Contents

1 Question 1

1.1 Q1.a Revenue based on FCFS

The study discusses the topic of a single-resource and two-fare class, specifically a lower bound for expected revenue or a First-Come First-Serve (FCFS). In this instance, a café that operates during the breakfast and lunch times, does not reserve any croissants for sandwiches at lunch. In order to conduct this FCFS approach, we formulate the parameters and make the subsequent assumptions as below:

- Expected demands: The distributions of demands for breakfast and lunch are assumed to follow the Poisson distribution. The mean demand for croissants at breakfast is $mL = 50$; the mean demand for croissants at lunch is $mH = 20$.
- Prices: Low-fare price is the price of a single croissant sold at breakfast $pL = 1$; high-fare price is the price of a single sandwich croissant sold at lunch $pH = 1.5$.
- Capacity: This is the daily number of units (croissants) the café receives $capacity = 50$.

The FCFS formulation is defined as follows:

1. Croissants protected for lunch set to 0.
2. When a certain value of demand for breakfast croissants is received, compare the available croissants and demands. If availability is larger than demand, sell demanded croissants; otherwise sell all the remaining croissants.
3. Any leftover croissants that are not sold during the breakfast rush can be sold to customers at lunch under the same logic of step 2.
4. Repeat through the loop for all possible demands from 0 to 100 croissants for breakfast and lunch each. Calculate the expected revenue according to Poisson distribution and associated prices.

The expected revenue based on FCFS is 51.39, which is the minimum expected revenue that the café can receive due to the lack of revenue management.

1.2 Q1.b Optimal protection for lunch

In this instance, the same single-resource and two-fare-class problem is addressed except that the protection level for high-fare croissants is involved. This means that the café would reserve some croissants for lunch time in order to maximise returns generated. The problem uses the same parameters as discussed in part (a). Unlike FCFS, a variety of protection levels are considered to understand the subsequent expected revenue. A vector of revenues for each protection level is initialised to zero and will be then updated in each iteration. By iterating protection level from 0 to 50 (the capacity of croissants) and updating the available croissants for breakfast and lunch, the maximum expected revenue and associated protection level can be obtained. The optimal protection level for lunch is 18. It indicates that the café would achieve the optimal revenue if 18 croissants are saved for lunch.

1.3 Q1.c Optimal revenue

As mentioned previously, the FCFS policy allows the café to sell croissants without reserving any for the lunch time rush, meaning there is no protection level and therefore generating an expected revenue of 51.39. After implementing the reserve policy in part (b) with a protection level of 18, the optimal expected daily revenue is improved to 57.61. Comparing the expected revenue outcomes in (a) and (b), it is evident that the introduction of the protection level brings a 12% improvement in the revenue. This is due to the fact that setting protection level allows more sales revenue from high-fare croissants. From Figure 1, we can see that initial additional reserved croissants for lunch results in minor increases in revenue to the café. Once 18 croissants are reserved, it becomes disadvantageous to reserve more than that, as revenue starts to follow an approximately linear decrease. This means the café would benefit the most from reserving 18 croissants for lunch time and selling up to 32 croissants at breakfast.

1.4 Q1.d Change in allocation parameters

Considering the allocation of croissants, it is important to understand the setting of protection level may vary due to changes in parameters such as croissants the café receives. We have

