Online Supplement to "Anticipatory Packing"

Tongwen Wu, Zuo-Jun Max Shen, Yanzhi Li

EC.1. Omitted Details in Section 4

In this section, we first present the proof of hardness result for the evaluation problem and then provide details of its approximation algorithm.

EC.1.1. Hardness of Evaluation Problem

Proof of Theorem 1. We give an approximation-preserving reduction from any instance $\mathcal{I}$ of the maximum 3-bounded 3-dimensional matching (3DM-3) problem to an instance $\Phi(\mathcal{I})$ of Problem (Eva-SAP). The 3DM-3 problem is defined as follows:

Definition EC.1 (3DM-3). Given an edge set $E \subseteq V_1 \times V_2 \times V_3$ where V_1 ,V_2 and V_3 are disjoint vertex sets and $|V_1| = |V_2| = |V_3| = \mathfrak{n}$. And the number of occurrences of each element of $V_1 \cup V_2 \cup V_3$ in E is at most 3. A matching M is a subset of E such that no elements of M agree in any coordinate.

And the APX-hardness has been established by Kann (1991) as follows:

Lemma EC.1. *The 3DM-3 problem is APX-hard. Specifically, there exists a constant ϵ_0 such that it is NP-hard to decide whether an instance has a matching of size $\mathfrak{n}$ or every matching has size at most $(1-\epsilon_0)\mathfrak{n}$.*

Next, at a high level, our reduction is based on transforming edge e_j of E to order O_j of $\mathcal{G}$ and vertex v of $V_1 \cup V_2 \cup V_3$ to prepackage S of $\mathcal{P}$. The transformation is described as follows:

- Let $V_t = \{v_{t,1}, ..., v_{t,\mathfrak{n}}\}$ for $t = 1, 2, 3$ and $E = \{e_1, ..., e_m\}$.
- Let $N = 3(\mathfrak{n}+1)$ and for notation brevity, we use $V_1 \cup V_2 \cup V_3 \cup \{v_{1,0}, v_{2,0}, v_{3,0}\}$ to denote the item set. The vertex $v_{t,0}$ can be regarded as the auxiliary vertex not in 3DM-3.
- In (Eva-SAP), we are given a prepacking scheme $\mathcal{P}$ and an order set $\mathcal{G}$. Let $\mathcal{P} = \{(S_i, n_i)\}_{i=1,...,L}$ and $\mathcal{G} = \{O_j\}_{j=1,...,m}$ where $L = 3\mathfrak{n} + m$. Let $S_{(t-1)\mathfrak{n}+i} = \{v_{t,0}, v_{t,i}\}$ and $n_{(t-1)\mathfrak{n}+i} = 1$ for $t = 1, 2, 3$ and $i \in [\mathfrak{n}]$ and $S_{3\mathfrak{n}+j} = \{v_{1,0}, v_{2,0}, v_{3,0}\}$ for $j \in [m]$. Let $O_j = \{v_{1,0}, v_{2,0}, v_{3,0}\} \cup e_j$ for $j \in [m]$.
- We utilize the cost structure described in Example 1. Specifically, we set $w = 0, c = 1$ and p as an arbitrary nonnegative value.

Under this construction, the total prepackages assigned to each order contribute to the objective value of (Eva-SAP) either 3 or 2, that is, either assigning 3 prepackages of size 2 or 1 prepackage of size 3 to one order. For the case where only 2 or 1 prepackages of size 2 are assigned to one order, we can just replace them with one prepackage of size 3 without decreasing the objective value. Next, we show there is a mapping between matching and assignments of prepackages. On the one hand, for any matching M of size x, for each $e_j \in M$, we can assign $S_{(t-1)\mathfrak{n}+i}$ for $v_{t,i} \in e_j$ to O_j and $S_{3\mathfrak{n}+j}$ to O_j for $e_j \notin M$, which achieves the objective value of (Eva-SAP) of $3x + 2(m-x)$. On the other hand, for any assignment with objective value $3x+2(m-x)$, we can efficiently find orders assigned with 3 prepackages and then construct the matching M of size x . Since it is NP-hard to decide whether there is a matching of size $\mathfrak{n}$ or every matching has at most size $(1-\epsilon_0)\mathfrak{n}$, it is NP-hard to decide whether there is an assignment of value $\mathfrak{n}+2m$ or any assignment is of at most value $(1-\epsilon_0)\mathfrak{n}+2m$. Combining with the fact that $m \le 3\mathfrak{n}$ (from Definition EC.1), we complete the proof. □

EC.1.2. Constant-Factor Approximation Algorithms for Evaluation Problem

In this subsection, we first describe a $\frac{e}{e-1}$-approximation algorithm for Problem (Eva-SAP), followed by its performance analysis. The randomized rounding algorithm is formally presented in Algorithm 1.

Algorithm 1 Randomized approximation algorithm for (Eva-SAP) with bounded order size

Input: $\mathcal{P}=\{(S_i,n_i)\}_{i\in[L]}, \mathcal{G}=\{O_j\}_{j\in[M]}$

Output: Assignment decision $\mathbf{h}$

1: Solve the linear relaxation of (Eva-SAP) and let $\mathbf{h}^*$ be the optimal solution.
2: Set $\mathbf{y}=\{y_{ij}\}_{i\in[L],j\in[M]} \leftarrow \mathbf{0}$.
3: **for all** $j\in[M]$ **do**
4: Randomly sample $\mathcal{U}$ from $\mathscr{U}_j$ with probability $h_j^{*\mathcal{U}}$.
5: **for all** $S_i\in\mathcal{U}$ **do**
6: **if** $\sum_{j'\in[M]} y_{ij'} < n_i$ **then**
7: $y_{ij}\leftarrow 1$.
8: Construct the integral assignment decision $\mathbf{h}$ such that $h_j^{\mathcal{U}}=1$ if and only if $y_{ij}=1$ $\forall S_i\in\mathcal{U}$ and $y_{ij}=0$ $\forall S_i\notin\mathcal{U}$.

The linear relaxation is of polynomial size when the maximum order size is bounded, so Algorithm 1 can be executed in polynomial time. In addition, we leverage the assignment decisions $\mathbf{y}$ from prepackages to orders to ensure that the capacity constraint (1b) is satisfied.

Proof of Theorem 2. To establish the approximation ratio, we consider the expected contribution of each prepackage to the objective value. Define $\mathbf{y}^*$ by letting $y^*_{ij}=\sum_{\mathcal{U}\in\mathscr{U}_j:S_i\in\mathcal{U}} h_j^{*\mathcal{U}}$. Then

$$f(\mathcal{P},\mathcal{G}) \le \sum_{j\in[M],\mathcal{U}\in\mathscr{U}_j} d^{\mathcal{U}} h_j^{*\mathcal{U}} = \sum_{i\in[L],j\in[M]} q_{S_i} y^*_{ij}, \tag{EC.1}$$

and Pr[the sampled $\mathcal{U}$ from $\mathscr{U}_j$ contains S_i] $=y^*_{ij}$.

We apply the following technical lemma in our analysis.

LEMMA EC.2. *Let $X_1,\dots,X_n$ be independent Bernoulli variables, with $\Pr(X_i=1)=a_i$ for $1\le i\le n$, and $\sum_{i=1}^n a_i\le b$, where $b\in\mathbb{Z}_{>0}$, then we have*

$$\mathbb{E}\left\{\min(\sum_{i=1}^n X_i, b)\right\} \ge (1-\frac{1}{e})(\sum_{i=1}^n a_i). \tag{EC.2}$$

Proof of Lemma EC.2: Let Z_i be the random choice of i among b indices, satisfying $\Pr(Z_i = j) = 1/b$ for any $j \in [b]$. And we define $Y_{ij} = \mathbb{1}[X_i = 1 \text{ and } Z_i = j]$. It is easy to see that $\Pr(Y_{ij} = 1) = a_i/b$ and they are independent across i. Therefore, we have

$$\mathbb{E}\left\{\min(\sum_{i=1}^{n} X_i, b)\right\} = \mathbb{E}\left\{\min(\sum_{i=1}^{n}\sum_{j=1}^{b} Y_{ij}, b)\right\} \geq \mathbb{E}\left\{\sum_{j=1}^{b}\min(\sum_{i=1}^{n} Y_{ij}, 1)\right\} = \sum_{j=1}^{b}\mathbb{E}\left\{\min(\sum_{i=1}^{n} Y_{ij}, 1)\right\}$$
$$\geq \sum_{j=1}^{b}(1-\frac{1}{e})(\sum_{i=1}^{n} a_i)/b = (1-\frac{1}{e})(\sum_{i=1}^{n} a_i),$$

where the first equality uses $X_i = \sum_{j=1}^{b} Y_{ij}$, the first inequality follows from that $\min\{a+b, c+d\} \geq \min\{a,c\} + \min\{b,d\}$ for any a,b,c,d, and the second inequality is due to that

$$\mathbb{E}\left\{\min(\sum_{i=1}^{n} Y_{ij}, 1)\right\} = \Pr[\sum_{i=1}^{n} Y_{ij} \geq 1] = 1 - \Pr[Y_{ij} = 0\ \forall 1 \leq i \leq n] = 1 - \prod_{1\leq i\leq n}(1 - a_i/b)$$
$$\geq 1 - \exp(-\sum_{1\leq i\leq n} a_i/b) \geq (1-\frac{1}{e})\sum_{1\leq i\leq n} a_i/b.$$

