimensionality due to the fact that the number of potential inventory states grows exponentially in τ. Consequently, both of the approaches that we develop begin by showing how this difficulty can be overcome by proving the optimality of greedy policies, which can then be exploited in solving the inventory selection and trailer type selection problems.

In the first setting, we show that the trailer matching problem can be solved optimally via a simple greedy algorithm, which always matches the heaviest feasible trailer for which there is available inventory. We then show how to cast the inventory selection problem as a problem of maximizing a submodular montone set function subject to a cardinality constraint, for which it is well documented (see G. L. Nemhauser and Fisher (1978)) that a greedy procedure produces a solution within a factor of $1-\frac{1}{e}$ of optimal. Our proof of this result involves a sample path argument that relies critically on the optimality of the aforementioned greedy policy. Further, implementing the greedy procedure of G. L. Nemhauser and Fisher (1978) requires access to an oracle which can efficiently evaluate the expected revenue for any choice of initial trailers and inventory levels. Again, the optimality of a greedy policy is what allows us to develop such an oracle via Monte Carlo simulation. We wrap up our analysis in this first part by showing how to identify a reasonably small set of trailer types from an arbitrary universe of potential trailer types at a cost of only a factor of ϵ of optimal for any $\epsilon > 0$.

In the second setting, we first prove a modified greedy policy for the trailer matching problem. We then employ the piecewise linear approximation approach of Kunnumkal and Talluri (2015) for the network revenue management problem to our problem setting. To do so, we begin by considering the linear programming formulation of the dynamic program for the trailer matching problem, which has a decision variable and constraint for each potential inventory state in each time period. Following Kunnumkal and Talluri (2015), we then approximate each decision variable by a separable (by trailer types) piecewise linear function. This technique reduces the number of decision variables, but the number of constraints still grows exponentially in τ. Following the common practice in the revenue management literature, we show that this difficulty can be overcome by providing an efficient way to implement constraint generation. We then show how to extend this linear program to account for the inventory and trailer type selection problems. We accomplish the former by exploiting properties of the multiple choice knapsack problem. Ultimately we give a single linear program that simultaneously approximates each of the three problems. We show that this linear program can be efficiently solved and that it provides a tight upper bound on the optimal expected revenue, which can be used to measure the efficacy of various heuristic approaches.

We conclude with a series of computational experiments which test the performance of the approaches that we develop as well as the tightness of the upper bound that we propose. All test cases are implemented using historical truck arrival data from two North American warehouses of ABI along with accurate revenue and cost parameter estimates that are provided by the company as well. We find that in most of the test cases, both of our proposed approaches produce solutions to the ABI trailer problem that are within 1% of optimality on average.

Related Literature. The stream of literature that most closely resembles our work is that of approximate techniques for the network revenue management problem. The seminal approach of Simpson (1989) proposes a linear programming based approximation of the problem, known as the deterministic linear program (DLP), where the demand for each product is assumed to take on its expected value. Later on, Talluri and van Ryzin (1999*a*) and Talluri and van Ryzin (1999*b*) study the performance of bid price policies that can be derived from an optimal solution to the DLP. Topaloglu (2009) proposes an alternative way to derive bid prices, which employs a Lagrangian relaxation to decouple decisions across resources.

These previous works assume that the demand for a particular product is independent of the availability of the other products. To combat this issue, Gallego et al. (2004) propose the Choice-Based DLP, where the demand for each product is governed by some underlying customer choice model and there is a decision variable for the fraction of time to offer each assortment of products over the selling horizon. Gallego et al. (2004) show that the optimal objective value of the Choice-Based DLP is an upper bound on the optimal expected revenue and hence its value can be used to benchmark heuristics. Liu and van Ryzin (2009) extend this work by characterizing the general structure of optimal offer decisions and showing that the optimal objective of the Choice-Based DLP approaches the expected revenue of an optimal policy as the capacities and length of the time horizon are scaled up. Méndez-Díaz et al. (2010), Gallego et al. (2014) and Feldman and Topaloglu

(2017) present approaches for solving the Choice-Based DLP under various popular choice models which are based on developing efficient algorithms for the column generation subproblem.

Another popular approach, which more closely resembles the direction that we take, is to write the dynamic programming formulation of the network revenue problem as an equivalent linear program. This linear program has a decision variable and constraint for each potential inventory state and hence the size of this linear program grows exponentially in the length of the selling horizon. Adelman (2007) proposes approximating each decision variable as a linear combination of the corresponding inventory levels of each resource. This technique is often referred to as the affine approximation to the value functions. This approximation reduces the number of decision variables in the linear program, however, the number of constraints remains exponential in the length of the selling horizon. Consequently, Adelman (2007) propose an integer programming formulation for the constraint generation subproblem and also provide a way to get an upper bound on the optimal objective value of the full linear program, which can then be used to check the optimality gap at any point during constraint generation. In our setting, we give a similar upper bound, but prove the result in a different manner. Tong and Topaloglu (2014) extend this result by showing that column generation subproblem can actually be relaxed to a linear program.

With regards to the affine approximation, the previously stated results assume an independent demand model. Zhang and Adelman (2009), Meissner and Strauss (2012), and Vossen and Zhang (2015) apply the affine approximation to network revenue management problems with customer choice. They show that the linear program that results from variations of this approximation can be solved efficiently and the upper bound provided by the affine approximation is tighter than that provided by the Choice-Based DLP. More recently, Sumida et al. (2017) show that the affine approximations can be used to derive policies with constant factor performance guarantees for the network revenue management problem with parallel flight legs.

To the best of our knowledge, Kunnumkal and Talluri (2015) is the first to consider the piecewise linear approximation to the value functions that we also employ. They show that this approach provides a tighter upper bound on the expected revenue than both the DLP and the affine approximation. Talluri and Kunnumkal (2016) formalizes this notion by providing theoretical bounds on the gap between the DLP and the affine and piecewise linear approximations. We show that this approximate dynamic programming technique can be generalized to our setting and even extended to simultaneously account for all three problems that make up the ABI trailer problem. We note that the affine approximation could potentially be useful for developing upper bounds and heuristic policies for the trailer matching problem. However, it will not be hard to see later on, that extending this approach to the inventory selection problem in the same way as we do with the piecewise linear approximations would result in a trivial inventory vector that only chooses one trailer type.

The remainder of this paper is organized as follows. In section 2 we formalize the three problems that make up the ABI trailer problem. Then, sections 3 and 4 give our solution approaches for the two settings that we consider. In section 5, we present a series of computational experiments that measure the efficacy of our proposed solution approaches using real truck arrival data from ABI. Finally, we conclude and provide avenues for future work in section 6.

2. Problem Formulation

In what follows, we formalize the three problems that make up the ABI trailer problem progressing backwards with regards to the order in which they must be solved in practice. We begin by formulating the trailer matching problem and then move to the inventory selection problem and then finish with the trailer type selection problem. We discretize our planning horizon into time periods $t = 1, \dots, \tau$, where in each time period exactly one truck arrives. We denote the set of m potential types of trucks as $\mathcal{B} = \{1, \dots, m\}$. For a truck type $b \in \mathcal{B}$, let p_b and w_b be the arrival probability and truck weight respectively. Since exactly one truck arrives to the warehouse in each time period, we have that $\sum_{b \in \mathcal{B}} p_b = 1$. The collection of n types of pre-loaded trailers is given by $\mathcal{L} = \{1, \dots, n\}$, where the loaded weight of trailer type $l \in \mathcal{L}$ is given by w_l. We assume the trailer types are indexed in increasing order of weight. For ease of presentation, we create a dummy trailer $0 \in \mathcal{L}$ that has infinite capacity and generates zero revenue. The revenue from assigning trailer type l to a truck b is denoted as $r_{l,b}$. The exact form of $r_{l,b}$ will depend on whether or not we consider scaleback events, and hence we delay presenting an explicit expression for this term until later sections.

We use $x_t \in \mathbb{Z}_+^n$ to denote the remaining number of unassigned trailers at the beginning of time period t, where x_t^l gives the number of remaining units of trailer type l. We assume that the manager chooses a total of τ units across all n types of trailers and hence the initial inventory level x_1 must be chosen from the set $\mathcal{X}_1 = \{x \in \mathbb{Z}_+^n : \sum_{l \in \mathcal{L} \setminus \{0\}} x^l = \tau\}$. For $t > 1$, the inventory levels of all trailers must satisfy $x_t \in \mathcal{X}_t = \{x \in \mathbb{Z}_+^n : \sum_{l \in \mathcal{L} \setminus \{0\}} x^l \geq \tau - (t-1)\}$, since before time period t at most $(t-1)$ units of trailers have been assigned. We note that the inequality in the definition of $\mathcal{X}_t$ is a result of the fact that when scalebacks are not allowed, it is possible that there will be no feasible trailer that can be assigned to the arriving truck.

For fixed initial inventory vector $x_1 \in \mathcal{X}_1$, the trailer matching problem can be formulated as a simple dynamic program whose value function $V_t(x_t)$ represent the maximum expected revenue that can be derived from time periods $t, t+1, \dots, \tau$ given the current inventory levels x_t. Before giving the Bellman equations of the dynamic program we first develop a bit of notation that simplifies our exposition. For given inventory levels x_t, we let $\mathcal{L}(x_t) = \{l \in \mathcal{L} : x_t^l > 0\}$ denote the set of trailer types that have remaining inventory. Further, we let $e_l \in \{0,1\}^n$ donate the unit vector whose l^{th} component is equal to 1 and whose other components are 0. With this notation in hand, the dynamic programming formulation for the trailer matching problem is given below:

$$V_t(x_t) = \sum_{b \in \mathcal{B}} p_b \max_{l \in \mathcal{L}(x_t)} \left\{ r_{l,b} + V_{t+1}(x_t - e_l) \right\}, \tag{1}$$

with base cases

$$V_{\tau+1}(\cdot) = 0.$$

The optimal expected revenue for initial inventory vector x_1 is therefore given by $V_1(x_1)$ and hence the preceding inventory selection problem can be written as

$$Z^*(\mathcal{L}) = \max_{x_1 \in \mathcal{X}_1} V_1(x_1). \tag{2}$$

Our final task is to choose the trailer types $\mathcal{L}$ from the universe of all possible choices $\mathcal{U}$ in order to maximize $Z^*(\mathcal{L})$

$$OPT = \max_{\mathcal{L} \subset \mathcal{U}} Z^*(\mathcal{L}) \tag{3}$$

We first note that computing $V_1(x_1)$ is no simple task, since the number of feasible inventory vector grows exponentially in τ. Further, even if given a blackbox oracle that can compute $V_1(x_1)$ for any initial inventory level x_1, problem (2) remains non-trivial due its combinatorial nature. In what follows, we address problems (1) - (3) under two different assumptions regarding the set of feasible trailers that can be assigned to arriving trucks. In both cases, we show how our approaches can be extended to approximate the inventory selection and trailer selection problems both accurately and efficiently.

3. First Setting: No scalebacks

In the first setting, we assume that trailer weights cannot be altered in order to meet the gross weight limit of 80,000 lbs. In other words, a trailer $l \in \mathcal{L}$ can only be assigned to an arriving truck $b \in \mathcal{B}$ if $x_t^l > 0$ (inventory availability) and $w_l + w_b \leq 80,000$ (weight requirement). This setting is particularly relevant when scaleback events are costly or time consuming. Further, the policies and ideas that we develop in this setting inspire heuristics for the general setting, which we eventually show to be quite profitable. To encode this restriction, we set

$$r_{l,b} = \begin{cases} w_l \cdot r, & \text{if } w_l + w_b \leq 80,000 \\ 0, & \text{o.w.} \end{cases} \tag{4}$$

where r is the revenue gained per pound of weight loaded onto a truck. Due to the presence of the dummy trailer, it will never be optimal to assign a trailer $l \geq 1$ to arriving bus $b \in \mathcal{B}$ with $r_{l,b} = 0$. Consequently, in solving the dynamic program in (1), the structure of $r_{l,b}$ is enough to ensure that we do not make an infeasible assignment in this setting. We begin our analysis by focusing on the trailer matching problem in this setting.

3.1. The Trailer Matching Problem

In this section, we show that a simple greedy policy is optimal for (1) when the revenues take the form of (4). To do so, we prove the following proposition.

PROPOSITION 1. *For any time period t, inventory vector x_t and trailers $l_1, l_2 \in \mathcal{L}(x_t)$ such that $l_2 > l_1$, we have that*

$$V_t(x_t - e_{l_1}) - V_t(x_t - e_{l_2}) \leq (w_{l_2} - w_{l_1})r.$$

Proof. We prove the result by induction over the time periods. The result is trivially true for time period $T+1$. Next, we assume that the result holds up to time period $t+1$ and prove the result for time period t. To do so, we consider the optimal trailer assigned to each arriving truck for a fixed time period t with inventory vectors $x_t - e_{l_2}$ and $x_t - e_{l_1}$. For $b \in \mathcal{B}$, let

$$k_2^* = \underset{k_2 \in \mathcal{L}(x_t - e_{l_2})}{\arg\max} \; r_{k_2,b} + V_{t+1}(x - e_{l_2} - e_{k_2})$$

and

$$k_1^* = \underset{k_1 \in \mathcal{L}(x_t - e_{l_1})}{\arg\max} \; r_{k_1,b} + V_{t+1}(x - e_{l_1} - e_{k_1})$$

be such trailer assignments. Since the arriving truck $b \in \mathcal{B}$ has been fixed, we drop the dependence of b on these optimal trailer assignments to avoid cluttered notation. By the induction hypothesis we have that

$$k_2^* = \underset{k_2 \in \mathcal{L}(x_t - e_{l_2})}{\arg\max} \; r_{k_2,b}$$

and

$$k_1^* = \underset{k_1 \in \mathcal{L}(x_t - e_{l_1})}{\arg\max} \; r_{k_1,b}$$

since $r_{k_1,b} = w_{k_1} r$ and $r_{k_2,b} = w_{k_2} r$ as long the trailers can be feasibly matched with the given truck. Next, still fixing the arriving truck $b \in \mathcal{B}$, we consider the three possible scenarios for how k_1^* and k_2^* relate to each other. In each case, we let

$$\Delta(x_t, k_1^*, k_2^*) = (r_{k_1^*,b} + V_{t+1}(x - e_{l_1} - e_{k_1^*})) - (r_{k_2^*,b} + V_{t+1}(x - e_{l_2} - e_{k_2^*})),$$

and we show that $\Delta_t(x_t, k_1^*, k_2^*) \leq (w_{l_2} - w_{l_1})r$.

Case 1: $k_1^* = k_2^* = k^*$: In this first case, we have that

$$\begin{aligned}
\Delta(x_t, k^*, k^*) =& (r_{k^*,b} + V_{t+1}(x_t - e_{l_1} - e_{k^*})) - (r_{k^*,b} + V_{t+1}(x_t - e_{l_2} - e_{k^*})) \\
&= V_{t+1}(x_t - e_{l_1} - e_{k^*}) - V_{t+1}(x_t - e_{l_2} - e_{k^*}) \\
&= V_{t+1}(x'_t - e_{l_1}) - V_{t+1}(x'_t - e_{l_2}) \\
&\leq (w_{l_2} - w_{l_1})r.
\end{aligned}$$

The second equality follows by setting $x'_t = x_t \setminus \{k^*\}$ and the inequality follows by the induction hypothesis.

