ISL PROJECT
STOCK CONTROL PROBLEM

Tilburg University
Econometrics and Operations Research

Authors
Kenneth Rory Kolster
u736326, 2070539
Andrei Agapie
u461825, 2075694
Nikodem Baehr
u459229, 2076515
Josse van Baar
u875460, 2061725

26.05.2023

Contents

1 Introduction and Problem Definition

1.1 Introduction

We introduce our client, *'De Betrouwbare Computer'*, the company at the centre of our stock control problem. We have been tasked to assess the operational logistics of the hardware company and make an informed strategic decision about the improvement of their systems, according to models and underlying research. Our problem focuses on the company with its flow of goods and interactions between numerous location nodes.

In this report, we begin by mapping out our problem with location nodes and interactions, giving insight about the intricacies. We then give our conclusions and recommendations for our client, followed by the argumentation, explaining how we came to our solutions. The appendices contain computer module data, like its demand distribution, and technical terms used.

We initially consider for one type of module, the data for which is given in the Appendix A, but the way in which we construct our models and derive our optimal solutions allows us to apply more module types with different data and determine their respective reorder points and order quantities.

1.2 Problem Mapping

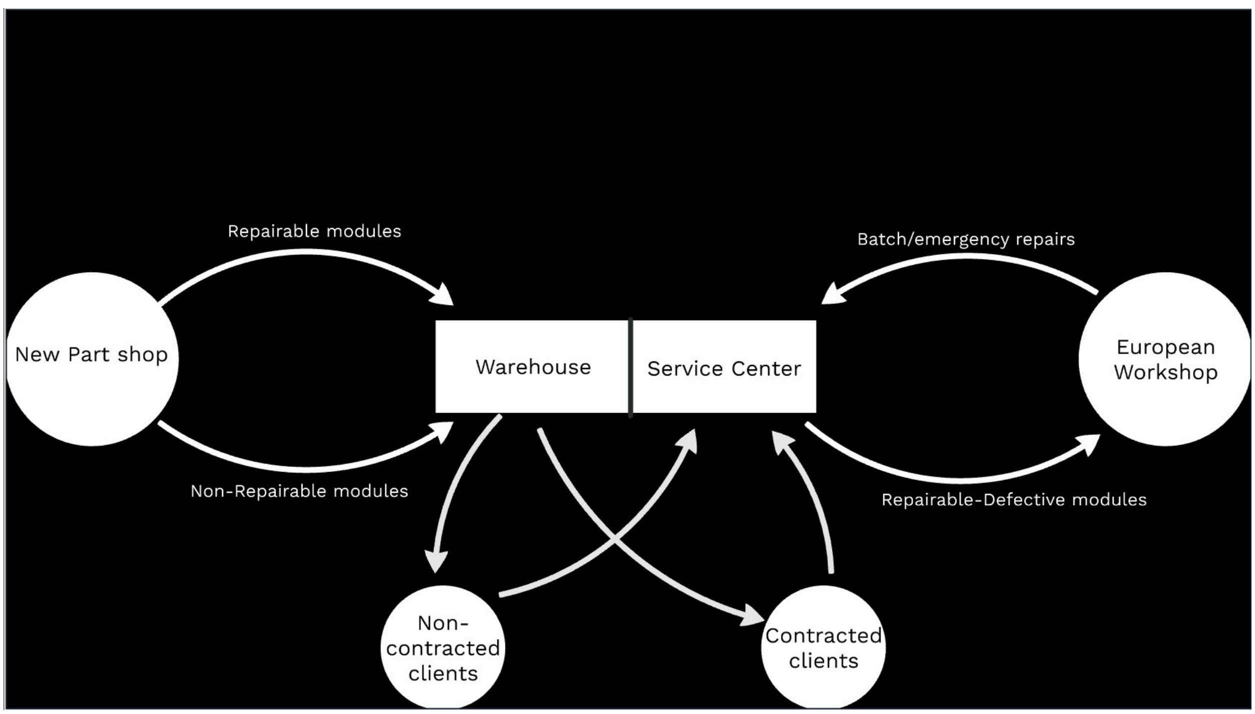

Figure 1: Problem Map

1.3 Definition and Assumptions

The locational nodes are as follows: the *Warehouse*, the *Service Centre*, the locations of the clients, the *European Workshop* (EW), and the *New Parts Shop* (NPS). The company houses a stock of different computer modules at its Warehouse. At the Warehouse, there are three types of modules: non-repairable modules represent 20% of the total stock, and the rest consists of the repairable, working modules, and repairable, defective modules. The defective, non-repairable modules are not stored at the Warehouse as they are simply discarded, while the defective, repairable modules are only kept until the next batch is sent to the European Workshop. The Service Centre of *'De Betrouwbare Computer'* works closely with the Warehouse and fields calls from its clients.

Assumptions of the *Warehouse* and the *Service Center*:

1. The company only hires engineers. They are working both in the field and in the *Service Center*. In the *Service Center*, they replace non-contracted clients' faulty products and take phone calls from contracted customers. In-field engineers, visit the contracted customers and replace their faulty products.

The following two nodes are not under the operations of the company, yet still play an important role in the flow of goods. The *European Workshop* is a centralised workshop where *'De Betrouwbare Computer'* sends its faulty, repairable modules. These components are repaired in batches, despite exceptional emergency repairs, and are sent back to the *Warehouse*.

Assumptions of the *European Workshop*:

1. Demand regarding the European Workshop model, explained later, is deterministic.

2. The repair time of the modules is stipulated in the appendix and so delivery time is negligible.

The *New Parts Shop* is from where the company orders its new modules. Non-repairable parts are already ordered following an Economic Order Quantity (EOQ) model, with an exponentially smoothed demand distribution, which runs satisfactorily. Thus we do not have to consider these types for this problem. The EOQ model establishes the number of parts that should be ordered based on the demand for the specific parts (per unit of time), the order costs (delivery, fixed fee for placing an order) and holding costs (the cost of storing the parts in the *Warehouse*). We shall focus on the repairable modules ordered from this shop as one aspect of the larger problem.

2 Argumentation

2.1 Warehouse Stock Level

We considered the above-described problem as an inventory management problem with two specific facets. We need to determine how much to obtain from the New Parts Shop (NPS) and from the European Workshop (EW). Consequently we have the following two models:

i We model the repair batches received from the EW as an "in-house production run" which had a production rate roughly equal to the demand. From this model we determine the number of modules to collect at the Warehouse until we send it to the EW for them to be repaired. In the meantime, we continue to receive demand during the stochastic lead time. In this model, we assume deterministic demand for simplicity.

ii We model the ordering from the NPS as an EOQ model with uncertain demand and stockouts. In the application of the EOQ model for the NPS, we implement the altered demand from the continual repair batches received. This model will tell us how much to order from the NPS and also an optimal reorder time.

The entire derivations, tailored to this specific case, can be found in Appendix B.

The New Parts Shop and the European Workshop:

Now we relate the formulas to our case. We consider the one repairable module with the known data provided in Appendix A.

