

---

# 40.012 MANUFACTURING AND SERVICE OPERATIONS

---

## Assignment 3

**Michael Hoon**  
1006617

June 4, 2024

## Question 1

(a)

Since we are working with discrete demand here, we will employ the optimal policy newsvendor model for discrete demand. We define the underage cost  $c_u$  as the cost per unit of unsatisfied demand, and the overage cost  $c_o$  to be the cost per unit of positive inventory remaining at the end of the project. Thus,  $c_u = \$10 - \$5 = \$5$  (lost profit from not selling a drink that could have been sold), and  $c_o = \$5 - \$2 = \$3$  (difference between the cost of producing a drink and the salvage value of an unsold drink).

In the continuous model, the optimal solution is the value of decision variable  $Q$  that makes the distribution equal to the critical ratio, defined as

$$\begin{aligned} F(Q^*) &= \frac{c_u}{c_u + c_o} \\ &= \frac{5}{5 + 3} \\ &= 0.625 \end{aligned}$$

In the discrete case however, the critical ratio will generally fall between two values of  $F(Q)$ , and the optimal solution procedure is to local the critical ratio between two values of  $F(Q)$  and choose the  $Q$  corresponding to the **higher value** (i.e. we need to find the smallest order quantity for which the cumulative probability equals or exceeds the critical ratio). The table of cumulative probability distribution is given below:

Piney Drinks	30	40	50	60	<b>70</b>	80	90
PDF	0.05	0.12	0.20	0.24	0.17	0.14	0.08
CDF	0.05	0.17	0.37	0.61	<b>0.78</b>	0.92	1.00

Table 1: Probability Distribution Tables

From Table 1, we can see that the smallest order quantity where the CDF exceeds the critical ratio  $F(Q^*) = 0.625$  is 0.78, which corresponds to 70 drinks. Thus, Piney Pineapple Drinks should plan on taking **70 drinks to the beach each day**.

(b)

Since we are now working with a normally-distributed demand with mean 60 and variance 81, demand is now a random variable with distribution  $D \sim \mathcal{N}(60, 81)$ . Using the same critical ratio as before, we obtain a standardized value of  $Z = 0.32$  from the Normal Distribution table. The optimal  $Q^*$  is then obtained by scaling:

$$\begin{aligned} Q^* &= \sigma Z + \mu \\ &= \sqrt{81} \times 0.32 + 60 \\ &= 62.88 \\ &\approx \boxed{63} \end{aligned}$$

Thus, they should plan on taking **63 drinks to the beach every day** assuming a normally-distributed demand.

(c)

If it is equally likely that they get orders between 30 and 90 units on Sundays, we are working with a Uniformly Distributed Demand, with  $D \sim U(30, 90)$ . Using the formula for the CDF of a Uniform Distribution,

$$F(x) = \begin{cases} 0, & x < 30 \\ \frac{x - 30}{90 - 30}, & 30 \leq x \leq 90 \\ 1, & x \geq 90 \end{cases}$$

To find the smallest order quantity where  $F(x)$  exceeds the critical ratio  $F(Q^*) = 0.625$ , we solve accordingly:

$$\begin{aligned} 0.625 &= \frac{Q^* - 30}{90 - 30} \\ Q^* &= 67.5 \\ &\approx \boxed{68} \end{aligned}$$

after rounding to the next upper nearest integer, we find that the **optimal number of Piney drinks to bring to the beach every day is 68**.

## Question 2

(a)

Considering the usual assumptions of the lot-size reorder point system, we define the following parameters:

1. Proportional Order Cost  $c = \$10$  per unit
2. Lead time  $\tau = 6$  months
3. Setup Cost  $K = \$50$  per order
4. Holding Cost  $h = Ic = 0.20 \times \$10 = \$2$  per unit
5. Stock-out Cost  $p = \$25$  per unit

The demand is uniformly distributed from 270 to 330, so we have that  $D \sim U(270, 330)$  for 6 months. We assume that the mean rate of demand is thus  $\lambda = 300 \times 2 = 600$  units per year. Defining  $G(Q, R)$  as the expected average annual cost of holding, setup, and shortages, we have:

$$G(Q, R) = h \left( \frac{Q}{2} + R - \lambda\tau \right) + \frac{K\lambda}{Q} + p \left( \frac{\lambda n(R)}{Q} \right) \quad (1)$$

