Seat Planning and Assignment with Social Distancing

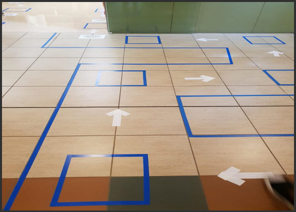
IEDA
The Hong Kong University of Science and Technology

Table of Contents

Introduction

Social Distancing under Pandemic

- Social distancing measures.



Social Distancing under Pandemic

- Social distancing in seating areas.

Situations

- Deterministic Demand
  Make the seat planning with the known the specific demand for each group.
- Stochastic Demand
  Make the seat planning with the known demand distribution before the demand realization.
- Dynamic Demand
  1. Assign seats to each group under fixed seat planning.
  2. Assign seats to each group under flexible seat planning.

  2.1: Assign seats to each group after its realization.

  2.2: Accept or reject each group after its realization, but assign them later.

Literature Review

Seat Planning with Social Distancing

- Allocation of seats on airplanes (Ghorbani et.al 2020), classroom layout planning (Bortolete et al. 2022), seat planning in long-distancing trains (Haque & Hamid 2022).

- Social distancing can be enforced in different groups (Moore et al. 2021).

- Seating planning for known groups in amphitheaters (Haque & Hamid 2022), airplanes (Salari et al. 2022), theater (Blom et al. 2022).

Dynamic Seat Assignment

- Related to multiple knapsack problem (Pisinger et al. 1999) and dynamic knapsack problem (Kleywegt et al. 1998).
- Dynamic seat assignment on airplane (Hamdouch et al. 2011), train (Berge et al. 1993).
- Assign-to-seat: dynamic capacity control for selling high-speed train tickets (Zhu et al. 2023).

Seat Planning with Social Distancing

Prerequisite: people in one group should sit together.

- 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: $L_j^0, j \in \mathcal{N}$.
- $L_j = L_j^0 + \delta$: the length of row j for each $j \in \mathcal{N}$.

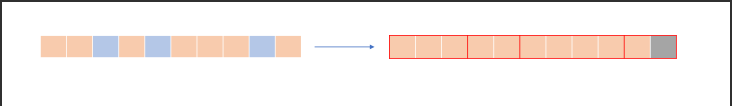


Figure: Problem Conversion with One Seat as Social Distancing

Basic Concepts

- Pattern: $\boldsymbol{h} = (h_1, \ldots, h_M)$, the seat planning for one row whose length is L.
  - The number of people accommodated:
  $|\boldsymbol{h}| = \sum_{i=1}^{M} ih_i = qM + \max\{r - \delta, 0\}$, where $q = \lfloor L/(M+\delta) \rfloor$, $r \equiv L \bmod (M+\delta)$.
- $\boldsymbol{h}$ is a largest pattern if $|\boldsymbol{h}| \geq |\boldsymbol{h}'|$ for any feasible $\boldsymbol{h}'$.
- $\boldsymbol{h}$ is a full pattern if $\sum_{i=1}^{M} n_i h_i = L$.

- Example:
  $\delta = 1,\ M = 4,\ L = 21;\ n_1 = 2, n_2 = 3, n_3 = 4, n_4 = 5,$
  Largest patterns: $(0,0,0,4), (0,0,4,1), (0,2,0,3).$
  Largest may not be full: $(0,0,0,4).$
  Full may not be largest: $(1,1,4,0).$

Seat Planning with Deterministic Demand

Deterministic Formulation

Seat planning problem with given demand $\boldsymbol{d}$:

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

Objective: maximize the number of people accommodated.

Property

In the LP relaxation of problem (1), there exists an index v such that the optimal solutions satisfy the following conditions:

- For $i = 1, \ldots, v-1$ and for all j, $x_{ij}^* = 0$.
- For $i = v+1, \ldots, M$, $\sum_j x_{ij}^* = d_i$.
- For $i = v$, $\sum_j x_{ij}^* = \frac{L - \sum_{i=v+1}^{M} d_i n_i}{n_v}$

When the demand is way smaller than the number of seats, the seat planning obtained from problem (1) does not utilize all seats. We aim to generate a seat planning utilizing all seats while ensuring that the demand requirements are met.

