Real-Time Omnichannel Fulfillment Optimization

Zuguang Gao, Zi (Elaine) Ling, Varun Gupta, and Linwei Xin

The University of Chicago Booth School of Business
Overview

1. Introduction

2. Model

3. Algorithms
   - N-store identical inventories
   - 2-store non-identical inventories
   - Randomized algorithms

4. Conclusion
Introduction

Trend for retailers to implement omnichannel strategies.
Motivation

- Nowadays, it is just less competitive to run a pure-online or pure-offline business.
Motivation

Challenges on the integration of online and offline businesses:

• Cost of in-store picking is higher than picking at fulfillment centers.
• In-store picking is less efficient than warehouse-picking.
• Prioritize and protect offline customers from stock-outs due to over-fulfillment of online customer.

Research Question

When should we fulfill online orders by using in-store inventory?

Our answer: algorithms for real-time omnichannel fulfillment with performance guarantees.

• $N$-stores, identical inventory
• 2-stores, asymmetric inventory
• Randomized algorithms


• **Competitive analysis:** Ball and Queyranne (2009), Golrezaei, Nazerzadeh and Rusmevichientong (2014), Chen et al. (2016), Ma, Simchi-Levi, and Teo (2021), Manshadi et al. (2022)
Model

- $N$ bricks-and-mortar stores, where the $i$-th store has initial inventory $I_i$.
- One type of product. No inventory replenishment.
- Profit of selling one-unit of product through the offline (resp., online) channel is $p_1$ (resp., $p_0$), with $r := \frac{p_0}{p_1} \leq 1$.
- Customers arrive one by one, each placing one order.
- Offline customers arriving at store $i$ is of type $i$. Online customers are of type $0$.
- The $t$-th order is represented as $(i_t, \ell_t)$.

**Competitive Ratio**

Goal: develop online algorithms to maximize the following competitive ratio.

$$\text{RATIO(ALG)} = \min_{T \geq 1} \min_{(i_1, \ell_1), \ldots, (i_T, \ell_T)} \frac{\text{ALG}[(i_1, \ell_1), (i_2, \ell_2), \ldots, (i_T, \ell_T)]}{\text{OPT}[(i_1, \ell_1), (i_2, \ell_2), \ldots, (i_T, \ell_T)]}$$
Protection Level Algorithm

The algorithm $\text{ALG}(\alpha_1, \ldots, \alpha_N)$ fulfills orders as follows.

- Upon seeing an offline order at store $i$, fulfill it as much as possible by the remaining available inventory at store $i$.

- Let $S \subset [N]$ be the set of stores that have available inventories and their so far used inventories for online orders have not exceeded $\alpha_i I_i$ for all $i \in S$. Upon seeing an online order, the order is fulfilled by all stores in $S$, keeping that the amount fulfilled from store $i$ is proportional to $\alpha_i I_i$. If at any point the total fulfilled online demand from store $j$ reaches $\alpha_j I_j$, then $S \leftarrow S \setminus \{j\}$, and the fulfillment continues until either we have fulfilled all demands in this online order, or $S$ becomes an empty set.
N-store Identical Inventories

Upper Bound

For the continuous $N$-store problem ($N \geq 2$) with identical inventory $I$, no deterministic online algorithm has a competitive ratio larger than

$$UB_1 \triangleq \min \left\{ \frac{1}{2 - r}, \min_{K \in \{1, \ldots, N-1\}} \min_{K \geq (1-r)N} \left\{ 1 - \frac{(N-K)K}{N(Kr + N - K)} \right\} \right\}.$$  

Proof sketch:

- Consider the arrival sequences: For $K = 1, \ldots, N$, 1) $KI$ online orders; 2) $KI$ online orders followed by $I$ offline orders at each of the $N$ stores.
- Suppose the algorithm fulfills $x$ units of online orders from each store.
- Write the profit expression for the arrival sequences and find the max-min.
Recall the algorithm $\text{ALG}_{(\alpha_1, \ldots, \alpha_N)}$. Consider only $\alpha_1 = \cdots = \alpha_N = \alpha$.