Above, the first inequality holds since $1 - x \leq \exp(-x)$ if $0 \leq x \leq 1$. The second inequality holds since $\exp(-x) \leq 1 - (1-\frac{1}{e})x$ if $0 \leq x \leq 1$. □

Therefore, we establish that

$$\mathbb{E}[\sum_{j\in[M],\mathcal{U}\in\mathscr{U}_j} d^{\mathcal{U}} h_j^{\mathcal{U}}] = \mathbb{E}[\sum_{i\in[L],j\in[M]} q_{S_i} y_{ij}] = \sum_{i\in[L]} q_{S_i}\mathbb{E}[\sum_{j\in[M]} y_{ij}]$$
$$= \sum_{i\in[L]} q_{S_i}\mathbb{E}\Big[\min\Big\{\sum_{j\in[M]} \mathbb{1}[\text{ the sampled } \mathcal{U} \text{ from } \mathscr{U}_j \text{ contains } S_i], n_i\Big\}\Big]$$
$$\geq (1-\frac{1}{e})\sum_{i\in[L]} q_{S_i}\cdot\sum_{j\in[M]} y^*_{ij} \geq (1-\frac{1}{e}) f(\mathcal{P},\mathcal{G}),$$

where the first equality is due to line 8 of Algorithm 1, the third equality follows from lines 6-7 of Algorithm 1, first inequality follows from Lemma EC.2, and the second inequality uses (EC.1). □

EC.2. Omitted Details in Section 5

In this section, we first present the proof of hardness results for the prepacking problem and then provide details of its approximation algorithms.

EC.2.1. Hardness of Prepacking Problem

We first give the proof of Theorem 3 and then establish APX-hardness even when the maximum order size is bounded.

Proof of Theorem 3. At a high level, we show that the maximum independent set (MIS) problem can be reduced to a special case of the (SAA) model with $M_t = 1$ for any $t \in [T]$, and we then analyze the performance upper bound. The MIS problem and its inapproximability are shown as follows:

Definition EC.2 (MIS). Given a graph $G = (V, E)$, the goal is to find a maximum subset X of V such that there are no two vertices of X connected by an edge in E.

LEMMA EC.3 (**Zuckerman** (**2006**)). *For all $\epsilon > 0$, it is NP-hard to approximate MIS within $\mathfrak{n}^{1-\epsilon}$, where $\mathfrak{n} = |V|$.*

In what follows, we describe an approximation-preserving reduction from an instance $\mathcal{I}$ of MIS with $\mathfrak{n} \geq 3$ to an instance $\Phi(\mathcal{I})$ of (SAA). Fix $\epsilon > 0$.

- Let $V = \{v_1, \ldots, v_\mathfrak{n}\}$ and $E = \{e_j\}_{j=1}^{|E|}$, where each e_j is a pair of vertices, and let d_i denote the degree of vertex v_i.
- Let each observation $\mathcal{G}_t$ contain just one order, that is $M_t = 1$ for any $t \in [T]$.
- Set $N = \mathfrak{n} + 1$ and $T = 2\mathfrak{n}^2 - |E|$. For the first $|E|$ observations, we have that the single order $O_1^t = \{\mathfrak{n}+1\} \cup e_t$ for $t \in [|E|]$. For the rest of the sample, we have that the single order $O_1^{|E|+\sum_{1\leq j\leq i-1}(2\mathfrak{n}-d_j)+k} = \{\mathfrak{n}+1, i\}$ for each $v_i \in V$ and any $k \in [2\mathfrak{n} - d_i]$.
- We utilize the cost structure described in Example 1. We set parameters p, w, c in order to satisfy the following inequalities:

$$\begin{cases} -(p+w) + \frac{2\mathfrak{n}}{T}(c+w) > 0 \\ -(p+w) + \frac{2\mathfrak{n}-1}{T}(c+w) < 0 \\ -(p+w) + \frac{1}{T}(2c+w) < 0 \\ p, w, c \geq 0. \end{cases}$$

In the proof, we use the following specific solution: $p = 2\mathfrak{n} - 1, w = 0, c = \frac{2\mathfrak{n}-1}{2}(\frac{T}{2\mathfrak{n}} + \frac{T}{2\mathfrak{n}-1})$. In this way, it is easy to verify that the effective prepackage can only be of the type $\{\mathfrak{n}+1, i\}$ for some $i \leq \mathfrak{n}$. In addition, Δ is equal to $\max_{1\leq i\leq \mathfrak{n}} d_i$.

To complete the proof, we first show for any independent set $I \subseteq V$, there is a solution to (SAA) with objective value of at least $\frac{|I|}{2}$. Specifically, for each $v_i \in I$, we make a prepackage $\{\mathfrak{n}+1, i\}$ and assign it to order O_1^t with $t \in \{j | v_i \in e_j\} \cup \{|E| + \sum_{1\leq j\leq i-1}(2\mathfrak{n} - d_j) + k | k \in [2\mathfrak{n} - d_i]\}$. Each prepackage contributes $-(p+w) + \frac{2\mathfrak{n}}{T}(c+w) = \frac{1}{2}$ to the objective value. On the other hand, we show for any prepacking scheme to (SAA) with objective value `val`, we can efficiently obtain an independent set I with $|I| \geq 2 \cdot \texttt{val}$. That is, because the effective prepackage $\{\mathfrak{n}+1, i\}$ in any solution with a positive benefit must be assigned to $2\mathfrak{n}$ observations, and contribute $\frac{1}{2}$ to `val`, we can construct an independent set composing of i with $\{\mathfrak{n}+1, i\}$ is included in the prepacking scheme with size of $2 \cdot \texttt{val}$. Since $T \leq 2\mathfrak{n}^2$ and $\Delta \leq \mathfrak{n}$, Problem (SAA) with maximum order size bounded by three is NP-hard to approximate within $\max\{(\frac{T}{2})^{\frac{1-\epsilon}{2}}, \Delta^{1-\epsilon}\} \leq \mathfrak{n}^{1-\epsilon}$. □

THEOREM EC.1. *Problem (SAA) is APX-hard to approximate when the maximum order size $\max_{j\in[M_t], t\in[T]}\{|O_j^t|\}$ and sample size T are both bounded.*

Proof of Theorem EC.1. We reduce the instance $\mathcal{I}$ of the 3DM-3 problem (see Definition EC.1) in an approximation-preserving manner to the instance of (SAA) with $T = 3$ and $\max_{j\in[M_t], t\in[T]}\{|O_j^t|\} = 4$. We give the reduction as follows:

- Let $V_t = \{v_{t,1}..., v_{t,\mathfrak{n}}\}$ for any $t = 1, 2, 3$ and $E = \{e_i\}_{i=1,\ldots,|E|}$.

- Set $T=3$, $K=4$, $M_t=\mathfrak{n}$, and $N=|E|+1$.
- For any $t\in[T]$ and any $j\in[\mathfrak{n}]$, we set O_j^t as $\{|E|+1\}\cup\{i|v_{t,j}\in e_i\}$.
- We still utilize the cost structure described in Example 1. We set parameters to satisfy the following inequalities:

$$\begin{cases} -(p+w)+\frac{3}{3}(c+w)>0 \\ -(p+w)+\frac{2}{3}(3c+w)<0 \\ p,w,c\geq 0. \end{cases}$$

In the following proof, we set the parameters as $p=1, w=10, c=2$.

Similar to the proof of Theorem 1, we first show for any matching M, there is a solution to (SAA) with objective value of at least $|M|$. For each $e_i\in M$, we make a prepackage $\{|E|+1,i\}$ and assign it to order O_j^t with $v_{t,j}\in e_i$. Since M is a matching, no conflict occurs with this assignment. Each prepackage contributes $-(p+w)+\frac{3}{3}(c+w)=1$ to the objective value.