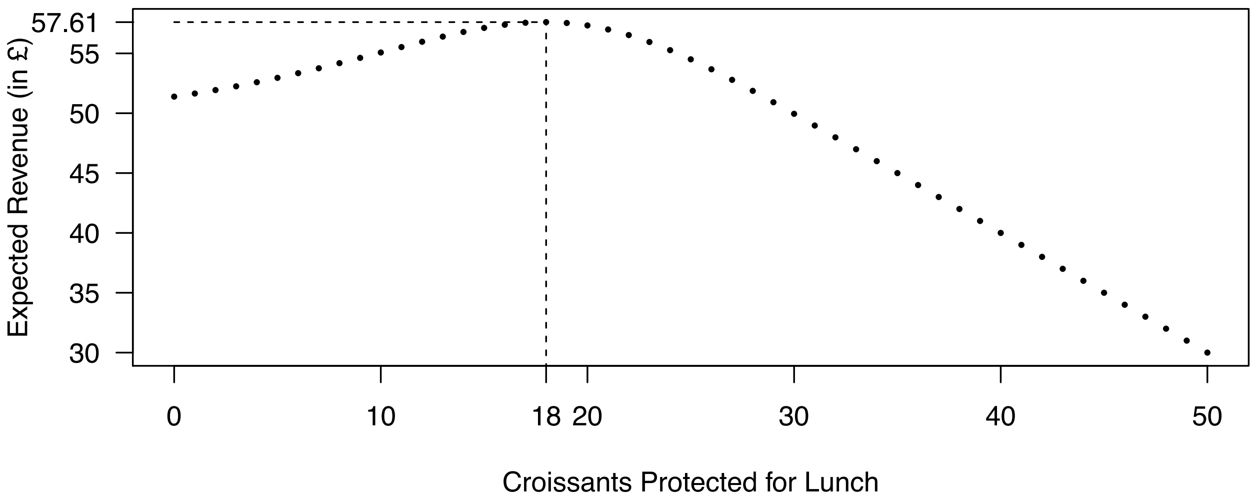

Figure 1: Change of the expected revenue along with each protection level for lunch.

conducted an analysis by changing parameters of the expected demand for sandwiches, the revenue generated by each croissant during breakfast or lunch and the capacity of croissants to explain the influence of those parameters on the revenue as well as the protection level, aiming to extract insights that could benefit the café.

To verify the impact of each variable (expected demand for sandwiches, the prices of an individual and sandwich croissants and daily capacity) on allocation decisions, the control variate method is applied. In this case, values of a variable are altered while the other variables are kept constant. The results from testing on all these four variables are listed below.

- Expected demand for sandwiches:

The value of expected demand for sandwiches is increased by 25% each time until it reaches 60. Considering that daily capacity of croissants (50) in the initial setting could be a key threshold for analysis, the range of this variable is set between 20 and 60. From Table 1, it is observed that the optimal protection level and optimal expected revenue increase proportionally with the expected demand, as long as the demand does not exceed the daily capacity. This result is reasonable as keeping more croissants to accommodate higher demand for sandwiches brings more revenue for the café. It is also observed that once the expected demand for croissant exceeds the capacity, optimal protection level remains the same as capacity, and the optimal expected revenue then increases at a much lower rate compared to the situation that demand is lower than capacity.

Table 1: Impact of demand variable on optimal protection level and expected revenue.

	optimal protection level	optimal expected revenue
demand*1.00	18	57.61
demand*1.25	23	59.82
demand*1.5	28	62.05
demand*1.75	32	64.32
demand*2.00	37	66.60
demand*2.25	42	68.89
demand*2.5	47	71.19
demand*2.75	50	73.40
demand*3.00	50	74.52

- Price of breakfast croissant (p_C) and price of sandwich croissant(p_S):

The prices of both breakfast and sandwich croissants are set in a range between £1-£1.5. Assuming that p_S is always higher than p_C, p_C is increased by £0.1 each time from £1 until £1.5, while p_S is decreased from £1.5 by £0.1 each time until £1.0. Based on Littlewood's rule, a new metric called price ratio is introduced and calculated in Table 2 and Table 3 to explain the impact of price changes. The ratio is calculated by dividing low fare price p_C with high fare price p_S in this instance. It is observed that the optimal protection level decreased with the price ratio. For the optimal expected revenue, the impact is more complicated as it increases with the price ratio if the increase of price ratio is due to the increase in p_C but decreased with the price ratio if the increase of price ratio is due to the decrease in p_S. According to Littlewood's rule $F_f(y) = 1 - p_C/p_S$ which implies that the probability that high-fare demand will exceed the protection level should equal the price ratio p_C/p_S to maximize expected revenue. Thus, a lower price ratio is an implication of high protection level for lunch, verifying the conclusion drawn from the results presented in the two tables.