Case 2: $k_1^* \neq k_2^* = l_1$: In this second case, the revenue difference can be written as

$$\begin{aligned}\Delta(x_t,k_1^*,l_1) &= (r_{k_1^*,b}+V_{t+1}(x_t-e_{l_1}-e_{k_1^*}))-(r_{l_1,b}+V_{t+1}(x_t-e_{l_2}-e_{l_1}))\\ &\leq (w_{l_2}r+V_{t+1}(x_t-e_{l_1}-e_{l_2}))-(w_{l_1}r+V_{t+1}(x_t-e_{l_2}-e_{l_1}))\\ &= (w_{l_2}-w_{l_1})r,\end{aligned}$$

where the first inequality follows by the induction hypothesis due to the fact that $k_1^* \leq l_2$, since otherwise k_1^* would also be optimal for the $x_t - e_{l_2}$ case.

Case 3: $k_2^* \neq k_1^* = l_2$: In this final case, the revenue difference can be written as

$$\begin{aligned}\Delta(x_t,l_2,k_2^*) &= (r_{l_2,b}+V_{t+1}(x_t-e_{l_1}-e_{l_2}))-(r_{k_2^*,b}+V_{t+1}(x_t-e_{l_2}-e_{l_1}))\\ &\leq (w_{l_2}r+V_{t+1}(x_t-e_{l_1}-e_{l_2}))-(w_{l_1}r+V_{t+1}(x_t-e_{l_2}-e_{l_1}))\\ &= (w_{l_2}-w_{l_1})r,\end{aligned}$$

where the inequality follows by the induction hypothesis due to the fact that $k_2^* \geq l_1$ since l_2 was optimal for the $x_t - e_{l_1}$ case.

Combining everything, we get that

$$\begin{aligned}V_t(x_t-e_{l_1})-V_t(x_t-e_{l_2}) &= \sum_{b\in\mathcal{B}} p_b \Delta(x_t,k_1^*,k_2^*)\\ &\leq (w_{l_2}-w_{l_1})r,\end{aligned}$$

where the inequality follows due to the three case argument made above.

Applying this result for time period $t+1$ gives that $w_{l_2}r + V_{t+1}(x_t - e_{l_2}) \geq w_{l_1}r + V_{t+1}(x_t - e_{l_1})$ for $l_1, l_2 \in \mathcal{L}$ such that $l_2 > l_1$. Further, since for any feasible matching, we have that $r_{l,b} = w_l r$, this immediately yields the following corollary, which states that a greedy policy is optimal.

Corollary 1. *For any time period t, inventory level x_t and truck arrival $b \in B$, we have that*

$$\arg\max_{l\in\mathcal{L}(x_t)} \left\{r_{l,b}+V_{t+1}(x_t-e_l)\right\} = \arg\max_{l\in\mathcal{L}(x_t)} r_{l,b}.$$

Not only does corollary 1 show that a simple, easy to implement policy is optimal, but it also makes approximating $V_1(x_1)$ far easier. To see this, consider any sample path of truck arrivals $P = \{b_1,\ldots,b_\tau\}$ and let $V_1^P(x_1)$ be the optimal revenue under sample path P. Since the greedy policy described in corollary 1 only depends on the truck arrival in each time period, we can compute $V_1^P(x_1)$ without recursively computing any of the value functions. Noting that $V_1(x_1) = \mathbb{E}[V_1^P(x_1)]$, where the expectation is taking with respect to the the sample path P, we can easily approximate $V_1(x_1)$ to arbitrary precision using Monte Carlo simulation. Consequently, when we tackle the inventory selection problem in the next section, we assume that $V_1(x_1)$ can be computed exactly for any starting inventory vector $x_1 \in \mathcal{X}$.

3.2. The Inventory Selection Problem

In this section, we focus on the inventory selection problem given in (2). We show that $V_1(x_1)$ can be recast as a monotone submodular set function. Recall that a set function $f(\cdot)$ is monotone if for all $S \subseteq T$ we have that $f(S) \leq f(T)$. Further, the set function $f(\cdot)$ is submodular if for all S and $i, j \notin S$, we have that $f(S \cup \{i\}) + f(S \cup \{j\}) \geq f(S \cup \{i, j\}) + f(S)$.

We begin by showing how to recast the value function $V_1(x_1)$ as a set function. To do so, we duplicate each trailer $l \in \mathcal{L}$ a total of τ times and define $\mathcal{N}(\mathcal{L})$ to be the set of all $n\tau$ trailers. The one-to-one correspondence is now easy to see; choosing an initial inventory vector x_1 is equivalent to choosing an initial subset of trailers $S_1 \subset \mathcal{N}(\mathcal{L})$ that contains x_1^l copies of trailer l. For initial set of trailers S_1, we let $S_1^l \subseteq S_1$ be the subset of type l trailers. With this notation in hand, we can reformulate problem (2) as

$$\max_{S_1 \subset \mathcal{N}(\mathcal{L}) : |S_1| = \tau} \mathbb{E}[V_1^P(S_1)],$$

where the expectation is taken again with respect to the sample path of truck arrivals P. In the proceeding proposition, we show that the submodularity results hold for any sample path of bus arrivals, which in turn immediately gives the more general submodularity of $V_1(x_1)$.

Proposition 2. *For arbitrary sample $P = (b_1, \ldots, b_\tau)$ of truck arrivals over the τ time periods and for any $S \subset \mathcal{N}(\mathcal{L})$ and trailers $i, j \in \mathcal{N}(\mathcal{L}) \setminus S$, we have that*

$$V_1^p(S \cup \{i\}) + V_1^p(S \cup \{j\}) \geq V_1^p(S \cup \{i, j\}) + V_1^p(S).$$

Proof. Throughout the proof, we work under general P and hence to simplify notation, we remove the dependence on P for each of the sets that we create. Let $\mathcal{L}^{i,j} = (l_1^{i,j}, \ldots, l_\tau^{i,j})$ and $\mathcal{L}^\emptyset = (l_1^\emptyset, \ldots, l_\tau^\emptyset)$ be the optimal trailer assignments for initial inventories $S \cup \{i, j\}$ and S respectively. To show the desired result, we construct feasible trailer assignments $\mathcal{L}^i = (l_1^i, \ldots, l_\tau^i)$ and $\mathcal{L}^j = (l_1^j, \ldots, l_\tau^j)$ for initial inventories $S \cup \{i\}$ and $S \cup \{j\}$ respectively that match the sum of the revenue accrued by $\mathcal{L}^{i,j}$ and $\mathcal{L}$. We let $S_t^{i,j}$, S_t^i, S_t^j, and $S_t^\emptyset$ represent the remaining inventory levels at time t for initial inventory levels of $S \cup \{i, j\}$, $S \cup \{i\}$, $S \cup \{j\}$ and S respectively. We show that if the time period t inventories satisfy

- $S_t^{i,j} = S_t^\emptyset \cup \{m_1, m_2\}$
- $S_t^i = S_t^\emptyset \cup \{m_1\}$
- $S_t^j = S_t^\emptyset \cup \{m_2\}$

for trailers $m_1, m_2 \in \mathcal{N}(\mathcal{L})$, then we can choose l_t^i and l_t^j such that $l_t^{i,j} \cup l_t^\emptyset = l_t^i \cup l_t^j$ and

- $S_{t+1}^{i,j} = S_{t+1}^\emptyset \cup \{m_1', m_2'\}$
- $S_{t+1}^i = S_{t+1}^\emptyset \cup \{m_1'\}$
- $S_{t+1}^j = S_{t+1}^\emptyset \cup \{m_2'\}$

for trailers $m_1', m_2' \in \mathcal{N}(\mathcal{L})$. We note that having $l_t^{i,j} \cup l_t^{\emptyset} = l_t^i \cup l_t^j$ for each time period t is enough to show that the sum of the revenue accrued in each time period under initial inventories $S \cup \{i,j\}$ and S is equal to the sum of the revenue accrued in each time period under initial inventories $S \cup \{i\}$ and $S \cup \{j\}$, hence proving the proposition.

The inventory condition is trivially satisfied for time period 1 and hence we prove the result for general time period t. We consider three different cases based on all the possible values that $l_t^{i,j}$ can take on.

Case 1: $l_t^{i,j} = l \in S_t^{\emptyset}$*:* In this case, we must have $l_t^{\emptyset} = l$ due to the fact that a greedy policy is optimal. We can then set $l_t^i = l_t^j = l$ to satisfy $l_t^{i,j} \cup l_t^{\emptyset} = l_t^i \cup l_t^j$. Further, letting, $m_1' = m_1$ and $m_2' = m_2$ and that noting that $S_{t+1}^{\emptyset} = S_t^{\emptyset} \setminus \{l\}$ gives the desired conditions on the inventories for time period $t+1$.

Case 2: $l_t^{i,j} = m_1 \neq l_t^{\emptyset} = k$*:* In this case, setting $l_t^i = m_1$ and $l_t^j = k$ is feasible and satisfies $l_t^{i,j} \cup l_t^{\emptyset} = l_t^i \cup l_t^j$. Further, letting $m_1' = k$ and $m_2' = m_2$ and noting that $S_{t+1}^{\emptyset} = S_t^{\emptyset} \setminus \{k\}$ gives the desired conditions on the inventories for time period $t+1$.

Case 3: $l_t^{i,j} = m_2 \neq l_t^{\emptyset} = k$*:* This case is symmetric to Case 2.

Since $V_1(S_1)$ is trivially monotone due to that fact that additional trailers can be added at no cost, we can apply the classic result of G. L. Nemhauser and Fisher (1978) to find an initial set of trailers that achieves an expected revenue of at least $(1 - \frac{1}{e})Z^*$. The algorithm described in G. L. Nemhauser and Fisher (1978), when applied in our setting, will build the initial inventory by continuously adding the trailer type that gives the greatest marginal gain in revenue. We conclude our analysis in this first setting by showing how we can efficiently approximate the initial trailer selection problem given our results for the the inventory selection and trailer matching problems.

3.3. The Trailer Selection Problem

In this section, we assume that warehouse manager can select trailer types with any weight in the interval $[w_{min}, w_{max}]$. We let $\mathcal{U}$ index the universe of possible trailer with such weights. The restriction that we impose is $w_{min} = 1$. We note that this restriction is not necessary but aids in simplifying the exposition of the gridding approach that we eventually propose. For a chosen set of trailer types $\mathcal{L} \subset \mathcal{U}$, we use $\mathcal{N}(\mathcal{L})$ to denote the set of the τ copies of each trailer type in $\mathcal{L}$. We are now interested in the solving the following problem:

$$OPT = \max_{\mathcal{L} \subset \mathcal{U}} \max_{S_1 \subset \mathcal{N}(\mathcal{L})} V_1(S_1), \tag{5}$$

where OPT is the optimal expected revenue that can be achieved by optimizing over all three problems. We let

$$(\mathcal{L}^*, S_1^*) = \arg\max_{\mathcal{L} \subset \mathcal{U}} \arg\max_{S_1 \subset \mathcal{N}(\mathcal{L})} V_1(S_1),$$

be the optimal set of trailer types and initial inventories.

For fixed $\epsilon > 0$, consider the set $\hat{\mathcal{L}}_\epsilon = \{l \in \mathcal{U} : w_l = (1+\epsilon)^k, k = 0, \ldots, \lceil \log(w_{max})/\log(1+\epsilon) \rceil\}$ of trailer types, where $\lceil\ \rceil$ is the operator that rounds up to the nearest integer. Note that $|\hat{\mathcal{L}}_\epsilon| = O(\log(w_{max})/\epsilon)$. Let

$$Z^*(\hat{\mathcal{L}}_\epsilon) = \max_{S_1 \subset \mathcal{N}(\hat{\mathcal{L}}_\epsilon)} V_1(S_1),$$

represent the optimal expected revenue over the τ truck arrivals when the set of trailer types is fixed at $\hat{\mathcal{L}}_\epsilon$. The following proposition relates OPT to $Z^*(\hat{\mathcal{L}}_\epsilon)$.

PROPOSITION 3. *For any $\epsilon > 0$, we have that $Z^*(\hat{\mathcal{L}}_\epsilon) \geq (1-\epsilon)OPT$.*

Proof. We prove the result by constructing a feasible initial inventory of trailers $\tilde{S}_1 \subset \mathcal{N}(\hat{\mathcal{L}}_\epsilon)$ that satisfies $V_1(\tilde{S}_1) \geq (1-\epsilon)OPT$. To do so, for trailer $l^* \in \mathcal{L}^*$, let $\lfloor l^* \rfloor_\epsilon = \arg\max_{l \in \hat{\mathcal{L}}_\epsilon : w_l \leq w_{l^*}} w_l$ represent w_{l^*} rounded down to the closest trailer weight of any trailer $l \in \hat{\mathcal{L}}_\epsilon$. We then construct $\tilde{S}_1$ by including $|S_1^{l^*}|$ copies of $\lfloor l^* \rfloor_\epsilon$ for each $l^* \in \mathcal{L}^*$. For fixed sample P of truck arrivals, we show that $V_1^P(\tilde{S}_1) \geq (1-\epsilon)V_1^P(S_1^*)$, which establishes the desired claim. To do so, we construct a feasible trailer matching policy under initial inventory $\tilde{S}_1$. In time period t, if $l_t^* \in \mathcal{L}^*$ is the trailer matched when the starting inventory is S_1^*, then match trailer $\tilde{l}_t = \lfloor l_t^* \rfloor_\epsilon$ in time period t when the initial inventory is $\tilde{S}_1$. Under this feasible policy we get that

$$\begin{aligned} V_1^P(\tilde{S}_1) &\geq \sum_{t=1}^{\tau} w_{\tilde{l}_t} r \\ &\geq \frac{1}{1+\epsilon} \sum_{t=1}^{\tau} w_{l_t^*} r \\ &= \frac{1}{1+\epsilon} V_1^P(S_1^*) \\ &\geq (1-\epsilon) V_1^P(S_1^*), \end{aligned}$$

where the second inequality follows by definitions of $\tilde{l}_t$.

For any choice of trailer types $\mathcal{L} \subseteq \mathcal{U}$, we denote the initial inventory obtained by applying the greedy algorithm in G. L. Nemhauser and Fisher (1978) using $\mathcal{N}(\mathcal{L})$ as $\mathcal{G}(\mathcal{N}(\mathcal{L}))$. As a last result in this setting, we give the performance of the initial inventory $G(\mathcal{N}(\hat{\mathcal{L}}_\epsilon))$ in the following theorem

THEOREM 1. *For any $\epsilon > 0$, the initial inventory $G(\mathcal{N}(\hat{\mathcal{L}}_\epsilon))$ satisfies*

$$V_1(G(\mathcal{N}(\hat{\mathcal{L}}_\epsilon))) \geq (1-\epsilon)(1-\frac{1}{e})OPT.$$

Proof. We have that

$$\begin{aligned} V_1(G(\mathcal{N}(\hat{\mathcal{L}}_\epsilon))) &\geq (1-\frac{1}{e}) Z^*(\hat{\mathcal{L}}_\epsilon) \\ &\geq (1-\epsilon)(1-\frac{1}{e})OPT. \end{aligned}$$

The first inequality follows by Proposition 2 and the second inequality follows by Proposition 3.