Generate The Full or Largest Pattern

Specifically, we can convert a given specific pattern into a largest or full pattern while ensuring that the original group type requirements are met. Our objective is to generate the pattern with maximal people. Mathematically, for any pattern $\boldsymbol{h} = (h_1, \ldots, h_M)$, we seek to find a pattern $\boldsymbol{h}' = (h_1{}', \ldots, h_M{}')$ that satisfies the following programming.

$$
\begin{aligned}
\max \quad & |\boldsymbol{h}'| \\
\text{s.t.} \quad & h_1{}' \geq h_1 \\
& h_1{}' + h_2{}' \geq h_1 + h_2 \\
& \ldots \\
& h_1{}' + \ldots + h_M{}' \geq h_1 + \ldots + h_M.
\end{aligned}
$$

Algorithm

For each row j, let $\beta = L_j - \sum_i n_i x_{ij}$. We aim to allocate the remaining unoccupied seats(β) in row j in a way that maximizes the number of planned groups that become the largest in size.

- k: the smallest group type. If $k \neq M$, $h_k \leftarrow h_k - 1$, $h_{\min\{(k+\beta),M\}} \leftarrow h_{\min\{(k+\beta),M\}} + 1$, $\beta \leftarrow \beta - \max\{1, M-k\}$. Continue this procedure until the pattern is largest or $\beta = 0$.
- If $k = M$ and $\beta > 0$, assign β in a greedy way.

- When $\beta \geq n_M$, $q \leftarrow \lfloor \frac{\beta}{n_M} \rfloor$, $\beta \leftarrow \beta - q n_M$, $h_M \leftarrow h_M + q$.
- When $n_1 \leq \beta < n_M$, $h_{\beta - n_1 + 1} \leftarrow h_{\beta - n_1 + 1} + 1$, $\beta \leftarrow 0$.
- When $0 < \beta < n_1$, $\beta \leftarrow 0$.

Seat Planning with Stochastic Demand

Method Flow

We aim to obtain a seat planning with known demand scenarios before the demand realization.

- Build the formulation of scenario-based stochastic programming(SSP).
  - Consider the nested relation: a smaller group can take the larger seats.
- Reformulate SSP to the benders master problem(BMP) and subproblem.
- The optimal solution can be obtained by solving BMP iteratively.

Scenario-based Stochastic Programming(SSP)

Objective: maximize the expected number of people
$y_{i\omega}^{+}$: excess supply for i, ω. $y_{i\omega}^{-}$: shortage of supply for i, ω.
$d_{i\omega}$: demand of group type i for scenario ω

$$
\begin{aligned}
\max \quad & E_{\omega}\left[\sum_{i=1}^{M-1}(n_i-\delta)\left(\sum_{j=1}^{N} x_{ij}+y_{i+1,\omega}^{+}-y_{i\omega}^{+}\right)+(n_M-\delta)\left(\sum_{j=1}^{N} x_{Mj}-y_{M\omega}^{+}\right)\right] \\
\text{s.t.} \quad & \sum_{j=1}^{N} x_{ij}-y_{i\omega}^{+}+y_{i+1,\omega}^{+}+y_{i\omega}^{-}=d_{i\omega}, \quad i=1,\ldots,M-1, \omega\in\Omega \\
& \sum_{j=1}^{N} x_{ij}-y_{i\omega}^{+}+y_{i\omega}^{-}=d_{i\omega}, \quad i=M, \omega\in\Omega \\
& \sum_{i=1}^{M} n_i x_{ij}\leq L_j, j\in\mathcal{N} \\
& y_{i\omega}^{+}, y_{i\omega}^{-}\in\mathbb{Z}_{+}, \quad i\in\mathcal{M}, \omega\in\Omega \\
& x_{ij}\in\mathbb{Z}_{+}, \quad i\in\mathcal{M}, j\in\mathcal{N}.
\end{aligned}
\tag{2}
$$