- Daily capacity:

Croissants received daily are set to vary in a range between 15 and 50. The purpose is to measure the impact of daily capacity on allocation decisions when its value situates below or

Table 2: Impact of pC and price ratio on optimal protection level and expected revenue.

	optimal protection level	optimal expected revenue	price ratio
pC=1.0	18	57.61	0.67
pC=1.1	17	60.86	0.73
pC=1.2	16	64.19	0.80
pC=1.3	15	67.62	0.87
pC=1.4	14	71.17	0.93
pC=1.5	0	74.97	1.00

Table 3: Impact of pS and price ratio on optimal protection level and expected revenue.

	optimal protection level	optimal expected revenue	price ratio
pS=1.5	18	57.61	0.67
pS=1.4	17	55.92	0.71
pS=1.3	17	54.28	0.77
pS=1.2	16	52.70	0.83
pS=1.1	14	51.23	0.91
pS=1.0	0	49.99	1.00

between the demand for breakfast (50) and the demand for sandwiches (20). From Table 4, it is observed that the optimal protection level remains the same as long as the capacity is larger than the demand of sandwich croissants (20). Once the capacity decreases to a level below 20, the optimal protection level is equal to the capacity. This result can be explained by the fact that keeping more sandwich croissants than demand is not rational as it won't generate more revenue for the café. The optimal expected revenue still increases with the capacity due to the increased revenue from breakfast croissants.

Overall, when capacity is higher than the expected demand of sandwich croissants, it is ideal for the café to reserve more croissants for lunch time if the price of sandwich croissants or expected demand of sandwich croissants increases. However, it is ideal for the café to reserve more croissants for breakfast time if the price of breakfast croissants increases. When capacity is lower than the expected demand of sandwich croissants, the optimal protection level always equals the capacity as it cannot sell more than it receives.

Table 4: Impact of daily capacity on optimal protection level and expected revenue.

	optimal protection level	optimal expected revenue
capacity=15	15	22.12
capacity=20	18	27.61
capacity=25	18	32.61
capacity=30	18	37.61
capacity=35	18	42.61
capacity=40	18	47.61
capacity=45	18	52.61
capacity=50	18	57.61

2 Question 2

2.1 Q2.a Optimal expected total revenue

The problem in question 2 is classified as a two-resource and multi-fare-class with mixed arrivals problem. To implement the Dynamic Programming algorithm, we assign the parameters and make the assumptions as follows:

- Two resources: Resource 1 is seats available on Day 1 with initial capacity $N_1 = 100$; Resource 2 is seats available on Day 2 with initial capacity $N_2 = 100$.
- Products: Product x_1 is high fare for Day 1, product x_2 is low fare for Day 1, product x_3 is high fare for Day 2, product x_4 is low fare for Day 2, product x_5 is high fare for a combined Day 1 and Day 2 ticket, product x_6 is low fare for a combined Day 1 and Day 2 ticket.
- Prices of products: Product x_k has price p_k given as (150,100,120,80,250,150) and the request for x_k arrives with probability λ_k given as (1/10,1/5,1/10,1/5,1/10,1/5) ($k \in [1, 6]$). λ_0 is the probability that no request arrives and $\lambda_0 + \lambda_1 + ... + \lambda_k = 1$.
- Time periods: TT$= 300$ as the reservation horizon is divided into 300 smaller time periods. It is assumed that at most one request arrives in each period.
- Value function: Value function at remaining capacity i, j and at period t is represented as $V[i, j, t-1]$.

The DP formulation is defined as follows:

Initialisation: Terminal values $V[i, j, 1] = 0$ as unsold seats are worthless at the end of time.

DP recursion:

1. Check the remaining capacity of each resource.
2. When a booking request for product x_k is received, verify whether the remaining capacity of correspondent resource is sufficient.
3. If the remaining capacity of correspondent resource is not sufficient, reject the request.
4. If the remaining capacity of correspondent resource is sufficient, compare the value function of accepting the request and the value function of rejecting it and leaving the seat to the next time period.
5. If the sum of immediate revenue in this period and the optimal revenue for all the remaining periods by accepting one single unit of product x_k exceeds the optimal revenue of leaving the seat to next period, accept the request otherwise reject it.
6. Repeat through the loop to the last time period. Calculate the total expected revenue.

The optimal expected total revenue equals 22,679.83.

2.2 Q2.b Expected revenue on FCFS

To implement first-come first-served rule in the DP recursion constructed in part (a), the DP recursion is changed as the algorithm accepts any requests as long as there is available capacity. The expected total revenue from FCFS policy is calculated as 19,719.77.