We conclude this first section with a quick analysis of the computation time needed to find $G(\mathcal{N}(\hat{\mathcal{L}}_\epsilon))$. We assume access to an oracle that can compute $V_1(S_1)$ for any $S_1 \subset \mathcal{N}(\hat{\mathcal{L}}_\epsilon))$ in $O(1)$. Given access to this oracle, the total runtime of the greedy procedure is $O(\tau \frac{\log(w_{max})}{\epsilon})$, which is polynomial in the input and $\frac{1}{\epsilon}$ and hence can be implemented quite efficiently as we later demonstrate in our computational experiments.

4. Second Setting: Scalebacks permitted

In this second setting, we assume that trailer weight can be altered with a fixed per-pound removal cost. The cost reflects operational cost associated with removing beer from preloaded trailers (e.g., labor, space), and we denote the unit cost by c_o. Consequently the revenue for matching trailer l to truck b is given by

$$r_{l,b} = \begin{cases} w_l \cdot r, & \text{if } w_l + w_b \leq 80,000 \\ (80,000 - w_b) \cdot r - [(w_l + w_b) - 80,000] \cdot c_o & \text{o.w.} \end{cases} \tag{6}$$

The term $r_{l,b}$ is the net revenue calculated as the revenue earned from shipping $\min\{w_l, 80,000 - w_b\}$ pounds of beer minus any removal cost if the particular matching results in a violation of the 80,000 lbs. weight limit. In this section, we utilize techniques from approximate dynamic programming to eventually build a linear program whose solution simultaneous approximates all three problems at once. As was done in section 3, we consider each of the three underlying problems sequentially, each time adding components to our linear program. We again initially fix the trailer types $\mathcal{L}$ when considering the trailer matching and inventory selection problem. Then, we show how to optimally choose a reasonable small set of trailer types in the trailer selection problem . Also, we revert back to encoding the remaining inventories in time period t as the vector $x_t \in \mathbb{Z}_+^n$. We begin by addressing the trailer matching problem.

4.1. The Trailer Matching Problem

In this new setting, the greedy policy given in Corollary 1 can easily be shown to be sup-optimal. Nonetheless, we are able to show a similar greedy-like structure on the optimal policy. The following proposition, whose proof we delay to Appendix A due to its similarity to the proof for Proposition 1, allows us characterize the optimal matching policy.

PROPOSITION 4. *For any time period t and inventory $x_t \in \mathcal{X}$, $l_1, l_2 \in \mathcal{L}(x_t)$ such that $l_2 > l_1$, we have that*

$$(w_{l_1} - w_{l_2})c_o \leq V_t(x_t - e_{l_1}) - V_t(x_t - e_{l_2}) \leq (w_{l_2} - w_{l_1})r.$$

Noting that the above proposition holds for any time period t, we get that the optimal trailer to assign to each bus $b \in \mathcal{B}$ in time period t when the inventory is x_t must be either the highest revenue trailer $l \in \mathcal{L}(x_t)$ whose weight satisfies $w_l + w_b \leq 80,000$ or the highest revenue trailer $l \in \mathcal{L}(x_t)$ whose weight satisfies $w_l + w_b > 80,000$. More formally, we let

$$\mathcal{L}_1(x_t, b) = \underset{l \in \mathcal{L}(x_t): w_l + w_b \leq 80,000}{\arg\max} w_l$$

and

$$\mathcal{L}_2(x_t, b) = \underset{l \in \mathcal{L}(x_t): w_l + w_b > 80,000}{\arg\min} w_l.$$

Finally, letting $\mathcal{L}(x_t, b) = \mathcal{L}_1(x_t, b) \cup \mathcal{L}_2(x_t, b)$, the following corollary of Proposition 4 provides some structure on the optimal matching policy.

COROLLARY 2. *For any time period t, inventory level x_t and truck arrival $b \in B$, we have that*

$$\underset{l \in \mathcal{L}(x_t)}{\arg\max} \left\{ r_{l,b} + V_{t+1}(x_t - e_l) \right\} = \underset{l \in \mathcal{L}(x_t, b)}{\arg\max} \left\{ r_{l,b} + V_{t+1}(x_t - e_l) \right\}.$$

It is important to note that even with Corollary 2, in order to find an optimal policy, we still must recursively compute each of the value functions. However, as we will go on to show, this tidbit of structure on the optimal policy plays a critical role within our approximate dynamic programming framework, which we present next.

The approximate dynamic programming framework. Our approach is based on the linear programming formulation of the dynamic program for the trailer matching problem given in (1). The linear program that we consider is given below.

$$\begin{aligned} V_1(x_1) = \min_{\mathcal{V}_t(\cdot), z} \; & \mathcal{V}_1(x_1) && \text{(LP-EXACT)} \\ \text{s.t.} \; & \mathcal{V}_t(x_t) \geq \sum_{b \in \mathcal{B}} p_b \cdot z(b, x_t) && x_t \in \mathcal{X}_t. \\ & z(b, x_t) \geq r_{l^*,b} + \mathcal{V}_{t+1}(x_t - e_{l^*}) && \forall x_t \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t, b). \end{aligned}$$

Note that the two constraints together enforce that

$$\mathcal{V}_t(x_t) \geq \sum_{b \in \mathcal{B}} p_b \max_{l \in \mathcal{L}(x_t, b)} \left\{ r_{l,b} + \mathcal{V}_{t+1}(x_t - e_l) \right\}.$$

The number of constraints and decision variables in LP-EXACT grows exponentially in τ and thus it is computationally intensive to solve directly. To partially side-step this issue, we borrow the approach of Kunnumkal and Talluri (2015) who approximate the value functions through the following piecewise linear approximation

$$\mathcal{V}_t(x_t) \approx \sum_{l \in \mathcal{L}} q_{l,t}(x^l),$$

where $q_{l,t}(x^l)$ can be interpreted as the value of of having x_l type l trailers in time period t. Plugging this approximation into (LP-EXACT) yields the following linear program

$$J(x_1) = \min_{q_{l,t}(\cdot),z} \sum_{l\in\mathcal{L}} q_{l,1}(x_1^l) \qquad \text{(LP-APPROX)}$$

$$\text{s.t.} \sum_{l\in\mathcal{L}} q_{l,t}(x^l) \geq \sum_{b\in\mathcal{B}} p_b \cdot z_{b,x}^t \qquad \forall t, x \in \mathcal{X}_t \qquad (7)$$

$$z_{b,x}^t \geq r_{l^*,b} + \sum_{l\in\mathcal{L}} q_{l,t+1}(x^l - \mathbb{1}_{l=l^*}) \qquad \forall x \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t, b) \qquad (8)$$

Additionally, we incorporate the following constraints, which ensure that the value of each trailer's inventory is marginally decreasing and that more inventory units is always more valuable than fewer.

$$q_{l,t}(y) - q_{l,t}(y-1) \geq q_{l,t}(y+1) - q_{l,t}(y) \qquad \forall t, l \in \mathcal{L}, y \in \{1,\ldots,\tau\} \qquad (9)$$

$$q_{l,t}(y) \geq q_{l,t}(y-1) \qquad \forall t, l \in \mathcal{L}, y \in \{1,\ldots,\tau\} \qquad (10)$$

Not only do these additional constraints help to generate a more accurate approximation, but they will also be critical when we move to using LP-APPROX to approximate the inventory selection problem. We also enforce that $q_{l,\tau+1}(\cdot) = 0$, which is a natural assumption on the terminal condition. Similar to Kunnumkal and Talluri (2015), it turns out that adding these constraints to the piecewise linear approximation does not affect its optimal solution. In Appendix A, we show that these constraints are satisfied by any optimal solution of LP-APPROX (without enforcing these two sets of constraints) given that we can enforce constraint (9) only for $y = 1$. Consequently, for the remainder of this paper, we assume that these constraints are implicitly a part of all future linear program that result from piecewise linear approximations.

An optimal solution $\{q_{l,t}^*(y) : \forall t, l \in \mathcal{L}, y \in \{0,\ldots,\tau\}\}$ to LP-APPROX can be useful in a few ways. First, the optimal objective $J(x_1) = \sum_{l\in\mathcal{L}} q_{l,1}^*(x_1^l)$ provides an upper bound for $V_1(x_1)$. To see this, note that the solution $\mathcal{V}_t(x_t) = \sum_{l\in\mathcal{L}} q_{l,t}^*(x^l)$ is trivially feasible to LP-EXACT and achieves an objective of $J(x_1)$. We then can use this upper bound to measure the efficacy of any approach for the trailer matching problem such as the heuristic policy that results from approximating the value functions $V_t(x_t)$ in (1) with $\sum_{l\in\mathcal{L}} q_{l,t}^*(x^l)$. Finding this optimal solution, however, is no simple task. Note that while the piecewise linear programming approximation helps reduce the number of decision variables to $O(\tau^2 n)$, constraints (7) and (8) are required for every potential inventory vector and hence the number of these constraints grows exponentially in τ. Consequently, in order to solve LP-APPROX, we must develop an efficient way to employ constraint generation.

The constraint generation procedure. In order to efficiently solve LP-APPROX, we develop a simple dynamic program to solve the subproblem that results from a constraint generation approach. Before describing our constraint generation approach, we note that for any inventory vector $x \in \mathcal{X}_t$, constraints (7) and (8) can be combined into the following single nonlinear constraint:

$$\sum_{l\in\mathcal{L}} q_{l,t}(x^l) \geq \sum_{b\in\mathcal{B}} p_b \max_{l^*\in\mathcal{L}(x_t,b)} \left\{ r_{l^*,b} + \sum_{l\in\mathcal{L}} q_{l,t+1}(x^l - \mathbb{1}_{l=l^*}) \right\}. \tag{11}$$

Our constraint generation procedure begins by formulating LP-APPROX with only a small subset of the constraints (7) and (8). We refer to this linear problem as the master linear problem, to which we will sequentially add violated constraints. In each iteration of the constraint generation, we solve the master problem, which yields optimal decision variables $\{\hat{q}_{l,t}(y) : \forall t, l \in \mathcal{L}, y \in \{0, \dots, \tau\}\}$. Given this optimal solution, we can find a violated constraint by solving the following subproblem for each time period t.

$$\theta_t^* = \max_{x\in\mathcal{X}_t} \theta_t(x), \tag{CG}$$

where

$$\theta_t(x) = \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\hat{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} (\hat{q}_{l,t+1}(x^l) - \hat{q}_{l,t}(x^l)). \tag{12}$$

We use the shorthand $\Delta\hat{q}_{l,t+1}(x^l) = \hat{q}_{l,t+1}(x^l) - \hat{q}_{l,t+1}(x^l - 1)$ to represent the marginal value of the trailer l's x^l-th unit in time period t. Note that (12) is obtained by subtracting $\sum_{l\in\mathcal{L}} \hat{q}_{l,t+1}(x^l)$ from both sides of (11) and noting that $\sum_{b\in\mathcal{B}} p_b = 1$. If $\theta_t^* > 0$, then the associated constraint for inventory vector

$$x^* = \arg\max_{x\in\mathcal{X}_t} \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\hat{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} (\hat{q}_{l,t+1}(x^l) - \hat{q}_{l,t}(x^l)),$$

is violated. We then add this constraint to the master linear program and repeat the procedure.

The main difficulty in solving subproblem (CG) is that the inner maximization over the trailers to assign to each potential arriving bus $b \in \mathcal{B}$ can only seemingly be determined once the inventory vector $x \in \mathcal{X}_t$ is fully specified. However, by exploiting the special structure of the two trailers specified by $\mathcal{L}(x, b)$, we are able to derive a dynamic programming formulation of (CG) that can be solved in polynomial time.

Before giving our dynamic programming formulation, we introduce an alternative, representation of any inventory vector, which helps illuminate the recursive manner in which we solve (CG). Any inventory vector $x \in \mathbb{Z}_+^n$ can equivalently be represented as a vector indicating trailer types in $\mathcal{L}(x)$ with non-zero inventory, which we denote as $(l^{(1)}, \dots, l^{(\gamma)})$. In addition, we must also specify the corresponding inventory levels of each of these trailer types. We do so using the vector $(x^{l^{(1)}}, \dots, x^{l^{(\gamma)}})$. With regards to these two vectors, we have that $\gamma(x) = \sum_{l=1}^n \mathbb{1}_{\{x^l>0\}}$ gives the total number of trailers types with non-zero inventory and we use the convention that $l^{(i)} \in \mathcal{L}(x)$ gives the i-th smallest indexed trailer in $\mathcal{L}(x)$ with inventory level $x^{l^{(i)}}$. Unless otherwise stated, we work under a fixed inventory vector x, and hence to avoid cluttered notation, we drop the dependence of the above parameters on x. In addition to trailer type 0, we introduce another dummy trailer type $n+1$. We set $l^{(0)}=0$, $l^{(\gamma+1)}=n+1$ and also enforce that $\hat{q}_{0,t}(\cdot)=\hat{q}_{n+1,t}(\cdot)=0$ and $r_{n+1,b}=-\infty$ which makes sure trailer type $n+1$ will never be assigned to any truck.