Reformulation

Problem (2) is equivalent to the following master problem

$$\begin{aligned} \max \quad & \mathbf{c}^\intercal \mathbf{x} + z(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{n}\mathbf{x} \leq \mathbf{L} \\ & \mathbf{x} \in \mathbb{Z}_+^{M \times N}, \end{aligned} \tag{3}$$

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

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

and for each scenario $\omega \in \Omega$, we have the subproblem

$$\begin{aligned} z_\omega(\mathbf{x}) := \max \quad & \mathbf{f}^\intercal \mathbf{y} \\ \text{s.t.} \quad & \mathbf{x}\mathbf{1} + \mathbf{V}\mathbf{y} = \mathbf{d}_\omega \\ & \mathbf{y} \geq 0. \end{aligned} \tag{4}$$

Solution to Subproblem

Problem (4) is easy to solve with a given $\mathbf{x}$ from the perspective of the dual problem:

$$\begin{aligned} \min \quad & \alpha_\omega^\intercal(\mathbf{d}_\omega - \mathbf{x}\mathbf{1}) \\ \text{s.t.} \quad & \alpha_\omega^\intercal \mathbf{V} \geq \mathbf{f}^\intercal \end{aligned} \tag{5}$$

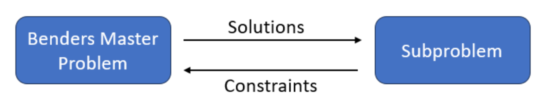
- The feasible region of problem (5), $P = \{\alpha | \alpha^\intercal V \geq \mathbf{f}^\intercal\}$, is bounded. In addition, all the extreme points of P are integral.
- The optimal solution to this problem can be obtained directly according to the complementary slackness property.

Benders Decomposition Procedure

Let z_ω be the lower bound of problem (5), SSP can be obtained by solving following restricted benders master problem:

$$\begin{aligned} \max \quad & \mathbf{c}^\intercal \mathbf{x} + \sum_{\omega \in \Omega} p_\omega z_\omega \\ \text{s.t.} \quad & \mathbf{n}\mathbf{x} \le \mathbf{L} \\ & (\alpha^k)^\intercal (\mathbf{d}_\omega - \mathbf{x}\mathbf{1}) \ge z_\omega, \alpha^k \in \mathcal{O}, \forall \omega \\ & \mathbf{x} \in \mathbb{Z}_+ \end{aligned} \tag{6}$$

Constraints will be generated from problem (5) until an optimal solution is found.



Issue of Optimality

- In most cases, the optimal solution can be obtained.
- For the extreme case where the optimal solution cannot be obtained, obtaining a feasible solution did not take too much time.
- It is more time-consuming to verify that it is the optimal solution. Thus, we set a time limit to obtain a feasible solution.
- There exists an optimal solution to SSP such that the patterns associated with this optimal solution are composed of the full or largest patterns under any given scenarios.

Running time of Benders Decomposition and IP

Parameters:
(150, 350): the demand of each group type is randomly sampled from (150, 350).
(21, 50): the number of seats of each row is randomly sampled from (21, 50).

# of scenarios	Demands	# of rows	# of groups	# of seats	Running time of IP(s)	Benders (s)
1000	(20, 30)	10	4	(21, 30)	1.6	0.1
1000	(20, 30)	10	4	(21, 40)	1.6	0.1
1000	(150, 350)	30	8	(21, 50)	4.1	0.23
5000					28.73	0.47
10000					66.81	0.91
50000					925.17	4.3
1000	(1000, 2000)	200	8	(21, 50)	5.88	0.29
5000					30.0	0.62
10000					64.41	1.09
50000					365.57	4.56
1000	(150, 250)	30	16	(41, 60)	17.15	0.18
5000					105.2	0.67
10000					260.88	1.28
50000					3873.16	6.18

Seat Assignment with Dynamic Demand

Seat Assignment under Fixed Seat Planning

In certain situations, it is necessary to determine the seat planning prior to the actual demand realization. When each group arrives, we make decisions regarding whether to accept or reject them based on the predetermined seat planning.