By comparing the expected total revenue from (a) and (b), we find that the optimal expected total revenue is 15% larger than the expected total revenue based on FCFS policy. The FCFS policy achieves the lowest revenue among all the other methods for revenue optimization. Due to its major drawback of accepting too many low-value requests, FCFS is the only method whose revenue decreases when the number of seats increases, that is, when flexibility increases. In contrast, DP never gets hurt by additional flexibility and revenues increase with the increasing degree of flexibility as it guarantees that the new request of product x_k is accepted only in case that the additional revenue by accepting the request of one single unit of product x_k exceeds the revenue of leaving the seat to the next period. In this case, following FCFS policy, the algorithm accepts more requests of products with low fares, such as x_2, x_4 and x_6, which does not achieve the expected total revenue as the optimal policy.

2.3 Q2.c LP bid prices heuristics

To apply a heuristic policy using the bid prices to the Dynamic Programming recursion, a linear programming is conducted to obtain the bid prices when demands are assumed to be deterministic. In this case, the linear programming is defined as an approach to achieve optimisation where a linear objective function is to be maximised subject to linear constraints. Specifically, the linear objective function (1) and linear constraints are indicated as below:

$$Maximise \quad 150x_1 + 100x_2 + 120x_3 + 80x_4 + 250x_5 + 150x_6 \tag{1}$$

subject to

$$x_1 + x_2 + x_5 + x_6 \leq 100$$

$$x_3 + x_4 + x_5 + x_6 \leq 100$$

$$x_1 \leq 30, x_2 \leq 60, x_3 \leq 30, x_4 \leq 60, x_5 \leq 30, x_6 \leq 60$$

$$x_k \geq 0 \quad for\ k = 1, 2, ..., 6$$

The linear programming is solved by using `lp` function of R package `lpSolve` and generates marginal values for capacity, which can serve as an estimate of network bid prices. Hence, the bid prices obtained for Day 1 and Day 2 seat are 100 and 80 respectively, which indicate that we can realise 100 in additional revenue from adding another seat to Day 1 and similarly, additional revenue of 80 for Day 2. If a booking request for a one-day seat arrives, we only accept the request when the fare exceeds or equals the bid price for that seat. Equivalently, if a request involving both Day 1 and Day 2 seats arrives, we only accept the request when the fare exceeds or equals the sum of the bid prices, which is equal to 180 in this case. Overall, the bid price indicates the minimum price we should accept for a request on a seat. Given this idea behind the bid prices, the heuristic policy using bid prices is set as follows:

- At each arrival of high fare for Day 1, we accept the request as the fare 150 is larger than the bid price 100;
- At each arrival of low fare for Day 1, we accept the request as the fare 100 is equal to the bid price 100;
- At each arrival of high fare for Day 2, we accept the request as the fare 120 is larger than the bid price 80;
- At each arrival of low fare for Day 2, we accept the request as the fare 80 is equal to the bid price 80;
- At each arrival of high fare for a combined of Day 1 and 2, we accept the request as the fare 250 is larger than the bid price 180;
- At each arrival of low fare for a combined of Day 1 and 2, we reject the request as the fare 150 is lower than the bid price 180.

It might be surprising that we reject the low fare for a combined ticket as the price is the second highest among the products. However, given the deterministic demands, the contribution to the total revenue will be greater by rejecting the discounted 2-day tickets and probably passing on to customers who are in favor of 1-day tickets. By implementing the heuristic policy to the DP recursion, the total of expected revenue is 21,915.60, which is much higher (+11.1%) than the revenue based on FCFS and slightly lower (-3.4%) than the optimal revenue resulted from dynamically accepting or rejecting requests. In conclusion, bid prices can provide useful and intuitive insights into seats allocation, and the heuristic policy based on the bid prices can achieve results close to the optimal total revenue. Nevertheless, this heuristic rule also raises a problem of ignoring uncertainty as it assumes deterministic anticipated demands as the LP formulation.

2.4 Q2.d Design our own heuristic solution

Our heuristic policy is designed as that we accept all the high fare class requests for Day 1 ticket, Day 2 ticket or a combined 2-day ticket (that is, the product x_1, x_3 and $x5$), and dynamically reject or accept the remaining products, which are associated with low fare classes. The logic of the algorithm implemented in DP recursion is described as (initialisation is the same as part (a)):

1. When a booking request for product x_k is received, verify whether the remaining capacity of correspondent resource is sufficient. If no remaining capacity, reject the request; otherwise go to step 2.