Given this notation, we know that if $\mathcal{L}_1(x,b)=l^{(i)}$, then we must have $\mathcal{L}_2(x,b)=l^{(i+1)}$ for any truck arrival $b\in B$. Consequently, for trailers $l^{(i)},l^{(i+1)}\in\mathcal{L}(x)$ we define $B(l^{(i)},l^{(i+1)})=\{b\in\mathcal{B}: 80,000-w_{l^{(i+1)}}<w_b\le 80,000-w_{l^{(i)}}\}$ to be the buses that satisfy $\mathcal{L}_1(x,b)=l^{(i)}$ and $\mathcal{L}_2(x,b)=l^{(i+1)}$. Noting that

$$B=\cup_{i=0}^{\gamma}B(l^{(i)},l^{(i+1)}),$$

it is easy to see that (12) can be rewritten as

$$\theta_t(x)=\sum_{i=0}^{\gamma}\left\{\sum_{b\in B(l^{(i)},l^{(i+1)})}p_b\max_{l\in\{l^{(i)},l^{(i+1)}\}}\left\{r_{l,b}-\Delta\hat{q}_{l,t+1}(x^l)\right\}+\right.$$
$$\left.\sum_{l^{(i)}<l<l^{(i+1)}}[\hat{q}_{l,t+1}(0)-\hat{q}_{l,t}(0)]+[\hat{q}_{l^{(i+1)},t+1}(x^{l^{(i+1)}})-\hat{q}_{l^{(i+1)},t}(x^{l^{(i+1)}})]\right\}.$$

For any $i\in\{0,\dots,\gamma\}$, the inner sum can be computed with only knowledge of the trailers $l^{(i)}$ and $l^{(i+1)}$ and their respective inventory levels $x^{l^{(i)}}$ and $x^{l^{(i+1)}}$. Consequently, to help with proving the correctness of the dynamic program that we present for (CG), we define

$$\begin{aligned}\theta_t(l^{(k)},\dots,l^{(\gamma)},x^{l^{(k)}},\dots,x^{l^{(\gamma)}})&=\sum_{i=k}^{\gamma}f(l^{(i)},x^{l^{(i)}},l^{(i+1)},x^{l^{(i+1)}})\\&=f(l^{(k)},x^{l^{(k)}},l^{(k+1)},x^{l^{(k+1)}})+\theta_t(l^{(k+1)},\dots,l^{(\gamma)},x^{l^{(k+1)}},\dots,x^{l^{(\gamma)}}),\end{aligned}$$

where we define $\theta_t(n+1,\cdot)=0$ and

$$f(l,x,l',x')=\sum_{b\in B(l,l')}p_b\max_{l^*\in\{l,l'\}}\left\{r_{l^*,b}-\mathbb{1}_{l^*=l}\Delta\hat{q}_{l,t+1}(x)-\mathbb{1}_{l^*=l'}\Delta\hat{q}_{l,t+1}(x')\right\}+$$
$$\sum_{l<l<l'}[\hat{q}_{l,t+1}(0)-\hat{q}_{l,t}(0)]+[\hat{q}_{l',t+1}(x')-\hat{q}_{l',t}(x')]$$

for any trailers $l,l'\in\mathcal{L}$ that satisfy $l<l'$ and non-zero inventories $x,x'>0$. The term $\theta_t(l^{(k)},\dots,l^{(\gamma)},x^{l^{(k)}},\dots,x^{l^{(\gamma)}})$ can be interpreted as the contribution of the $\gamma-k$ heaviest trailers in $\mathcal{L}(x)$ to (12) and hence we have that $\theta_t(x)=\theta_t(l^{(0)},\dots,l^{(\gamma)},x^{l^{(0)}},\dots,x^{l^{(\gamma)}})$.

The Bellman equations of our dynamic programming formulation of (CG) are presented below

$$J(l,x,c)=\max_{l<l'\le n+1,\,x'>0}f(l,x,l',x')+J(l',x',c+x'\mathbb{1}_{\{l'\neq n+1\}})\tag{CG-DP}$$

where the elements of the state space have the following meaning.

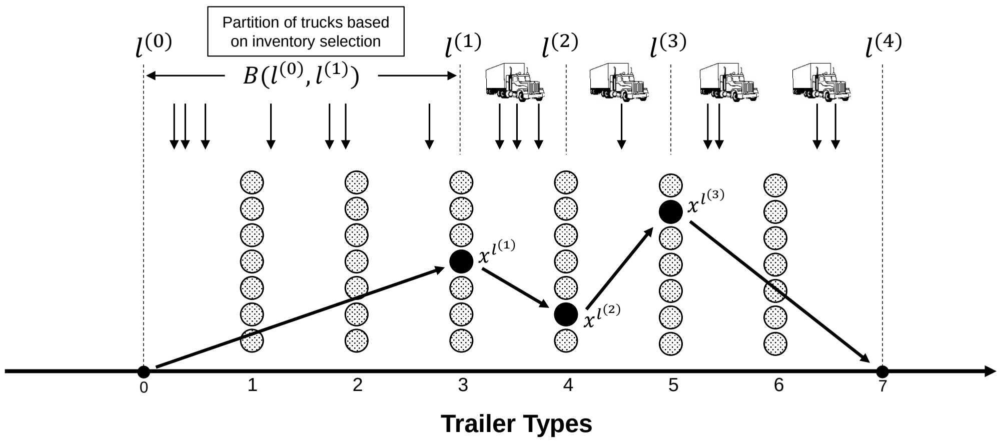

Figure 1 Illustration of Dynamic Program (CG-DP)

In this example with $n = 6$, each circle represents a pair (l, x) - trailer type and its inventory level. An edge between two dark circles represents the step reward which is given by $f(l, x, l', x')$. A chosen sequence of trailer types and inventory levels as highlighted in this picture is $(3, 4), (4, 2), (5, 6)$. At the terminal node 7, the total assigned inventory is $\sum_{i=1}^{3} x^{l(i)} = 12$ which is used to assess feasibility to the original problem. The optimal solution given by the dynamic program corresponds to the maximum revenue path from circle $(0, 0)$ to $(7, 0)$.

- $l \leq n+1$: a trailer $l \in \mathcal{L}$.
- x: the inventory level of trailer l.
- c: total inventories assigned up to trailer l (exclusive of virtual trailer $n+1$).

Additionally, we have the following two terminal conditions. For $\tau - (t-1) \leq c \leq \tau$, we have that $J(n+1, \cdot, c, \cdot) = 0$ and for $c < \tau - (t-1)$ and $c > \tau$ we have that $J(n+1, \cdot, c, \cdot) = -\infty$. The latter case ensures that we choose inventory levels across all trailers that are feasible for time period t. Since the inventory state in period t must satisfy $\tau - (t-1) \leq \sum_{l=1}^{n} x_t^l \leq \tau$, we need to keep track of cumulative inventory assigned in our dynamic program and impose a high penalty at terminal state to enforce feasibility. Figure 1 illustrates the dynamic program for $n = 6$.

The following proposition proves the correctness of our dynamic programming formulation.

Proposition 5. *For any $l \in \mathcal{L}(x)$, $c, x \geq 0$ and $1 \leq l \leq n$ we have that*

$$J(l, x, c) = \max_{\substack{(l^{(k+1)}, \ldots, l^{(\gamma)}, x^{l^{(k+1)}}, \ldots, x^{l^{(\gamma)}}): \\ \tau - (t-1+c) \leq \sum_{i=k+1}^{\gamma} x^{l^{(k)}} \leq \tau - c}} \theta_t(l^{(k)} = l, l^{(k+1)}, \ldots, l^{(\gamma)}, x^{l^{(k)}} = x, x^{l^{(k+1)}}, \ldots, x^{l^{(\gamma)}}).$$

Proposition 5, which is proved in Appendix A, immediately yields $\theta_t^* = J(0, 0, 0)$. Next, we consider overall runtime of our dynamic programming approach. The sets $B(l_1, l_2)$ can be precomputed in $O(n^2 m)$ since there are $O(n^2)$ pairs of trailers $l_1, l_2 \in \mathcal{L}$ and m total trucks. Each value function

$J(\cdot,\cdot,\cdot)$ can computed in $O(n\tau)$ by simply enumerating over all possible choices for l' and x'. Further, since the dynamic program has at most $O(n\tau^2)$ states, the total runtime to solve problem (CG) is $O(n^2(\tau^3+m))$.

We conclude this section by showing how we can obtain an upper bound on $J(x_1)$ at each iteration of the constraint generation using the values of θ_t^*. This upper bound allows us to compute an optimality gap that can be used as a stoping criterion during constraint generation. Recall that we denote the solution to the master problem in each iteration by $\{\hat{q}_{l,t}(y) : \forall t, l \in \mathcal{L}, y \in \{0,\ldots,\tau\}\}$ whose corresponding objective value we denote as $\hat{J}(x_1)$. The upper bound presented in the following proposition resemble the bound presented in Adelman (2007) for the affine approximations to the value functions in the network revenue management setting, however the proof that we present for establishing these bounds is quite different from Adelman (2007).

Proposition 6. *The optimal solution $J(x_1)$ to (LP-APPROX) is bounded by:*

$$\hat{J}(x_1) \le J(x_1) \le \hat{J}(x_1) + \sum_{t=1}^{\tau} \theta_t^* \tag{13}$$

ff $\hat{J}(x_1) \le J(x_1)$ is trivial since the master problem in each iteration employs a subset of the constraints in (LP-APPROX) which yields a lower bound. To prove $J(x_1) \le \hat{J}(x_1) + \sum_{t=1}^{\tau} \theta_t^*$, by (CG) and (12), we have

$$\theta_t^* \ge \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\hat{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} (\hat{q}_{l,t+1}(x^l) - \hat{q}_{l,t}(x^l)).$$

for any t, l, x^l, which is equivalent to

$$\theta_t^* + \sum_{l\in\mathcal{L}} \hat{q}_{l,t}(x^l) \ge \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\hat{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} \hat{q}_{l,t+1}(x^l) \tag{14}$$

for any $t \in \{1.\cdots,\tau\}$, $x \in \mathcal{X}_t$. We construct a feasible solution $\{\bar{q}_{l,t}(y) : \forall t, l \in \mathcal{L}, y \in \{0,\ldots,\tau\}\}$ to problem (LP-APPROX) as following:

$$\bar{q}_{l,t}(x^l) \triangleq \hat{q}_{l,t}(x^l) + \frac{1}{n}\sum_{s=t}^{\tau} \theta_s^*$$

for any t, l, x^l. To show $\bar{q}_{l,t}(x^l)$ is a feasible solution, we just need to verify that (9), (10) and (11) hold.

First, (9) and (10) hold trivially since the additional part $\frac{1}{n}\sum_{s=t}^{\tau} \theta_s^*$ cancels out on both sides of the constraints in (9) and (10). Next we show (11) holds for all t, l, x^l, which is

$$\sum_{l\in\mathcal{L}} \bar{q}_{l,t}(x^l) \ge \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\bar{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} \bar{q}_{l,t+1}(x^l)$$

where $\Delta\bar{q}_{l,t+1}(x^l) = \bar{q}_{l,t+1}(x^l) - \bar{q}_{l,t+1}(x^l - 1)$. Based on the definition, we have

$$\begin{aligned}
\sum_{l\in\mathcal{L}} \bar{q}_{l,t}(x^l) &= \sum_{l=1}^{n} \left(\hat{q}_{l,t}(x^l) + \frac{1}{n}\sum_{s=t}^{\tau} \theta_s^* \right) \\
&= \sum_{l=1}^{n} \hat{q}_{l,t}(x^l) + \theta_t^* + \sum_{s=t+1}^{\tau} \theta_s^* \\
&\geq \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\hat{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} \hat{q}_{l,t+1}(x^l) + \sum_{s=t+1}^{\tau} \theta_s^* \\
&= \sum_{b\in\mathcal{B}} p_b \max_{l\in\mathcal{L}(x_t,b)} \left\{ r_{l,b} - \Delta\bar{q}_{l,t+1}(x^l) \right\} + \sum_{l\in\mathcal{L}} \bar{q}_{l,t+1}(x^l)
\end{aligned}$$

The inequality holds by (14) and the rest are by definition of $\bar{q}^*_{l,t}(x^l)$.

The objective value obtained at this feasible solution is

$$\sum_{l\in\mathcal{L}} \bar{q}_{l,1}(x_1^l) = \sum_{l\in\mathcal{L}} \hat{q}_{l,1}(x_1^l) + \sum_{s=1}^{\tau} \theta_s^* = \hat{J}(x_1) + \sum_{t=1}^{\tau} \theta_t^*.$$

Since it is a feasible solution, we have $J(x_1) \leq \hat{J}(x_1) + \sum_{t=1}^{\tau} \theta_t^*$.

4.2. The Inventory Selection Problem

In this section, we show that the linear program that arises through our piecewise linear approximation to the value functions can be extended to select the initial inventory levels for fixed set of trailer types $\mathcal{L}$. To do so, we first extend LP-EXACT as follows

$$\begin{aligned}
Z^* = &\min_{\mathcal{V}_t(\cdot),z,Q} Q && \text{(EXACT-INV)} \\
\text{s.t. } & Q \geq \mathcal{V}_1(x_1) \qquad \forall x_1 \in \mathcal{X}_1 && (15) \\
& \mathcal{V}_t(x_t) \geq \sum_{b\in\mathcal{B}} p_b \cdot z_{b,x}^t \qquad \forall t, x \in \mathcal{X}_t \\
& z(b,x_t) \geq r_{l^*,b} + \mathcal{V}_{t+1}(x_t - e_{l^*}) \qquad \forall x_t \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t,b).
\end{aligned}$$

It is easy to see that the optimal objective to EXACT-INV is the optimal revenue of the inventory selection problem given in (2) since the addition of constraint in (15) ensures that $Q = \max_{x_1\in\mathcal{X}_1} \mathcal{V}_1(x_1)$ at optimality. Plugging the piecewise linear approximation into EXACT-INV yields the following second version of our approximate linear program

$$\begin{aligned}
J^* = &\min_{q_{l,t}(\cdot),z,Q} Q && \text{(APPROX-INV)} \\
\text{s.t. } & Q \geq \sum_{l\in\mathcal{L}} q_{l,1}(x_1^l) \qquad \forall x_1 \in \mathcal{X}_1 && (16) \\
& \sum_{l\in\mathcal{L}} q_{l,t}(x^l) \geq \sum_{b\in\mathcal{B}} p_b \cdot z_{b,x}^t \qquad \forall t, x \in \mathcal{X}_t. \\
& z_{b,x}^t \geq r_{l^*,b} + \sum_{l\in\mathcal{L}} q_{l,t+1}(x^l - \mathbb{1}_{l=l^*}) \qquad \forall x \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t,b).
\end{aligned}$$

The inventory vector $x_1^* = \arg\max_{x_1 \in \mathcal{X}_1} \sum_{l \in \mathcal{L}} q_{l,1}(x_1^l)$ can then be selected as the initial inventory. Further, it is again trivial to see that $J^* \geq Z^*$ and hence J^* can be used to bound the optimal gap of any policy for the inventory selection problem.

The tractability of APPROX-INV. Since the constraint generation procedure described in the previous section does not apply to the constraints in (16), we no longer have an efficient way to solve APPROX-INV since $|\mathcal{X}_1|$ is exponential in τ. In what follows, we show that we can formulate an equivalent version of APPROX-INV where the constraints in (16) are recast as equivalent set of just $O(n\tau)$ constraints. This more concise version linear program is given below.

$$
\begin{aligned}
J_c^* = \min_{q_{l,t}(\cdot), z, \alpha_l, \beta} & \sum_{l \in \mathcal{L}} \alpha_l + \tau\beta && && \text{(CONCISE-INV)} \\
\text{s.t. } & \alpha_l + x^l \beta \geq q_{l,1}(x^l) && \forall x^l \in \{0, \dots, \tau\}, l \in \mathcal{L} && (17) \\
& \sum_{l \in \mathcal{L}} q_{l,t}(x^l) \geq \sum_{b \in \mathcal{B}} p_b \cdot z_{b,x}^t && \forall t, x \in \mathcal{X}_t. \\
& z_{b,x}^t \geq r_{l^*,b} + \sum_{l \in \mathcal{L}} q_{l,t+1}(x^l - \mathbb{1}_{l=l^*}) && \forall x \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t, b).
\end{aligned}
$$

First, we show that the two linear programs have the same optimal objective value and then we show how to recover x_1^* from CONCISE-INV. We remind the reader that constraints (9) and (10) continue to be enforced. The following proposition accomplishes the first task.

Proposition 7. $J_c^* = J^*$.