- There is one and only one group arrival at each period, $t = 1, \ldots, T+1$.
- The probability of an arrival of group type i: p_i.

After acceptance, customers have more freedom to choose their seats with the seat planning.

Group-type Control

- Obtain the seat planning from stochastic programming. Suppose the corresponding supply is $[X_1, \ldots, X_M]$.
  - When the supply for the current group is sufficient, accept it directly.
  - When the supply is insufficient, decide which group type to be assigned to.

Let D_j^t be the random variable indicates the number of group type j in t periods.
$P(D_i^{T-t} \geq x_i)$ is the probability that the demand of group type i in $(T-t)$ periods is no less than x_i.
$d^t(i,j) = \underbrace{i + (j-i-\delta) P(D_{j-i-\delta} \geq x_{j-i-\delta} + 1; T-t)}_{\text{acceptance}}$ - $\underbrace{jP(D_j \geq x_j; T-t)}_{\text{rejection}}$.
For all $j > i$, find the maximum value denoted as $d^t(i, j^*)$.
If $d^t(i, j^*) > 0$, we will place the group of i in $(j^* + \delta)$-size seats. Otherwise, reject the group.

Performance

# of samples	T	probabilities	# of rows	IP
1000	45	[0.4,0.4,0.1,0.1]	8	85.3
	50			97.32
	55			102.40
	60			106.02
	65			108.84
1000	35	[0.25,0.25,0.25,0.25]	8	87.08
	40			101.24
	45			110.52
	50			114.39
	55			117.26
5000	300	[0.25,0.25,0.25,0.25]	30	749.76
	350			866.42
	400			889.44
	450			916.66

Each entry of people served is the average of 50 instances.

Assign seats to each group after its realization

Dynamic Seat Assignment Problem

- There is one and only one group arrival at each period, $t = 1, \ldots, T+1$.
- The probability of an arrival of group type i: p_i.
- $\mathbf{L} = (l_1, l_2, \ldots, l_N)$, where $l_j = 0, \ldots, L_j, j \in \mathcal{N}$: Remaining capacity.
- $u_{i,j}^t$: Decision. Assign group type i to row j at period t, $u_{i,j}^t = 1$.
- $U^t(\mathbf{L}) = \{u_{i,j}^t \in \{0,1\}, \forall i, j | \sum_{j=1}^N u_{i,j}^t \leq 1, \forall i, n_i u_{i,j}^t \mathbf{e}_j \leq \mathbf{L}, \forall i, j\}$.
- $\mathbf{e}_j$: Unit column vector with j-th element being 1.
- $V^t(\mathbf{L})$: Value function at period t, given remaining capacity, $\mathbf{L}$.

$$V^t(\mathbf{L}) = \max_{u_{i,j}^t \in U^t(\mathbf{L})} \left\{ \sum_{i=1}^M p_i \left(\sum_{j=1}^N i u_{i,j}^t + V^{t+1}\left(\mathbf{L} - \sum_{j=1}^N n_i u_{i,j}^t \mathbf{e}_j\right)\right) + p_0 V^{t+1}(\mathbf{L}) \right\}$$

Method 1

- Obtain the seat planning after the realization of group.