Where  $n(R)$  is the expected number of stock-outs:

$$n(R) = \mathbb{E}(\max(D - R, 0)) = \int_R^\infty (x - R)f(x) dx \quad (2)$$

Since we have a uniformly distributed demand  $D$ , the PDF of the annual demand  $f(x)$  is given by:

$$f(x) = \begin{cases} \frac{1}{330 - 270} = \frac{1}{60}, & 270 \leq x \leq 330 \\ 0 & \text{otherwise} \end{cases}$$

and the CDF  $F(x)$  is given by:

$$F(x) = \begin{cases} 0 & x < 270 \\ \frac{x - 270}{60} & 270 \leq x \leq 330 \\ 1 & x > 330 \end{cases}$$

The objective now is to find the optimal order quantity  $Q^*$  and  $R^*$  that minimises  $G(Q, R)$ . Employing the iterative algorithm, we have that

$$\begin{aligned} Q_0 = \text{EOQ} &= \sqrt{\frac{2K\lambda}{h}} \\ &= \sqrt{\frac{2 \times 50 \times 600}{2}} \\ &= 100\sqrt{3} \\ &= 173.20508 \end{aligned}$$

Next, we need to find  $R_0$  from the equation

$$1 - F(R) = \frac{Qh}{p\lambda} \quad (3)$$

Using  $Q_0 = 100\sqrt{3}$ , we have:

$$\begin{aligned} 1 - F(R_0) &= \frac{100\sqrt{3} \times 2}{25 \times 600} \\ R_0 &= 60 \times \left( 1 - \frac{100\sqrt{3} \times 2}{25 \times 600} \right) + 270 \\ &= 328.614359 \end{aligned}$$

To find  $n(R_0)$ , we use Equation 2:

$$\begin{aligned} n(R_0) &= \int_{R_0}^{\infty} (x - R_0) f(x) dx \\ &= \int_{R_0}^{330} \frac{x - R_0}{60} dx \\ &= 0.016 \end{aligned}$$

with this, we can find  $Q_1$  with the equation:

$$Q = \sqrt{\frac{2\lambda[K + pn(R)]}{h}} \quad (4)$$

$$\begin{aligned} Q_1 &= \sqrt{\frac{2 \times 600 \times [50 + 25 \times 0.016]}{2}} \\ &= 12\sqrt{210} \\ &= 173.89652 \end{aligned}$$

for  $R_1$ , we repeat with Equation 3

$$\begin{aligned} 1 - F(R_1) &= \frac{Q_1 h}{p\lambda} \\ R_1 &= 60 \times \left( 1 - \frac{21\sqrt{210} \times 2}{25 \times 600} \right) + 270 \\ &= 328.60882 \end{aligned}$$

Comparing the obtained values, we have

$$\begin{array}{ll} Q_0 = 173.20508 & R_0 = 328.614359 \\ Q_1 = 173.89652 & R_1 = 328.60882 \end{array}$$

we see that the values are very similar, thus we can conclude that the algorithm has converged, and the optimal order quantity  $Q^*$  and reorder point  $R^*$  is  $(Q_1, R_1) \approx (174, 329)$ .

**(b)**

We consider a Type I service level in a  $(Q, R)$  system, where probability of not stocking out during the order arrival lead-time is given by  $\alpha$ . Since  $\alpha$  completely determines  $R^*$ , we can decouple the computations of  $Q^*$  and  $R^*$ . Since this is the proportion of order cycles without stockouts, we have  $P(D \leq R) = F(R)$ . To find  $\alpha$ , we just set  $F(R) = \alpha$  as follows:

$$\begin{aligned}\alpha &= F(R_1) \\ &= \frac{329 - 270}{60} \\ &= \frac{59}{60} \\ &\approx \boxed{0.983}\end{aligned}$$

The probability of not stocking out during the order arrival lead-time is thus around 98.3%.

**(c)**

We now consider a Type II service, where the proportion of demand that is met from stock is defined as  $\beta$  (Fill Rate). Since  $n(R)/Q$  is the average fraction of demands that stock out each cycle, then we have that

$$\frac{n(R)}{Q} = 1 - \beta \tag{5}$$

Using Equation 5, we have

$$\begin{aligned}\beta &= 1 - \frac{n(R_1)}{Q_1} \\ &= 1 - \frac{\int_{329}^{330} \left( \frac{x - 329}{60} \right) dx}{174} \\ &= 0.999952107\end{aligned}$$

Thus, around 99.995% of the demand is met from stock.