W e first show that $J_c^* \leq J^*$. Let Q^*, $\{q_{l,t}^*(x^l) : \forall t, l \in \mathcal{L}, x^l \in \{0, \dots, \tau\}\}$, and $\{z_{b,x}^{*t} : \forall t, x \in \mathcal{X}_t, b \in \mathcal{B}\}$ be the optimal solution to APPROX-INV. We construct a feasible solution $\hat{\beta}$, $\{\hat{\alpha}_l : \forall l \in \mathcal{L}\}$, $\{\hat{q}_{l,t}(x^l) : \forall t, l \in \mathcal{L}, x^l \in \{0, \dots, \tau\}\}$, and $\{\hat{z}_{b,x}^t : \forall t, x \in \mathcal{X}_t, b \in \mathcal{B}\}$ to CONCISE-INV that achieves an objective of J^* To start, we set $\hat{q}_{l,t}(x^l) = q_{l,t}^*(x^l)$ and $\hat{z}_{b,x}^t = z_{b,x}^{*t}$ and hence we trivially satisfy the bottom two constraints of CONCISE-INV. Next, we go about choosing $\hat{\beta}$, $\{\hat{\alpha}_l : \forall l \in \mathcal{L}\}$ that is feasible for constraint (17) and satisfies $\sum_{l \in \mathcal{L}} \hat{\alpha}_l + \tau\hat{\beta} = Q^* = J^*$.

Recall that we have defined $x_1^* = \arg\max_{x_1 \in \mathcal{X}_1} \sum_{l \in \mathcal{L}} q_{l,1}^*(x_1^l)$. We set

$$\hat{\beta} = \max_{l \in \mathcal{L}} q_{l,1}^*(x_1^{*l} + 1) - q_{l,1}^*(x_1^{*l}), \quad \hat{\alpha}_l = q_{l,t}(x_1^{*l}) - x_1^{*l}\hat{\beta}$$

and define

$$l_\Delta = \arg\max_{l \in \mathcal{L}} q_{l,1}^*(x_1^{*l} + 1) - q_{l,1}^*(x_1^{*l}).$$

If the solution defined above is feasible, then its objective value satisfies

$$
\begin{aligned}
J_c^* &\leq \sum_{l \in \mathcal{L}} (q_{l,t}^*(x_1^{*l}) - x_1^{*l}\hat{\beta}) + \hat{\beta}\tau \\
&= \sum_{l \in \mathcal{L}} (q_{l,t}^*(x_1^{*l}) \\
&= J^*,
\end{aligned}
$$

where the inequality follows because $\sum_{l\in\mathcal{L}} x_1^{*l} = \tau$. To show that our proposed solution satisfies constraint (17), we define

$$\tilde{x}^l = \underset{x^l\in\{0,\dots,\tau\}}{\arg\max}\; q_{l,1}^*(x^l) - x^l\hat{\beta}$$

and we show that $\tilde{x}^l = x_1^{*l}$ for each $l\in\mathcal{L}$.

First, we assume by way of contradiction, that there exists a $k\in\mathcal{L}$ such that $\tilde{x}^k > x_1^{*k}$. In this case, we have that $q_{k,1}^*(\tilde{x}^k) - q_{k,1}^*(x_1^{*k}) > (\tilde{x}^k - x_1^{*k})\hat{\beta} \ge \hat{\beta}$, which is not possible given our definition of $\hat{\beta}$. Next, we again assume by way of contradiction that there exists $k\in\mathcal{L}$ such that $\tilde{x}^k < x_1^{*k}$. In this case, we get that

$$\begin{aligned}\hat{\beta} = q_{l_\Delta,1}(x_1^{*l_\Delta}+1) - q_{l_\Delta,1}(x_1^{*l_\Delta}) &> \frac{q_{k,1}^*(x_1^{*k}) - q_{k,1}^*(\tilde{x}^k)}{(x_1^{*k}-\tilde{x}^k)}\\ &\ge \min_{x\in\{\tilde{x}^k+1,\dots,x_1^{*k}\}} q_{k,1}^*(x) - q_{k,1}^*(x-1)\\ &= q_{k,1}^*(x_1^{*k}) - q_{k,1}^*(x_1^{*k}-1).\end{aligned}$$

The last inequality follows due to constraint (9). With this is hand, we elucidate the contradiction by first noting that $k\ne l_\Delta$, since otherwise we would have violated (9). Consequently, we have that

$$\begin{aligned}Q^* &= \sum_{l\in\mathcal{L}} q_{l,1}^*(x_1^{*l})\\ &< \sum_{l\in\mathcal{L}} q_{l,1}^*(x_1^{*l}) + (q_{l_\Delta,1}(x_1^{*l_\Delta}+1) - q_{l_\Delta,1}^*(x_1^{*l_\Delta})) - (q_{k,1}^*(x_1^{*k}) - q_{k,1}^*(x_1^{*k}-1))\\ &= \sum_{l\in\mathcal{L}\setminus\{l_\Delta,k\}} q_{l,1}^*(x_1^{*l}) + q_{l_\Delta,1}^*(x_1^{*l_\Delta}+1) + q_{k,1}^*(x_1^{*k}-1).\end{aligned}$$

So that constraint (16) is violated for initial inventory vector x_1 that satisfies

$$x_1^l = \begin{cases} x_1^{*l}, \text{ if } l\notin\{l_\Delta s, k\}\\ x_1^{*l}-1, \text{if } l=k\\ x_1^{*l}+1, \text{if } l=l_\Delta.\end{cases} \tag{18}$$

Next, we show that $J^*\le J_c^*$. To do so, we start with an optimal solution β^*, $\{\alpha_l^* : \forall l\in\mathcal{L}\}$, $\{q_{l,t}^*(x^l) : \forall t, l\in\mathcal{L},\ x^l\in\{0,\dots,\tau\}\}$, and $\{z_{b,x}^{*t} : \forall t, x\in\mathcal{X}_t, b\in\mathcal{B}\}$ to CONCISE-INV and then construct a feasible solution $\hat{Q}$, $\{\hat{q}_{l,t}(x^l) : \forall t, l\in\mathcal{L},\ x^l\in\{0,\dots,\tau\}\}$, and $\{\hat{z}_{b,x}^t : \forall t, x\in\mathcal{X}_t, b\in\mathcal{B}\}$ to APPROX-INV that has objective $J_c^* = \sum_{l\in\mathcal{L}}\alpha_l^* + \tau\beta^*$. We again set $\hat{q}_{l,t}(x^l) = q_{l,t}^*(x^l)$ and $\hat{z}_{b,x}^t = z_{b,x}^{*t}$ and hence we trivially satisfy the bottom two constraints of APPROX-INV. Next, we set $\hat{Q} = \alpha_l^* + \tau\beta^*$ and that if this assignment is feasible, we get that $J^*\le\hat{Q} = J_c^*$. To show that constraint ((16)) is satisfied, consider arbitrary $x_1\in\mathcal{X}_1$ and note that

$$\sum_{l\in\mathcal{L}} q_{l,1}^*(x_1^l) \le \sum_{l\in\mathcal{L}} \alpha_l^* + x_1^l\beta^*$$

$$
\begin{aligned}
&= \tau \beta^* + \sum_{l \in \mathcal{L}} \alpha_l^* \\
&= \hat{Q},
\end{aligned}
$$

where the inequality follows since for any $x_1 \in \mathcal{X}_1$, we must have that $\sum_{l \in \mathcal{L}} x_1^l = \tau$.

While Proposition 7 establishes that we recover the upper bound for the inventory selection problem J^* by solving CONCISE-INV, it does not immediately give us a way to recover the initial inventory vector x_1^*. However, given optimal decision variables $\{q_{l,1}^*(x^l) : \forall l \in \mathcal{L},\, x^l \in \{0, \ldots, \tau\}\}$ to CONCISE-INV, we show that we can recover x_1^* by solving the following linear program, which is a special case of the classic multiple choice knapsack problem (MCKP) described in Sinha and Zoltners (1979).

$$
\begin{aligned}
\max_{y_{l,x} \in [0,1]} \quad & \sum_{l \in \mathcal{L}} \sum_{x=0}^{\tau} q_{l,1}^*(x) y_{l,x} && \text{(KNAP)} \\
\text{s.t.} \quad & \sum_{x=0}^{\tau} y_{l,x} = 1 && \forall l \in \mathcal{L} \setminus \{0\} \\
& \sum_{l \in \mathcal{L} \setminus \{0\}} \sum_{x=0}^{\tau} x y_{l,x} = \tau.
\end{aligned}
$$

While we allow $y_{l,x}$ to vary continuously over the interval $[0,1]$, we eventually show that there always exists an optimal integral solution. As a result, the decision variable $y_{l,t}$ can be interpreted as an indicator of whether or not we set the initial inventory level of $l \in \mathcal{L}$ to x. The first set of constraints in KNAP ensure that we only choose a single inventory level for each trailer type while the second constraint ensures that we choose exactly τ trailers to preload across all trailer types.

We begin with a structural result regarding the number of fractional decision variables of an optimal solution to KNAP. Let $y_{l,t}^*$ be the optimal decision variables to this linear program and $\mathcal{F} = \{(l,x) : 0 < y_{l,x}^* < 1\}$ give the tuples of indices for fractional optimal decision variables. The following lemma restates a classic result of Sinha and Zoltners (1979), who show that the optimal solution of the linear programming relaxation of any MCKP must contain at most two fractional variables. For completeness, we include the proof in Appendix A.

Lemma 1. *The optimal decision variables to the linear programming relaxation of KNAP must satisfy $|\mathcal{F}| \leq 2$. Moreover if $(l_1, x_1), (l_2, x_2) \in \mathcal{F}$, then we must have $l_1 = l_2$.*

Building on this general result for the MCKP, we show that for our special case, the linear programming relaxation is tight.

Lemma 2. *There exists optimal decision variables to the linear programming relaxation of KNAP that satisfies $|\mathcal{F}| = 0$.*

L emma 1 states that the optimal decision variables of KNAP must satisfy $|\mathcal{F}| < 2$. The first constraint of KNAP ensures that $|\mathcal{F}| \neq 1$. To eliminate the only remaining case, we assume by way of contradiction that $|\mathcal{F}| = 2$, where y^*_{l,x_1} and y^*_{l,x_2} are the two fraction decision variables. We define $x^* = x_1 y^*_{l,x_1} + x_2 y^*_{l,x_2}$. Since x^* is a convex combination of x_1 and x_2, we know that $x_1 \leq x^* \leq x_2$. Further, since $\sum_{l\in\mathcal{L}} \sum_{x=1}^{\tau} x y^*_{l,t} = \tau$, we must have that $x^* \in \mathbb{Z}_+$. As a result, the alternative solution, which sets $\hat{y}_{l,x^*} = 1, \hat{y}_{l,x_1} = \hat{y}_{l,x_2} = 0$ and $\hat{y}_{l,x} = y^*_{l,x}$ otherwise, is feasible to KNAP. Further, since the $q^*_{l,1}(x)$ are marginally decreasing in x, we must have that $q^*_{l,1}(x^*) \geq q^*_{l,1}(x_1) y^*_{l,x_1} + v_{l,1}(x_2) y^*_{l,x_2}$, which shows that this new solution achieves at least as high an objective value while maintaining integrality .

Let $\{y^*_{l,x} : \forall l \in \mathcal{L}, x \in \{0,\ldots,\tau\}\}$ denote the optimal decision variables of KNAP. Lemma 2 shows choosing an initial inventory vector that satisfies $x^l_1 = \sum_{l\in\mathcal{L}\setminus\{0\}} \sum_{x=0}^{\tau} x y^*_{l,x}$, we recover recover x^*_1. We conclude our theoretical analysis with a simple extension of our approach to approximate the trailer selection problem.

4.3. The Trailer Selection Problem

In this section, we assume again that ABI can select trailer types with any weight in the interval $[w_{min}, w_{max}]$. We let $\mathcal{U}$ index the universe of possible trailer with such weights. We still impose the restriction that $w_{min} = 1$, which is again not necessary, but dramatically simplifies our exposition. We now write the objective of EXACT-INV as a function of our choice of the trailer types $\mathcal{L}$

$$\begin{aligned}
Z^*(\mathcal{L}) = \min_{\mathcal{V}_t(\cdot),z,Q} \; & \mathcal{V}_1(x_1) \\
\text{s.t. } & Q \geq \mathcal{V}_1(x_1) && \forall x_1 \in \mathcal{X}_1 \\
& \mathcal{V}_t(x_t) \geq \sum_{b\in\mathcal{B}} p_b \cdot z^t_{b,x} && \forall t, x \in \mathcal{X}_t \\
& z(b, x_t) \geq r_{l^*,b} + \mathcal{V}_{t+1}(x_t - e_{l^*}) && \forall x_t \in \mathcal{X}_t, b \in \mathcal{B}, l^* \in \mathcal{L}(x_t, b).
\end{aligned}$$

The trailer selection problem can then be written as

$$OPT = \max_{\mathcal{L}\subseteq\mathcal{U}} Z^*(\mathcal{L}).$$

For fixed $\epsilon > 0$, we again consider the set $\hat{\mathcal{L}}_\epsilon = \{l \in \mathcal{U} : w_l = (1+\epsilon)^k, k = 0,\ldots,\lceil \log(w_{max})/\log(1+\epsilon)\rceil\}$ of trailer types, where $\lceil\ \rceil$ is the operator that round up to the nearest integer. Our final theorem shows an an upper bound for OPT can be derived by solving CONCISE-INV with $\mathcal{L} = \hat{\mathcal{L}}_\epsilon$.

Theorem 2. *Let $J^{*\epsilon}_c$ be the optimal solution of CONCISE-INV when with $\mathcal{L} = \hat{\mathcal{L}}_\epsilon$, for any $\epsilon > 0$. We have that*

$$J^{*\epsilon}_c (1+\epsilon) \geq OPT.$$

W e have that $J_c^{*\epsilon} \geq Z(\hat{L}_\epsilon)$ and by Proposition 3, we have that $Z(\hat{L}_\epsilon)(1+\epsilon) \geq OPT$. Combining these two claims yields the desired claim.

Moreover, we can obtain $J_c^{*\epsilon}$ in time that is polynomial in the input and $\frac{1}{\epsilon}$ based on our discussion throughout this section. The remaining sections are devoted to testing the efficacy of the algorithms proposed and the tightness of the upper bounds developed on real truck arrival data from AB.

5. Computational Experiments

In this section, we test the efficacy of the approaches developed above for the ABI trailer problem using real data from two brewery warehouses in Cartersville, Georgia (CRTV) and Fort Collins, Colorado (FCL). We only consider instances of the ABI trailer problem where scalebacks are allowed. We begin by providing a detailed overview of the data set given to us, which guides our parameter selection and gives a sense of the scale of the problems that we are able to solve.