On the other hand, we show for any solution to (SAA) with objective value `val`, we can efficiently obtain a matching M with $|M|\geq$ `val`. Specifically, under this reduction, the prepackage making positive benefit must be assigned in all three observations. Furthermore, such a prepackage has at most 2 items, and each item i, except $|E|+1$, corresponds to the edge e_i. Therefore, the prepackage in any solution with a positive benefit must have exactly 2 items, be assigned to all samples, and contribute 1 to `val`. Thus, for any solution to (SAA), we can find prepackages with positive benefit and transform them to a matching M with $|M|\geq$ `val`. Therefore, by lemma EC.1 this special case is APX-hard to approximate. □

EC.2.2. General Approximation Algorithms

EC.2.2.1. Performance Analysis for Marginal-Based Rounding.

Proof of Proposition 1. Let $(\mathbf{x}^*,\mathbf{y}^*)$ be the optimal solution to the linear relaxation of (SAA-K). For brevity, define the total cost savings of any solution $(\mathbf{x},\mathbf{y})$ as:

$$\mathrm{CS}(\mathbf{x},\mathbf{y})=\sum_{t\in[T]}\omega_t\sum_{i\in[Q]}\sum_{j\in\mathcal{J}_i^t}q_{R_i}\cdot y_{ij}^t-\sum_{i\in[Q]}p_{R_i}\cdot x_i. \tag{EC.3}$$

By the definition of Δ, we can partition the effective prepackages into $\Delta+1$ buckets $\{\mathcal{B}_k\}_{k=1}^{\Delta+1}$, as done in the marginal-based rounding scheme. For each $\hat{\mathbf{x}}^k$, we construct an integral assignment $\hat{\mathbf{y}}^k$ as follows:

$$\hat{y}_{ij}^{kt}=\begin{cases}1, & \lceil\sum_{j'<j}y_{ij'}^{*t}\rceil<\lceil\sum_{j'\leq j}y_{ij'}^{*t}\rceil\leq\hat{x}_i^k \\ 0, & \text{otherwise}\end{cases}\quad\forall i\in[Q],\ t\in[T],\ j\in\mathcal{J}_i^t.$$

Without loss of generality, we can assume that $\sum_{t\in[T]}\omega_t q_{R_i}\cdot\mathbb{1}\left\{\sum_{j\in\mathcal{J}_i^t}y_{ij}^{*t}=x_i^*\right\}\geq p_{R_i}$ for all $i\in[Q]$; otherwise, we could reduce the value of x_i^* to obtain a higher objective value. Thus, by construction of $\hat{\mathbf{x}}^k$ and $\hat{\mathbf{y}}^k$, we then have:

$$\mathrm{CS}(\hat{\mathbf{x}}^k,\hat{\mathbf{y}}^k)\geq\sum_{t\in[T]}\omega_t\sum_{i\in\mathcal{B}_k}\sum_{j\in\mathcal{J}_i^t}q_{R_i}\cdot y_{ij}^{*t}-\sum_{i\in\mathcal{B}_k}p_{R_i}\cdot x_i^*.$$

Let $\hat{\mathbf{y}}$ be the optimal integral assignment decision corresponding to the prepacking scheme $\hat{\mathbf{x}}$. Then we obtain:

$$\mathrm{CS}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \geq \frac{1}{\Delta+1} \sum_{k=1}^{\Delta+1} \mathrm{CS}(\hat{\mathbf{x}}^k, \hat{\mathbf{y}}^k) \geq \frac{1}{\Delta+1} \mathrm{CS}(\mathbf{x}^*, \mathbf{y}^*).$$

□

EC.2.2.2. Algorithm Description for Correlated Rounding. At a high level, our correlated rounding scheme operates by first constructing a set of bundles and then performing randomized selection with a scaling parameter to avoid conflicts. The procedure is formally described as follows:

Bundle construction. Bundles are constructed separately for each effective prepackage type. Fix a prepackage type R_i. We begin by computing a set of breaking points:

$$P_i \triangleq \left\{ a \,\middle|\, a = \sum_{j' \leq j} y_{ij'}^{*t} \text{ for some } t \in [T],\ j \in \mathcal{J}_i^t \right\} \cup \left[\lfloor x_i^* \rfloor \right] \cup \{0, x_i^*\}.$$

We arrange the elements in P_i in ascending order as $a^{(1)}, a^{(2)}, \ldots, a^{(|P_i|)}$. For each interval $[a^{(l)}, a^{(l+1)})$ with $1 \leq l \leq |P_i| - 1$, we construct a bundle. If the resulting bundle is the kth one, denoted by $B_k = (z_k, \tilde{i}_k, (\tilde{j}_k^t)_{t\in[T]})$, then its components are defined as follows:

$$z_k = a^{(l+1)} - a^{(l)}, \quad \tilde{i}_k = i, \quad \tilde{j}_k^t = \begin{cases} j, & \exists j \text{ such that } \sum_{j'<j} y_{ij'}^{*t} \leq a^{(l)} \text{ and } \sum_{j' \leq j} y_{ij'}^{*t} \geq a^{(l+1)} \\ 0, & \text{otherwise} \end{cases} \quad \forall t \in [T].$$

We denote the total number of constructed bundles across all prepackage types by Q'.

Example EC.1. We illustrate the bundle construction with a simple example. As shown in Figure EC.1, consider a prepackage R_i with $T = 2$, $x_i^* = 1.8$, and assignment variables given by $y_{i1}^{*1} = y_{i2}^{*1} = y_{i3}^{*1} = 0.6$, and $y_{i1}^{*2} = y_{i2}^{*2} = 0.8$. The resulting set of breaking points is $P_i = \{0, 0.6, 0.8, 1, 1.2, 1.6, 1.8\}$, and we construct six bundles. For example, for the interval $[1.6, 1.8)$, we construct a bundle B_k with:

$$z_k = 0.2, \quad \tilde{i}_k = i, \quad \tilde{j}_k^1 = 3, \quad \tilde{j}_k^2 = 0,$$

since the cumulative assignment in $t = 2$ is $y_{i1}^{*2} + y_{i2}^{*2} = 1.6 < x_i^* = 1.8$, meaning no additional assignment is made in $t = 2$ for this interval.

Randomized selection. Next, we randomly select from the constructed bundles to generate a feasible solution to (SAA-K) through a delicate procedure that leverages the structure of bundles sharing the same $\tilde{i}_k$ to proactively avoid conflicts among them.

Specifically, given a **scaling parameter** α, we randomly select bundles for each prepackage type R_i, independently across different prepackage types but in a correlated manner among bundles of the same type. The parameter α controls the selection probability of each bundle, effectively scaling down the expected number of selected bundles to reduce the likelihood of conflicts. A smaller α reduces the likelihood of selecting multiple conflicting bundles simultaneously, thereby improving the feasibility of the resulting solution. After the selection process, all bundles are combined, and any remaining conflicting

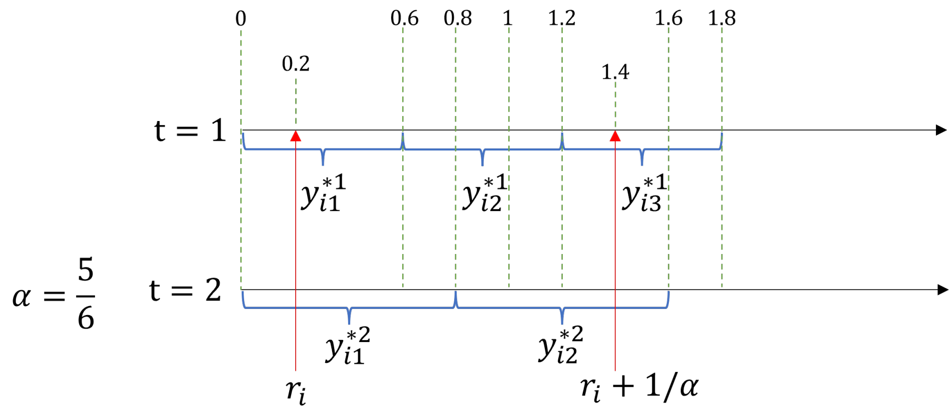


Figure EC.1 An Illustration for Bundle Construction

bundles are discarded, ensuring that the final solution is feasible for (SAA-K). In the experiment, we set $\alpha = 2$ to enhance practical performance, although our subsequent analysis is based on the condition $\alpha \le 1$.