Denote the following variables:

$\boldsymbol{D_{OLD}}$ = daily demand of the module[1]
$\boldsymbol{D_{NEW}}$ = daily demand of the module discounting repair batches from European Workshop
r = inventory level at which order is placed
$\boldsymbol{L_{EW}}$ = lead time for a repair batch
$\boldsymbol{L_{NPS}}$ = lead time for a New Parts Shop order
$\boldsymbol{X}$ = demand during lead time

Denote the following parameters:

K_{EW} = fixed cost of a repair batch
K_{NPS} = fixed cost of a repair batch
h_1 = holding cost/unit/day for deliverable (working) modules
h_2 = holding cost/unit/day for non-deliverable (defective) modules
q_{EW} = number of units stored and sent to the European Workshop
q_{NPS} = number of units ordered from the New Parts Shop
δ = percentage of defective, repairable modules that fail to be repaired at the European Workshop
c_B = penalty cost for being one unit short[2]

[1] Random variables are denoted by a bold capital letter

[2] Since non-contracted clients refer to 10% of demand for repairable modules, note that we take 90% of c_B for this value in the model.

We first determine the ordering system for the European Workshop as the continuous rate EOQ. We defined a total cost function, $TC(q_{EW})$ = (holding costs) + (ordering costs), determined the respective terms from it and minimized it in Appendix B. It then gave us an optimal value to store and send to the EW:

$$q^*_{EW} = \sqrt{\frac{2K_{EW}\mathbb{E}[\boldsymbol{D_{OLD}}] - 2\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D_{OLD}}]\right)^2}{h_1 + h_2 - 2\delta}}$$

In this case, the module has two holding costs: if it is a working module then it is €0.66 per unit per day, and if it is defective it costs €0.33 per unit per day. K_{EW} is the fixed repair batch costs, $\mathbb{E}[\boldsymbol{D}]$ is the expected demand found to be 3.03 and $\delta = 0.05$ is the proportion of modules unable to be repared at the EW. Finally, $\mathbb{E}[\boldsymbol{L}]$ is the expected lead time, equal to 5. This yields:

$$q^*_{EW} = \sqrt{\frac{2 \cdot 50 \cdot 3.03 - 10 \cdot (3.03)^2}{0.66 + 0.33 - 2 \cdot 0.05}}$$
$$q^*_{EW} \approx 15.404$$

This means that we collect and store defective modules at the Warehouse until we obtain an amount of 15.404 units, which we then send to the EW. This number is obviously not viable in the real world so we input values 15 and 16 into the Total Cost function in (1) to determine which integer to take. This turns out to be 16 units, as it has a lower total cost. Therefore, we have $q^*_{EW} = 16$. The consequence of this result generates the following stock level figure:

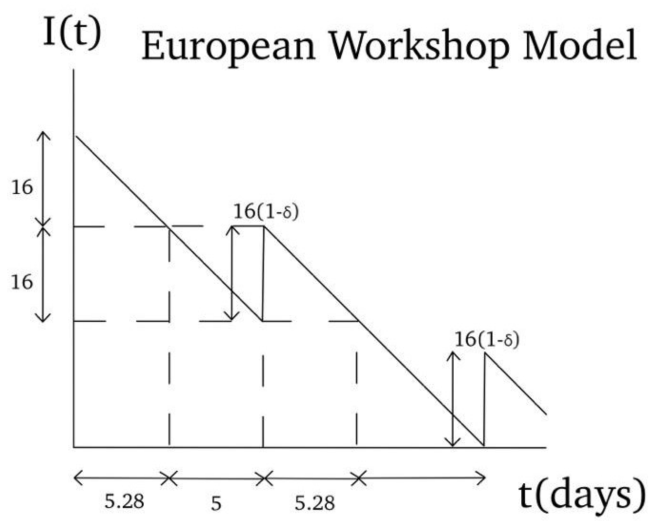

Figure 2: Stock Level with European Workshop (Not to scale)

Before we start solving the ordering system for the New Parts Shop, we need to derive demand during lead time $\boldsymbol{X}$, as we will require it to determine the optimal reorder point r. We took the demand distribution given to us in Appendix A and the following plot shows the probability density of the demand per day:

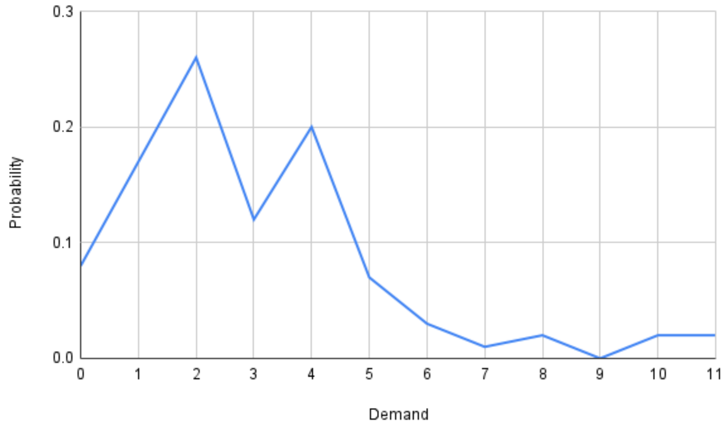

Figure 3: Demand distribution (per day)

With this distribution, we used methods of **linear interpolation** and *random variable scaling* to determine the demand distribution during the lead time (time period during which a new order is being delivered from the New Parts Shop). We developed python code (provided in Appendix C) to scale the demand depending on the lead time, and then create the resulting Probability Density Function (PDF) and Cumulative Distribution Function (CDF). This means that if provided with another demand distribution for another module, or even a different lead time distribution, we can still generate the respective demand during lead time, for that module. The following plot shows the probability density of the demand during lead time, $\boldsymbol{X}$.

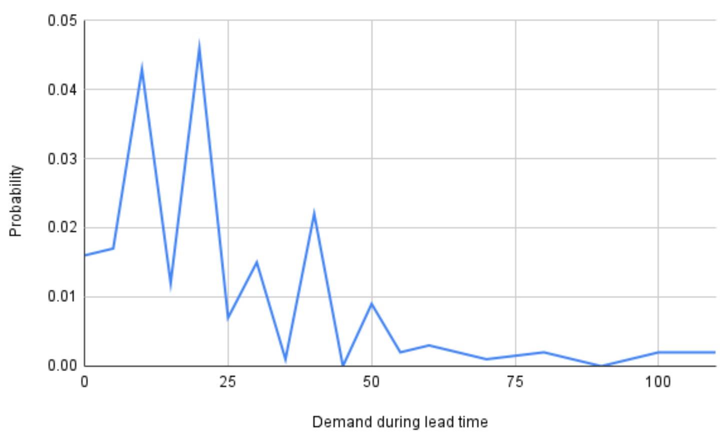

Figure 4: Demand during lead time

Table 1 containing the PDF and CDF of this demand distribution during the lead time can be found in Appendix A.