ABI Data Description. For each of the two warehouses, we have access to a bevy of information that allows us to simulate realistic instances of the ABI trailer problem. More specifically, we have been given historical third party truck arrival data from February to July of 2016 at both warehouses. For each truck arrival, we have a timestamp giving the truck's arrival time and date, the weight of the truck and its associated carrier, which denotes the unique third party delivery service to which it belongs. Since ABI typically preloads a distinct set of trailers for each carrier at the beginning of the day, we solve separate ABI trailer problems for each warehouse-carrier pair. The distribution $\mathcal{B}$ of truck arrivals for each carrier is taken to be the empirical distribution derived from the full six month history of arrivals. In creating this distribution for each carrier, we round the weights of the trucks to the closest multiple of 100 to keep the cardinality of $\mathcal{B}$ tractable. Since the average weight of each truck exceeds 18,000 lbs., this rounding is not likely to have a dramatic effect on our results. The number of arrivals τ is assumed to be the maximum number of arrivals that are observed in any single day day over the six months of historical arrivals. Finally, estimates of the revenue per pound r and the overage cost c_o were given to us for each carriers. A full summary of these parameters for each carrier at each warehouse is given in Tables 1a and 1b.

Computational setup. For each carrier at each warehouse, we test the following three policies for the corresponding ABI trailer problem, where a policy refers to an approach to solve the inventory selection and the trailer matching problems. To choose the collection of trailer types , we use the exponential spaced set of grid points over the interval $[80K - \max_{b\in\mathcal{B}} w_b, 80K - \min_{b\in\mathcal{B}} w_b]$ for both policies. It is easy to see that this range of potential trailer types includes the smallest and largest trailer weights that one would ever consider choosing. We created instances of this exponential grid with $\epsilon \in \{0.02, 0.05\}$. For $\epsilon = 0.05$, the average cardinality over all carriers of $\hat{\mathcal{L}}_{0.05}$ was 2.83 and 3.18 for CRTV and FCL respectively. For $\epsilon = 0.02$, these numbers increased to 4.92 and 5.54 for warehouses CRTV and FCL respectively.

carrier	τ	$\lvert\mathcal{B}\rvert$	c_o	r
AHLY	13	25	0.047	0.006
WENX	34	33	0.047	0.012
GTGA	23	30	0.047	0.007
PRIJ	33	38	0.047	0.015
MTNF	22	24	0.047	0.008
WENL	13	25	0.047	0.012
TAMI	13	23	0.047	0.020
AIOE	12	38	0.047	0.015
WSXI	16	24	0.047	0.008
PASC	11	27	0.047	0.009
MTEN	10	33	0.047	0.009
JBHI	12	41	0.047	0.018

(a) CRTV

carrier	τ	$\lvert\mathcal{B}\rvert$	c_o	r
WERD	38	35	0.050	0.004
WENZ	16	39	0.050	0.013
CRTC	19	39	0.050	0.009
WERS	24	31	0.050	0.005
TAMI	25	28	0.050	0.005
WENP	36	47	0.050	0.008
AQIR	14	31	0.050	0.008
MVT	16	34	0.050	0.011
CRFR	16	20	0.050	0.008
SWFT	14	34	0.050	0.007
VYGR	15	23	0.050	0.008
TMXI	18	32	0.050	0.013

(b) FCL

Table 1 Parameters for each instance of the ABI trailer problem at warehouses CRTV and FCL.

1. *Piecewise linear approximation*: This policy is developed using the piecewise linear approximation that is detailed in section 4. More specifically, we begin by solving CONCISE-INV via the constraint generation procedure that we outline above. We randomly generate an initial set of constraints so as to ensure that each trailer type is represented at least once for each possible inventory type at each time period. Then, using Proposition 6, we measure the optimality gap of our current solution at each iteration of the constraint generation procedure and stop constraint generation either when the optimality gap is within 1% or after 50 iterations of constraint generation. For each instance, we store the upper bound that is developed in Proposition 6, which we use to test the efficacy of all of the heuristics that we test. We find that even when the constraint generation terminates after 50 iterations, our upper bounds remain quite tight. After solving CONCISE-INV, we then solve KNAP to recover x_1^*, which we use as our initial inventory vector. Given this initial inventory state, we implement the policy that rolls forward the dynamic program given in (1) using the piecewise linear approximation to the value functions:

$$\mathcal{V}_t(x_t) \approx \sum_{l \in \mathcal{L}} q^*_{l,t}(x^l).$$

We abbreviate this policy as PL for short.

2. *Greedy*: This policy will always match the available trailer type that has the largest immediate revenue to the arriving truck. Namely, in time period t with truck arrival $b \in \mathcal{B}$, the trailer $l \in \mathcal{L}(x_t, b)$ with the larger revenue will be matched. Since Corollary 2 shows that one of these two trailers must be optimal, this greedy policy is a natural heuristic to investigate. It is not too difficult to see that the submodularity result of Proposition 2 continues to hold when the value functions in the proposition statement are computed through the aforementioned greedy policy. Consequently, we select the initial inventory levels of each trailer type by employing the greedy procedure that is described at the end of section 3. Implementing this greedy procedure requires

access to the expected revenue of any given initial inventory vector, which we estimate using Monte Carlo simulation with 10,000 trials. We abbreviate this policy as GR for short.

3. *Single*: This policy chooses the optimal single trailer type to stock through complete enumeration over the possible trailer types in $\hat{\mathcal{L}}_\epsilon$. We abbreviate this policy as SG for short.

For policy $\pi \in \{\mathrm{PL}, \mathrm{GR}, \mathrm{SG}\}$ we let x_1^π be the initial inventory vector suggested by the given policy and $\tilde{V}_1^\pi(x_1^\pi)$ be the expected revenue of this initial inventory decision under the given policy. Since $\tilde{V}_1^\pi(x_1^\pi)$ cannot be computed exactly, we estimate its value via Monte Carlo simulation with 10,000 trials. All experiments used Python 2.7 on an Intel Core i5 with 3.2 GHz CPU and 32GB of RAM and Gurobi 6.5.1 as the linear programming solver.

Results. The performance of the three policies at each of the warehouses is reported in Tables 2a and 2b. Columns 1 and 2 of these tables give the carrier and value of ϵ used to derive the set of trailers $\hat{\mathcal{L}}_\epsilon$ for each instance, respectively. Columns 3 through 5 report the optimality gap of the three policies, which for policy $\pi \in \{\mathrm{PL}, \mathrm{GR}, \mathrm{SG}\}$ is computed as $100 \times (UB - \tilde{V}_1^\pi(x_1^\pi))/UB$, where UB is the upper bound on the optimal expected revenue that can be derived using Proposition 6. We note that this optimality gap does not take into account the gap from choosing a potentially suboptimal collection of trailer types in the trailer selection problem. The final column in this table gives the number of iterations of constraint generation that are implemented before termination.

We observe that all three policies perform quite well over all carriers at both warehouses, indicating that simple policies can perform quite well for the ABI trailer problem. Nonetheless, the performance of the greedy policy provides a clear improvement over the other two policies. For $\epsilon = 0.05$, the average optimality gap for PL over all carriers is 1.03% and 0.95% at CRTV and FCL respectively. For policy GR, the average optimality gaps are 0.75% and 0.83% at CRTV and FCL respectively. For $\epsilon = 0.02$, the average performance of all policies drops, but the dip is more pronounced for PL, whose average optimality gap is 2.44% for warehouse FCL. On the other hand, the average optimality gap for GR never exceeds 1% in either warehouse for $\epsilon = 0.02$. When $\epsilon = 0.05$, the policy GR outperforms policy SG by over 0.8% and 0.65% on average at warehouses CRTV and FCL respectively, for carriers in which the greedy policy selected more than one trailer. The improvements in performance of policy GR over SG for the instances in which $\epsilon = 0.02$ all exceed 0.4%.

We note that the increase in the optimality gap as ϵ decreases is due to the fact that the reported optimality gaps do not incorporate the loss in revenue due to a suboptimal choice of trailer types. In other words, the optimality gaps reported are conditioned on the choice of trailer types, and since a smaller ϵ leads to more trailer types and hence a more difficult problem, it is no surprise that the optimality gaps increase as ϵ is decreased. Interestingly, the performance of PL is still reasonably good even when constraint generation is terminated early due to hitting the 50 iteration limit.

carrier	ϵ	% Opt. Gap PL	% Opt. Gap GR	% Opt. Gap SG	CG iters.
AHLY	0.05	0.53	0.53	0.53	18
AHLY	0.02	1.86	0.75	0.76	26
WENX	0.05	0.88	0.87	1.68	31
WENX	0.02	2.85	2.40	3.05	50
GTGA	0.05	1.07	0.89	2.32	24
GTGA	0.02	2.43	0.91	1.74	50
PRIJ	0.05	0.71	0.76	1.56	22
PRIJ	0.02	1.61	0.90	1.49	50
MTNF	0.05	0.85	0.91	1.81	15
MTNF	0.02	1.18	0.73	1.05	26
WENL	0.05	0.97	0.97	0.97	11
WENL	0.02	2.37	0.61	1.14	39
TAMI	0.05	1.45	0.23	0.23	4
TAMI	0.02	0.97	0.89	0.97	11
AIOE	0.05	1.11	0.74	1.25	9
AIOE	0.02	1.28	0.76	1.39	26
WSXI	0.05	0.75	0.75	0.75	4
WSXI	0.02	0.93	0.79	1.55	10
PASC	0.05	2.09	0.81	0.81	11
PASC	0.02	1.14	0.82	0.98	38
MTEN	0.05	0.91	0.91	0.91	14
MTEN	0.02	1.37	0.91	1.00	35
JBHI	0.05	1.11	0.67	1.09	13
JBHI	0.02	1.42	0.75	1.22	23

(a) CRTV

carrier	ϵ	% Opt. Gap PL	% Opt. Gap GR	% Opt. Gap SG	CG iters.
WERD	0.05	0.88	0.87	1.75	47
WERD	0.02	4.53	3.79	5.18	50
WENZ	0.05	1.58	0.83	1.16	45
WENZ	0.02	2.87	2.04	2.36	50
CRTC	0.05	0.91	0.89	1.87	33
CRTC	0.02	1.26	1.28	1.66	50
WERS	0.05	0.98	0.98	1.41	30
WERS	0.02	4.13	2.95	4.18	50
TAMI	0.05	0.99	0.99	0.99	23
TAMI	0.02	0.85	0.78	0.94	43
WENP	0.05	0.82	0.90	1.70	47
WENP	0.02	4.61	2.69	3.65	50
AQIR	0.05	0.57	0.61	0.61	19
AQIR	0.02	2.21	0.80	1.06	39
MVT	0.05	1.37	0.78	0.9	16
MVT	0.02	3.20	0.72	0.97	35
CRFR	0.05	1.04	0.91	2.29	16
CRFR	0.02	2.52	0.81	1.93	42
SWFT	0.05	0.96	0.91	0.97	17
SWFT	0.02	0.77	0.68	1.01	36
VYGR	0.05	0.63	0.63	0.63	18
VYGR	0.02	0.98	0.85	1.42	22
TMXI	0.05	0.65	0.67	1.56	17
TMXI	0.02	1.44	1.07	1.53	50

(b) FCL

Table 2 Performance metrics for the two heuristic policies at warehouses CRTV and FCL.

The worst observed optimality gap when constraint generation terminates early is 4.61% for PL. Finally, it is worth noting that while the policy produced from the piecewise linear approximation is generally not as lucrative as the greedy approach, there is still great value in being able to solve CONCISE-INV efficiently, as our results show that the upper bound provided by the optimal objective are generally very tight and hence prove useful in measuring the quality of any possible heuristic.

6. Conclusion

In this paper, we introduce the ABI trailer problem to model how ABI delivers its beer to wholesalers via third party delivery trucks. First, ABI must solve the trailer type selection problem, which involves choosing the set of preloaded trailer weights that ABI will consider loading. Next, they must choose the inventory levels of each trailer type, which we aptly denote as the inventory selection problem. In the third and final problem that makes up the ABI trailer problem, the trailers must be matched to arriving third party trucks so as to maximize expected revenue for a fixed number of truck arrivals. We develop two approaches, which simultaneously address all three problems under two different setting that are distinguished by whether or not ABI is allowed to alter preloaded trailers in an online fashion. Through a series of computational experiments using real data from AB, we show that the approaches that we develop lead to solutions that are within fractions of a percent of optimal.

There are several interesting avenues for future research. Specific to the ABI trailer problem, a compelling direction for future work could involve developing a policy for the trailer matching problem with scalebacks that has a provable performance guarantee. A more general direction for future work could consider incorporating inventory decisions into classic online resource allocation problems as described in the introduction. For example, it would be interesting to see if an initial inventory decision can be incorporated into the personalized assortment problem studied by Rusmevichietong et al. (2014) or the classic network revenue management problem on parallel flight legs originally studied by Zhang and Cooper (2004).

Acknowledgments

The authors gratefully acknowledge Anheuser Busch InBev for providing logistic data for this research project.

References

Adelman, D. (2007), 'Dynamic bid prices in revenue management', *Operations Research* **55**, 647–661.

Aouad, A., Levi, R. and Segev, D. (2018), 'Greedy-like algorithms for dynamic assortment planning under multinomial logit preferences', *Operations Research* . forthcoming.

Feldman, J., Korula, N., Mirrokni, V., Muthukrishnan, S. and Pal, M. (2009), 'Online ad assignment with free disposal', *WINE* pp. 374–385.

Feldman, J. and Topaloglu, H. (2017), 'Revenue management under the Markov chain choice model', *Operations Research* .

G. L. Nemhauser, L. A. W. and Fisher, M. L. (1978), 'An analysis of approximations for maximizing submodular set functions', *Mathematical Programming* **14**, 265–294.

Gallego, G., Iyengar, G., Phillips, R. and Dubey, A. (2004), 'Managing flexible products on a network'.

Gallego, G., Ratliff, R. and S.Shebalov (2014), 'A general attraction model and sales-based linear program for network revenue management under customer choice', *Operations Research* .

Goyal, V., Levi, R. and Segev, D. (2016), 'Near-optimal algorithms for the assortment planning problem under dynamic substitution and stochastic demand', *Operations Research* **64**, 219–235.

Honhon, D., Gaur, V. and Seshadri, S. (2010), 'Assortment planning and inventory decisions under stockout-based substitution', *Operations Research* **58**, 1364–1379.

Kunnumkal, S. and Talluri, K. (2015), 'On a piecewise-linear approximation for network revenue management', *Mathematics of Operations Research* **41**, 72–91.

Liu, Q. and van Ryzin, G. (2009), 'On the choice based linear programming model for network revenue management', *Manufacturing Service Operations Management* **10**, 288–310.

Ma, W. and Levi, D. S. (2017), Online resource allocation under arbitrary arrivals: Optimal algorithms and tight competitive rations. Unpublished Manuscript.

Mehta, A. (2013), 'Online matching and ad allocation', *Foundations and Trends in Theoretical Computer Science* **8**, 265–368.

Meissner, J. and Strauss, A. (2012), 'Network revenue management with inventory-sensitive bid prices and customer choice', *European Journal of Operations Research* **216**, 459–468.

Méndez-Díaz, I., Miranda-Bront, J., Vulcano, G. and Zabala, P. (2010), 'A branch-and-cut algorithm for the latent-class logit assortment problem', *Discrete Applied Mathematics* **36**, 383–390.