We begin by fixing a prepackage type R_i. A random number r_i is drawn uniformly from the interval $[0, 1/\alpha)$. Then, we select all bundles B_k whose corresponding interval is $[a^{(l)}, a^{(l+1)})$, such that:

$$\tilde{i}_k = i \quad \text{and} \quad a^{(l)} \le r_i + X/\alpha < a^{(l+1)} \quad \text{for some } X \in \mathbb{Z}_{\ge 0}.$$

This selection process is illustrated in Figure EC.1. In the example, with $\alpha = 5/6$ and $r_i = 0.2$, we select two bundles: $B_k = (z_k = 0.6 - 0 = 0.6,\ \tilde{i}_k = i,\ \tilde{j}^1_k = 1,\ \tilde{j}^2_k = 1)$ corresponding to r_i, and $B_{k'} = (z_{k'} = 1.6 - 1.2 = 0.4,\ \tilde{i}_{k'} = i,\ \tilde{j}^1_{k'} = 3,\ \tilde{j}^2_{k'} = 2)$ corresponding to $r_i + 1/\alpha$. It is easy to see that the selected bundles for the same prepackage type will not conflict when $\alpha \le 1$. This is because the difference between values $r_i + X/\alpha$ for different X is at least 1 when $\alpha \le 1$, while each $y^{*t}_{ij} \le 1$. This ensures that, for any $t \in [T]$, the selected bundles with $\tilde{i} = i$ have distinct values of $\tilde{j}^t_k$ (if nonzero).

Next, we combine the selected bundles across all prepackage types. To ensure feasibility, we discard any bundle that conflicts with previously selected ones. Formally, the final set of selected bundles is given by:

$$\mathcal{S} = \left\{ B_k \ \text{ that was previously selected} \ \middle| \ \begin{array}{l} \tilde{j}^t_k \ne \tilde{j}^t_{k'} \quad \forall t \in [T],\ k' \ne k, \\ B_{k'} \text{ is selected, and } R_{\tilde{i}_k} \cap R_{\tilde{i}_{k'}} \ne \emptyset \end{array} \right\}.$$

This construction ensures that for any two bundles in $\mathcal{S}$, if their corresponding prepackage types share at least one item, then they do not include the same order. From this set $\mathcal{S}$, we construct a feasible solution $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ to (SAA-K) as follows:

$$\tilde{x}_i = \sum_{B_k \in \mathcal{S}} \mathbb{1}\{\tilde{i}_k = i\}, \quad \tilde{y}^t_{ij} = \mathbb{1}\left\{\exists B_k \in \mathcal{S} \text{ such that } \tilde{j}^t_k = j\right\}, \quad \forall i \in [Q],\ t \in [T],\ j \in \mathcal{J}^t_i.$$

Remark EC.1 (A Heuristic Improvement). In the above description, for ease of analysis, we choose a subset $\mathcal{S}$ of selected bundles, which yields no conflict, and construct a feasible solution $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ directly. However, a more delicate procedure can be applied in practice to eliminate conflicts more effectively and improve solution quality. At a high level, this procedure first determines a preliminary assignment decision $\dot{\mathbf{y}}$ from the selected bundles while avoiding conflicts, then optimizes the prepacking decision $\ddot{\mathbf{x}}$ based on $\dot{\mathbf{y}}$, and finally determines the assignment $\ddot{\mathbf{y}}$. The steps are as follows: (1) Initialize $\dot{\mathbf{y}} = \mathbf{0}$. (2) Iterate over bundles $k = 1 \ldots Q'$ if it is selected: for each $t \in [T]$, set $\dot{y}^t_{\tilde{i}_k \tilde{j}^t_k} = 1$ if adding this assignment keeps $\dot{\mathbf{y}}$ feasible under Constraint (3a). (3) Optimize $\ddot{\mathbf{x}}$ by

$$\ddot{x}_i = \arg\max_{x_i \in \mathbb{Z}_{\geq 0}} \Big\{ -p_{R_i} x_i + \sum_{t \in [T]} \omega_t \cdot q_{R_i} \cdot \min\{x_i, \sum_{j \in \mathcal{J}^t_i} \dot{y}^t_{ij}\} \Big\}, \quad \forall i \in [Q].$$

(4) Set $\ddot{\mathbf{y}}$ by

$$\ddot{y}^t_{ij} = \mathbb{1}\Big[\dot{y}^t_{ij} = 1 \text{ and } \sum_{l \leq j} \dot{y}^t_{il} \leq \ddot{x}_i \Big], \quad \forall i \in [Q], t \in [T], j \in \mathcal{J}^t_i.$$

It is obvious to see that $\ddot{\mathbf{y}}$ satisfies Constraint (3a), since $\dot{\mathbf{y}}$ satisfies this constraint and $\ddot{\mathbf{y}} \leq \dot{\mathbf{y}}$. And $(\ddot{\mathbf{x}}, \ddot{\mathbf{y}})$ must satisfy Constraint (3b) by the construction of $\ddot{\mathbf{y}}$. Therefore, $(\ddot{\mathbf{x}}, \ddot{\mathbf{y}})$ is a feasible solution to (SAA-K). Moreover, this heuristic yields a solution at least as good as $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$. That is,

$$\begin{aligned}
\mathrm{CS}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) &\leq \sum_{t \in [T]} \omega_t \sum_{i \in [Q]} \sum_{j \in \mathcal{J}^t_i} q_{R_i} \cdot \tilde{y}^t_{ij} - \sum_{i \in [Q]} p_{R_i} \cdot \max_{t \in [T]} \{ \sum_{j \in \mathcal{J}^t_i} \tilde{y}^t_{ij} \} \\
&= \sum_{i \in [Q]} \sum_{t \in [T]} \omega_t \cdot q_{R_i} \cdot \min\Big\{ \max_{t' \in [T]} \{ \sum_{j \in \mathcal{J}^{t'}_i} \tilde{y}^{t'}_{ij} \}, \sum_{j \in \mathcal{J}^t_i} \tilde{y}^t_{ij} \Big\} - \sum_{i \in [Q]} p_{R_i} \cdot \max_{t \in [T]} \{ \sum_{j \in \mathcal{J}^t_i} \tilde{y}^t_{ij} \} \\
&\leq \sum_{i \in [Q]} \sum_{t \in [T]} \omega_t \cdot q_{R_i} \cdot \min\Big\{ \max_{t' \in [T]} \{ \sum_{j \in \mathcal{J}^{t'}_i} \tilde{y}^{t'}_{ij} \}, \sum_{j \in \mathcal{J}^t_i} \dot{y}^t_{ij} \Big\} - \sum_{i \in [Q]} p_{R_i} \cdot \max_{t \in [T]} \{ \sum_{j \in \mathcal{J}^t_i} \tilde{y}^t_{ij} \} \\
&\leq \sum_{i \in [Q]} \Big(-p_{R_i} \ddot{x}_i + \sum_{t \in [T]} \omega_t \cdot q_{R_i} \cdot \min\{ \ddot{x}_i, \sum_{j \in \mathcal{J}^t_i} \dot{y}^t_{ij} \} \Big) = \mathrm{CS}(\ddot{\mathbf{x}}, \ddot{\mathbf{y}}),
\end{aligned}$$

where the first inequality uses $\tilde{x}_i \geq \max_{t \in [T]} \{ \sum_{j \in \mathcal{J}^t_i} \tilde{y}^t_{ij} \}$, the second inequality is due to $\tilde{\mathbf{y}} \leq \dot{\mathbf{y}}$, and the third inequality follows from the construction of $\ddot{\mathbf{x}}$.

EC.2.2.3. Performance Analysis for Correlated Rounding. In this subsection, we prove the approximation ratios stated in Proposition 2 with specific scaling parameters.

By the construction of $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, we know

$$\mathrm{CS}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \sum_{t \in [T]} \omega_t \sum_{i \in [Q]} \sum_{j \in \mathcal{J}^t_i} q_{R_i} \cdot \tilde{y}^t_{ij} - \sum_{i \in [Q]} p_{R_i} \cdot \tilde{x}_i = \sum_{B_k \in \mathcal{S}} \gamma_k,$$

where

$$\gamma_k = q_{R_i} \Big(\sum_{t \in [T]} \omega_t \mathbb{1}\{ \tilde{j}^t_k \neq 0 \} \Big) - p_{R_i}$$

can be thought of as the contribution to the total cost savings once bundle B_k is selected into $\mathcal{S}$. Note that, without loss of generality, we can assume $\sum_{t \in [T]} \omega_t q_{R_i} \cdot \mathbb{1}\Big\{ \sum_{j \in \mathcal{J}^t_i} y^{*t}_{ij} = x^*_i \Big\} \geq p_{R_i}$ for all $i \in [Q]$, ensuring that $\gamma_k \geq 0$ by the bundle construction. We now prove the two approximation ratios separately.