Now for the ordering system from the New Parts Shop, we defined the total cost function as $TC(q_{NPS}, r)$ = (expected daily holding cost) + (expected daily ordering cost) + (expected daily cost due to shortages). We also minimized this in Appendix B, giving us the following equations. In the derivations, we used h arbitrarily as the holding costs, but now we can implement the two different holding costs of h_1 and h_2. Given that our engineers swap working parts for defective parts, the amount of stock in the Warehouse between cycles remains the same. This then applied to the formulas 6 and 7 obtains:

$$
\begin{aligned}
q^*_{NPS} &= \sqrt{\frac{2K_{NPS}\mathbb{E}[\boldsymbol{D_{NEW}}]}{h_1 + h_2}} \\
\mathbb{P}(\boldsymbol{X} \geq r^*) &= \frac{(h_1 + h_2)q^*_{NPS}}{c_B\mathbb{E}[\boldsymbol{D_{NEW}}]} \\
\mathbb{P}(\boldsymbol{X} < r^*) &= 1 - \frac{(h_1 + h_2)q^*_{NPS}}{c_B\mathbb{E}[\boldsymbol{D_{NEW}}]}
\end{aligned}
$$

Note that from the continuous rate EOQ model of the repair batches from the European Workshop, the daily demand is altered. We calculate the new expected demand to be:

$$
\begin{aligned}
\mathbb{E}[\boldsymbol{D_{NEW}}] &= \frac{q^*_{EW} + \mathbb{E}[\boldsymbol{D_{OLD}}] \cdot \mathbb{E}[\boldsymbol{L_{EW}}] - q^*_{EW}(1-\delta)}{\frac{q^*_{EW}}{\mathbb{E}[\boldsymbol{D_{OLD}}]} + \mathbb{E}[\boldsymbol{L_{EW}}]} \\
\mathbb{E}[\boldsymbol{D_{NEW}}] &= \frac{16 + 3.03 \cdot 5 - 16(1 - 0.05)}{5.28 + 5} \\
\mathbb{E}[\boldsymbol{D_{NEW}}] &\approx 1.55
\end{aligned}
$$

We now plug in our values in the equations above and obtain our optimal values:

$$
\begin{aligned}
q^*_{NPS} &= \sqrt{\frac{2 \cdot 150 \cdot 1.55}{0.66 + 0.33}} \\
&\approx 21.68
\end{aligned}
$$

We input 21 and 22 into the Total Cost function (3) and receive 22 as the optimal choice since it yielded a lower cost. As we don't have the exact value of c_B (penalty cost of failing to supply a client) this up to further deliberation, but for now we assume it to be €100. Now we compute the optimal reorder time equation.

$$
\begin{aligned}
\mathbb{P}(\boldsymbol{X} < r^*) &= 1 - \frac{0.99 \cdot 22}{c_B \cdot 1.55} \\
&= 1 - \frac{0.99 \cdot 22}{100 \cdot 1.55} \\
\mathbb{P}(\boldsymbol{X} < r^*) &\approx 0.8596
\end{aligned}
$$

Using the CDF table we obtain the optimal reorder value of $r^* = 43$. Thus together with an optimal order quantity of 22, the ordering system from the New Parts Shop may look similarly to the following sketch:

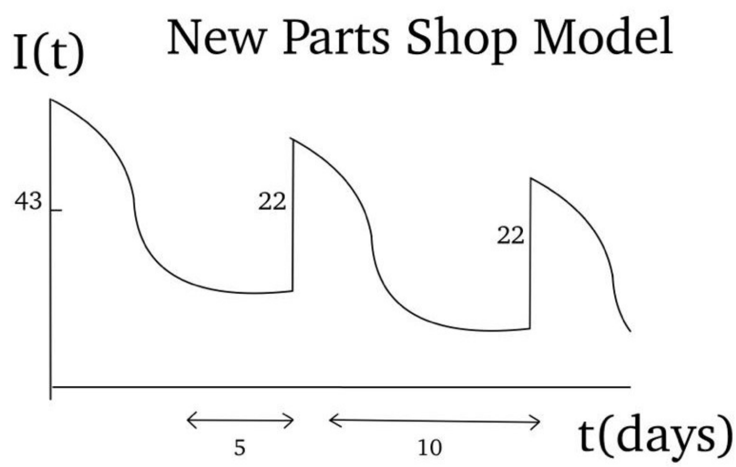

Figure 5: Stock Level with New Parts Shop (Not to scale)

2.2 Engineer Markov Chain

We consider the task of staffing our company in order to service clients with a faulty device. As described in the Definition and Assumptions section, our company has engineers that visit and service clients who call the Service Center. We decided to model the calls from clients as a Markov birth and death process **(M/M/c)**, with the states symbolising the amount of calls outstanding for parts. We did this because we viewed the situation as a system with c engineers (servers) who have to deal with customers whose calls can become a queue. We take the rate at which clients call the Service Center to be the demand for the module, and we assume calls to arrive as a Poisson process so that we are able to apply this **M/M/c** model. Given that the expected daily demand is approximately 3, and that there are 8 working hours in a day, we compute the arrival rate (birth rate) λ to be equal to 3/8. We also then assume our engineers to have an exponential service time with a mean of 1 hour. This service time will also include travel time. Thus our service rate (death rate) μ is equal to 1.

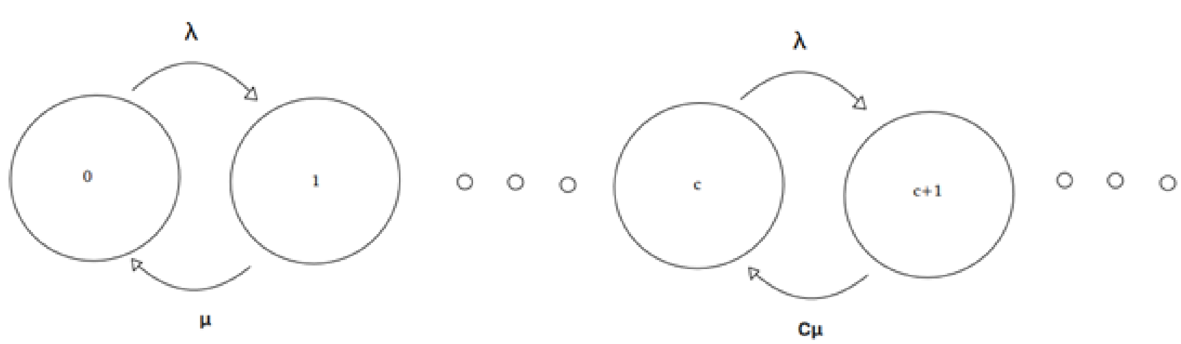

Figure 6: Markov Chain

Our aim is to hire enough engineers to service our clients, with the restriction that at least 95% of them are serviced within 8 hours of their call. This allows us to activate a maximum of 5% of penalty clauses for all calls that the Service Center receives. In order to model this birth and death process we apply the formulas for the limiting distribution that we have been taught as part of our Stochastic Operations Research course, namely:

$$p_n = \begin{cases} \frac{c^n \cdot \rho^n}{n!} p_0, & \text{for } n \leq c \\ \frac{c^c \cdot \rho^n}{c!} p_0, & \text{for } n > c \end{cases}$$

with

$$p_0 = \left[\sum_{n=0}^{c-1} \left(\frac{c^n \cdot \rho^n}{n!} \right) + \frac{c^c}{c!} \frac{\rho^c}{1-\rho} \right]^{-1}$$

where $\rho = \frac{\lambda}{c \cdot \mu}$.

Given these formulas and the model we have chosen, we can determine the probability of a customer waiting in the queue using Little's Law. The derivation can be found in Appendix B. We have the final equation:

$$\mathbb{P}[W_q \leq 8] = 1 - \frac{p_c}{1-\rho} + (\frac{p_c}{1-\rho}) \times [1 - e^{-c\mu(1-\rho)\cdot 8}]$$

Now we set up our objective function and subsequent constraint. We want to have the number of engineers minimal since we want to minimize the cost of employee wages, but we also require the waiting probability of less than 8 hours to be at least 95%. This gives us:

$$\begin{aligned} &\arg\min_c \ \mathbb{P}[W_q(c) \leq 8] \\ s.t. \ \ &\mathbb{P}[W_q(c) \leq 8] \geq 0.95 \end{aligned}$$

We consider the case for only one module, where the call rate is $\frac{3}{8}$ and the service rate is 1. Then by using a minimization function in python (code provided in Appendix C), we found the optimal number of engineers for the system with only one module to be 1, with the percentage of customers waiting longer than 8 working hours being $\mathbb{P}[W_q(c) \leq 8] \approx 0.9975$.