Rusmevichietong, P., Golrezaei, N. and Nazerzadeh, H. (2014), 'Real-time optimization of personalized assortments', **63**(9), 3128–3145. Management Science.

Simpson, R. (1989), *Using Network Flow Techniques to Find Shadow Prices for Market Demands and Seat Inventory Control*, FTL memorandum, MIT, Department of Aeronautics and Astronautics, Flight Transportation Laboratory.
URL: *https://books.google.com/books?id=GUuvHAAACAAJ*

Sinha, P. and Zoltners, A. (1979), 'The multiple choice knapsack problem', *Operations Research* **37**(3), 503–515.

Stein, C., Truong, V. and Wang, X. (2016), Advance reservations with heterogeneous customers. Unpublished Manuscript.

Sumida, M., Topaloglu, H. and Rusmevichientong, P. (2017), Dynamic assortment optimization for reusable products with random usage durations. Unpublished Manuscript.

Talluri, K. and Kunnumkal, S. (2016), ‘Technical note: A note on relaxations of the choice network revenue management dynamic program’, *Operations Research* **64**, 158–166.

Talluri, K. and van Ryzin, G. (1999*a*), ‘An analysis of bid-price controls for network revenue management’, *Management Science* **44**, 1577–1593.

Talluri, K. and van Ryzin, G. (1999*b*), ‘A randomized linear programming method for computing network bid prices’, *Transportation Science* **33**, 207–216.

Tong, C. and Topaloglu, H. (2014), ‘On the approximate linear programming approach for network revenue management problems’, *INFORMS Journal of Computing* **26**, 131–134.

Topaloglu, H. (2009), ‘Using lagrangian relaxation to compute capacity-dependent bid prices in network revenue management’, *Operations Research* **57**, 637–649.

Vossen, T. and Zhang, D. (2015), ‘Reductions of approximate linear programs for network revenue management’, *Operations Research* **63**, 1352–1371.

Zhang, D. and Adelman, D. (2009), ‘An approximate dynamic programming approach to network revenue management with customer choice’, *Transportation Science* **43**, 381–394.

Zhang, D. and Cooper, W. (2004), ‘Revenue management for parallel flights with customer choice behavior’, *Operations Research* **53**, 415–431.

Appendix A: Additional Proofs

A.1. Proof of Proposition 4

We prove by induction. It holds for $t=\tau+1$ since by definition $V_{\tau+1}(\cdot)=0$. Suppose it holds for $t+1$, and we show it holds for t.

Define

$$k_1^* = \underset{l\in\mathcal{L}(x-e_{l_1})}{\arg\max}\ r_{l,b}+V_{t+1}(x-e_{l_1}-e_l)$$
$$k_2^* = \underset{l\in\mathcal{L}(x-e_{l_2})}{\arg\max}\ r_{l,b}+V_{t+1}(x-e_{l_2}-e_l)$$

and

$$\Delta_b(x,k_1^*,k_2^*)=[r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*})]$$

We show that $c_o(w_{l_1}-w_{l_2})\le\Delta_b(x,k_1^*,k_2^*)\le r(w_{l_2}-w_{l_1})$ for any b. Once this is shown, by $V_t(x-e_{l_1})-V_t(x-e_{l_2})=\sum_{b\in\mathcal{B}}p_b\Delta_b(x,k_1^*,k_2^*)$, we prove the proposition.

1. We show the following result first: for any $l_2>l_1$, it always holds that

$$c_o(w_{l_1}-w_{l_2})\le r_{l_2,b}-r_{l_1,b}\le r(w_{l_2}-w_{l_1}) \tag{19}$$

To prove this, we show for three possible scenarios of b:

(a) $w_b\in[0,80,000-w_{l_2}]$. In this case, $r_{l_2,b}=rw_{l_2}$ and $r_{l_1,b}=rw_{l_1}$. We have $r_{l_2,b}-r_{l_1,b}=r(w_{l_2}-w_{l_1})$. Also by definition $w_{l_2}>w_{l_1}$. Hence (19) holds.

(b) $w_b\in[80,000-w_{l_2},80,000-w_{l_1}]$. In this case, $r_{l_2,b}=r(80,000-w_b)-c_o(w_{l_2}+w_b-80,000)$ and $r_{l_1,b}=rw_{l_1}$. Notice $r_{l_2,b}\le rw_{l_2}$ which yields $r_{l_2,b}-r_{l_1,b}\le r(w_{l_2}-w_{l_1})$. On the other hand, $r_{l_2,b}-r_{l_1,b}=r(80,000-w_b-w_{l_1})-c_o(w_{l_2}+w_b-80,000)=r(80,000-w_b-w_{l_1})-c_o(w_{l_1}+w_b-80,000)+c_o(w_{l_1}-w_{l_2})=(r+c_o)(80,000-w_b-w_{l_1})+c_o(w_{l_1}-w_{l_2})\ge c_o(w_{l_1}-w_{l_2})$. Hence (19) holds.

(c) $w_b\in[80,000-w_{l_1},80,000]$. In this case, $r_{l_2,b}=r(80,000-w_b)-c_o(w_{l_2}+w_b-80,000)$ and $r_{l_1,b}=r(80,000-w_b)-c_o(w_{l_1}+w_b-80,000)$. We have $r_{l_2,b}-r_{l_1,b}=c_o(w_{l_1}-w_{l_2})$. Also by definition $w_{l_2}>w_{l_1}$. Hence (19) holds.

2. We show $\Delta_b(x,k_1^*,k_2^*)\le r(w_{l_2}-w_{l_1})$.

(1) If $k_1^*\in\mathcal{L}(x-e_{l_2})$: by optimal choice of k_2^*, we have

$$r_{k_1^*,b}+V_{t+1}(x-e_{l_2}-e_{k_1^*})\le r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*}) \tag{20}$$

With this,

$$\begin{aligned}\Delta_b(x,k_1^*,k_2^*)=&[r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*})]\\ =&[r_{k_1^*,b}+V_{t+1}(x-e_{l_2}-e_{k_1^*})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*})]\\ &+V_{t+1}(x-e_{l_1}-e_{k_1^*})-V_{t+1}(x-e_{l_2}-e_{k_1^*})\\ \le&V_{t+1}(x-e_{l_1}-e_{k_1^*})-V_{t+1}(x-e_{l_2}-e_{k_1^*})\\ \le&r(w_{l_2}-w_{l_1})\end{aligned}$$

where the first inequality is due to (20) and the second inequality holds by induction.

(2) if $k_1^* \notin \mathcal{L}(x-e_{l_2})$, it must be $k_1^*=l_2$.

$$\begin{aligned}\Delta_b(x,k_1^*,k_2^*)=&[r_{l_2,b}+V_{t+1}(x-e_{l_1}-e_{l_2})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-k_2^*)]\\=&r_{l_2,b}-r_{l_1,b}+[r_{l_1,b}+V_{t+1}(x-e_{l_1}-e_{l_2})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*})]\\\le& r_{l_2,b}-r_{l_1,b}\\\le& r(w_{l_2}-w_{l_1})\end{aligned}$$

where the first inequality is due to optimality of k_2^* and second inequality is due to (19).

3. We show $\Delta_b(x,k_1^*,k_2^*)\ge c_o(w_{l_1}-w_{l_2})$.

(1) If $k_2^*\in\mathcal{L}(x-e_{l_1})$: by optimal choice of k_1^*, we have

$$r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})\ge r_{k_2^*,b}+V_{t+1}(x-e_{l_1}-e_{k_2^*}) \tag{21}$$

With this,

$$\begin{aligned}\Delta_b(x,k_1^*,k_2^*)=&[r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_2}-e_{k_2^*})]\\=&[r_{k_1^*,b}+V_{t+1}(x-e_{l_2}-e_{k_1^*})]-[r_{k_2^*,b}+V_{t+1}(x-e_{l_1}-e_{k_2^*})]\\&+V_{t+1}(x-e_{l_1}-e_{k_2^*})-V_{t+1}(x-e_{l_2}-e_{k_2^*})\\\ge& V_{t+1}(x-e_{l_1}-e_{k_2^*})-V_{t+1}(x-e_{l_2}-e_{k_2^*})\\\ge& c_o(w_{l_1}-w_{l_2})\end{aligned}$$

where the first inequality is due to (21) and the second inequality holds by induction.

(2) if $k_2^*\notin\mathcal{L}(x-e_{l_1})$, it must be $k_2^*=l_1$. Therefore,

$$\begin{aligned}\Delta_b(x,k_1^*,k_2^*)=&[r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})]-[r_{l_1,b}+V_{t+1}(x-e_{l_2}-e_{l_1})]\\=&r_{l_2,b}-r_{l_1,b}+[r_{k_1^*,b}+V_{t+1}(x-e_{l_1}-e_{k_1^*})]-[r_{l_2,b}+V_{t+1}(x-e_{l_2}-e_{l_1})]\\\ge& r_{l_2,b}-r_{l_1,b}\\\ge& c_o(w_{l_1}-w_{l_2})\end{aligned}$$

where the first inequality is due to optimality of k_1^* and second inequality is due to (19).

A.2. Proof of Concave and Increasing Property

We prove that (9) and (10) hold for $y\ge 2$, and by imposing (9) to hold at $y=1$ we conclude that (10) holds at $y=1$ as well. We first prove the following Lemma which accomplishes the first part. We then enforce (9) to hold at $y=1$ and show it yields (10) to hold at $y=1$ as well. Our proof borrows the same approach as used in Kunnumkal and Talluri (2015).

LEMMA 3. *There exists an optimal solution* $\{q_{l,t}^*(x^l)\}$ *to* (APPROX-INV) *such that*

1. For any $x^l\in\{2,\cdots,\tau\}$,

$$q_{l,t}^*(x^l)-q_{l,t}^*(x^l-1)\ge q_{l,t+1}^*(x^l)-q_{l,t+1}^*(x^l-1) \tag{22}$$

2. for any $x^l \in \{2, \cdots, \tau\}$, where $q^*_{l,t}(\tau+1) \triangleq q^*_{l,t}(\tau)$,

$$q^*_{l,t}(x^l) - q^*_{l,t}(x^l - 1) \geq q^*_{l,t}(x^l + 1) - q^*_{l,t}(x^l) \tag{23}$$

D efine for any $l \in \mathcal{L}$

$$\mathcal{R}_l(x^l) = \{\mathbf{q} \in \mathcal{X} | q^l = x^l\}$$

be the set of inventory vectors which satisfy the inventory level of trailer l is fixed at x^l. Given a separable piecewise-linear approximation $\mathcal{Q} = \{q_{l,t}(x^l_t) | \forall t, l \in \mathcal{L}, x^l_t \in \{0, \cdots, \tau\}\}$, we let

$$\begin{aligned}\epsilon_{l,t}(x^l, \mathcal{Q}) = \min_{x \in \mathcal{R}_l(x^l), l^b_x \in \mathcal{L}(x,b)} & \sum_{l \in \mathcal{L}} q_{l,t}(x^l) - \sum_{l \in \mathcal{L}} q_{l,t+1}(x^l) \\ & - \sum_{b \in \mathcal{B}} \left[p_b \cdot \left(r_{l^b_x, b} + q_{l^b_x, t+1}(x^{l^b_x} - 1) - q_{l^b_x, t+1}(x^{l^b_x}) \right) \right]\end{aligned} \tag{24}$$

Note that if $\mathcal{Q}$ is feasible to problem (APPROX-INV), $\epsilon_{l,t}(x^l, \mathcal{Q}) \geq 0$ for all l, t, x^l. We begin with a preliminary result.

We first prove the following lemma, by which we prove Lemma 3:

LEMMA 4. *There exists an optimal solution $\mathcal{Q}^* = \{q^*_{l,t}(x^l_t) | \forall t, l \in \mathcal{L}, x^l_t \in \{0, \cdots, \tau\}\}$ such that $\epsilon_{l,t}(x^l, \mathcal{Q}^*) = 0$ for all l, t and x^l.*

Proof of Lemma 4: Let $\mathcal{Q} = \{q_{l,t}(x^l)\}$ be an optimal solution to problem (APPROX-INV). Let s be the largest time period such that there exists l_0 and x^{l_0} with $\epsilon_{l_0,s}(x^{l_0}, \mathcal{Q}) > 0$. This means that $\epsilon_{l,t}(x^l, \mathcal{Q}) = 0$ for any l, x^l and $t > s$. We define another solution $\mathcal{Q}^*$ as following:

$$q^*_{l,t}(x) = \begin{cases} q_{l,t}(x) - \epsilon_{l,t}(x^l, \mathcal{Q}) & \text{if } l = l_0, t = s, x = x^l \\ q_{l,t}(x) & \text{otherwise} \end{cases} \tag{25}$$

Obviously $q^*_{l,t}(x) \leq q_{l,t}(x)$ for all l, t, x. Therefore $\sum_{l \in \mathcal{L}} q^*_{l,1}(x^l_1) \leq \sum_{l \in \mathcal{L}} q_{l,1}(x^l_1)$. Next we show that $\mathcal{Q}^*$ is feasible to problem (APPROX-INV). Since there is only one instance of $q^*_{l,t}(x)$ that $q^*_{l,t}(x) \neq q_{l,t}(x)$, all constraints in problem (APPROX-INV) are satisfied by $\mathcal{Q}^*$ except those that contain $q^*_{l_0,s}(x^{l_0})$. Note $q^*_{l_0,s}(x^{l_0})$ only appears in constraints corresponding to time periods $s-1$ and s.

For periods s, $\forall x = \{x^1, .., x^{l_0}, .., x^n\} \in \mathcal{X}$ and $\{l^b_x \in \mathcal{L}(x,b) : b = 1, ..., m\}$,

$$\begin{aligned}\sum_{l \in \mathcal{L}} q^*_{l,s}(x^l) &= \sum_{l \in \mathcal{L}} q_{l,s}(x^l) - \epsilon_{l,s}(x^{l_0}, \mathcal{Q}) \\ &\geq \sum_{b \in \mathcal{B}} \left[p_b \cdot \left(r_{l^b_x, b} + q_{l^b_x, s+1}(x^{l^b_x} - 1) - q_{l^b_x, s+1}(x^{l^b_x})) \right) \right] + \sum_{l \in \mathcal{L}} q_{l,s+1}(x^l) \\ &= \sum_{b \in \mathcal{B}} \left[p_b \cdot \left(r_{l^b_x, b} + q^*_{l^b_x, s+1}(x^{l^b_x} - 1) - q^*_{l^b_x, s+1}(x^{l^b_x})) \right) \right] + \sum_{l \in \mathcal{L}} q^*_{l,s+1}(x^l)\end{aligned}$$

where the first equality holds by (25), second inequality holds by (24), and last equality holds by (25).