Approximation ratio $O(TK)$. For the general case, we set the scaling parameter $\alpha = \frac{1}{2TK}$. To establish the approximation ratio, we lower bound the probabilities of being in $\mathcal{S}$ for each bundle B_k. First, since r_i is uniformly drawn from the interval $[0, 1/\alpha)$, we know that for each bundle B_k,

$$\Pr[B_k \text{ was previously selected}] = \alpha \cdot z_k.$$

We then bound the probability that $B_k \in \mathcal{S}$ by:

$$\begin{aligned}
\Pr[B_k \in \mathcal{S}] = &\Pr[B_k \text{was previously selected}] \\
&\cdot \Pr\Big[\tilde{j}_k^t \neq \tilde{j}_{k'}^t \ \forall t \in [T], k' \in [Q'], \tilde{i}_k \neq \tilde{i}_{k'},\ B_{k'} \text{ was previously selected, and } R_{\tilde{i}_k} \cap R_{\tilde{i}_{k'}} \neq \emptyset\Big] \\
\geq &\alpha \cdot z_k \cdot \Big(1 - \sum_{k' \in [Q']: \tilde{i}_{k'} \neq \tilde{i}_k, R_{\tilde{i}_k} \cap R_{\tilde{i}_{k'}} \neq \emptyset, \text{ and } \tilde{j}_{k'}^t = \tilde{j}_k^t \exists t \in [T]} \Pr[B_{k'} \text{ was previously selected}]\Big) \\
\geq &\alpha \cdot z_k \cdot \Big(1 - \sum_{t \in [T], l \in R_{\tilde{i}_k}} \ \sum_{k' \in [Q']: R_{\tilde{i}_{k'}} \ni l, \tilde{j}_{k'}^t = \tilde{j}_k^t} \alpha \cdot z_{k'}\Big) \geq \alpha \cdot z_k \cdot (1 - \alpha \cdot TK) \geq \frac{1}{4TK} \cdot z_k.
\end{aligned}$$

Above, the equality follows from the construction of $\mathcal{S}$, the fact that selected bundles with same $\tilde{i}_k$ will have no conflict when $\alpha < 1$, and the independence of selection among the bundles with different $\tilde{i}_k$. The first inequality uses the union bound. The second enumerates potential conflicts by overlapping items l of order $O^t_{\tilde{j}_k^t}$, possibly overcounting. The third inequality follows from the fact that,

$$\sum_{k' \in [Q']: R_{\tilde{i}_{k'}} \ni l, \tilde{j}_{k'}^t = j} z_{k'} \leq \sum_{i \in [Q]: \mathcal{J}_i^t \ni j, l \in R_i} y_{ij}^{*t} \leq 1, \quad \forall t \in [T], j \in [M_t], l \in O_j^t.$$

And the last inequality uses $\alpha = \frac{1}{2TK}$.

Therefore, we conclude that

$$\mathbb{E}[\mathrm{CS}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})] = \sum_{k \in [Q']} \gamma_k \Pr[B_k \in \mathcal{S}] \geq \frac{\sum_{k \in [Q']} \gamma_k \cdot z_k}{4TK} = \frac{\mathrm{CS}(\mathbf{x}^*, \mathbf{y}^*)}{4TK},$$

where the last equality is due to

$$\begin{aligned}
\sum_{k \in [Q']} \gamma_k \cdot z_k &= \sum_{k \in [Q']} \sum_{t \in [T]: \tilde{j}_k^t \neq 0} \omega_t \cdot q_{R_{\tilde{i}_k}} \cdot z_k - \sum_{k \in [Q']} p_{R_{\tilde{i}_k}} \cdot z_k \\
&= \sum_{t \in [T]} \omega_t \sum_{i \in [Q]} \sum_{j \in \mathcal{J}_i^t} q_{R_i} \cdot y_{ij}^{*t} - \sum_{i \in [Q]} p_{R_i} \cdot x_i^* = \mathrm{CS}(\mathbf{x}^*, \mathbf{y}^*).
\end{aligned} \tag{EC.4}$$

The approximation of $O(TK)$ for Problem (SAA) with a fixed K is established. □

Approximation ratio $O(\sqrt{TC})$ for singleton order sets. In this part, we show that when each order set $\mathcal{G}_T$ contains exactly one order and the maximum order size is bounded by a constant C, the approximation ratio can be improved to $O(\sqrt{TC})$ by setting $\alpha = \frac{1}{e\sqrt{5TC}}$. The key difference from the general case lies in the analysis: rather than bounding the probability that each individual bundle $B_k \in \mathcal{S}$, we analyze the joint probability that all selected bundles are conflict-free.

Before delving into the analysis, we first give a Chernoff bound (see Motwani and Raghavan (1995) for the proof) for the tail probabilities of the sum of independent random variables.

Lemma EC.4 (Chernoff Bound). *Let $X_1, X_2, \ldots, X_l$ be l independent random variables, not necessarily identically distributed, such that each X_i takes either the value 0 or the value a_i for some $0 \le a_i \le 1$. Then for $R = \sum_{i=1}^{l} X_i$, $\mathbb{E}[R] = \mu$, and any $\delta \ge 0$, we have*

$$\Pr[R \ge \mu(1+\delta)] < \mathbb{E}[(1+\delta)^{R-\mu(1+\delta)}] \le (\exp(\delta)/(1+\delta)^{(1+\delta)})^{\mu} \le (e/(1+\delta))^{(1+\delta)\mu}, \tag{EC.5}$$

and if $0 < \delta < 1$,

$$\Pr[R \le \mu(1-\delta)] < \mathbb{E}[(1-\delta)^{R-\mu(1-\delta)}] \le \exp(-\mu\delta^2/2). \tag{EC.6}$$

Let the binary random variable Z_k indicate whether bundle B_k is selected and $E_{t,l}$ be the event that item l in order O_1^t is assigned to more than one selected bundle, i.e., a conflict occurs. Note that $\{Z_k\}_{k\in[Q']}$ are independent among bundles with different $\tilde{i}_k$. Then,

$$\Pr[E_{t,l}] = \Pr[\sum_{k\in[Q']:l\in R_{\tilde{i}_k}, \tilde{j}_k^t=1} Z_k \ge 2] \le (e\alpha/2)^2 = \frac{1}{20TC}. \tag{EC.7}$$

The inequality holds by taking $\mu \le \alpha$ and $\mu(1+\delta) = 2$ in inequality (EC.5).

Next, we bound the probability that the total cost savings from selected bundles falls below a constant fraction of the expected cost savings. To apply Lemma EC.4, we assume $\max_{k\in[Q']}\{\gamma_k\} = 1$ and $\alpha\sum_{k\in[Q']}\gamma_k z_k \ge 1$ without loss of generality. If the latter is not true, then simply selecting the bundle B_k with the largest γ_k achieves the desired bound. Now consider:

$$\Pr[\sum_{k\in[Q']}\gamma_k Z_k \le \frac{\alpha}{2}\sum_{k\in[Q']}\gamma_k z_k] \le \exp(-\alpha\sum_{k\in[Q']}\gamma_k z_k/8) \le \exp(-1/8). \tag{EC.8}$$

The first inequality holds by taking $\delta = 1/2$ and $\mu = \alpha\sum_{k\in[Q']}\gamma_k z_k$ in Inequality (EC.6). The second inequality holds since $\mu \ge 1$.

We now consider the total failure probability:

$$\sum_{t\in[T]}\sum_{l\in O_1^t}\Pr[E_{t,l}] + \Pr[\sum_{k\in[Q']}\gamma_k Z_k \le \frac{\alpha}{2}\sum_{k\in[Q']}\gamma_k z_k] \le \frac{1}{20} + 0.88 \approx 0.93. \tag{EC.9}$$

When there are no conflicts on any items, $\mathcal{S}$ equals the full set of selected bundles. Thus, with constant probability, $\mathrm{CS}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ is at least $\Omega(\frac{1}{\sqrt{TC}})$ of $\mathrm{CS}(\mathbf{x}^*, \mathbf{y}^*)$. □

EC.2.3. Approximation Algorithms under Laminar Structure

In this subsection, we establish the approximation ratios under the weakly laminar structure defined in Assumption 1. The approximation ratio of K under this assumption, when the maximum prepackage size K is fixed, directly follows from the algorithm described in Section 5.3. Therefore, we focus on the case where both K and the conflict degree Δ are fixed.

