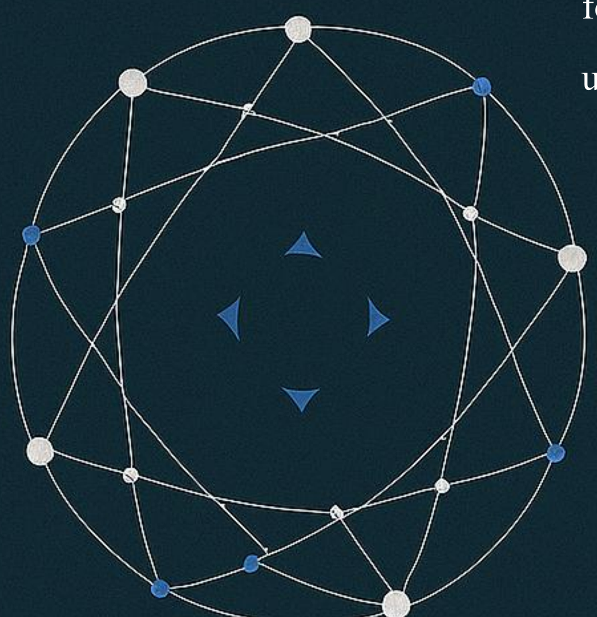
Applied Problem Solving Techniques

for Industrial Engineering

using Python



Assistant Professor
Bülent Şimşek

Istanbul Arel University

Course Notes

Contents

Applied Problem Solving Techniques

1. Introduction to Python for Industrial Engineering

a. Economic Order Quantity (EOQ) (Kumar & Suresh, 2008)

Inventory models address the management of idle resources such as labor, machinery, capital, and materials. These models primarily focus on two critical decisions: (i) determining the optimal order quantity (whether to purchase or produce), and (ii) identifying the appropriate time to place the order, with the overarching goal of minimizing total inventory-related costs.

Regarding the first decision—how much to order—two key cost components are considered: inventory carrying (or holding) costs and ordering (or acquisition) costs. As the order quantity increases, inventory carrying costs rise due to higher stock levels, while ordering costs decline because fewer orders are required. The term ‘order quantity’ refers to the amount of goods procured or produced within a single replenishment cycle.

The Economic Order Quantity (EOQ) represents the optimal order size that minimizes the total cost of inventory management by balancing carrying costs and ordering costs. Mathematically, the minimum total cost is achieved when:

$$Inventory\ Carrying\ Cost = Ordering\ Cost$$

There are two widely recognized methods for determining EOQ:

1. The Tabulation Method
2. The Algebraic Method

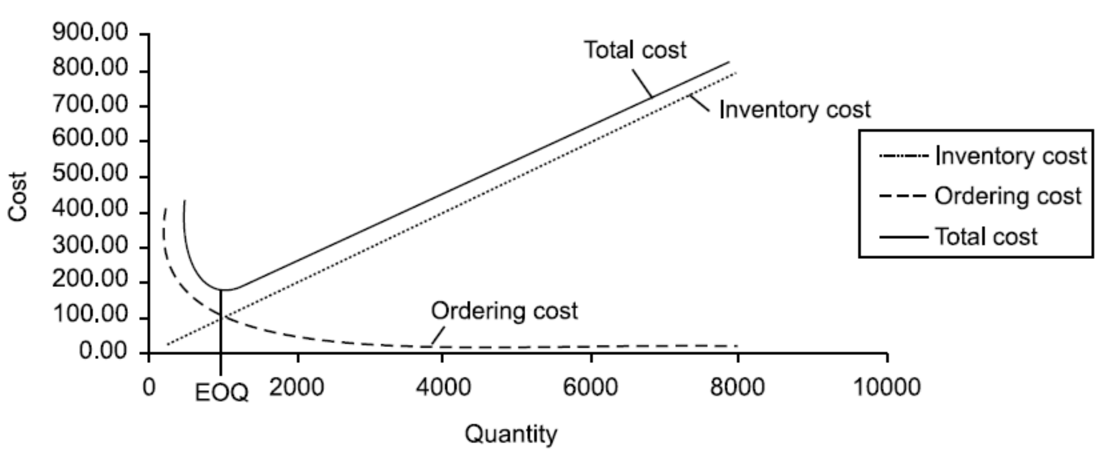


Figure 1-1 Inventory Cost Curve

i. Determination of Economic Order Quantity (EOQ) Using the Tabulation (Trial & Error) Method

This method provides a practical, step-by-step approach to identify the optimal order quantity by testing different lot sizes and evaluating their total cost implications. The following procedure is followed:

1. Select multiple possible lot sizes to be evaluated (e.g., 100 units, 200 units, 300 units, etc.).
2. Calculate the average inventory for each lot size, which is typically half of the order quantity.
3. Compute the inventory carrying cost by multiplying the average inventory with the unit holding cost.
4. Determine the ordering cost by dividing annual demand by the lot size and multiplying it by the cost per order.
5. Add the carrying and ordering costs to obtain the total cost for each lot size.
6. Identify the order quantity that results in the minimum total cost. This quantity is considered the Economic Order Quantity (EOQ).