Finding the best $\alpha$

For each $N \geq 3$, define $\hat{K}^* \triangleq \min \left\{ \left\lfloor \frac{N}{2-r} \right\rfloor + 1, N \right\}$. Let $\alpha^* = \min \left\{ \frac{1}{2-r}, \frac{\hat{K}^*}{N} \right\}$, then, we have that

$$\text{RATIO} \left( \text{ALG}_{\alpha^*} \right) = \text{UB}_1.$$

Proof sketch:

• Consider general arrival sequence: $(i_0, \ell_0), (i_1, \ell_1), \ldots, (i_N, \ell_N)$ with $i_k = k$.

• In $\text{ALG}_{\alpha^*}$, store $k$ fulfills $\min \left\{ \frac{\ell_0}{N}, \alpha \ell \right\}$ online demands and $\min \left\{ l - \min \left\{ \frac{\ell_0}{N}, \alpha \ell \right\}, \ell_k \right\}$ offline demands.

• Write the profit expression and find the max-min (take min over $\{\ell_k\}$ then max over $\alpha$).
Suppose store 1 has initial inventory $I$ and store 2 has initial inventory $qI$, where $q \geq 1$.

### Upper Bound

No deterministic online algorithm has a competitive ratio larger than

$$\text{UB}_2 \triangleq \min_{1 \leq t \leq (1+q)I} \max_{x+y \leq t} \min \left\{ \frac{x+y}{t}, \frac{(x+y)r + I - x}{I + (t \wedge qI)r}, \frac{(x+y)r + qI - y}{qI + (t \wedge I)r}, \frac{(x+y)(r-1) + (1+q)I}{(1+q)I} \right\}$$

#### Proof sketch:

- Consider the arrival sequences: 1) $t$ online orders; 2) $t$ online orders followed by $I$ offline orders at store 1; 3) $t$ online orders followed by $qI$ offline orders at store 2; 4) $t$ online orders followed by $I$ offline orders from store 1 and $qI$ offline orders from store 2.
- Suppose that after the first $t$ online orders, the algorithm fulfilled $x$ units from store 1 and $y$ units from store 2.
- Write the profit expression for the arrival sequences and find the min-max-min.
Finding the best $\alpha$

Recall the algorithm $\text{ALG}(\alpha_1, \alpha_2)$. Consider only $\alpha_1 = \alpha_2 = \alpha$. Given fixed $q$ and $r$,

- $0 \leq r \leq \left[ \frac{1}{1+q} \vee \left(1 - \frac{1}{q}\right) \right]$, then, $\alpha^* = \frac{1}{2-r}$ and $\text{RATIO}(\text{ALG}_{\alpha^*}) = \frac{1}{2-r}$;
- $\left[ \frac{1}{1+q} \vee \left(1 - \frac{1}{q}\right) \right] < r \leq 1$, then, $\alpha^* \in \left[ 1 - \frac{q}{(1+qr)(1+q)}, \min \left\{ \frac{q}{(1-r)(1+qr)(1+q)}, 1 \right\} \right]$, and $\text{RATIO}(\text{ALG}_{\alpha^*}) = 1 - \frac{q}{(1+qr)(1+q)}$.