If we were to consider the case of two modules, one repairable and one non-repairable for instance, the new birth rate of the process is one divided by the sum of the two different rates. This is because the two processes are both Poisson counting processes, and since we are interested in the outstanding number of calls for both modules, we can sum the processes, resulting in each "birth" (incoming call probability) being equal to the chance of either module breaking first, which is the inverse of the sum of their two rates. Let λ_r be the rate at which calls about the repairable module arrive and λ_{nr} the rate at which calls about the non-repairable module arrive. The service time, μ, remains the same. The limiting distribution is now:

$$p_n = \begin{cases} \frac{((\lambda_r + \lambda_{nr})/\mu)^n}{n!} p_0, & \text{for } n \leq c \\ \frac{(\lambda_r + \lambda_{nr})^n}{\mu^n} \frac{c^c}{c!} p_0, & \text{for } n > c \end{cases}$$

with

$$p_0 = \left[\sum_{n=0}^{c-1} \left(\frac{c^n((\lambda_r + \lambda_{nr})/c \cdot \mu)^n}{n!} \right) + \frac{((\lambda_r + \lambda_{nr})/c \cdot \mu)^c}{c!} \frac{1}{1-\rho} \right]^{-1}$$

The above simply shows that the rates of calls can be summed. Thus for any number of modules, provided that we know their mean time of breaking, we can determine the number of engineers required to satisfy our constraint of 95% service rate.

Moreover, consider a larger number of modules, which intuitively increases the number of engineers as well. For example, if the summed call rate of numerous module types was $\frac{100}{8}$, the optimal number of engineers is 14, which gives $\mathbb{P}[W_q(c) \leq 8] \approx 0.9999$. On the other hand, 13 engineers in this situation gives $\mathbb{P}[W_q(c) \leq 8] \approx 0.9001$, which may be more inclined to the companies wishes, depending on the penalty costs and the costs of employee wages.

3 Conclusions and Recommendations

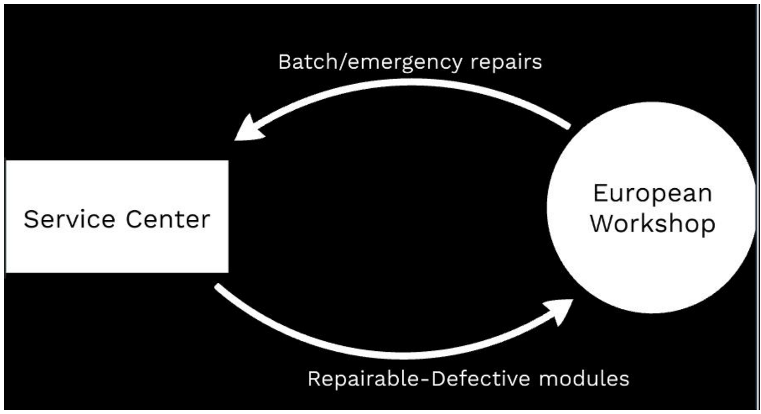

Figure 7: European Workshop (EW)

To solve the European Workshop problem of when to send the faulty modules to the Workshop after collecting and storing them at the Warehouse for a certain time, we recommend that the optimal quantity to do that is when q_{EW} is equal to 16. Given the quantity and the time it takes to repair the modules, we find it to be the best solution to keep the stock at the optimal level and minimising storage costs. Following this advice would mean sending a batch to the European Workshop roughly every 5.28 days, costing €$50 + 16 \cdot$ €(price of repair per unit).

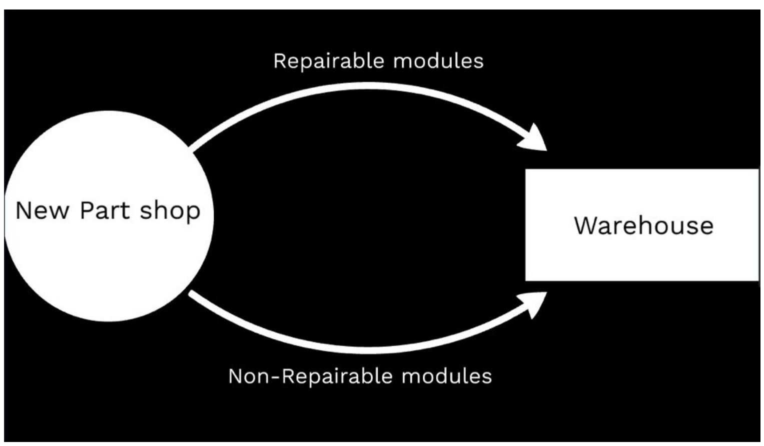

Figure 8: New Parts Shop (NPS)

The optimal quantity from the European Workshop problem, leads to a solution of the New Parts Shop problem of when and how many new modules to order. We recommend to order a new shipment of 22 modules from the New Parts Shop, when the inventory level of the module in the Warehouse is equal to 43 units. This will allow us to minimise the total costs relating to holding and purchasing the modules, as well as incurring penalty costs for not supplying a client with a replacement. Each order would cost €$150 + 22 \cdot$ €(price per new module unit), which would happen approximately 14 days. This sounds often, however when considering more modules with different demands, we could minimize costs by arranging combined purchase orders, resulting in shared deliveries. This would also decrease the optimal frequency of orders and thus r^* and q^*_{NPS} would increase. Combining the two models for the European Workshop and the New Parts Shop leads to a potential stock level graph like the following:

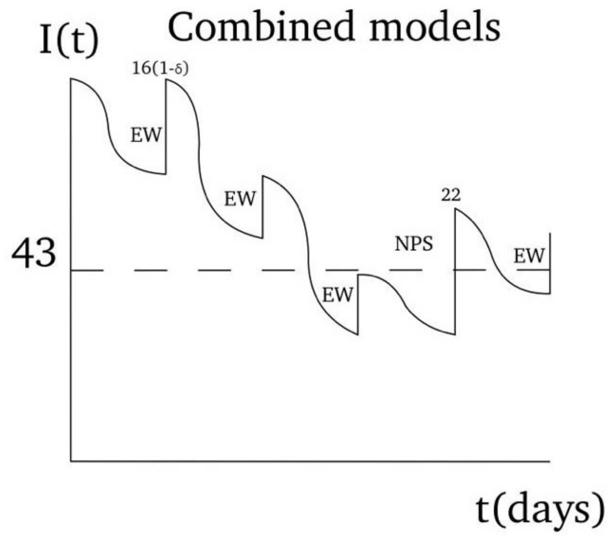

Figure 9: Combined Stock Level (Not to scale)

To solve the problem of staffing to meet the number of calls, we recommend hiring 1 engineer (for the case of only 1 module). This ensures that at least 95% of the contracted clients are serviced before 8 hours, which prevents the penalty clause from being activated. In fact, as in the Argumentation section 99.75% of customers wait less than 8 working hours. Of course, if we are considering more modules and thus a higher call rate, this number of engineers increases, as explained in the Argumentation section.