For periods $s-1$, $\forall x = \{x^1, ..., x^n\} \in \mathcal{X}$ and $\{l^b_x \in \mathcal{L}(x,b) : b = 1, ..., m\}$,

$$\begin{aligned}\sum_{l \in \mathcal{L}} q^*_{l,s-1}(x^l) &= \sum_{l \in \mathcal{L}} q_{l,s-1}(x^l) \\ &\geq \sum_{b \in \mathcal{B}} \left[p_b \cdot \left(r_{l^b_x, b} + q_{l^b_x, s}(x^{l^b_x} - 1) - q_{l^b_x, s}(x^{l^b_x})) \right) \right] + \sum_{l \in \mathcal{L}} q_{l,s}(x^l)\end{aligned}$$

$$
\begin{aligned}
&= \sum_{b\in\mathcal{B}} \Big[p_b \cdot \Big(r_{l_x^b,b} + q_{l_x^b,s}(x^{l_x^b}-1)\Big)\Big] + \sum_{l\in\mathcal{L}}[(1-\sum_{\{b:l_x^b=l\}} p_b)q_{l,s}(x^l)] \\
&\geq \sum_{b\in\mathcal{B}} \Big[p_b \cdot \Big(r_{l_x^b,b} + q^*_{l_x^b,s}(x^{l_x^b}-1)\Big)\Big] + \sum_{l\in\mathcal{L}}[(1-\sum_{\{b:l_x^b=l\}} p_b)q^*_{l,s}(x^l)] \\
&= \sum_{b\in\mathcal{B}} \Big[p_b \cdot \Big(r_{l_x^b,b} + q^*_{l_x^b,s}(x^{l_x^b}-1) - q^*_{l_x^b,s}(x^{l_x^b})\Big)\Big] + \sum_{l\in\mathcal{L}} q^*_{l,s}(x^l)
\end{aligned}
$$

where the first equality holds by (25), second inequality holds because $\{q_{l,t}\}$ are feasible to problem (APPROX-INV), third and last equality holds by rearranging terms, and fourth inequality holds by $q^*_{l,t}(x) \leq q_{l,t}(x)$ and all the coefficients in front of $q_{l,t}(x)$ are non-negative.

So far we have constructed a $\mathcal{Q}^*$ such that $\epsilon_{l_0,t}(x^{l_0},\mathcal{Q}^*)=0$ and $\sum_{l\in\mathcal{L}} q^*_{l,1}(x_1^l) \leq \sum_{l\in\mathcal{L}} q_{l,1}(x_1^l)$. We can continue such an approach by looping through the periods backwards (till period 1) and in every period checking for all $l\in\mathcal{L}$ that the condition $\epsilon_{l,t}(x^l,\mathcal{Q}^*)=0$ is met. Lemma 4 proof is done.

Now we are ready to prove Lemma 3. It holds for $\tau+1$ trivially. We assume that Lemma 3 holds for period $t+1$, and next prove it holds for t.

(1) We prove (22) for period t. Pick any trailer type l^* with $x^{l^*}>0$, by Lemma 4, there exist an $x\in\mathcal{R}_{l^*}(x^{l^*})$ such that $\epsilon_{l^*,t}(x^{l^*},\mathcal{Q})=0$. In other words there exist $x=\{x^1,..,x^{l^*},..x^n\}\in\mathcal{R}_{l^*}(x^{l^*})$ and $\{l_x^1,...,l_x^m\}: l_x^b\in\mathcal{L}(x,b)$ such that

$$
\begin{aligned}
&\sum_{l\neq l^*} q_{l,t}(x^l) + q_{l^*,t}(x^{l^*}) \\
=&\sum_{b\in\mathcal{B}} \Big[p_b\cdot\Big(r_{l_x^b,b} + q_{l_x^b,t+1}(x^{l_x^b}-1) - q_{l_x^b,t+1}(x^{l_x^b})\Big)\Big] + \sum_{l\in\mathcal{L}} q_{l,t+1}(x^l) \\
=&\sum_{\{b\in\mathcal{B}|l_x^b\neq l^*\}} \Big[p_b\cdot\Big(r_{l_x^b,b} + q_{l_x^b,t+1}(x^{l_x^b}-1) - q_{l_x^b,t+1}(x^{l_x^b})\Big)\Big] + \\
&\sum_{\{b\in\mathcal{B}|l_x^b= l^*\}} \big[p_b\cdot\big(r_{l^*,b} + q_{l^*,t+1}(x^{l^*}-1) - q_{l^*,t+1}(x^{l^*})\big)\big] + \sum_{l\neq l^*} q_{l,t+1}(x^l) + q_{l^*,t+1}(x^{l^*})
\end{aligned}
$$

Now let $\bar{x}=x+e_{l^*}$. The l_x^b defined above satisfies $l_x^b\in\mathcal{L}(\bar{x},b)$ too, because $x^{l^*}>0$. Therefore it must satisfy the constraint:

$$
\begin{aligned}
&\sum_{l\neq l^*} q_{l,t}(\bar{x}^l) + q_{l^*,t}(\bar{x}^{l^*}) \\
\geq &\sum_{\{b\in\mathcal{B}|l_x^b\neq l^*\}} \Big[p_b\cdot\Big(r_{l_x^b,b} + q_{l_x^b,t+1}(\bar{x}^{l_x^b}-1) - q_{l_x^b,t+1}(\bar{x}^{l_x^b})\Big)\Big] + \\
&\sum_{\{b\in\mathcal{B}|l_x^b= l^*\}} \big[p_b\cdot\big(r_{l^*,b} + q_{l^*,t+1}(\bar{x}^{l^*}-1) - q_{l^*,t+1}(\bar{x}^{l^*})\big)\big] + \sum_{l\neq l^*} q_{l,t+1}(\bar{x}^l) + q_{l^*,t+1}(\bar{x}^{l^*})
\end{aligned}
$$

Notice that $\bar{x}^l=x^l$ if $l\neq l^*$ and $\bar{x}^l=x^l+1$ if $l=l^*$. By taking the difference,

$$
\begin{aligned}
&q_{l^*,t}(x^{l^*}+1) - q_{l^*,t}(x^{l^*}) \\
\geq &\sum_{\{b\in\mathcal{B}|l_x^b= l^*\}} \big[p_b\cdot\big(r_{l^*,b} + q_{l^*,t+1}(\bar{x}^{l^*}-1) - q_{l^*,t+1}(\bar{x}^{l^*})\big)\big] + q_{l^*,t+1}(\bar{x}^{l^*}) - \\
&\sum_{\{b\in\mathcal{B}|l_x^b= l^*\}} \big[p_b\cdot\big(r_{l^*,b} + q_{l^*,t+1}(x^{l^*}-1) - q_{l^*,t+1}(x^{l^*})\big)\big] - q_{l^*,t+1}(x^{l^*}) \\
=&[\sum_{\{b\in\mathcal{B}|l_{x,b}= l^*\}} p_b][q_{l^*,t+1}(x^{l^*}) - q_{l^*,t+1}(x^{l^*}+1) - q_{l^*,t+1}(x^{l^*}-1) + q_{l^*,t+1}(x^{l^*})] +
\end{aligned}
$$

$$\begin{aligned}&q_{l^*,t+1}(x^{l^*}+1)-q_{l^*,t+1}(x^{l^*})\\ \geq& q_{l^*,t+1}(x^{l^*}+1)-q_{l^*,t+1}(x^{l^*})\end{aligned}$$

where the last inequality is by induction.

(2) We prove (23) for period t for all $x^l > 1$. Pick any l^* and $x^{l^*} > 1$. By Lemma 4, there exist an $\bar{x} \in \mathcal{R}_{l^*}(x^{l^*}+1)$ such that $\epsilon_{l^*,t}(x^{l^*}+1,\mathcal{Q})=0$. In other words there exist $\bar{x}=\{x^1,..,x^{l^*}+1,..x^n\}\in\mathcal{R}_{l^*}(\bar{x}^{l^*})$ and $\{l^1_{\bar{x}},...,l^m_{\bar{x}}\}: l^b_{\bar{x}}\in\mathcal{L}(\bar{x},b)$ such that

$$\begin{aligned}&\sum_{l\neq l^*} q_{l,t}(\bar{x}^l)+q_{l^*,t}(\bar{x}^{l^*})\\ =&\sum_{b\in\mathcal{B}}\left[p_b\cdot\left(r_{l^b_{\bar{x}},b}+q_{l^b_{\bar{x}},t+1}(\bar{x}^{l^b_{\bar{x}}}-1)-q_{l^b_{\bar{x}},t+1}(\bar{x}^{l^b_{\bar{x}}})\right)\right]+\sum_{l\in\mathcal{L}} q_{l,t+1}(\bar{x}^l)\\ =&\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}\neq l^*\}}\left[p_b\cdot\left(r_{l^b_{\bar{x}},b}+q_{l^b_{\bar{x}},t+1}(\bar{x}^{l^b_{\bar{x}}}-1)-q_{l^b_{\bar{x}},t+1}(\bar{x}^{l^b_{\bar{x}}})\right)\right]+\\ &\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}=l^*\}}\left[p_b\cdot\left(r_{l^*,b}+q_{l^*,t+1}(\bar{x}^{l^*}-1)-q_{l^*,t+1}(\bar{x}^{l^*})\right)\right]+\sum_{l\neq l^*} q_{l,t+1}(\bar{x}^l)+q_{l^*,t+1}(\bar{x}^{l^*}+1)\end{aligned}$$

Let $x=\bar{x}-e^{l^*}$. The above defined $l^b_{\bar{x}}$ also satisfies $l^b_{\bar{x}}\in\mathcal{L}(x,b)$ because $x^{l^*}>1$. Therefore it must be feasible and satisfy:

$$\begin{aligned}&\sum_{l\neq l^*} q_{l,t}(x^l)+q_{l^*,t}(x^{l^*})\\ \geq&\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}\neq l^*\}}\left[p_b\cdot\left(r_{l^b_{\bar{x}},b}+q_{l^b_{\bar{x}},t+1}(x^{l^b_{\bar{x}}}-1)-q_{l^b_{\bar{x}},t+1}(x^{l^b_{\bar{x}}})\right)\right]+\\ &\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}=l^*\}}\left[p_b\cdot\left(r_{l^*,b}+q_{l^*,t+1}(x^{l^*}-1)-q_{l^*,t+1}(x^{l^*})\right)\right]+\sum_{l\neq l^*} q_{l,t+1}(x^l)+q_{l^*,t+1}(x^{l^*})\end{aligned}$$

Notice that $\bar{x}^l=x^l$ if $l\neq l^*$ and $\bar{x}^l=x^l+1$ if $l=l^*$. By taking the difference,

$$\begin{aligned}&q_{l^*,t}(\bar{x}^{l^*})-q_{l^*,t}(x^{l^*})\\ \leq&\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}=l^*\}}\left[p_b\cdot\left(r_{l^*,b}+q_{l^*,t+1}(\bar{x}^{l^*}-1)-q_{l^*,t+1}(\bar{x}^{l^*})\right)\right]+q_{l^*,t+1}(\bar{x}^{l^*})-\\ &\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}=l^*\}}\left[p_b\cdot\left(r_{l^*,b}+q_{l^*,t+1}(x^{l^*}-1)-q_{l^*,t+1}(x^{l^*})\right)\right]-q_{l^*,t+1}(x^{l^*})\\ =&[\sum_{\{b\in\mathcal{B}|l^b_{\bar{x}}=l^*\}}p_b][q_{l^*,t+1}(x^{l^*})-q_{l^*,t+1}(x^{l^*}+1)-q_{l^*,t+1}(x^{l^*}-1)+q_{l^*,t+1}(x^{l^*})]+\\ &q_{l^*,t+1}(x^{l^*}+1)-q_{l^*,t+1}(x^{l^*})\\ =&[\sum_{\{b\in\mathcal{B}|l_{x,b}=l^*\}}p_b][q_{l^*,t+1}(x^{l^*})-q_{l^*,t+1}(x^{l^*}-1)]+[1-\sum_{\{b\in\mathcal{B}|l_{x,b}=l^*\}}p_b][q_{l^*,t+1}(x^{l^*}+1)-q_{l^*,t+1}(x^{l^*})]\\ \leq& q_{l^*,t+1}(x^{l^*})-q_{l^*,t+1}(x^{l^*}-1)\\ \leq& q_{l^*,t}(x^{l^*})-q_{l^*,t}(x^{l^*}-1)\end{aligned}$$

where the second to last inequality is due to induction assumption for period $t+1$ and the last inequality is using results in (1).

It is obvious that (23) is equivalent to (9) for $y \geq 2$. To see (10) holds for $y \geq 2$, we first apply $t = \tau$ to (22) and conclude (10) holds for $y \geq 2$ at $t = \tau$, and then we roll t backwards and apply to (22) which yields (10) holds for $y \geq 2$ at all t.

If we enforce that (9) holds at $y = 1$ for all t, then (10) holds at $y = 1$ for all t as well by applying the enforcement and (10) holding at $y = 2$.

A.3. Proof of Proposition 5

We prove the results via induction over k. The result holds trivially when $l = \gamma$. Next, we assume the result holds for all $k+1 \leq k \leq \gamma$ and show that the result holds for $k = k$.

$$\max_{\substack{(l^{(k+1)},\ldots,l^{(\gamma)},x^{l^{(k+1)}},\ldots,x^{l^{(\gamma)}}): \\ \tau-(t-1+c)\leq\sum_{i=k+1}^{\gamma} x^{l^{(k)}} \leq \tau - c}} \theta_t(l^{(k)} = l, l^{(k+1)}, \ldots, l^{(\gamma)}, x^{l^{(k)}} = x, x^{l^{(k+1)}}, \ldots, x^{l^{(\gamma)}}) =$$

$$\max_{\substack{(l^{(k+1)},\ldots,l^{(\gamma)},x^{l^{(k+1)}},\ldots,x^{l^{(\gamma)}}): \\ \tau-(t-1+c)\leq\sum_{i=k+1}^{\gamma} x^{l^{(k)}} \leq \tau - c}} f(l, x, l^{(k+1)}, x^{l^{(k+1)}}) + \theta_t(l^{(k+1)}, \ldots, l^{(\gamma)}, x^{l^{(k+1)}}, \ldots, x^{l^{(\gamma)}}) =$$

$$\max_{l<l'\leq n+1,\, x'>0} \left\{ f(l, x, l', x') + \max_{\substack{(l^{(k+2)},\ldots,l^{(\gamma)},x^{l^{(k+2)}},\ldots,x^{l^{(\gamma)}}): \\ \tau-(t-1+c+x')\leq\sum_{i=k+2}^{\gamma} x^{l^{(k)}} \leq \tau - c - x'}} \theta_t(l^{(k+1)} = l', \ldots, l^{(\gamma)}, x^{l^{(k+1)}} = x', \ldots, x^{l^{(\gamma)}}) \right\} =$$

$$\max_{l<l'\leq n+1,\, x'>0} \{ f(l, x, l', x') + J(l', x', c + x' \mathbb{1}_{\{l' \neq n+1\}}) \}$$

where the third equality follows by the induction hypothesis.

A.4. Proof of Lemma 1

Since there are $n+1$ constraints, any basis can contain at most $n+1$ positive variables. Moreover, the first constraint of KNAP ensures that each of the n groups must contribute at least one basic variables. Further, if a group contains a fractional variable, then there must be another fractional variable in this group due again to the first constraint. Consequently, there can be at most one group with two fractional variables, otherwise the basis would contain more than $n+1$ variables.