Proof sketch:

- Consider general arrival sequence: $(i_0, \ell_0), (i_1, \ell_1), (i_2, \ell_2)$ with $i_k = k$.
- $\text{ALG}_\alpha$ fulfills $\min \left\{ \frac{\alpha l}{\alpha l + \alpha q l} \ell_0, \alpha l \right\} =: A$ online demands from store 1,
  $$\min \left\{ \frac{\alpha ql}{\alpha l + \alpha ql} \ell_0, \alpha ql \right\} =: B$$ online demands from store 2, $\min\{l - A, \ell_1\}$ offline demands from store 1, and $\min\{ql - B, \ell_2\}$ offline demands from store 2.
- Write the profit expression and find the max-min (take min over $\{\ell_k\}$ then max over $\alpha$).
2-store Non-identical Inventories

**Upper Bound**

No deterministic online algorithm has a competitive ratio larger than

\[ UB_2 \triangleq \min_{1 \leq t \leq (1+q)l} \max_{x+y \leq t} \min \left\{ \frac{x+y}{t}, \frac{(x+y)r + l - x}{l + (t \wedge ql)r}, \frac{(x+y)r + ql - y}{ql + (t \wedge l)r}, \frac{(x+y)(r-1) + (1+q)l}{(1+q)l} \right\} \]

**Finding the best \( \alpha \)**

Recall the algorithm \( \text{ALG}_{(\alpha_1, \alpha_2)} \). Consider only \( \alpha_1 = \alpha_2 = \alpha \). Given fixed \( q \) and \( r \),

- \( 0 \leq r \leq \left[ \frac{1}{1+q} \vee \left( 1 - \frac{1}{q} \right) \right] \), then, \( \alpha^* = \frac{1}{2 \cdot r} \) and \( \text{RATIO} (\text{ALG}_{\alpha^*}) = \frac{1}{2 \cdot r} \);
- \( \left[ \frac{1}{1+q} \vee \left( 1 - \frac{1}{q} \right) \right] < r \leq 1 \), then, \( \alpha^* \in \left[ 1 - \frac{q}{(1+qr)(1+q)}, \min \left\{ \frac{q}{(1-r)(1+qr)(1+q)}, 1 \right\} \right] \), and \( \text{RATIO} (\text{ALG}_{\alpha^*}) = 1 - \frac{q}{(1+qr)(1+q)} \).

Do they match? No... Is there hope? Yes!! How? Dynamic thresholds!
Adaptive Algorithm for 2 Stores

WLOG, all orders have size 1. The adaptive algorithm, $\text{ALG}_{\text{adap}}$, works as follows.

- Suppose the number of online items seen so far is $t - 1$ and the next arriving customer is online (making it a total of $t$ online orders), where $t \in [1, (1 + q)n]$.
- Compute $x(t)$ and $y(t)$, the amount of inventory used to fulfill the first $t$ online orders from stores 1 and 2, respectively, by solving

$$\max_{x+y \leq t} \min \left\{ \frac{x+y}{t}, \frac{(x+y)r + n - x}{n + (t \land qn)r}, \frac{(x+y)r + qn - y}{qn + (t \land n)r}, \frac{(x+y)(r - 1) + (1 + q)n}{(1 + q)n} \right\}.$$

- Fulfill $x(t) - x(t - 1)$ amount of online demand from store 1 (if available) and fulfill $y(t) - y(t - 1)$ amount of online demand from store 2 (if available).

Theorem

$\text{ALG}_{\text{adap}}$ is feasible, and $\text{RATIO}(\text{ALG}_{\text{adap}}) = \text{UB}_2$. 
Randomized Algorithm (N-store)

Discrete setting: all orders are of size 1, cannot do fractional fulfillment.
Randomized algorithm:

- Upon seeing an offline order at location \( i \), the order is fulfilled if there is remaining available inventory at location \( i \).
- Upon seeing an online order, first decide the probability of accepting, then if accepted, randomly pick a store to fulfill the order while trying to keep the number of online order fulfilment even across stores.