$$
\begin{aligned}
\max \quad & \sum_j k a_j + E_\omega \left[\sum_{i=1}^{M-1} (n_i - \delta)(\sum_{j=1}^{N} x_{ij} + y_{i+1,\omega}^{+} - y_{i\omega}^{+}) + (n_M - \delta)(\sum_{j=1}^{N} x_{Mj} - y_{M\omega}^{+}) \right] \\
\text{s.t.} \quad & \sum_{j=1}^{N} x_{ij} - y_{i\omega}^{+} + y_{i+1,\omega}^{+} + y_{i\omega}^{-} = d_{i\omega}, \quad i = 1, \ldots, M-1, \omega \in \Omega \\
& \sum_{j=1}^{N} x_{ij} - y_{i\omega}^{+} + y_{i\omega}^{-} = d_{i\omega}, \quad i = M, \omega \in \Omega \\
& \sum_{i=1}^{M} n_i x_{ij} \leq L_j - n_k a_j, j \in \mathcal{N} \\
& \sum_j a_j \leq 1 \\
& x_{ij} \in \mathbb{Z}_{+}, \quad i \in \mathcal{M}, j \in \mathcal{N}, y_{i\omega}^{+}, y_{i\omega}^{-} \in \mathbb{Z}_{+}, \quad i \in \mathcal{M}, \omega \in \Omega, a_j \in \{0,1\}, j \in \mathcal{N}.
\end{aligned} \tag{7}
$$

- Suppose the supply is $[X_1, \ldots, X_M]$.
  - For the arriving group type i, if $X_i > 0$, accept it.
  - If $X_i = 0$, regenerate a seat planning.

Method 2 Overview

- Obtain the seat planning composed of full or largest patterns.
  - Linear seat planning from relaxed stochastic programming
  - Integral seat planning from deterministic model
  - Construct largest or full patterns.
- Dynamic seat assignment
  - Determine the group type by group-type control
  - Decision on assigning the group to a specific row

Obtain Seat Planning Composed of Full or Largest Patterns

Step 1. Obtain the solution, $\mathbf{x}^*$, 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 (1) to obtain the optimal solution, $\mathbf{x}^1$.

Step 3. Construct the full or largest patterns with $\mathbf{x}^1$.

Policies

We have the following policies when we make the instant allocation.

- Dynamic seat assignment(Method 2)
- Bid-price control
- Dynamic programming based heuristic
- Booking limit control
- First come first served

Dynamic Seat Assignment(DSA)

1. Determine the group type by the group-type control.
2. Make the decision on assigning the group to a specific row.
   - Break tie for determining a specific row
   - Decision on assigning the group
   - Value of Acceptance(VoA): approximation of $V_t(\mathbf{L} - n_i \mathbf{e}_j) + i$. (Find a pattern containing group type j^*)
   - Value of Rejection(VoR): approximation of $V_t(\mathbf{L})$.
   - If VoA is no less than VoR, accept group type i, otherwise, reject it.

Regenerate the seat planning

- When $X_M = 0$
- When comparing VoA and VoR

Bid-price Control

The dual problem of LP relaxation of problem (1) 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 (1) 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: $\{\max \sum_{i=1}^{M} (n_i - \delta) x_i : x_i \leq d_i, i \in \mathcal{M}, \sum_{i=1}^{M} n_i x_i \leq L, x_i \in \mathbb{Z}_+\}$.
- Decision: u. If we accept a request in period t, $u^t = 1$; otherwise, $u^t = 0$.
  - DP with one row can be expressed as:

$$V^t(l) = \max_{u^t \in \{0,1\}} \left\{ \sum_i p_i [V^{t+1}(l - n_i u^t) + i u^t] + p_0 V^{t+1}(l) \right\}, l \geq 0$$

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

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

Booking limit Control

Basic idea: for every type of requests, we only allocate a fixed amount according to the static solution and reject all other exceeding requests.

1. Observe the arrival group type i.
2. Solve problem (1) using the expected demand.
3. Obtain the optimal solution, x^*_{ij} and the aggregate optimal solution, $\mathbf{X}$.
4. If $X_i > 0$, accept the arrival and assign the group to row k where $x_{ik} > 0$, update $\mathbf{L}^{t+1} = \mathbf{L}^t - n_i \mathbf{e}_k$; otherwise, reject it, let $\mathbf{L}^{t+1} = \mathbf{L}^t$.