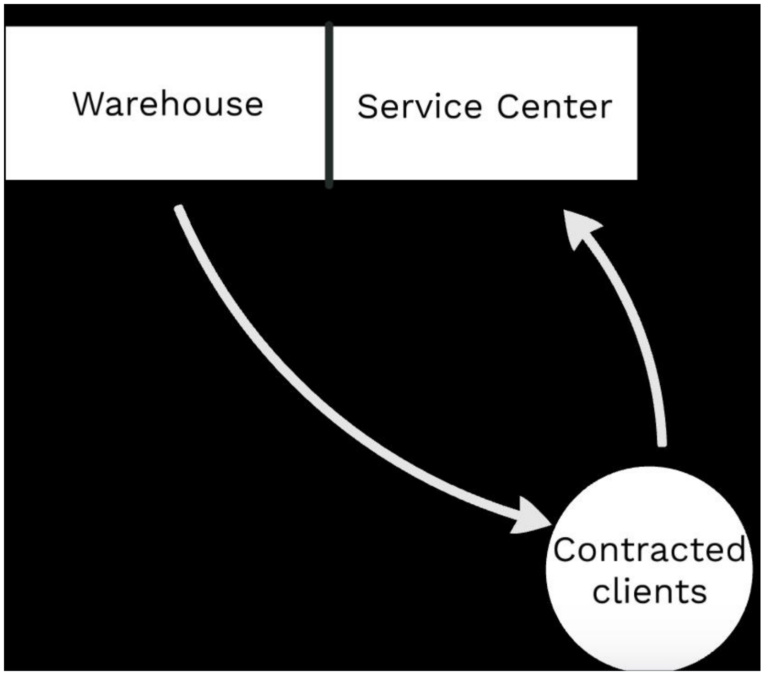

Figure 10: Engineers and Contracted clients

4 Bibliography

J.S.H. van Leeuwaarden (2023). Stochastic Operations Research Models. Part I. Department of Econometrics and Operations Research Tilburg University, 20-25

J.P.C. Blanc and J.S.H. van Leeuwaarden (2023). Stochastic Operations Research Models Part II: Queueing Systems and Inventory Management. Department of Econometrics and Operations Research Tilburg University, 44-68

Wayne L.Winston (2004). Operations Research Applications and Algorithms, Third Edition.

Brown, R. Decision Rules for Inventory Management. New York: Holt, Rinehart and Winston, 1967.

5 Appendices

5.1 Appendix A

The following data for one of the repairable modules are known:

Daily distribution of demand:

Demand	Probability	Demand	Probability
0	0.08	6	0.03
1	0.17	7	0.01
2	0.26	8	0.02
3	0.12	9	0.00
4	0.20	10	0.02
5	0.07	11	0.02

Other data with regard to this module

Fraction of non-repairable modules:	0.05
Time required for a repair batch:	4 or 16 days, 50% chance either way
Delivery times of new module orders:	5 or 10 days, 50% chance either way
Fixed costs in the event of repair batch:	€50
Fixed ordering costs:	€150
Stock costs deliverable module:	€0.66 (per unit per day)
Stock costs defective module:	€0.33 (per unit per day)
Fraction of unused modules:	0.33[3]
Percentage delivered from stock:	81.3% (including emergency repairs)
	50% (excluding emergency repairs)
Average number of outstanding backorders:	1.01

We are neglecting the last 3 data points.

The table on the next page contains the Probability Density Function (PDF) and Cumulative Distribution Function (CDF) values for the demand during lead time of an order from the New Parts Shop. We created it via the method explained in the Argumentation section.

[3]This means that of every three modules given to the field engineers, one is returned by the engineers after three days (as a surplus item).

Table 1: Demand during lead time

(a) Part 1 (b) Part 2 (c) Part 3

X	PDF	CDF
0	0.016	0.016
1	0.0162	0.0322
2	0.0164	0.0486
3	0.0166	0.0652
4	0.0168	0.082
5	0.017	0.099
6	0.0222	0.1212
7	0.0274	0.1486
8	0.0326	0.1812
9	0.0377	0.2189
10	0.0429	0.2618
11	0.0367	0.2985
12	0.0306	0.3291
13	0.0244	0.3535
14	0.0182	0.3717
15	0.012	0.3837
16	0.0188	0.4025
17	0.0256	0.4281
18	0.0324	0.4605
19	0.0391	0.4996
20	0.0459	0.5455
21	0.0381	0.5836
22	0.0304	0.614
23	0.0226	0.6366
24	0.0148	0.6514
25	0.007	0.6584
26	0.0086	0.667
27	0.0102	0.6772
28	0.0118	0.689
29	0.0134	0.7024
30	0.015	0.7174
31	0.0122	0.7296
32	0.0094	0.739
33	0.0066	0.7456
34	0.0038	0.7494
35	0.001	0.7504
36	0.0052	0.7556
37	0.0094	0.765
38	0.0136	0.7786
39	0.0178	0.7964
40	0.022	0.8184
41	0.0176	0.836
42	0.0132	0.8492
43	0.0088	0.858
44	0.0044	0.8624
45	0	0.8624
46	0.0018	0.8642
47	0.0036	0.8678
48	0.0054	0.8732
49	0.0072	0.8804
50	0.009	0.8894
51	0.0076	0.897
52	0.0062	0.9032
53	0.0048	0.908
54	0.0034	0.9114
55	0.002	0.9134
56	0.0022	0.9156
57	0.0024	0.918
58	0.0026	0.9206
59	0.0028	0.9234
60	0.003	0.9264
61	0.0028	0.9292
62	0.0026	0.9318
63	0.0024	0.9342
64	0.0022	0.9364
65	0.002	0.9384
66	0.0018	0.9402
67	0.0016	0.9418
68	0.0014	0.9432
69	0.0012	0.9444
70	0.001	0.9454
71	0.0011	0.9465
72	0.0012	0.9477
73	0.0013	0.949
74	0.0014	0.9504
75	0.0015	0.9519
76	0.0016	0.9535
77	0.0017	0.9552
78	0.0018	0.957
79	0.0019	0.9589
80	0.002	0.9609
81	0.0018	0.9627
82	0.0016	0.9643
83	0.0014	0.9657
84	0.0012	0.9669
85	0.001	0.9679
86	0.0008	0.9687
87	0.0006	0.9693
88	0.0004	0.9697
89	0.0002	0.9699
90	0	0.9699
91	0.0002	0.9701
92	0.0004	0.9705
93	0.0006	0.9711
94	0.0008	0.9719
95	0.001	0.9729
96	0.0012	0.9741
97	0.0014	0.9755
98	0.0016	0.9771
99	0.0018	0.9789
100	0.002	0.9809
101	0.002	0.9829
102	0.002	0.9849
103	0.002	0.9869
104	0.002	0.9889
105	0.002	0.9909
106	0.002	0.9929
107	0.002	0.9949
108	0.002	0.9969
109	0.002	0.9989
110	0.002	1.0009

5.2 Appendix B

Derivation of the Continuous rate EOQ (production run):

This derivation follows the continuous rate EOQ derivation in Winston (2004), tailored to our case project. We assume now stockouts for this part of the stock model as they are included in the EOQ model with uncertain demand provided hereafter. We also deviate from the derivation in Winston by including uncertain demand.