2. If receiving the request x_k with $k \in 1, 3, 5$, go to step 3; otherwise go to step 4.

3. Accept the request and assign the sum of immediate revenue and the optimal revenue for all the remaining periods to the associated value function.

4. If the sum of immediate revenue and the optimal revenue for all the remaining periods by accepting one single unit of product x_k ($k \in 2, 4, 6$) exceeds the optimal revenue of leaving the seat to next period, accept the request otherwise reject it.

5. Repeat through the loop to the last time period. Calculate the total expected revenue.

The expected total revenue generated from this heuristic policy is 22,677.20. The objective of our design of this heuristic policy is to account for the demand constraints of low fare tickets, which are product 2, 4 and 6 specifically. Thus, it can be seen that this heuristic rule results in a total expected revenue higher than the revenue from heuristic using bid prices, which do not account for uncertainty and is less flexible. Additionally, it is observed that this revenue is quite close to the optimal revenue obtained in question (a). More discussions can be found in the next subsection.

2.5 Q2.e Comparison of the expected revenues

Compared with the optimal expected total revenue obtained by dynamically rejecting or accepting requests in question (a), the revenue resulted from implementing self-designed heuristic policy is close to the optimal revenue with only $22,679.83 - 22,677.20 = 2.63$ of difference and 0.012% decrease from the optimal revenue. As discussed above, the DP algorithm achieves optimisation by taking into consideration the complexity of mixed-arrival requests for different products and relaxing the demand constraints. The reasons that our designed heuristic policy accepts high fares as long as the request arrives are based on two aspects. The bid prices obtained from LP solutions confirm that the price of each high fare product is higher than the associated bid price, which suggests the minimum price that we should accept a product. Additionally, the step to decide whether accept or reject a request by comparing the expected value for the product being sold and for leaving the seat for next iteration gives an intuition that selling the seat to that high fare ticket would generate more revenue than rejecting it in this case. Thus, it is not surprising that the difference between our heuristic revenue and the optimal revenue is extremely small.

Overall, Figure 2 shows the comparison of the total expected revenues calculated in the question 2, which are resulted from FCFS rule, the bid-price based heuristic policy, our own heuristic and the DP algorithm, as well as the LP solution. The FCFS rule and the LP solution provide theoretically the lower and upper bound respectively for the total revenue that can be obtained in this study. When looking at the potential revenue from FCFS to LP solution, we find that the heuristic policy using bid price has achieved approximately 71% of the potential revenue, while our heuristic policy and the DP recursion have both reached approximately 96% of the potential revenue. It can be concluded that revenue management can achieve significant increase in expected revenue and appropriately designed heuristic policy is useful for not only saving computation cost but also reaching a good but not necessarily optimal revenue.

3 Question 3

3.1 Q3.a Introduction

The car rental industry involves agencies or companies that loan their numerous portfolio of vehicles to travelers for business or leisure. Before the impact of Covid-19, the global travel sector was booming, with all aspects of the industry generating large sums of revenue, for example, revenue of traveling and tourism industry in countries like the United Kingdom amounting to £145.9 billion (TourismAlliance, 2019). Additionally, it was estimated that UK customers accumulated a large number of car rental days and this was expected to grow.