The inventory cost and ordering cost curve is as shown in Figure 1-1, with respect to quantity ordered.

Example 1-1: EOQ Calculation – XYZ Ltd.

XYZ Ltd. is a large retailer offering a wide range of products to its customers. Among these products, one product is attracting attention from the company's inventory management perspective. Annual demand for this product is around 8,000 units.

The cost of each order placed to supply the product is $12.50. This cost includes various expenses such as order preparation, tracking, and logistics.

The unit price of the product is $1.00, and the annual carrying cost ratio for on-hand inventory is 20%. This ratio reflects expenses such as rent, insurance, and the risk of spoilage incurred when the product remains in storage.

XYZ Ltd.'s goal is to find the answer to the question of how many units per order for this product will minimize the total cost.

Solution:

The table 1-1 and the graph in figure 1-1 indicate that an order size of 1000 units will gives the lowest total cost among the different alternatives. It also shows that minimum total cost occurs when carrying cost is equal to ordering cost.

#n of orders / year	Lot Size	Average Inventory	Inventory Cost	Ordering Cost	Total Cost
(1)	(2)	(3)	(4)	(5)	(6)=(4)+(5)
1	8000	4000	800.00	12,50	812.50
2	4000	2000	400.00	25	425.00
4	2000	1000	200.00	50	250.00
8	1000	500	100.00	100	200.00
12	666.67	333.33	66.67	150	216.67
16	500	250	50.00	200	250.00

Table 1-1 Tabulated Solution

b. Determination of Economic Order Quantity (EOQ) by Analytical Method

The analytical method of determining EOQ is based on a set of idealized assumptions that allow for a closed-form solution. These assumptions include:

1. **Constant and known demand**: The demand rate remains uniform and predictable over the planning horizon.
2. **Lot size and total demand**: Let D represent the total annual demand (units per year), and Q be the order quantity per procurement or production run.
3. **No shortages allowed**: Inventory is replenished precisely when it reaches zero, ensuring continuous availability.
4. **Instantaneous replenishment**: The entire order quantity is received or produced at once, without delay.
5. **Zero lead time**: The time between placing and receiving the order is assumed to be negligible.
6. **Fixed ordering (or setup) cost**: Each procurement or production run incurs a fixed cost, denoted as C_3.
7. **Inventory carrying cost**: The annual holding cost per unit is denoted as $C_1 = C \times I$, where C is the unit cost and I is the inventory carrying rate (expressed as a percentage of the unit cost).

Under these conditions, the inventory level follows a saw-tooth pattern over time, as visualized in a typical inventory-time diagram. The vertical axis represents inventory level (Q), while the horizontal axis represents time. The total planning period (typically one year) is divided into n equal replenishment cycles.

This model forms the basis for deriving the classical EOQ formula, which minimizes the sum of ordering and holding costs.

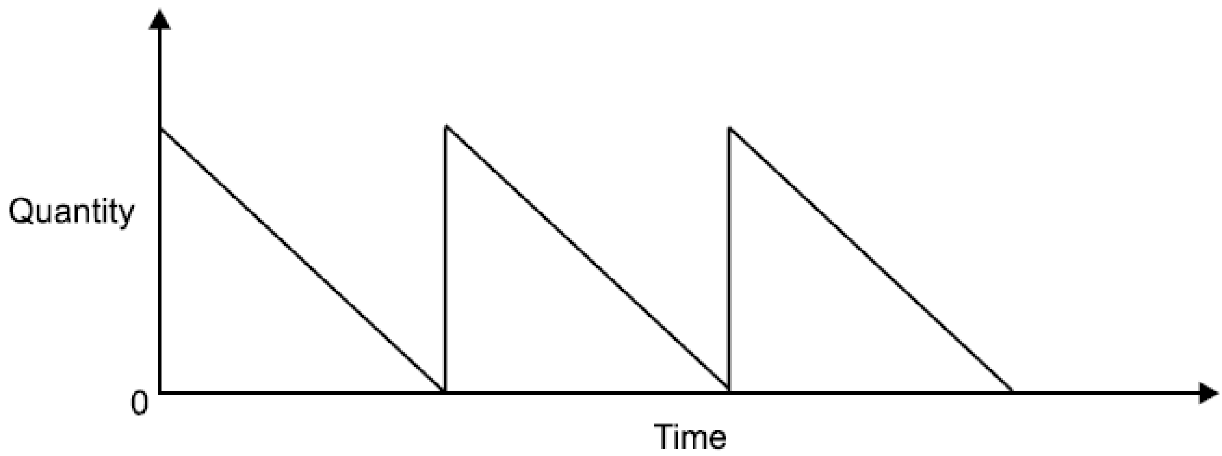


Figure 1-2 Quantity over Time

The most economic point in terms of total inventory cost exists where,

Inventory carrying cost = Annual ordering cost (set-up cost)