Denote the following parameters:

K = fixed cost of a repair batch
h_1 = holding cost/unit/day for deliverable (working) modules
h_2 = holding cost/unit/day for non-deliverable (defective) modules

Denote the following variables:

q_{EW} = number of units stored and sent to the European Workshop
$\boldsymbol{D}$ = daily demand of the module[4]
$\boldsymbol{L_{EW}}$ = lead time for a repair batch

We define the Total Cost as a function of q_{EW}, where $TC(q_{EW})$ = (holding costs) + (ordering costs). The holding cost is split into two terms: the stock of deliverable goods and non-deliverable goods. The average inventory level of non-deliverable goods is the average level of q_{EW} stocked at the Warehouse, and so can be computed as

$$\begin{aligned} \text{average } q_{EW} \text{ level} &= \frac{1}{2}[\text{amount at start of } \textbf{cycle} + \text{amount at end}] \\ &= \frac{1}{2}[0 + q_{EW}] \\ \implies \text{holding cost of non-deliverable} &= \frac{1}{2} q_{EW} \end{aligned}$$

The average inventory of the deliverable parts is more complex, since we have a non-consistent, non-stationary stock level with numerous cycles. This means that the inventory levels at the beginning and end of each cycle are not constant and hence has to be included in the calculations. We denote f arbitrarily as the stock level before beginning cycle between orders from the New Parts Shop. It's value is not significant in this derivation as it is a constant that cancels away. The following table contains the stock levels at the beginning and end of each cycle of repair batches, and the average level of the cycle.

Cycle	Beginning	End	Average
1st	f	$f - q_{EW} - \mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$	$f - \frac{1}{2}q_{EW} - \frac{1}{2}\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$
2nd	$f + \delta q_{EW} - q_{EW} - \mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$	$f + \delta q_{EW} - 2q_{EW} - 2\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$	$f + \delta q_{EW} - \frac{3}{2}q_{EW} - \frac{3}{2}\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$
3rd	$f + 2\delta q_{EW} - 2q_{EW} - 2\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$	$f + 2\delta q_{EW} - 3q_{EW} - 3\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$	$f + 2\delta q_{EW} - \frac{5}{2}q_{EW} - \frac{5}{2}\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$

We use $\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}}$ to denote the number of cycles in the model, because demand should be met. Thus depending on the optimal q^*_{EW}, which we solve for eventually, the number of cycles is then taken into account here.

[4]Random variables are denoted by a bold capital letter

This leads to the "n^{th}-term" formula of average deliverable stock level as:

$$f+\left(\delta\cdot\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}}-\delta\right)q_{EW}+\left(-\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}}+\frac{1}{2}\right)q_{EW}+\left(-\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}}+\frac{1}{2}\right)\cdot\mathbb{E}[\boldsymbol{L_{EW}}]\mathbb{E}[\boldsymbol{D}]$$
$$\implies f+\left(\delta-1+\frac{\mathbb{E}[\boldsymbol{L_{EW}}]}{2}\right)\mathbb{E}[\boldsymbol{D}]+\left(\frac{1}{2}-\delta\right)q_{EW}-\frac{\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{q_{EW}}$$

Finally, we require the ordering cost which is simply K multiplied by the number of orders (number of cycles), and that is equal to:

$$\text{ordering cost}=K\cdot\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}}$$

Now we bring all the computed terms together to form the Total Cost function which we then minimize with respect to q_{EW}.

$$TC(q_{EW})=h_1\left(f+\left(\delta-1+\frac{\mathbb{E}[\boldsymbol{L_{EW}}]}{2}\right)\mathbb{E}[\boldsymbol{D}]+\left(\frac{1}{2}-\delta\right)q_{EW}-\frac{\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{q_{EW}}\right)+h_2\frac{q_{EW}}{2}+K\frac{\mathbb{E}[\boldsymbol{D}]}{q_{EW}} \tag{1}$$

We take the first order conditions and set it to zero:

$$\frac{\partial TC(q_{EW})}{\partial q_{EW}}=h_1\left(\frac{1}{2}-\delta\right)+\frac{\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{(q_{EW})^2}+\frac{h_2}{2}-\frac{K\mathbb{E}[\boldsymbol{D}]}{(q_{EW})^2}$$
$$0=\frac{h_1+h_2-2\delta}{2}+\frac{K\mathbb{E}[\boldsymbol{D}]-\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{(q_{EW})^2}$$
$$\implies (q_{EW})^2=\frac{2K\mathbb{E}[\boldsymbol{D}]-2\cdot\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{h_1+h_2-2\delta}$$
$$\implies q^*_{EW}=\sqrt{\frac{2K\mathbb{E}[\boldsymbol{D}]-2\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{h_1+h_2-2\delta}} \tag{2}$$

We ignore the negative root of the solution because quantities cannot be negative. We now take the second order condition to confirm that the solution we have found is minimal.

$$\frac{\partial^2 TC(q_{EW})}{\partial q^2_{EW}}=-\frac{2\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{q^3_{EW}}+\frac{2K\mathbb{E}[\boldsymbol{D}]}{q^3_{EW}}>0$$

$$\iff \frac{2K\mathbb{E}[\boldsymbol{D}]}{q^3_{EW}}>\frac{2\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2}{q^3_{EW}}$$
$$\iff K\mathbb{E}[\boldsymbol{D}]>\mathbb{E}[\boldsymbol{L_{EW}}]\left(\mathbb{E}[\boldsymbol{D}]\right)^2$$
$$\iff K>\mathbb{E}[\boldsymbol{L_{EW}}]\cdot\mathbb{E}[\boldsymbol{D}]$$

Because K, $\mathbb{E}[\boldsymbol{D}]$, and $\mathbb{E}[\boldsymbol{L_{EW}}]$ are 50, 3, and 5, respectively, in our case the solution we have found is indeed minimal. This then concludes the derivation of the EOQ model for the European Workshop repair batches.

Derivation of the EOQ with Uncertain Demand:

This derivation also follows analogue to its counterpart in Winston (2004).

Denote the following variables:

$\boldsymbol{L_{NPS}}$ = lead time for each order from the New Parts Shop

q_{NPS} = quantity ordered each time an order takes place

$\boldsymbol{D}$ = daily demand of the module

$I(t)$ = Inventory level at time t

r = inventory level at which order is placed

$\boldsymbol{X}$ = demand during lead time

$\boldsymbol{B}_r$ = number of back orders during a cycle if the reorder point is r

Denote the following parameters:

K = fixed ordering cost

h = holding cost/unit/day

c_B = penalty cost for being one unit short[5]

Note: We define here holding cost h arbitrarily for the usage in derivation solely. In the project, as shown the Argumentation section, we apply our case of two separate holding costs.
We assume $\boldsymbol{D}$ and $\boldsymbol{X}$ to be identically distributed and demands at different points in time are independent. We also assume lead time $\boldsymbol{L}$ is independent of the demand per unit time during lead time. Then we have the following results:

$$\begin{aligned}\mathbb{E}[\boldsymbol{X}] &= \mathbb{E}[\boldsymbol{L_{NPS}}]\mathbb{E}[\boldsymbol{D}],\\ var\boldsymbol{X} &= \mathbb{E}[\boldsymbol{L_{NPS}}](var\boldsymbol{D}) + \mathbb{E}[(\boldsymbol{D})^2](var\{\boldsymbol{L_{NPS}}\})\end{aligned}$$

Since for the one module initial case, we have that each module is purchased for the same amount and so purchasing costs are fixed and hence ignored in this derivation.