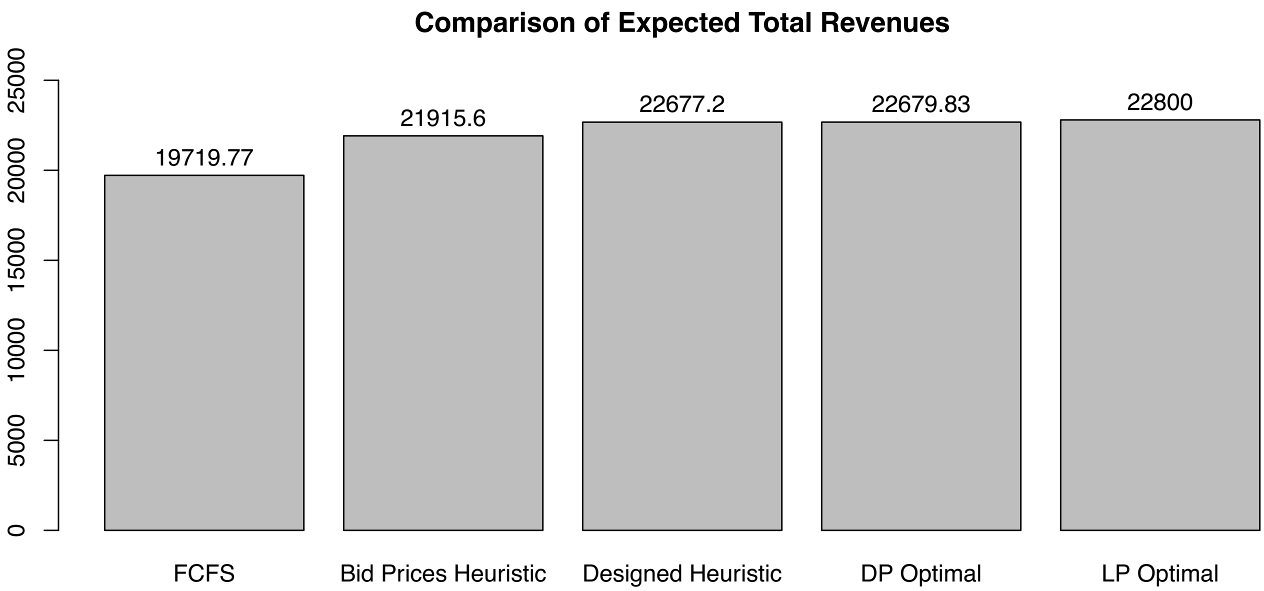

Figure 2: Comparison of the expected total revenues obtained from FCFS, the heuristic policy using bid prices, our own heuristic policy, the DP algorithm and the LP solution.

Therefore, this study aims to simulate a scenario of a small car rental company in the UK that offers rentals over 3 days with a capacity of 45 vehicles on each day and 12 products in total: a high and low fare for each day and also for combinations of the first two days, the last two days and all the three days. The 3-day offering is over a weekend period with the inclusion of a bank holiday on 25th May 2020 which influences the rates of the vehicles to rent. Based on the 3-day holiday setting, the rental price on a single day is set to decrease along with the holidays and as a result the company offers the lowest price on the last day. Currently, the company adopts a FCFS revenue policy which does not prove to be the most efficient compared with other competitors' policies. Furthermore, this study is carried out without the variable of Covid-19.

3.2 Q3.b Methodologies and assumptions

The problem is classified as a three-resource and multiple fare classes setting. The following parameters are set to simulate the scenario of the small car rental company based on current players and market trends (see references):

- Three resources: Resources 1, 2 and 3 refer to the number of cars available on Days 1, 2 and 3 respectively with capacity set as $N_1 = 45$, $N_2 = 45$ and $N_3 = 45$.

- Twelve products: Product x_1 and x_2 are Day 1 high and low fares, product x_3 and x_4 are Day 2 high and low fares, product x_5 and x_6 are Day 3 high and low fares, product x_7 and x_8 are high and low fares for the combined first two-day rental, product x_9 and x_{10} are high and low fares for the combined last two-day rental, product x_{11} and x_{12} are high and low fares for the combined 3-day rental.
- Prices of products: The product x_k ($k \in [1, 12]$) has price p_k presented as (49, 45, 44, 40, 42, 38, 90, 85, 83, 78, 130, 120) in £.
- Time periods: $TT = 100$ as the reservation horizon is divided into 100 smaller time periods. The assumption is that at most one rental request arrives in each period. In each time period, the request for x_k rentals arrives with probability 0.04 and 0.12 for high and low fares respectively across all products and thus the probability of no request arrives is 0.04.

In this study, the methodologies carried out use bid pricing, linear programming and dynamic programming as well as the results discussed in order to arrive at a conclusive way to improve revenue for the car rental company.

Initially, to improve the current FCFS policy set in place, we attempted a bid-price heuristic approach that compares the fare of a booking request to the hurdle rate and accepts the fare of the product if it exceeds that of the bid price. To apply the linear programming, the demands for each product are assumed to be 10 cars for all the high-fare products and 30 cars for low-fare. The bid prices obtained from LP are £45, £40 and £38 for car rental on each day respectively. Thus we propose a heuristic policy that the car rental company can accept all the requests except that for product 12, which means rejecting all the requests for low-fare 3 days rental, as long as the request arrives based on availability of cars.