Numerical Results

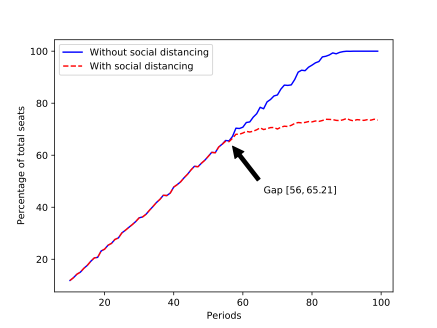
Performances of Different Policies

T	Probabilities	DSA(%)	DP1(%)	Bid(%)	Booking(%)	FCFS(%)
60	[0.25, 0.25, 0.25, 0.25]	99.12	98.42	98.38	96.74	98.17
70		98.34	96.87	96.24	97.18	94.75
80		98.61	95.69	96.02	98.00	93.18
90		99.10	96.05	96.41	98.31	92.48
100		99.58	95.09	96.88	98.70	92.54
60	[0.25, 0.35, 0.05, 0.35]	98.94	98.26	98.25	96.74	98.62
70		98.05	96.62	96.06	96.90	93.96
80		98.37	96.01	95.89	97.75	92.88
90		99.01	96.77	96.62	98.42	92.46
100		99.23	97.04	97.14	98.67	92.00
60	[0.15, 0.25, 0.55, 0.05]	99.14	98.72	98.74	96.61	98.07
70		99.30	96.38	96.90	97.88	96.25
80		99.59	97.75	97.87	98.55	95.81
90		99.53	98.45	98.69	98.81	95.50
100		99.47	98.62	98.94	98.90	95.25

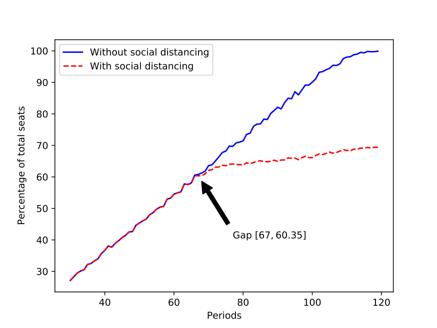
DSA has better performance than other policies under different demands.

Impact of Social Distancing as Demand Increases

$\gamma = p_1 * 1 + p_2 * 2 + p_3 * 3 + p_4 * 4$: the expected 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.

Estimation of Gap Point

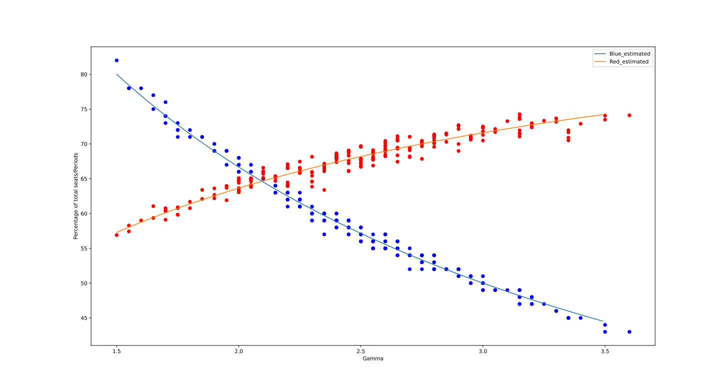


Figure: Gap points under 200 probabilities

Blue points: period of the gap point. Red points: occupancy rate of the gap point. Gap points can be estimated.

Make A Later Allocation

This setting is particularly applicable to larger venues, such as stadiums, where an immediate decision is made when a group arrives, but the actual allocation of seats for that group is deferred to a later time.

Policies:

- Dynamic programming based heuristic
- Bid-price control

Conclusion

Conclusion

- We address the problem of dynamic seat assignment with social distancing.
- Our approach, stochastic planning policy, provides a comprehensive solution for optimizing seat assignments while ensuring the safety of customers under dynamic situation.
- We can estimate the occupancy rate when applying SPP according to γ.

The End