The improved approximation ratio follows from Algorithm 2, which is outlined in Section 5.3. This algorithm computes the optimal solution to (SAA-K) when considering only the prepackage types in each subset $\mathcal{B}_k''$ separately. Among the $\lceil\log_2(K+1)\rceil$ candidate solutions, we select the best one, thereby obtaining the desired approximation guarantee.

Algorithm 2 $\lceil \log_2(K+1) \rceil$-Approximation Algorithm for Problem (SAA) under Assumption 1 with Fixed K and Δ

Input: Order sets $\{\mathcal{G}_t\}_{t=1}^T$

Output: Prepacking scheme $\mathcal{P}$

1: Initialize $\mathbf{x}^\dagger = \mathbf{0}$, $\mathbf{y}^\dagger = \mathbf{0}$.
2: **for all** $1 \le k \le \lceil \log_2(K+1) \rceil$ **do**
3: Define $\mathcal{B}_k'' = \{S \subseteq [N] \mid S \text{ is effective and } 2^{k-1} \le |S| < 2^k\}$.
4: For each $S \in \mathcal{B}_k''$, define $\mathcal{T}_S = \{S' \in \mathcal{B}_k'' \mid g(S') = S\}$, where $g(S) = \arg\min_{S' \in \mathcal{B}_k'', S' \subseteq S} |S'|$.
5: Decompose $\mathcal{B}_k'' = \bigcup_{S \in \mathcal{B}_k'' : \mathcal{T}_S \neq \emptyset} \mathcal{T}_S$.
6: Initialize $\mathbf{x}^\ddagger = \mathbf{0}$, $\mathbf{y}^\ddagger = \mathbf{0}$.
7: **for all** $S \in \mathcal{B}_k''$ with $\mathcal{T}_S \neq \emptyset$ **do**
8: Let $(\mathbf{x}', \mathbf{y}')$ be the optimal solution to (SAA-K) restricted to prepackage types in $\mathcal{T}_S$. ▷ It is polynomially solvable by brute force, because $|\mathcal{T}_S| \le \Delta$ and the evaluation problem for each prepacking scheme is a weighted bipartite matching problem.
9: Update $(\mathbf{x}^\ddagger, \mathbf{y}^\ddagger) \leftarrow (\mathbf{x}^\ddagger, \mathbf{y}^\ddagger) + (\mathbf{x}', \mathbf{y}')$. ▷ Feasibility maintained as $\mathcal{T}_S$ sets are conflict-free.
10: **if** $\mathrm{CS}(\mathbf{x}^\ddagger, \mathbf{y}^\ddagger) \ge \mathrm{CS}(\mathbf{x}^\dagger, \mathbf{y}^\dagger)$ **then**
11: Update $(\mathbf{x}^\dagger, \mathbf{y}^\dagger) \leftarrow (\mathbf{x}^\ddagger, \mathbf{y}^\ddagger)$.
12: **return** The prepacking scheme corresponding to $\mathbf{x}^\dagger$.

EC.3. Omitted Details in Section 6

Proof of Proposition 3. Regarding the feasibility, for constraint (3a), we have for any $t \in [T], j \in [M_t], k \in O_j^t$,

$$\sum_{i \in [Q]: j \in \mathcal{J}_i^t, k \in R_i} \check{y}_{ij}^t = \sum_{i \in [Q]: j \in \mathcal{J}_i^t, k \in R_i} \sum_{\mathcal{U} \in \mathscr{U}_{tj}: R_i \in \mathcal{U}} \check{h}_{tj}^{\mathcal{U}} = \sum_{\mathcal{U} \in \mathscr{U}_{tj}} \sum_{R_i \in \mathcal{U}: k \in R_i} \check{h}_{tj}^{\mathcal{U}} \le \sum_{\mathcal{U} \in \mathscr{U}_{tj}} \check{h}_{tj}^{\mathcal{U}} \le 1,$$

where the first inequality follows from the fact that prepackages in $\mathcal{U}$ are disjoint. For constraint (3b), we clearly have for any $i \in [Q], t \in [T]$,

$$\sum_{j \in \mathcal{J}_i^t} \check{y}_{ij}^t = \sum_{j \in \mathcal{J}_i^t} \sum_{\mathcal{U} \in \mathscr{U}_j^t : R_i \in \mathcal{U}} \check{h}_{tj}^{\mathcal{U}} \le \check{x}_i.$$

As for the objective value, by simple calculation, we have

$$\begin{aligned}
\sum_{t \in [T]} \omega_t \sum_{j=1}^{M_t} \sum_{\mathcal{U} \in \mathscr{U}_{tj}} d^{\mathcal{U}} \cdot \check{h}_{tj}^{\mathcal{U}} - \sum_{i \in [Q]} p_{R_i} \cdot \check{x}_i &= \sum_{t \in [T]} \omega_t \sum_{j=1}^{M_t} \sum_{\mathcal{U} \in \mathscr{U}_{tj}} \sum_{R_i \in \mathcal{U}} q_{R_i} \cdot \check{h}_{tj}^{\mathcal{U}} - \sum_{i \in [Q]} p_{R_i} \cdot \check{x}_i \\
&= \sum_{t \in [T]} \omega_t \sum_{i \in [Q]} \sum_{j \in \mathcal{J}_i^t} \sum_{\mathcal{U} \in \mathscr{U}_{tj}: R_i \in \mathcal{U}} q_{R_i} \cdot \check{h}_{tj}^{\mathcal{U}} - \sum_{i \in [Q]} p_{R_i} \cdot \check{x}_i = \sum_{t \in [T]} \omega_t \sum_{i \in [Q]} \sum_{j \in \mathcal{J}_i^t} q_{R_i} \cdot \check{y}_{ij}^t - \sum_{i \in [Q]} p_{R_i} \cdot \check{x}_i = \mathrm{CS}(\check{\mathbf{x}}, \check{\mathbf{y}}).
\end{aligned}$$

□

Proof of Proposition 4. Note that all the solutions $\{\bar{\mathbf{x}}^{(k)}, \bar{\mathbf{h}}^{(k)}\}$ can be bounded by a polytope so the L_2-norm of all subgradients $\nabla L(\boldsymbol{\lambda}^{(k)})$ are bounded, i.e. there exists a constant C such that $\sup\{\|\nabla L(\boldsymbol{\lambda}^{(k)})\|\} \leq C$. Therefore, we apply the following lemma:

Lemma EC.5 **(Proposition 3.2.6 in Bertsekas (2015)).** *Consider using subgradient methods for minimizing a real-valued convex function $f: \mathcal{R}^n \to \mathcal{R}$ over a closed convex set $\mathcal{X}$. In particular, we focus on the method*

$$\mathbf{x}_{k+1} = P_{\mathcal{X}}(\mathbf{x}_k - \alpha_k \mathbf{g}_k),$$

where $\mathbf{g}_k$ is any subgradient of f at $\mathbf{x}_k$, α_k is a positive step size, and $P_{\mathcal{X}}(\cdot)$ denotes the projection on the set $\mathcal{X}$ (with respect to the Euclidean norm). Assume that for some scalar c, we have $c \geq \sup\{\|\mathbf{g}_k\| | k = 0, 1, \ldots\}$. If $\lim_{k\to\infty} \alpha_k = 0, \sum_{k=0}^{\infty} \alpha_k = \infty, \sum_{k=0}^{\infty} \alpha_k^2 < \infty$, and the optimal solution set is not empty, then $\{\mathbf{x}_k\}$ converges to some optimal solution.

By setting $\alpha_k = 1/k$, the conditions of the lemma are satisfied, and it follows that $\{\boldsymbol{\lambda}^{(k)}\}$ converges to an optimal solution to the dual problem (6).

To recover the optimal primal solution to the linear relaxation of (SAA-SAP), we invoke the following result:

Lemma EC.6 **(Corollary 3 and Theorem 2 in Sherali and Choi (1996)).** *Consider a primal problem: minimize $\mathbf{c}^\top \mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X}$ is a nonempty polytope in $\mathcal{R}^n$, and its dual problem: maximize $\{\theta(\boldsymbol{\lambda}) : \boldsymbol{\lambda} \geq \mathbf{0}\}$, where $\theta(\boldsymbol{\lambda})$ is the Lagrangian subproblem: $\theta(\boldsymbol{\lambda}) = \min\{\mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\mathbf{A}\mathbf{x} - \mathbf{b}) | \mathbf{x} \in \mathcal{X}\}$. Consider the subgradient method as follows:*

$$\mathbf{g}_k = \mathbf{A}\mathbf{x}_k - \mathbf{b} \text{ where } \mathbf{x}_k \text{ solves for } \theta(\boldsymbol{\lambda}_k), \text{ and } \lambda_{k+1,i} = [\lambda_{ki} + \alpha_k g_{ki}]^+, \text{ where } \alpha_k \text{ is the step size.}$$

If $\alpha_k = 1/k$ and $\{\boldsymbol{\lambda}_k\}$ converges to some feasible solution $\bar{\boldsymbol{\lambda}}$, then $\bar{\boldsymbol{\lambda}}$ is the optimal solution to the dual problem and any accumulation point of $\left\{\frac{\sum_{j=1}^k \mathbf{x}_j}{k}\right\}$ is the optimal solution to the primal problem.