### Question 3

If the unmet demand is lost, we need to consider the extension to a *lost sales* model. Excess demand cannot be back-ordered here, so we have the following updated equations:

$$T = \frac{(Q + \mathbb{E}[(D - R)^+])}{\lambda} \quad (6)$$

$$s = R - \lambda\tau + n(R) \quad (7)$$

The cost function is given as

$$G(Q, R) = h \left( R - \lambda\tau + n(R) + \frac{Q}{2} \right) + \frac{K\lambda}{Q} + c\lambda + \frac{pn(R)\lambda}{Q} \quad (8)$$

taking the derivative, we have

$$\begin{aligned} \frac{\partial G(Q, R)}{\partial R} &= h + \left( h + \frac{p\lambda}{Q} \right) \frac{dn}{dR} = 0 \\ 1 - F(R) &= \frac{Qh}{Qh + p\lambda} \end{aligned} \quad (9)$$

With Equation 9 and  $Q_0 = \text{EOQ} = 100\sqrt{3}$ , we can now iteratively compute the optimal order quantity and reorder point:

$$\begin{aligned} 1 - F(R) &= \frac{100\sqrt{3} \times 2}{100\sqrt{3} \times 2 + 25 \times 600} \\ R_0 &= 60 \times \left( 1 - \frac{100\sqrt{3} \times 2}{100\sqrt{3} \times 2 + 25 \times 600} \right) + 270 \\ &= 328.645637 \end{aligned}$$

To find  $n(R_0)$ , we use Equation 2:

$$\begin{aligned} n(R_0) &= \int_{R_0}^{\infty} (x - R_0) f(x) dx \\ &= \int_{R_0}^{330} \frac{x - R_0}{60} dx \\ &= 0.0152858 \end{aligned}$$

with this, we can find  $Q_1$  with Equation 4:

$$\begin{aligned} Q_1 &= \sqrt{\frac{2\lambda[K + pn(R)]}{h}} \\ &= \sqrt{\frac{2 \times 600[50 + 25 \times 0.0152858]}{2}} \\ &= 173.86571 \end{aligned}$$

for  $R_1$ , we repeat with Equation 9

$$\begin{aligned}
 1 - F(R_1) &= \frac{Q_1 h}{Q_1 h + p\lambda} \\
 R_1 &= 60 \times \left( 1 - \frac{173.86571 \times 2}{173.86571 \times 2 + 25 \times 600} \right) + 270 \\
 &= 328.640588
 \end{aligned}$$

With this, we can compare the following values:

$$\begin{array}{ll}
 Q_0 = 173.20508 & R_0 = 328.645637 \\
 Q_1 = 173.86571 & R_1 = 328.640588
 \end{array}$$

we see that the values are relatively close, and we can conclude that the algorithm has converged and we terminate here. Thus, the optimal order quantity and reorder point for the case where unmet demand is lost is  $(Q^*, R^*) = (Q_1, R_1) \approx (174, 329)$  (rounded to nearest integer).



## Question 4

To determine a more accurate formulation for the optimal lot size, we start from Equation 3, where

$$\begin{aligned} 1 - F(R) &= \frac{Qh}{p\lambda} \\ p &= \frac{Qh}{\lambda(1 - F(R))} \end{aligned} \quad (10)$$

where  $p$  here is also known as the imputed shortage cost. We substitute Equation 10 into Equation 4 to get:

$$Q = \sqrt{\frac{2\lambda\{K + Qhn(R)/[\lambda(1 - F(R))]\}}{h}}$$

which is a quadratic equation in  $Q$ . Solving for the positive root, we get

$$Q = \frac{n(R)}{1 - F(R)} + \sqrt{\frac{2K\lambda}{h} + \left(\frac{n(R)}{1 - F(R)}\right)^2} \quad (11)$$

which is known as the Service Level Order Quantity (SOQ Formula). We use Equation 11 with the following equation

$$n(R) = (1 - \beta)Q \quad (12)$$

and solve simultaneously to obtain the optimal values of  $(Q, R)$ , satisfying a type II service constraint. Since we would like to achieve a 98.5% fill rate,  $\beta = 0.985$ , from Equation 5 we have the relation

$$\begin{aligned} n(R) &= (1 - 0.985)Q \\ &= 0.015Q \end{aligned}$$

Using this, we now solve for  $n(R_0)$  by first setting  $Q_0 = \text{EOQ} = 100\sqrt{3}$ :

$$\begin{aligned} n(R_0) &= 0.015 \times 100\sqrt{3} \\ &= 1.5\sqrt{3} \end{aligned}$$

