Dynamic Seat Assignment with Social Distancing

IEDA
The Hong Kong University of Science and Technology

Table of Contents

1. Introduction
2. Literature Review
3. Problem Definition
4. Seat Planning by Stochastic Programming
5. Dynamic Seat Assignment for Each Group Arrival
6. Numerical Results

Introduction

Social Distancing under Pandemic

- Governments issued the policy about social distancing constraint.

- Theater tickets booking: assign to seat.

Literature Review

Seat Planning with Social Distancing

- Seat planning on airplanes, classrooms, trains.

- Group seat assignment in amphitheaters, airplanes, theater.

Dynamic Seat Assignment

- Dynamic multiple knapsack problem

- Social distancing and group seats constraints make assign-to-seat necessary, which is different from traditional revenue management.

Problem Definition

Seat Planning with Social Distancing

- Group type $\mathcal{M} = \{1, \ldots, M\}$.
- Row $\mathcal{N} = \{1, \ldots, N\}$.
- The social distancing: δ seat(s).
- $n_i = i + \delta$: the new size of group type i for each $i \in \mathcal{M}$.
- The number of seats in row j: $S_j, j \in \mathcal{N}$.
- $L_j = S_j + \delta$: the length of row j for each $j \in \mathcal{N}$.

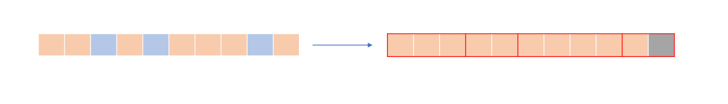


Figure: Problem Conversion with One Seat as Social Distancing

Basic Concepts

- Pattern: the seat planning for each group type in one row. Denoted by $P_k = (t_1, \ldots, t_M)$, where t_i is the number of group type i.
- For each pattern k, α_k, β_k indicate the number of groups and the left seats, respectively.
- Loss for pattern k: $\alpha_k \delta + \beta_k - \delta$.
- The largest patterns: patterns with the minimal loss.
- Full patterns: $\beta_k = 0$.
- Example:
  $\delta = 1$, $M = 4$, $n_i = i + 1, i \in \mathcal{M}$, $L = 21$.
  Largest patterns: $(0, 0, 0, 4), (0, 0, 4, 1), (0, 2, 0, 3)$.
  Not a Full pattern: $(0, 0, 0, 4)$.

Loss of The Largest Patterns

- A largest pattern can be obtained by the greedy way: select the maximal group size, n_M, as many times as possible, then $L = n_M \cdot q + r, 0 \leq r < n_M$, r is the number of empty seats.
- When $r > \delta$, these seats can be occupied by the group type $(r - \delta)$; when $r \leq \delta$, leave these seats empty.

* Loss of the largest patterns: $q\delta - \delta + f(r)$, where $f(r) = 0$ if $r > \delta$; $f(r) = r$ if $r \leq \delta$.

* For a seat layout, $\{S_1, S_2, \ldots, S_N\}$, the minimal total loss: $\sum_j (\lfloor \frac{S_j+\delta}{n_M} \rfloor - \delta + f((S_j + \delta) \mod n_M))$. The maximal number of people assigned: $\sum_j (S_j - \lfloor \frac{S_j+\delta}{n_M} \rfloor + \delta - f((S_j + \delta) \mod n_M))$.

Dynamic Seat Assignment Problem

- $T+1$ periods in total.
- There is one and only one group arrival each period.
- The probability of an arrival of group type i: p_i.
* Remaining capacity: $\mathbf{L} = (l_1, l_2, \ldots, l_N)$.
- Value function: $V_t(\mathbf{L})$.
- $\mathbf{e}_j^\intercal$: decision variable whether to assign the current group to row j.

$$V_t(\mathbf{L}) = \sum_i p_i \left[\max_{\substack{j \in \mathcal{N}: \\ L_j \geqslant n_i}} \left\{ V_{t+1}\left(\mathbf{L} - n_i \mathbf{e}_j^\intercal\right) + i, V_{t+1}(\mathbf{L}) \right\} \right], V_{T+1}(\mathbf{L}) = 0.$$

- DP is computationally complex caused by the curse of dimensionality.
- We give a seat planning by stochastic programming firstly, then apply stochastic planning policy to make the decision.

Seat Planning by Stochastic Programming

Method Flow

- The formulation of scenario-based stochastic programming(SSP).
- Transform (SSP) to stochastic programming. Problem (1) → Problem (2).
- Transform Problem (2) to a restricted benders master problem(RBMP) with generative constraints.
- (RBMP) can be solved iteratively.
- Obtain seat planning by deterministic model.