Average inventory = 1/2 (maximum level + minimum level)

$$= \frac{Q\ +\ 0}{2} = \frac{Q}{2}$$

Total inventory carrying cost = Average inventory × Inventory carrying cost per unit

i.e., Total inventory carrying cost $= \frac{Q}{2} \times C_1 = \frac{QC_1}{2}$ (1)

Total annual ordering costs = Number of orders per year × Ordering cost per order

Total annual ordering costs $= (D/Q) \times C\_3 = \left(\frac{D}{Q}\right) C_3$ (2)

Now, summing up the total inventory cost and the total ordering cost, we get the total inventory cost C(Q).

i.e., Total cost of production run = Total inventory carrying cost + Total annual ordering costs

$C(Q) = QC\_1/2 + \left(\frac{D}{Q}\right) C_3$ (cost equation) (3)

But, the total cost is minimum when the inventory carrying costs becomes equal to the total annual ordering costs. Therefore,

$$\frac{QC_1}{2} = \left(\frac{D}{Q}\right) C_3$$

or $QC_1 = \left(\frac{2D}{Q}\right) C_3$ or $Q^2 = \frac{2C_3D}{C_1}$

or $$Q = \sqrt{\frac{2C_3D}{C_1}}$$

i.e., Optimal quantity (EOQ) $$Q_0 = \sqrt{\frac{2C_3D}{C_1}} \tag{4}$$

Optimum number of orders, (N_0) = D/Q_0 (5)

Optimum order interval, (t_0) = $\frac{365}{N_0}$ in days = $\frac{1}{N_0}$ in years or $t_0 = \frac{Q_0}{D}$ (6)

Average yearly cost (TC) = $\sqrt{2C_3DC_1}$ (7)

c. Python Basics: Assignment, Conditionals, and Loops

In Python, basic programming structures such as assignment statements, conditional logic, and loops form the foundation of algorithmic thinking. These elements are essential for automating decision-making and repetitive tasks, especially in data-driven fields like Industrial Engineering.

i. Variables (Das, Lawson, Mayfield, & Norouzi, 2024)

In Python, variables are used to store data that can be manipulated throughout the program. The assignment operator = assigns a value to a variable. Once assigned, variables can be used to perform arithmetic operations such as addition, subtraction, multiplication, and division.

Python uses dynamic typing, so you do not need to declare variable types. Variable names should be meaningful and reflect the context (e.g., daily_demand, unit_cost).

- Assignment Operator (=)

```
x = 10          # Assigns integer value 10 to variable x
name = "Ali"    # Assigns a string to the variable name
```

- Addition (+)

```
a = 5
b = 3
total = a + b     # total = 8
```

- Subtraction (—)

```
stock = 50
used = 12
remaining = stock - used    # remaining = 38
```

- Multiplication (×)

```
unit_cost = 2.5
quantity = 100
total_cost = unit_cost * quantity    # total_cost = 250.0
```

- Division (÷)

```
total_items = 800
box_size = 50
number_of_boxes = total_items / box_size    # number_of_boxes = 16.0
```

ii. Conditional Statements (if, elif, else)

Conditional statements are used to perform different actions based on different conditions.

```
stock = 40
critical_level = 50

if stock < critical_level:
    print("Reorder needed.")
elif stock == critical_level:
    print("Stock is at critical level.")
else:
    print("Stock level is sufficient.")
```

iii. Looping Structures

- While Loop

Repeats a block of code as long as the condition is True.

```
day = 1
while day <= 5:
    print(f"Day {day}: Process running...")
    day += 1
```

- For Loop

Used to iterate over a sequence (e.g., list, range).

```
for i in range(1, 6):    # from 1 to 5, not 6
    print(f"Iteration {i}")
```

Example – Printing a cost table:

```
order_sizes = [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800]

for Q in order_sizes:
    avg_inventory = Q / 2
    ordering_cost = (8000 / Q) * 12.5
    carrying_cost = avg_inventory * 0.2
    total_cost = ordering_cost + carrying_cost
    print(f"Q: {Q}, Total Cost: {round(total_cost,2)}")
```

What is f-string (formatted string)?

The values of variables are placed directly into the string defined by the f"..." notation using curly brackets { }.

Example:

```
day = 3
stock = 52
print(f"Day {day}: Stock = {stock}")
```

How would it be written without the f-string?

```
# 1. Concatenation with the + operator
print("Day " + str(day) + ": Stock = " + str(stock))

# 2. with the format() function
print("Day {}: Stock = {}".format(day, stock))
```

Bibliographies

Das, U., Lawson, A., Mayfield, C., & Norouzi, N. (2024). *Introduction to Python Pogramming*. Houston: openstax.

Kumar, S. A., & Suresh, N. (2008). *Production and Operations Management*. New Delhi: New Age International Limited Publishers.