Similarly,

$$\begin{aligned} 1 - F(R_0) &= \frac{Q_0h}{p\lambda} \\ &= \frac{100\sqrt{3} \times 2}{25 \times 600} \\ &= \frac{\sqrt{3}}{75} \end{aligned}$$

To find  $R_0$ , we use the Equation 2:

$$\begin{aligned} n(R_0) &= \int_{R_0}^{\infty} \frac{x - R_0}{60} dx \\ 1.5\sqrt{3} &= \frac{1}{60} \left| \frac{x^2}{2} - R_0x \right|_{R_0}^{330} \\ 90\sqrt{3} &= \frac{330^2}{2} - 330R_0 - \frac{R_0^2}{2} + R_0^2 \end{aligned}$$

Solving quadratically, we obtain the result

$$R_0 = \mathbf{312.34301}$$

$$R_0 = 347.65698$$

The value 347.65 is **rejected** as it is greater than the demand 330. We can also use Equation 11 to obtain  $Q_1$ :

$$\begin{aligned} Q_1 &= \frac{1.5\sqrt{3}}{\sqrt{3}/75} + \sqrt{\frac{2 \times 50 \times 600}{2} + \left(\frac{1.5\sqrt{3}}{\sqrt{3}/75}\right)^2} \\ &= 319.033895 \end{aligned}$$

Since this significantly differs from  $Q_0$ , we continue the algorithm. To find  $n(R_1)$ , we have

$$n(R_1) = 0.015Q_1$$

Similarly,

$$\begin{aligned} 1 - F(R_1) &= \frac{Q_1 h}{p\lambda} \\ &= \frac{319.033895 \times 2}{25 \times 600} \\ &= 0.04253785 \end{aligned}$$

To find  $R_1$ , we use use Equation 2

$$\begin{aligned} n(R_1) &= \int_{R_1}^{330} \frac{x - R_1}{60} dx \\ 0.015(319.033895) &= \frac{1}{60} \left| \frac{x^2}{2} - R_1 x \right|_{R_0}^{330} \\ 60 \times 0.015(319.033895) &= \frac{330^2}{2} - 330R_1 - \frac{R_1^2}{2} + R_1^2 \end{aligned}$$

Solving quadratically, we get the values

$$R_1 = \mathbf{306.036256}$$

$$R_1 = 353.963743$$

Similarly, we reject the value 353.963743. We can use Equation 11 to obtain  $Q_2$ :

$$\begin{aligned} Q_2 &= \frac{0.015(319.033895)}{0.04253785} + \sqrt{\frac{2 \times 50 \times 600}{2} + \left(\frac{0.015(319.033895)}{0.04253785}\right)^2} \\ &= 319.033906 \end{aligned}$$

To find  $R_2$ , we similarly have

$$\begin{aligned} n(R_2) &= \int_{R_2}^{330} \frac{x - R_2}{60} dx \\ 0.015(319.033906) &= \frac{1}{60} \left| \frac{x^2}{2} - R_2 x \right|_{R_0}^{330} \\ 60 \times 0.015(319.033906) &= \frac{330^2}{2} - 330R_2 - \frac{R_2^2}{2} + R_2^2 \end{aligned}$$

Solving quadratically, we obtain the values

$$R_2 = \mathbf{306.036255}$$

$$R_2 = 353.963744$$

Similarly, the value 353.963744 is rejected. Lastly, comparing the different values obtained, we have

$$Q_0 = 173.20508$$

$$R_0 = 312.34301$$

$$Q_1 = 319.033895$$

$$R_1 = 306.036256$$

$$Q_2 = 319.033906$$

$$R_2 = 306.036255$$

we see that the values for  $Q_1, Q_2$  and  $R_1, R_2$  are relatively close, and we can conclude that the algorithm has converged, so we terminate here. Thus, the optimal order quantity and reorder point for the case where we want to achieve 98.5% fill rate with the accurate formulation is  $(Q^*, R^*) = (Q_2, R_2) \approx (319, 306)$  (rounded to nearest integer). For a sanity check, we calculate the fill rate  $\beta$ :

$$\begin{aligned} \beta &= 1 - \frac{n(R)}{Q} \\ &= 1 - \frac{\int_{306}^{330} \left( \frac{x-306}{60} \right) dx}{319} \\ &= 0.9849529 \\ &\approx 0.985 \end{aligned}$$

which indeed corresponds to a 98.5% fill rate.