Scenario-based Stochastic Programming

$$\begin{aligned}
(SSP)\max \quad & E_{\omega}\left[\sum_{i=1}^{M-1}(n_i-\delta)(\sum_{j=1}^{N}x_{ij}+y_{i+1,\omega}^{+}-y_{i\omega}^{+})+(n_M-\delta)(\sum_{j=1}^{N}x_{Mj}-y_{M\omega}^{+})\right] \\
\text{s.t.} \quad & \sum_{j=1}^{N}x_{ij}-y_{i\omega}^{+}+y_{i+1,\omega}^{+}+y_{i\omega}^{-}=d_{i\omega}, \quad i=1,\ldots,M-1,\omega\in\Omega \\
& \sum_{j=1}^{N}x_{ij}-y_{i\omega}^{+}+y_{i\omega}^{-}=d_{i\omega}, \quad i=M,\omega\in\Omega \\
& \sum_{i=1}^{M}n_i x_{ij}\leq L_j, j\in\mathcal{N} \\
& y_{i\omega}^{+},y_{i\omega}^{-}\in\mathbb{Z}_{+}, \quad i\in\mathcal{M},\omega\in\Omega \\
& x_{ij}\in\mathbb{Z}_{+}, \quad i\in\mathcal{M},j\in\mathcal{N}.
\end{aligned} \tag{1}$$

Stochastic Programming

$$\begin{aligned} \max \quad & \mathbf{c}^{\intercal}\mathbf{x} + z(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{n}\mathbf{x} \leq \mathbf{L} \\ & \mathbf{x} \in \mathbb{Z}_{+}^{M \times N}, \end{aligned} \tag{2}$$

where $z(\mathbf{x})$ is the recourse function defined as

$$z(\mathbf{x}) := E(z_{\omega}(\mathbf{x})) = \sum_{\omega \in \Omega} p_{\omega} z_{\omega}(\mathbf{x}),$$

and for each scenario $\omega \in \Omega$,

$$\begin{aligned} z_{\omega}(\mathbf{x}) := \max \quad & \mathbf{f}^{\intercal}\mathbf{y}_{\omega} \\ \text{s.t.} \quad & \mathbf{x}\mathbf{1} + \mathbf{V}\mathbf{y}_{\omega} = \mathbf{d}_{\omega} \\ & \mathbf{y}_{\omega} \geq 0. \end{aligned} \tag{3}$$

Solve The Second Stage Problem

Problem (3) is easy to solve with the given x which can be seen by the dual problem:
The dual of problem (3) is

$$\begin{aligned} \min \quad & \alpha_\omega^\intercal(\mathbf{d}_\omega - \mathbf{x}\mathbf{1}) \\ \text{s.t.} \quad & \alpha_\omega^\intercal \mathbf{V} \geq \mathbf{f}^\intercal \end{aligned} \tag{4}$$

Let $P = \{\alpha | \alpha^\intercal V \geq \mathbf{f}^\intercal\}$. The feasible region of problem (4), P, is bounded. In addition, all the extreme points of P are integral.

Delayed Constraint Generation

LP of problem (1) can be obtained by solving following restricted benders master problem(RBMP):

$$
\begin{aligned}
\max \quad & \mathbf{c}^{\intercal} x + \sum_{\omega \in \Omega} p_{\omega} z_{\omega} \\
\text{s.t.} \quad & \sum_{i=1}^{M} n_i x_{ij} \leq L_j, j \in \mathcal{N} \\
& (\alpha^k)^{\intercal} (\mathbf{d}_{\omega} - \mathbf{x1}) \geq z_{\omega}, \alpha^k \in \mathcal{O}^t, \forall \omega \\
& \mathbf{x} \geq 0
\end{aligned}
\tag{5}
$$

Constraints will be generated from problem (4) until the value of RBMP converges.

Benders Decomposition Algorithm

Step 1. Solve LP (5) with all $\alpha_\omega^0 = \mathbf{0}$ for each scenario. Then, obtain the solution $(\mathbf{x}_0, \mathbf{z}^0)$.

Step 2. Set the upper bound $UB = c'\mathbf{x}_0 + \sum_{\omega \in \Omega} p_\omega z_\omega^0$.

Step 3. For x_0, we can obtain α_ω^1 and $z_\omega^{(0)}$ for each scenario, set the lower bound $LB = c'x_0 + \sum_{\omega \in \Omega} p_\omega z_\omega^{(0)}$

Step 4. For each ω, if $(\alpha_\omega^1)'(\mathbf{d}_\omega - \mathbf{x}_0\mathbf{1}) < z_\omega^0$, add one new constraint, $(\alpha_\omega^1)'(\mathbf{d}_\omega - \mathbf{x}\mathbf{1}) \geq z_\omega$, to RBMP.

Step 5. Solve the updated RBMP, obtain a new solution (x_1, z^1) and update UB.

Step 6. Repeat step 3 until $UB - LB < \epsilon$.(In our case, UB converges.)