Example

Consider 3 stores with \( I = 10 \) and \( \alpha = 0.25 \). Then, with online order arrivals:

- \((0, 1, 0), (0, 1, 1), (1, 1, 1), (1, 1, 2), (2, 1, 2), (2, 2, 2)\)
- 7th online order: accept since \( N\alpha I = 7.5 > 7 \). Randomly from one store, e.g. \((3, 2, 2)\)
- 8th online order: accept with probability \( N\alpha I - 7 = 0.5 \). If accepted, randomly from store 2 or store 3, e.g. \((3, 2, 2)\) w.p. 0.5; \((3, 3, 2)\) w.p. 0.25; \((3, 2, 3)\) w.p. 0.25.
Randomized Algorithm (2-store)

Recall from $\text{ALG}_{\text{adap}}$ where we have a sequence $\{x(t), y(t)\}_{t \in [T]}$ satisfying:

- $x(0) = y(0) = 0$
- $x(t + 1) \geq x(t)$, $y(t + 1) \geq y(t)$ for all $t \in \{0, \ldots, T - 1\}$, and
- $x(t + 1) + y(t + 1) \leq x(t) + y(t) + 1$.

Goal: create stochastic processes $X(t)$, $Y(t)$ satisfying

- $\mathbb{E}[X(t)] = x(t)$, $\mathbb{E}[Y(t)] = y(t)$ for all $t \in [T]$,
- $\lfloor x(t) \rfloor \leq X(t) \leq \lceil x(t) \rceil$, $\lfloor y(t) \rfloor \leq Y(t) \leq \lceil y(t) \rceil$, almost surely.
Randomized Algorithm (2-store)

\[ x(t) + y(t) \leq \lfloor x(t) \rfloor + \lfloor y(t) \rfloor + 1 \]

\[ x(t) + y(t) > \lfloor x(t) \rfloor + \lfloor y(t) \rfloor + 1 \]
Randomized Algorithm (2-store)

- $(X(t), Y(t))$ on points $P_{LL}, P_{ML}, P_{LM}$. Probability masses $q_{LL}(t), q_{ML}(t), q_{LM}(t)$.
- $(X(t+1), Y(t+1))$ on points $P_{MM}, P_{ML}, P_{LM}$. Probability masses $q_{MM}(t+1), q_{ML}(t+1), q_{LM}(t+1)$.
- Given $x(t), y(t), x(t+1), y(t+1)$, find $q_{LL}(t), q_{ML}(t), q_{LM}(t), q_{MM}(t+1), q_{ML}(t+1), q_{LM}(t+1)$.

Given $(X(t), Y(t))$, $(x(t), y(t))$ and $(x(t+1), y(t+1))$, we need to define the joint distribution of $(X(t+1), Y(t+1))$. 

Figure: Illustration of flows
Randomized Algorithm (2-store)

- Source capacity constraints:
  \[ f_{LLLL} + f_{LLML} = q_{LL}(t) \]
  \[ f_{LMMM} + f_{LMML} = q_{LM}(t) \]
  \[ f_{MLMM} + f_{MLML} = q_{ML}(t) \]

- Sink demand constraints:
  \[ f_{LMMM} + f_{MLMM} = q_{MM}(t + 1) \]
  \[ f_{LLLL} + f_{LMMM} = q_{LM}(t + 1) \]
  \[ f_{LLML} + f_{MLML} = q_{ML}(t + 1) \]

One can verify the existence of the flows, which give a rounding mechanism, e.g., if \( X(t), Y(t) = P_{ML}(t) \), then \( X(t + 1), Y(t + 1) \) is chosen to be \( P_{MM} \) with probability \( \frac{f_{MLMM}}{q_{ML}(t)} \) and \( P_{ML} \) with probability \( \frac{f_{MLML}}{q_{ML}(t)} \).
Conclusion

Summary:

- Online protection level algorithm with a tight competitive ratio for the $N$-store problem with identical inventory.

- With static booking limits, the optimal competitive ratio cannot be achieved with non-identical inventory, in contrast to the one store case in Ball and Queyranne (2009).

- Deterministic online algorithm with a tight competitive ratio for the 2-store problem, with dynamic booking limits.

- Randomized online algorithms for the discrete problems with the same competitive ratios.
Thank you!

Questions? zuguang.gao@chicagobooth.edu