## Question 5

Multi-Echelon Inventory Management (MEIM) systems are crucial in optimising inventory levels (across multiple stages) of a supply chain to minimise overall costs while meeting customer demand efficiently. Instead of studying a classic paper on MEIM, we will conduct a short literature review on **more recent transformative and novel research** being done in this domain.

The paper titled "Multi-echelon inventory optimization using deep reinforcement learning" [1] published in the Central European Journal of Operations Research, studies the applicability of a **Deep Reinforcement Learning (DRL)** approach to different MEIM systems, with the objective of **minimizing the holding and backorder costs**. Recent advancements in IT and data exchange have also allowed for more sophisticated multi-echelon inventory optimization techniques. However, these optimisation models can **quickly become very complex**, and rely heavily on assumptions and simplifications.

A more promising method introduced by the authors involves using Reinforcement Learning (RL), which was originally developed to solve sequential decision making problems in dynamic environments (including Robotics). Due to recent advances in Deep Learning, RL is often combined with neural networks to form DRLs. The authors **review existing literature** on MEIM and the application of RL here, noting that quite some research has already been done. They highlight the **complexities involved** in coordinating inventory policies across multiple locations and how DRL, particularly **Proximal Policy Optimization (PPO)**, can address these challenges.

The study focuses on three specific network structures: linear, divergent, and general (based on real-life manufacturer). The problem is modelled via a **Markov Decision Process (MDP)** where all decisions are made centrally by a single decision maker. The PPO algorithm is implemented to solve the MDPs in an **approximate manner**, and handle the continuous action space inherent in inventory management problems. The steps include state space transformation, action space transformation, and reward function transformation to tailor the PPO for inventory management tasks. The state space includes variables like inventory levels and demand rates, while the action space pertains to order quantities. The reward function is designed to **penalize holding and backorder costs**, driving the optimization process towards cost-effective inventory management (via hyperparameter tuning through computing a loss function). The results are as follows: with a **Linear Network**, the PPO algorithm achieved an average improvement of **16.4%** over benchmark solutions, the **Divergent Network** with an **11.3%** improvement, and **General Network** with a **6.6%** improvement. This demonstrates the effectiveness of the PPO algorithm in **reducing costs** and improving efficiency across different network structures, and how DRL can **adapt to various complexities and dynamics in a MEIM system**.

The author's contribution to this research was threefold. A DRL method was applied to a continuous action space (not yet widely used) to a MEIM system, creating **better scalability** to larger problems. Second, they investigated a general inventory system, which has received less attention in literature due to its complexity and intractability, and showed that the PPO algorithm can **improve inventory policies**. Lastly, they demonstrated the flexibility of the PPO algorithm by applying it to a variety of cases **without major modifications** and **without any parameter tuning**. They also noted that more research should be done on the **explainability of the method** and DRL in general, as it is a key component of implementing such methods in practice.

## Some Accompanying Images for Reference

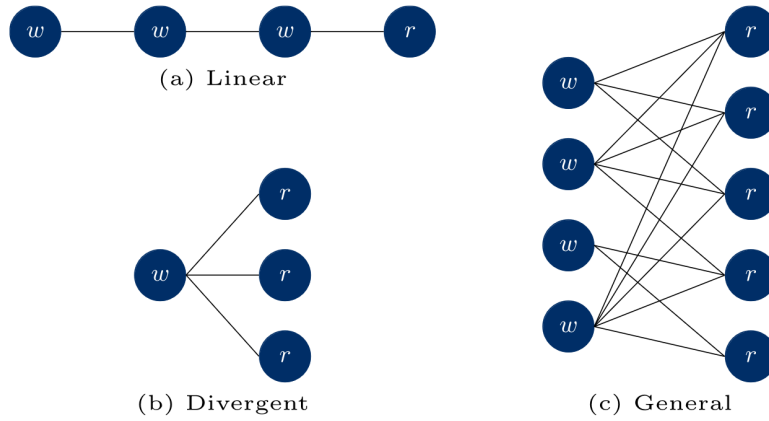


Figure 1: Three Types of Network Structures, with Warehouses  $w$  and Retailers  $r$