By this lemma, there exists a subsequence of $\{\bar{\mathbf{h}}^{(k)}\}$ that converges to $\mathbf{h}^*$ of some optimal LP solution $\{\mathbf{x}^*, \mathbf{h}^*\}$. Moreover, since the maximization is over the finite set, $\{\bar{\mathbf{x}}^{(k)}\}$ of this subsequence also converges to $\mathbf{x}^*$. □

EC.4. An Integer Program of (SAA) under Specific Cost Modeling

In this section, we present an integer programming (IP) formulation for the sample-average approximation of the prepacking problem under the specific cost model described in Example 1. This formulation does not rely on any restrictive assumptions about the parameters and maintains a polynomial number of variables and constraints with respect to the input size. Although developing exact IP approaches is not the main focus of this work, we provide this formulation as an alternative to (SAA-K), particularly because its size remains polynomial even when the prepackage size is unbounded.

We begin by using a set of binary vectors as decision variables to represent a prepacking scheme $\mathcal{P} = \{(S_i, n_i)\}_{i=1}^{L}$. Specifically, each prepackage S_i is encoded by a binary vector $\mathbf{s}_i \in \{0,1\}^N$, where $s_{ik} = 1$ indicates that item $k \in [N]$ is included in S_i. The number of prepackages L is also a decision variable, leading to a variable number of $\mathbf{s}_i$'s. The following lemma shows that L can be upper bounded by a quantity that is polynomial in the input size:

LEMMA EC.7. *Suppose the packing cost is given by $p_S = p + w$ and the cost savings from packing an order with prepackage S is $q_S = c(|S|-1) + w$, for any $S \subseteq [N]$. Assume, without loss of generality, that $\omega_1 \geq \omega_2 \geq \cdots \geq \omega_T \geq 0$ and $\sum_{t=1}^{T} \omega_t = 1$. Then the number of prepackages in the optimal solution to (SAA) is upper bounded by $\bar{L}$, where $\bar{L} = \frac{\sum_{t=1}^{T}\sum_{j=1}^{M_t}|O_j^t|}{\min_{k\in\mathbb{Z}_{>0}}\{k\cdot\zeta(k)\}}$, with $\zeta(k) = \arg\min_{t\in\mathbb{Z}_{>0}}\left\{\left(\sum_{t'=1}^{t}\omega_{t'}\right)\cdot(c(k-1)+w) > p+w\right\}$.*

Proof of Lemma EC.7. For a prepackage of size k to yield a positive total cost saving, it must be used in at least $\zeta(k)$ orders. Since each usage occupies at least k items, the total number of items occupied by this prepackage is at least $k\cdot\zeta(k)$. Therefore, the total number of prepackages in the optimal solution is upper bounded by the total number of items across all orders divided by $\min_{k\in\mathbb{Z}_{>0}}\{k\cdot\zeta(k)\}$. □

This result slightly improves upon the trivial upper bound $\frac{1}{2}\sum_{t=1}^{T}\sum_{j=1}^{M_t}|O_j^t|$, which follows directly from the fact that each prepackage must contain at least two items. Specifically, we observe that $\zeta(k) \geq 1$ for all $k \geq 2$, and $\zeta(1) = \infty$, implying $\min_{k\in\mathbb{Z}_{>0}}\{k\cdot\zeta(k)\} \geq 2$, and thus yielding a tighter bound.

Using Lemma EC.7, we can now transform (SAA) into the following model.

$$
\begin{aligned}
\max_{\mathbf{y},\mathbf{s},\mathbf{l},\mathbf{f},L} \quad & \sum_{t=1}^{T} \omega_t \cdot f^t - (w+p)L && && \text{(SAA-Vec)}\\
\text{s.t.} \quad & f^t \leq \sum_{i\in[\bar{L}],j\in[M_t]} (c(\sum_{k\in[N]} s_{ik} - 1) + w) y_{ij}^t, && \forall t \in [T] && \text{(EC.10a)}\\
& L = \sum_{i=1}^{\bar{L}} l_i && && \text{(EC.10b)}\\
& l_i \geq s_{ik}, && \forall i \in [\bar{L}], k \in [N] && \text{(EC.10c)}\\
& \sum_{i\in[\bar{L}]} s_{ik} y_{ij}^t \leq o_{jk}^t, && \forall j \in [M_t], k \in [N], t \in [T] && \text{(EC.10d)}\\
& \sum_{j=1}^{M_t} y_{ij}^t \leq l_i, && \forall i \in [\bar{L}], t \in [T] && \text{(EC.10e)}\\
& l_i \in \{0,1\}, y_{ij}^t \in \{0,1\}, s_{ik} \in \{0,1\}, && \forall i \in [\bar{L}], j \in [M_t], t \in [T] && \text{(EC.10f)}
\end{aligned}
$$

In (SAA-Vec), we use $\bar{L}$ binary variables (i.e., l_i) to indicate the number of prepackages, and $l_i = 1$ if S_i contains at least one item, as reflected by (EC.10c). The assignment variables $\mathbf{y}$ are the same as those in (SAA-K). The coefficient $o_{jk}^t = 1$ if $k \in O_j^t$ and 0 otherwise, and the bilinear term $s_{ik} y_{ij}^t$ is equal to 1 if prepackage S_i fulfills item k of order O_j^t and 0 otherwise. The constraint (EC.10a) is constructed for the value of evaluation problem.

(SAA-Vec) is a nonlinear model because of the bilinear terms $s_{ik}y_{ij}^t$ appearing in (EC.10a) and (EC.10d). This can be addressed by linearization with McCormick Relaxation (McCormick 1976). That is, the product of two binary variables can be replaced with four linear inequalities. We thus have the following linear integer program equivalent to (SAA-Vec).

$$\max_{\mathbf{z},\mathbf{y},\mathbf{s},\mathbf{l},\mathbf{f},L} \quad \sum_{t=1}^{T}\omega_t \cdot f^t - (w+p)L \tag{SAA-IP}$$

$$\text{s.t.} \quad f^t \le c \sum_{i\in[\bar{L}],j\in[M_t],k\in[N]} z_{ijk}^t + (w-c)\sum_{i\in[\bar{L}],j\in[M_t]} y_{ij}^t, \qquad \forall t\in[T] \tag{EC.11a}$$

$$\sum_{i=1}^{\bar{L}} z_{ijk}^t \le o_{jk}^t, \qquad \forall j\in[M_t], k\in[N], t\in[T] \tag{EC.11b}$$

$$\begin{cases} z_{ijk}^t \ge s_{ik} + y_{ij}^t - 1 \\ z_{ijk}^t \le y_{ij}^t \\ z_{ijk}^t \le s_{ik} \\ z_{ijk}^t \ge 0, \end{cases} \qquad \forall i\in[\bar{L}], j\in[M_t], k\in[N], t\in[T] \tag{EC.11c}$$

$$(EC.10b), (EC.10c), (EC.10e), (EC.10f),$$

In (SAA-IP), we introduce auxiliary variables z_{ijk}^t to replace the bilinear terms $s_{ik}y_{ij}^t$, with their equivalence enforced by constraints (EC.11c). This reformulation ensures that the size of the integer program remains polynomial in the input size. Specifically, the model includes $\bar{L}N$ variables $\{\mathbf{s}_i\}_{i=1}^{\bar{L}}$ and $\bar{L}$ variables $\{l_i\}_{i=1}^{\bar{L}}$ to represent prepackages, $O(\bar{L}\sum_{t=1}^{T} M_t)$ assignment variables $\mathbf{y}$, and $O(\bar{L}N\sum_{t=1}^{T} M_t)$ auxiliary variables $\mathbf{z}$.

While this formulation is relatively compact, its scale can still be computationally prohibitive for instances with a large number of items and orders. As such, developing more practical integer programming approaches remains a possible direction for future research.

EC.5. Robustness Check

In this section, we conduct robustness checks by varying the cost parameters and the maximum prepackage size K. We examine how these choices affect the performance of anticipatory packing and report additional statistics on the structure and usage of the resulting prepackages.