However, when this policy was implemented with the given setting, the outcome resulted in a revenue amount even lower than that of the FCFS policy. We assume that this may be due to the exclusion of the combined car rental product for Day 1 and Day 3, which corresponds to the intuition that customers going to a car rental company would not rent for the first day (Saturday) and then the third day (Monday), particularly during a bank holiday. This main feature of car rental business might result in that the bid-price based heuristic policy fails to improve the revenue and is unreasonable to be implemented in this case.

Additionally, we attempted to dynamically calculate bid prices obtained from linear programming by factoring in the availability of cars, and attempted to decide whether accept or reject a request under the guidance of those dynamic bid prices. Given the three-resource setting, however, this would be computationally unfeasible (Phillips, 2005) for the small car rental company, as the linear programming would be rerun and the bid prices would be recalculated within each iteration of available vehicles and time periods.

To take into consideration the main feature of car rental operating and to combat the issues presented by incorporating bid pricing, we decided to adopt a Dynamic Programming approach to help maximise the expected revenue that this small car rental company desires to achieve during this 3-day holiday. While these types of methods assume deterministic demands, Dynamic Programming does not assume any form of customer arrivals but instead forecasts by time frame and class to assign vehicles. Additionally, Dynamic Programming is able to produce optimal solutions with a faster computational speed compared to exhaustive search.

To implement the DP recursion, a 4-dimensional array, with the length of each dimension equal to 31, 31, 31 and 101 according to the capacities and time horizons, is initialised for value function and would then be updated in each iteration. Similarly to previous practices of this problem set, the comparison of value function between accepting the car rental request for a specific product and leaving the capacity to next time period is conducted for each recursion. Last but not least, the different terminal values are being tested to consider the possibility of renting vehicles at a more discounted price when there is no time period left.

3.3 Q3.c Analysis of the results

According to the analysis results, the expected total revenue for the car rental company are compared in Table 5. It is indicated that the company can only achieve £5041.34 of total revenue using their current FCFS rule, but it can obtain £5475.96 of total revenue when incorporating DP in their revenue management. By calculating the theoretical upper bound of expected total revenue using LP, under the assumption that demands are deterministic, implementing DP can achieve 55% of the gap between FCFS and upper bound.

Table 5: Comparison of the expected total revenues achieved by different revenue management methods.

	FCFS	DP algorithm	LP (upper bound)
Total revenue (£)	5041.34	5475.96	5825

Adopting DP as a solution to improve the expected total revenue for this car rental company, we decide to explore whether there is more space to increase the optimal revenue. It is noticeable that terminal values at the end of time horizon were set to 0 under our initial assumption. However, this assumption may be relaxed due to the flexibility of policies that the car rental company can adopt such as giving customers the option to book discounted vehicles "at last minute". In this situation, car rental booking before the last time period could still be valuable because customers might have specific demand at that time period. Thus, this assumption is relaxed by assuming different discounted prices for remaining vehicles on each day at the last time period, ranging from 0 to 35 with an interval of 5. The results are shown in Figure 3.