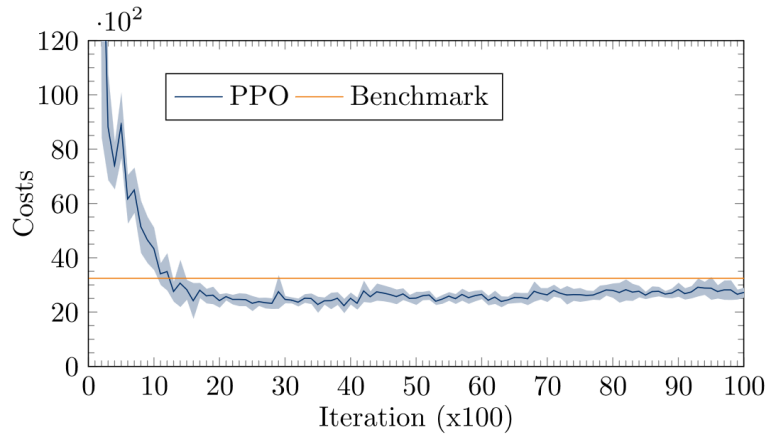


Figure 2: Performance of the PPO algorithm in the **linear** inventory system. The blue shade depicts the 95% confidence interval

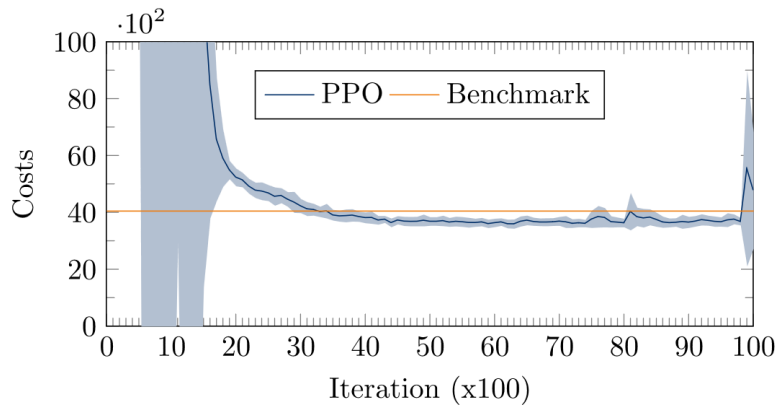


Figure 3: Performance of the PPO algorithm in the **divergent** inventory system. The blue shade depicts the 95% confidence interval

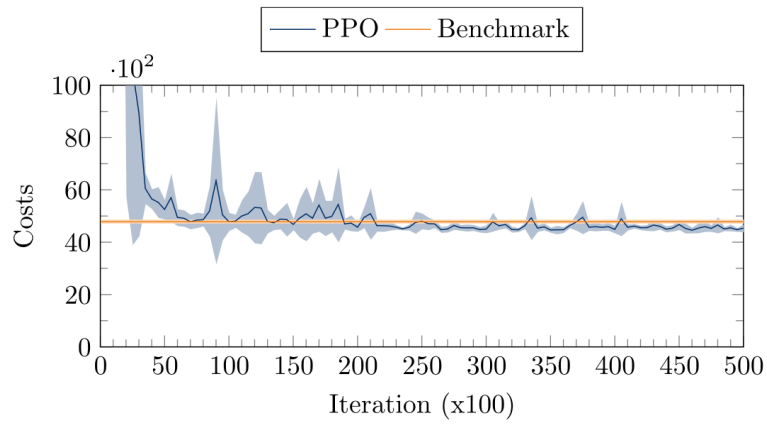


Figure 4: Results of the PPO algorithm on the case of manufacturing Company (**General Network**). The blue shade depicts the 95% confidence interval

## References

- [1] Kevin Gevers, Lotte van Hezewijk, and Martijn R. K. Mes. Multi-echelon inventory optimization using deep reinforcement learning. *Central European Journal of Operations Research*, July 2023.