We define the objective function for this model as $TC(q_{NPS}, r)$ = (expected daily holding cost) + (expected daily ordering cost) + (expected daily cost due to shortages). Then for the expected daily holding cost we calculate it to be h(expected value of $I(t)$).

$$\begin{aligned}\text{Expected value of } I(t) \text{ during a cycle} = &\frac{1}{2}[(\text{expected value of } I(t) \text{ at cycle beginning})\\ &+ (\text{expected value of } I(t) \text{ at cycle end}]\end{aligned}$$

At the end of the cycle the inventory level will equal the inventory level when the order is placed, r subtract the amount of demand that occurred in the lead time $\boldsymbol{X}$. At the beginning of the cycle the inventory level will equal the amount of stock at the end of the cycle together with the amount ordered q_{NPS}. Hence, this now yields:

[5] Since non-contracted clients refer to 10% of demand for repairable modules, note that we take 90% of c_B for this value in the model.

$$\begin{aligned}\text{Expected value of } I(t) \text{ during a cycle} &= \frac{1}{2}(r - \mathbb{E}[\boldsymbol{X}] + r - \mathbb{E}[\boldsymbol{X}] + q_{NPS}) \\ &= \frac{q_{NPS}}{2} + r - \mathbb{E}[\boldsymbol{X}]\end{aligned}$$

Therefore, the expected daily holding cost $\approx h(\frac{q_{NPS}}{2} + r - \mathbb{E}[\boldsymbol{X}])$. Now we calculate the expected daily shortage cost.

$$\text{Expected daily shortage cost} = (\text{expected shortage cost}) \times (\text{expected cycles})$$

An average of $\frac{\mathbb{E}[\boldsymbol{D}]}{q_{NPS}}$ orders will be placed each day (corrected in terms of scale) in order to meet all the demand. Then using the definition of $\boldsymbol{B}_r$:

$$\text{Expected shortage cost} = c_B \mathbb{E}[\boldsymbol{B}_r]$$
$$\implies \text{Expected daily shortage cost} = \frac{c_B \mathbb{E}[\boldsymbol{B}_r]\mathbb{E}[\boldsymbol{D}]}{q_{NPS}}$$

Finally, we have the expected daily order cost:

$$\text{Expected daily order cost} = K(\frac{\text{expected orders}}{\text{day}}) = \frac{K\mathbb{E}[\boldsymbol{D}]}{q_{NPS}}$$

Then all together, we have derived the objective function to be:

$$TC(q_{NPS}, r) = h(\frac{q_{NPS}}{2} + r - \mathbb{E}[\boldsymbol{X}]) + \frac{c_B \mathbb{E}[\boldsymbol{B}_r]\mathbb{E}[\boldsymbol{D}]}{q_{NPS}} + \frac{K\mathbb{E}[\boldsymbol{D}]}{q_{NPS}} \tag{3}$$

Then using a method called the **Golden Section Search**, we can find values for q_{NPS} and r that minimize equation (3), such that they satisfy the following:

$$\frac{\partial \text{TC}(q^*_{NPS}, r^*)}{\partial q_{NPS}} = \frac{\partial \text{TC}(q^*_{NPS}, r^*)}{\partial r} = 0$$

Brown (1967) showed that the EOQ can be used to approximate q^* in the majority of cases if EOQ $\leq \boldsymbol{\sigma_X}$ (standard deviation of demand during lead time). This value we computed to be roughly 22.1, hence this bound is satisfied in our problem. Therefore, we obtain for the optimal value of q_{NPS}:

$$q^*_{NPS} = \sqrt{\frac{2K\mathbb{E}[\boldsymbol{D}]}{h}} \tag{4}$$

Now that we have a given value for q^*_{NPS}, we can use **marginal analysis** to determine an optimal value for r, which minimizes $TC(q_{NPS}, r)$. Also expected daily ordering cost is independent of r, so we can focus on the expected daily holding and shortage costs. Then by keeping q_{NPS} fixed and increasing r by an arbitrarily small amount Δ we can see whether this results in an increase or decrease in $TC(q_{NPS}, r)$.

For expected daily holding cost, increasing the reorder point, r by Δ will result in an increase of:

$$h(\frac{q_{NPS}}{2} + r\Delta - \mathbb{E}[\boldsymbol{X}]) - h(\frac{q_{NPS}}{2} + r - \mathbb{E}[\boldsymbol{X}]) = h\Delta$$

For expected daily shortage costs, if we increase r by Δ, there will be a reduction. This is because if $\mathbb{P}(\boldsymbol{X} \geq r)$, then shortage costs decrease by $c_B\Delta$, since the number of shortages in the cycle will decrease by Δ units. With the average of $\frac{\mathbb{E}[\boldsymbol{D}]}{q_{NPS}}$ cycles per day, increasing the reorder point to $r + \Delta$ will reduce the expected daily shortage costs by

$$\frac{\Delta\mathbb{E}[\boldsymbol{D}]c_B\mathbb{P}(\boldsymbol{X} \geq r)}{q_{NPS}}$$

Due to the fact that $\mathbb{P}(\boldsymbol{X} \geq r)$ decreases when r increases, the above decreases when you increase r by Δ. Let r^* be the optimal value of r such that marginal benefit equals marginal cost, or in terms of the formulas:

$$\frac{\Delta\mathbb{E}[\boldsymbol{D}]c_B\mathbb{P}(\boldsymbol{X} \geq r^*)}{q_{NPS}} = h\Delta$$

$$\mathbb{P}(\boldsymbol{X} \geq r^*) = \frac{hq^*_{NPS}}{c_B\mathbb{E}[\boldsymbol{D}]} \tag{5}$$

If $\frac{hq^*_{NPS}}{c_B\mathbb{E}[\boldsymbol{D}]} > 1$, then (4) has no solution. However, in our case we will assume our penalty cost c_B to be large enough. That concludes the derivation of the EOQ with uncertain demand, with the following formulas:

$$q^*_{NPS} = \sqrt{\frac{2K\mathbb{E}[\boldsymbol{D}]}{h}} \tag{6}$$

$$\mathbb{P}(\boldsymbol{X} \geq r^*) = \frac{hq^*_{NPS}}{c_B\mathbb{E}[\boldsymbol{D}]} \tag{7}$$

Markov Chain derivations:

Since we have a Queueing System following M/M/c, we have the following formulas:

$$p_n = \begin{cases} \frac{c^n \cdot \rho^n}{n!} p_0, & \text{for } n \leq c \\ \frac{c^c \cdot \rho^n}{c!} p_0, & \text{for } n > c \end{cases}$$

with

$$p_0 = \left[\sum_{n=0}^{c-1}\left(\frac{c^n \cdot \rho^n}{n!}\right) + \frac{c^c}{c!}\frac{\rho^c}{1-\rho}\right]^{-1}$$

where $\rho = \frac{\lambda}{c \cdot \mu}$.