Deterministic Formulation

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{M}\sum_{j=1}^{N}(n_i - s)x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{N} x_{ij} \leq s_i, \quad i \in \mathcal{M}, \\
& \sum_{i=1}^{M} n_i x_{ij} \leq L_j, j \in \mathcal{N} \\
& x_{ij} \in \mathbb{Z}_+, \quad i \in \mathcal{M}, j \in \mathcal{N}.
\end{aligned} \tag{6}
$$

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{M}\sum_{j=1}^{N}(n_i - s)x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{N} x_{ij} \geq s_i, \quad i \in \mathcal{M}, \\
& \sum_{i=1}^{M} n_i x_{ij} \leq L_j, j \in \mathcal{N} \\
& x_{ij} \in \mathbb{Z}_+, \quad i \in \mathcal{M}, j \in \mathcal{N}.
\end{aligned} \tag{7}
$$

Generate integral seat planning.

Obtain The Feasible Seat Planning

Step 1. Obtain the solution, $\mathbf{x}^*$, from stochatic linear programming by benders decomposition. Aggregate $\mathbf{x}^*$ to the number of each group type, $s_i^0 = \sum_j x_{ij}^*, i \in \mathbf{M}$.

Step 2. Solve problem (6) to obtain the optimal solution, $\mathbf{x}^1$. Aggregate $\mathbf{x}^1$ to the number of each group type, $s_i^1 = \sum_j x_{ij}^1, i \in \mathbf{M}$.

Step 3. Obtain the optimal solution, $\mathbf{x}^2$, from problem (7) with supply $\mathbf{s}^1$. Aggregate $\mathbf{x}^2$ to the number of each group type, $s_i^2 = \sum_j x_{ij}^2, i \in \mathbf{M}$.

Step 4. For each row, construct a full pattern.

Dynamic Seat Assignment for Each Group Arrival

Stochastic Planning Policy

- Group-type control
  - Seat planning from stochastic programming.
  - Small group assigned to larger seats.
- Value of Acceptance and Rejection
  - Compare the value of stochastic programming when assigning in the row versus not assigning.

Bid-price Control

The dual problem of linear relaxation of problem (6) is:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{M} d_i z_i + \sum_{j=1}^{N} L_j \beta_j \\
\text{s.t.} \quad & z_i + \beta_j n_i \geq (n_i - \delta), \quad i \in \mathcal{M}, j \in \mathcal{N} \\
& z_i \geq 0, i \in \mathcal{M}, \beta_j \geq 0, j \in \mathcal{N}.
\end{aligned} \tag{8}
$$

There exists h such that the aggregate optimal solution to relaxation of problem (6) takes the form $xe_h + \sum_{i=h+1}^{M} d_i e_i$, $x = (L - \sum_{i=h+1}^{M} d_i n_i)/n_h$.

Dynamic Programming Based Heuristic

Relax all rows to one row with the same capacity by $L = \sum_{j=1}^{N} L_j$.
Deterministic problem is:
$\{\max \sum_{i=1}^{M} (n_i - \delta) x_i : x_i \leq d_i, i \in \mathcal{M}, \sum_{i=1}^{M} n_i x_i \leq L, x_i \in \mathbb{Z}_+\}$.
Let u denote the decision, where $u(t) = 1$ if we accept a request in period t, $u(t) = 0$ otherwise, the DP with one row can be expressed as:

$$V_t(L) = \mathbb{E}_{i \sim p}[\max_{u \in \{0,1\}} \{[V_{t+1}(L - n_i u) + iu]\}], L \geq 0$$

$$V_{T+1}(x) = 0, \forall x.$$

After accepting one group, assign it in some row arbitrarily when the capacity of the row allows.

Numerical Results

Running time of Benders Decomposition and IP

# of scenarios	demands	running time of IP(s)	Benders (s)	# of rows	# of groups	# of seats
1000	(150, 350)	5.1	0.13	30	8	(21, 50)
5000		28.73	0.47	30	8	
10000		66.81	0.91	30	8	
50000		925.17	4.3	30	8	
1000	(1000, 2000)	5.88	0.29	200	8	(21, 50)
5000		30.0	0.62	200	8	
10000		64.41	1.09	200	8	
50000		365.57	4.56	200	8	
1000	(150, 250)	17.15	0.18	30	16	(41, 60)
5000		105.2	0.67	30	16	
10000		260.88	1.28	30	16	
50000		3873.16	6.18	30	16	