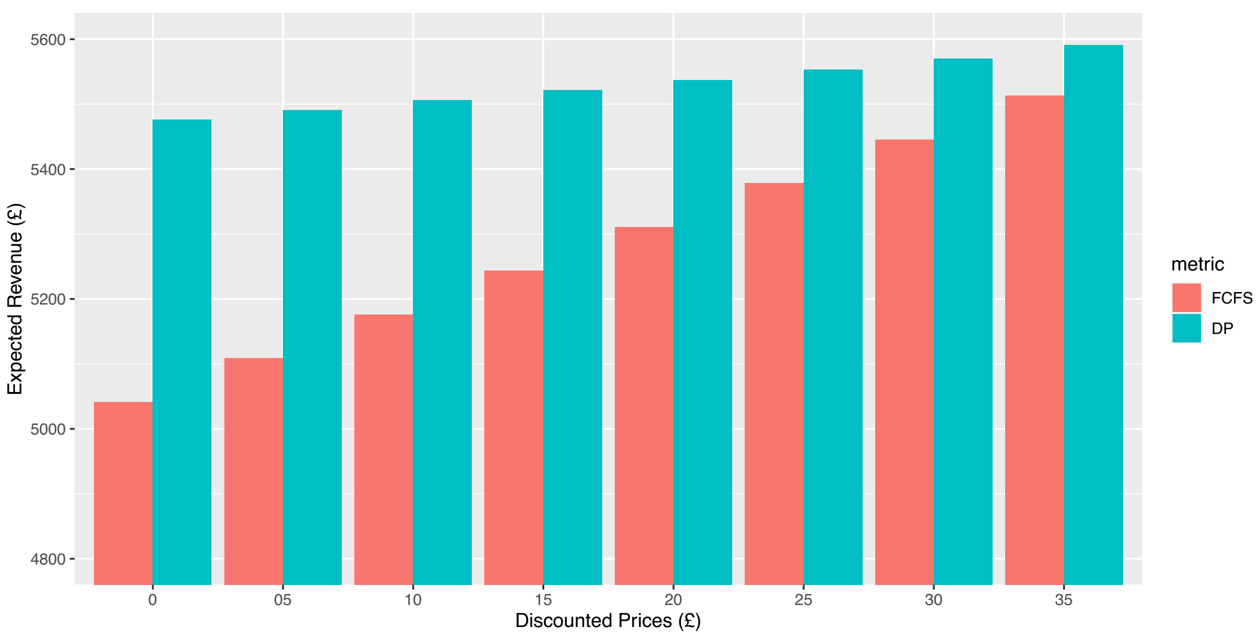

Figure 3: Comparison of revenue under FCFS and DP for different discounted price at the end of horizon

There are two insights observed from the figure. Firstly, the optimal expected revenue using either DP or FCFS method increases proportionally when the discounted price for remaining vehicles increases, following an approximately linear relationship. Secondly, varying discounted price has a large impact on FCFS revenue as FCFS revenue increases more rapidly along with the discounted prices increasing, compared to DP optimal revenue. This confirms that adopting the DP method as a revenue management solution can achieve the optimal expected revenue. Based on these two insights, we conclude that it will be beneficial for the car rental company with the goal of maximising the revenue to adopt DP for revenue management and give the remaining vehicles a discounted price of £35 at the last time period.

In conclusion, based on the results and analysis above, we strongly recommend the car rental company to incorporate DP in their revenue management as the company would achieve 55% of the gap between their current revenue using FCFS policy (lower bound) and the theoretical optimal revenue under LP (upper bound). Additionally, if the company wants to further improve their optimal revenue, sales measures should be taken to monetize the remaining vehicles on each holiday. By charging customers who show specific demand for discounted rental prices at the last time period, the company is able to reduce loss of unused capacities but it should be noticed that it contributes little to the total expected revenue under DP method.

4 References

Businesswire, (2019), “Key market trends and issues in the car rental industry, 2019 Report-ResearchAndMarkets.com”, [online], Available at: https://www.businesswire.com/news/home/20191119005898/en/Key-Market-Trends-Issues-in-the-Car-Rental-Industry-2019-Report---ResearchAndMarkets.com (Accessed: 19 February 2021)

EnterpriseRent-A-Car, (2021), “Car hire from Enterprise”, [online], Available at: https://www.enterprise.co.uk/en/car-hire.html?icid=header.hire.hires-_-car.hire-_-ENGB.NULL (Accessed: 19 February 2021)

Expedia.co.uk, (2021), “Car Search”, [online], Available at: https://www.expedia.co.uk/carsearch?acop=2&ageInRange=true&d1=2021-03-06&d2=2021-03-07&dagv=1&date1=6%2F3%2F2021&date2=7%2F3%2F2021&dpln=5392460&drid1=&fdrp=0&loc2=&locn=London%20%28LHR%20-%20Heathrow%29&pickupIATACode=LHR&rdct=1&rdus=10&returnIATACode=&styp=4&subm=1&time1=1030AM&time2=1030AM&ttyp=2&vend= (Accessed: 20 February 2021)

MordorIntelligence, (2021), “Car rental market - growth, trends, Covid-19 impact, and forecasts (2021 - 2026)” [online], Available at: https://www.mordorintelligence.com/industry-reports/car-rental-market (Accessed 20 February)

Phillips, Robert. Pricing and Revenue Optimization, Stanford University Press, 2005. ProQuest Ebook Central, https://ebookcentral.proquest.com/lib/city/detail.action?docID=730441

TourismAlliance, (2019), “UK Tourism Statistics 2019”, [online], Available at: https://www.tourismalliance.com/downloads/TA_408_435.pdf (Accessed: 19 February 2021)