Now we determine the waiting time distribution for someone waiting in the queue for longer than 8 hours. We us Little's Law and some other given identities from M/M/c queues. It is calculated as follows:

$$
\begin{aligned}
\mathbb{P}[T_q \leq y] &= w_0 + (1 - w_0) \times [1 - e^{-c\mu(1-\rho)\cdot y}], \\
w_0 &:= \mathbb{P}[T_q = 0], \\
&= 1 - p_B, \\
p_B &= \frac{p_c}{1-\rho}, \\
\implies w_0 &= 1 - \frac{p_c}{1-\rho}, \\
T_q &= \lambda W_q.
\end{aligned}
$$

From these, we can derive the probability of waiting time being more equal to or less than 8 hours.

$$
\begin{aligned}
\mathbb{P}[T_q \leq y] &= 1 - \frac{p_c}{1-\rho} + (\frac{p_c}{1-\rho}) \times [1 - e^{-c\mu(1-\rho)\cdot y}], \\
\mathbb{P}[\lambda \cdot W_q \leq y] &= 1 - \frac{p_c}{1-\rho} + (\frac{p_c}{1-\rho}) \times [1 - e^{-c\mu(1-\rho)\cdot y}], \\
\mathbb{P}[W_q \leq \frac{y}{\lambda}] &= 1 - \frac{p_c}{1-\rho} + (\frac{p_c}{1-\rho}) \times [1 - e^{-c\mu(1-\rho)\cdot \frac{y}{\lambda}}], \\
\mathbb{P}[W_q \leq 8] &= 1 - \frac{p_c}{1-\rho} + (\frac{p_c}{1-\rho}) \times [1 - e^{-c\mu(1-\rho)\cdot 8}] \qquad (8)
\end{aligned}
$$

Now we set up our objective function and subsequent constraint. We want to have the number of engineers minimal since we want to minimize the cost of employee wages, but we also require the waiting probability of less than 8 hours to be at least 95%. This gives us:

$$
\begin{aligned}
&\arg\min_{c} \ \mathbb{P}[W_q(c) \leq 8] \qquad (9) \\
&s.t. \ \ \mathbb{P}[W_q(c) \leq 8] \geq 0.95 \qquad (10)
\end{aligned}
$$

5.3 Appendix C

Python file used for generating scaled-distribution with linear interpolation:

```
import pandas as pd
import csv
from openpyxl import Workbook

leadtime_distribution1 = {5: 0.5, 10: 0.5}
demand_distribution1 = {0: 0.08, 1: 0.17, 2: 0.26,
                        3: 0.12, 4: 0.2, 5: 0.07,
                        6: 0.03, 7: 0.01, 8: 0.02,
                        9: 0, 10: 0.02, 11: 0.02}

def scaled_distribution(leadtime_distribution, demand_distribution):
    scaled_distribution1 = {}
```

```python
    for l, P_l in leadtime_distribution.items():
        for d, P_d in demand_distribution.items():
            if l*d in scaled_distribution1:
                scaled_distribution1[l*d] += P_l*P_d
            elif l*d not in scaled_distribution1:
                scaled_distribution1[l*d] = P_l*P_d

    return scaled_distribution1

def linear_interpolation(distribution):
    smoothed_distribution = {}

    keys = list(distribution.keys())
    values = list(distribution.values())

    for i in range(len(keys) - 1):
        current_key = keys[i]
        next_key = keys[i + 1]
        current_value = values[i]
        next_value = values[i + 1]

        smoothed_distribution[current_key] = current_value

        # Calculate the number of intermediary integers between current_key and next_key
        num_intermediary = next_key - current_key - 1

        if num_intermediary > 0:
            # Calculate the increment in probability for each intermediary integer
            increment = (next_value - current_value) / (num_intermediary + 1)

            # Distribute the probabilities evenly across the intermediary integers
            for j in range(1, num_intermediary + 1):
                intermediary_key = current_key + j
                intermediary_prob = current_value + (increment * j)
                smoothed_distribution[intermediary_key] = intermediary_prob

    # Add the last key-value pair from the original distribution
    smoothed_distribution[keys[-1]] = values[-1]

    return smoothed_distribution

def X(distribution):
    total = sum(distribution.values())

    X = {}
```

```
    for key, value in distribution.items():
        scaled_value = value / total
        X[key] = scaled_value

    return X

def rounded_distribution(distribution):
    rounded_distribution1 = {}
    for x, P_x in distribution.items():
        rounded_distribution1[x] = round(P_x, 4)

    return rounded_distribution1

scaled_distribution1 = scaled_distribution(leadtime_distribution1, demand_distribution1)
linear_interpolation1 = linear_interpolation(scaled_distribution1)
X1 = X(linear_interpolation1)
rounded_distribution1 = rounded_distribution(X1)
print("rounded X distribution: ", rounded_distribution1)


def dictionary_to_csv(dictionary, filename):
    with open(filename, 'w', newline='') as csvfile:
        writer = csv.writer(csvfile)
        writer.writerow(['Key', 'Value'])  # Write header row

        for key, value in dictionary.items():
            writer.writerow([key, value])  # Write key-value pairs

#dictionary_to_csv(rounded_distribution1, 'X dist csv')

def csv_to_excel(csv_filename, excel_filename):
    # Read the CSV file
    with open(csv_filename, 'r') as csv_file:
        csv_reader = csv.reader(csv_file)
        data = list(csv_reader)

    # Create an Excel workbook
    workbook = Workbook()
    sheet = workbook.active

    # Write data from CSV to Excel sheet
    for row in data:
        sheet.append(row)

    # Save the Excel file
```

```
    workbook.save(excel_filename)

csv_to_excel('X_dist_csv', 'X_dist_table')
```

Python file used for optimizing waiting time

```
from math import factorial, exp, ceil

def probability_Wq_le_8(c, arrival_rate, service_rate):
    rho = arrival_rate / (c * service_rate)

    p_0_denominator = sum((c ** n) * (rho ** n) / factorial(n) for n in range(c))
    p_0_denominator += ((c ** c) / factorial(c)) * ((rho ** c) / (factorial(c) * (1 - rho)
    p_0 = 1 / p_0_denominator

    p_c = (c ** c) * (rho ** c) / factorial(c) * p_0

    return 1 - (p_c / (1 - rho)) + (p_c / (1 - rho)) *
    (1 - exp(-c * service_rate * (1 - rho) * 8))

def objective_function(c, arrival_rate, service_rate):
    return probability_Wq_le_8(c, arrival_rate, service_rate)

def constraint(c, arrival_rate, service_rate):
    return probability_Wq_le_8(c, arrival_rate, service_rate) - 0.95

arrival_rate = 3/8  # lambda
service_rate = 1    # mu

# Iterate over integer values of c (starting with the lowest stable c) and find the
optimal value
c = ceil(arrival_rate/service_rate)

while True:
    if objective_function(c, arrival_rate, service_rate) < 0.95:
        c+=1
    else:
        min_probability = objective_function(c, arrival_rate, service_rate)
        optimal_c = c
        break



print("Optimal value of c:", optimal_c)
print("Maximum probability of Wq <= 8:", min_probability)
```

5.4 Appendix D

Linear interpolation: Mathematical tool used to estimate points between two discrete data points, with the use of linear polynomials.
Cycle: Any interval of time that begins with the arrival of an order and ends the instant before the next order is received. (Winston)
Golden Section Search: The Golden Section Search is a numerical optimization algorithm used to find the minimum (or maximum) of a unimodal function within a given interval. The algorithm utilizes the golden ratio to efficiently narrow down the search interval. (Winston)
Marginal Analysis: Method used in areas like finance and operations research to assess cost or revenue changes by minimal variations.
M/M/c model: Comes from the Kendall's classification (A/B/C), where A is the arrival process, B the service time distribution and C is the number of (identical) servers. M/M/c means that the model follows a Markov (Poisson) arrival process, Markov (exponential) service time distribution and c is the number of servers (in our case, engineers) (van Leeuwaarden)