Feasible Seat Planning versus IP Solution

# samples	T	probabilities	# rows	people served by FSP	IP
1000	45	[0.4,0.4,0.1,0.1]	8	85.30	85.3
1000	50	[0.4,0.4,0.1,0.1]	8	97.32	97.32
1000	55	[0.4,0.4,0.1,0.1]	8	102.40	102.40
1000	60	[0.4,0.4,0.1,0.1]	8	106.70	NA
1000	65	[0.4,0.4,0.1,0.1]	8	108.84	108.84
1000	35	[0.25,0.25,0.25,0.25]	8	87.16	87.08
1000	40	[0.25,0.25,0.25,0.25]	8	101.32	101.24
1000	45	[0.25,0.25,0.25,0.25]	8	110.62	110.52
1000	50	[0.25,0.25,0.25,0.25]	8	115.46	NA
1000	55	[0.25,0.25,0.25,0.25]	8	117.06	117.26
5000	300	[0.25,0.25,0.25,0.25]	30	749.76	749.76
5000	350	[0.25,0.25,0.25,0.25]	30	866.02	866.42
5000	400	[0.25,0.25,0.25,0.25]	30	889.02	889.44
5000	450	[0.25,0.25,0.25,0.25]	30	916.16	916.66

Each entry of people served is the average of 50 instances. IP will spend more than 2 hours in some instances, as ‘NA’ showed in the table.

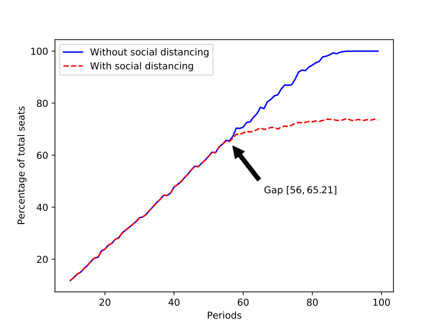
Performances of Different Policies

T	probabilities	Sto(%)	DP1(%)	Bid-price(%)	FCFS(%)
60	[0.25, 0.25, 0.25, 0.25]	99.12	98.42	98.38	98.17
70	[0.25, 0.25, 0.25, 0.25]	98.34	96.87	96.24	94.75
80	[0.25, 0.25, 0.25, 0.25]	98.61	95.69	96.02	93.18
90	[0.25, 0.25, 0.25, 0.25]	99.10	96.05	96.41	92.48
100	[0.25, 0.25, 0.25, 0.25]	99.58	95.09	96.88	92.54
60	[0.25, 0.35, 0.05, 0.35]	98.94	98.26	98.25	98.62
70	[0.25, 0.35, 0.05, 0.35]	98.05	96.62	96.06	93.96
80	[0.25, 0.35, 0.05, 0.35]	98.37	96.01	95.89	92.88
90	[0.25, 0.35, 0.05, 0.35]	99.01	96.77	96.62	92.46
100	[0.25, 0.35, 0.05, 0.35]	99.23	97.04	97.14	92.00
60	[0.15, 0.25, 0.55, 0.05]	99.14	98.72	98.74	98.07
70	[0.15, 0.25, 0.55, 0.05]	99.30	96.38	96.90	96.25
80	[0.15, 0.25, 0.55, 0.05]	99.59	97.75	97.87	95.81
90	[0.15, 0.25, 0.55, 0.05]	99.53	98.45	98.69	95.50
100	[0.15, 0.25, 0.55, 0.05]	99.47	98.62	98.94	95.25

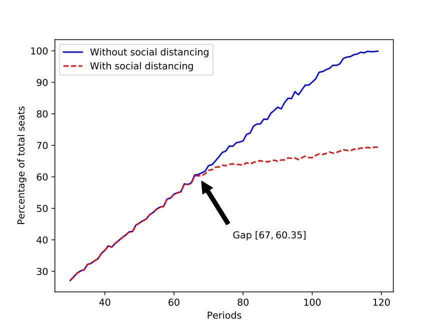
Conclusion:

Impact of Social Distance as Demands Increase

Let $\gamma = p_1 * 1 + p_2 * 2 + p_3 * 3 + p_4 * 4$ denote the number of people at each period.



(a) When $\gamma = 2.5$



(b) When $\gamma = 1.9$

The gap point represents the first period where the number of people without social distancing is larger than that with social distancing and the gap percentage is the corresponding percentage of total seats.

When Supply and Demand Are Close

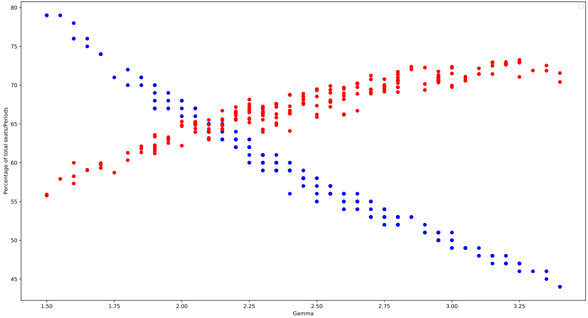


Figure: Gap points under 200 probabilities

Blue points: period of the gap point. Red points: occupancy rate of the gap point.

The End