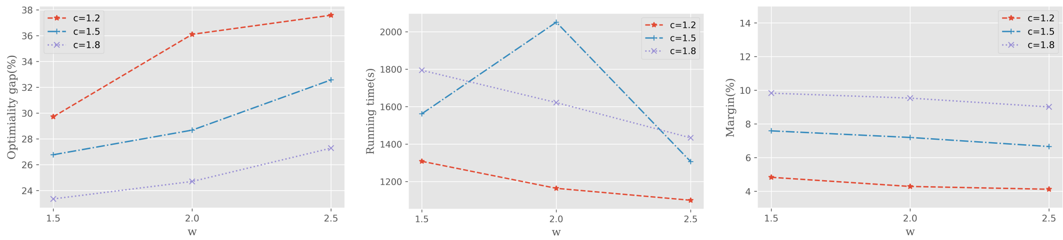
First, Figures EC.2 and EC.3 present the results for different values of w and c, based on the representative settings for both the real data $(K,T)=(3,15)$ and the synthetic data $(N,\mu)=(100,800)$ with $(K,T)=(5,30)$. Under the least favorable setting where $w=2.5$ and $c=1.2$, the average margin remains at 4.12% for the real data and 8.50% for the synthetic data, demonstrating the robustness of our approximation algorithm's performance. We observe that w and c exhibit a consistent pattern in their impact on the optimality gap, running time and margin. A few plausible explanations are as follows: Regarding the optimality gap, as w gets larger or c gets smaller, fewer prepackages will be packed, and a prepackage can potentially be used by more orders, which complicates the combinatorial nature and hence leads to a larger optimality gap. On the running time, as w increases or c decreases, the set of effective prepackages becomes smaller, potentially reducing the algorithm's running time due to a smaller search space. On the margin, intuitively, the cost benefit from anticipatory packing diminishes when w is higher or c is lower, which reduces the relative savings.

Second, we examine the effect of K for a fixed T. Recall that K denotes the maximum number of items allowed in a prepackage. To investigate the effect of K, we report the in-sample and out-of-sample performance of the greedy benchmark and our approximation algorithm on the real data for $T = 15$ and $K \in \{2,3,4,5,6,7\}$. The results are presented in Table EC.1. Increasing K has both benefits and costs. On the benefit side, a larger K enlarges the feasible solution space and can therefore potentially yield greater cost savings. On the cost side, there are two effects. First, from a computational standpoint, a larger K makes the SAA problem more difficult to solve, leading to a larger optimality gap. Second, from a statistical standpoint, larger prepackages are more specialized—they require a specific combination of items to co-occur in an order—and their estimated benefits are based on fewer matching orders in the sample. This makes the SAA solution more susceptible to overfitting the training scenarios, which can degrade out-of-sample performance. As shown in Table EC.1, these effects are reflected in the results. When $K = 2$, the optimality gap is the smallest, but the model excludes many practically useful prepackages, limiting the achievable cost savings. When $K > 3$, the gain from the enlarged solution space is minimal, because prepackages of size greater than three are rare in the optimal prepacking scheme (see Figure EC.4(a)); meanwhile, the increased computational difficulty and estimation noise erode the potential benefit. Consequently, a moderate value of K is sufficient: setting $K = 3$ strikes the best balance between solution quality, computational tractability, and statistical reliability, and delivers the best out-of-sample performance in our experiments.

Table EC.1 The Effect of Varying K on the Real Data $(p, w, c) = (1, 2, 1.5)$.

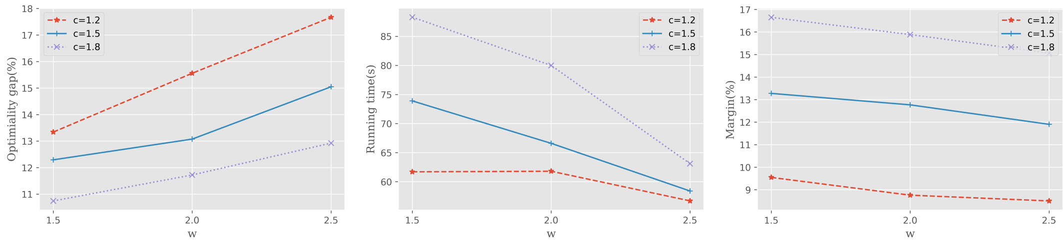
	GREEDY									APPROX								
(K,T)	Optimality gap(%)			Running time(s)			Margin(%)			Optimality gap(%)			Running time(s)			Margin(%)		
	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min
(2, 15)	40.4	41.9	38.7	35.3	55	14	4.50	5.91	-0.02	19.8	24.4	15.3	1319.5	1989	863	4.89	5.58	2.92
(3, 15)	54.9	56.8	52.7	43.5	72	23	5.38	7.63	0.35	28.7	47.3	22.8	2051.0	3460	1249	7.19	9.54	4.66
(4, 15)	57.3	63.3	54.2	110.5	198	43	5.23	7.32	-0.35	34.1	56.8	25.6	2417.1	4358	1433	6.99	9.25	4.94
(5, 15)	57.8	63.4	55.3	77.8	134	33	5.17	7.36	0.07	34.3	56.8	25.7	2675.9	4313	1650	7.00	9.32	5.03
(6, 15)	57.7	64.0	54.4	162.5	313	71	5.13	7.32	-0.07	34.2	56.9	25.7	2579.4	4830	1548	7.01	9.26	4.93
(7, 15)	57.5	63.6	54.4	93.5	156	44	5.19	7.71	-0.15	34.2	56.1	26.0	1987.1	4251	937	6.98	9.21	4.86

Third, we provide a detailed examination of how prepacking operates in practice. Figure EC.4 illustrates the distribution of prepackage types under different values of K for the real data with $T = 15$. As shown in Figure EC.4(a), most prepackages consist of two to three items, and the value of K has



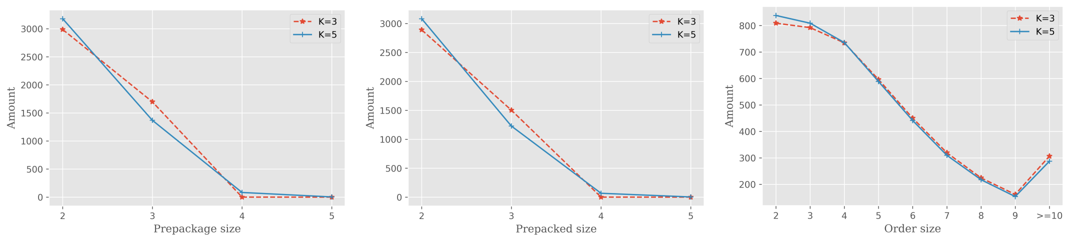
(a) (b) (c)

Figure EC.2 The Effect of Varying w and c on the Real Data



(a) (b) (c)

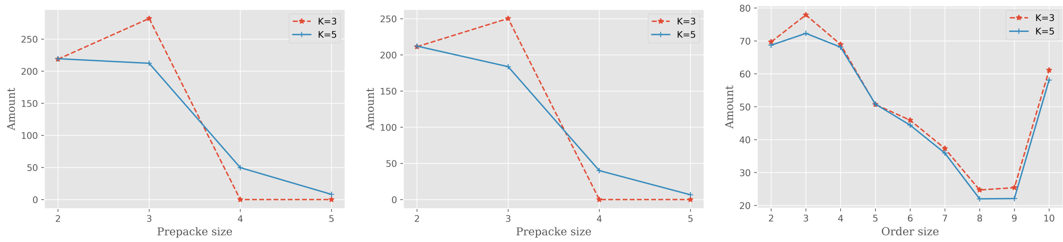
Figure EC.3 The Effect of Varying w and c on the Synthetic Data



(a) Prepackages distributed in package size (b) Used prepackages distributed in package size (c) Used prepackages distributed in order size

Figure EC.4 Statistics of Prepacked Packages on the Real Data

minimal impact on this distribution. A similar pattern is observed in the distribution of prepackages that are actually used, as depicted in Figure EC.4(b). Notably, a high proportion of prepackages are utilized during the peak period—for example, 94.55% under the setting $(K,T) = (5,15)$ on the real data—indicating that prepacking is executed in an effective manner. Figure EC.5 presents similar findings for the synthetic data under the setting $(N,\mu) = (100,800)$ and $T = 30$.



(a) Distribution of prepackage sizes (b) Distribution of the size of used prepackages (c) The size of orders that used prepackages

Figure EC.5 Statistics of Prepacked Packages on the Synthetic Data