Inventory System Simulation

MD. Irfanur Rahman Rafio
Department of Computer Science and Engineering
Islamic University of Technology
Dhaka, Bangladesh
ID: 190041125
irfanurrahmanrafio@gmail.com

***Abstract*—Inventory System simulation is a basic simulation model that can be used to compare alternative ordering policies for an Inventory system. Although simplistic, many of the elements of this model are representative of those found in actual inventories. In this experiment, we simulate an Inventory System of our own and calculate its results for different given policies. After that, we analyze the policies to determine the optimal one.**

***Index Terms*—Inventory, Supply, Demand, Evaluation**

I. System Description

A. Problem Statement

A company that sells a single product would like to decide how many items it should have in Inventory for each of the next n months (n is a fixed input parameter). The times between demands are IID exponential random variables with a mean of 0.1 month. The sizes of the demands, D, are IID random variables (independent of when the demands occur), with

$$D = \begin{cases} 1, & \text{with probability } 1/6 \\ 2, & \text{with probability } 1/3 \\ 3, & \text{with probability } 1/3 \\ 4, & \text{with probability } 1/6 \end{cases}$$

At the beginning of each month, the company reviews the Inventory level and decides how many items to order from its supplier. If the company orders Z items, it incurs a cost of K+iZ, where K=$32 is the setup cost and i=$3 is the incremental cost per item ordered. (If Z=0, no cost is incurred.) When an order is placed, the time required for it to arrive (called the delivery lag or lead time) is a random variable that is distributed uniformly between 0.5 and 1 month. The company uses a stationary (s, S) policy to decide how much to order, i.e.,

$$Z = \begin{cases} S - I, & I < S \\ 0, & I \geq S \end{cases}$$

where I is the Inventory level at the beginning of the month. When a demand occurs, it is satisfied immediately if the Inventory level is at least as large as the demand. If the demand exceeds the Inventory level, the excess of demand over supply is backlogged and satisfied by future deliveries. (In this case, the new Inventory level is equal to the old Inventory level minus the demand size, resulting in a negative Inventory level.) When an order arrives, it is first used to eliminate as much of the backlog (if any) as possible; the remainder of the order (if any) is added to the Inventory.

So far, we have discussed only one type of cost incurred by the Inventory system, the ordering cost. However, most real Inventory systems also have two additional types of costs, holding and shortage costs, which we discuss after introducing some additional notation. Let I(t) be the Inventory level at time t [I(t) could be positive, negative, or zero]; let $I^+(t) = max(I(t), 0)$ be the number of items physically on hand in the Inventory at time t [$I^+(t) \geq 0$]; and let $I^-(t) = max(-I(t), 0)$ be the backlog at time t [$I^-(t) \geq 0$]. For our model, we shall assume that the company incurs a holding cost of h=$1 per item per month held in (positive) Inventory. The holding cost includes such costs as warehouse rental, insurance, taxes, and maintenance, as well as the opportunity cost of having capital tied up in Inventory rather than invested elsewhere. We have ignored in our formulation the fact that some holding costs are still incurred when $I^+(t) = 0$. However, since our goal is to compare ordering policies, ignoring this factor, which after all is independent of the policy used, will not affect our assessment of which policy is best.

B. Input Data

Assuming that initially, the Inventory is full, and no order is outstanding. We simulate the Inventory system for n=120 months and use the average total cost per month (which is the sum of the average ordering cost per month, the average holding cost per month, and the average shortage cost per month) to compare the following nine Inventory policies:

TABLE I
List of Policies

Policy	Order Threshold, s	Inventory Capacity, S
1	20	40
2	20	60
3	20	80
4	20	100
5	40	60
6	40	80
7	40	100
8	60	80
9	60	100

C. State Variables

The state of the system in a certain point of time t is represented by the Inventory Level I(t).

D. State Space

The Inventory level can be any integer less than or equal to the Inventory capacity, S.

$$\text{State Space, } X = (-\infty, S] \cap \mathbb{Z}$$

E. Event Set

The system has four events: Evaluation(e), Demand(d), Supply(s) and Termination(t).

$$\text{Event Set, } E = \{a, d, s, t\}$$

F. Feasible Event Set

Any event is feasible in any point of time.

$$\text{Feasible Event Set, } F(t) = E$$

G. State Equations

The state equations of the system:

$$I(t^+) = \begin{cases} I(t) + Z, & \text{supply arrives at time t} \\ I(t) - D, & \text{demand arrives at time t} \\ I(t), & \text{otherwise} \end{cases}$$

H. Statistical and Output Variables

Time-average (per month) number of items held in Inventory,

$$\bar{I^+}(t) = \frac{\int_0^n I^+(t)dt}{n}$$

Time-average (per month) number of items in backlog,

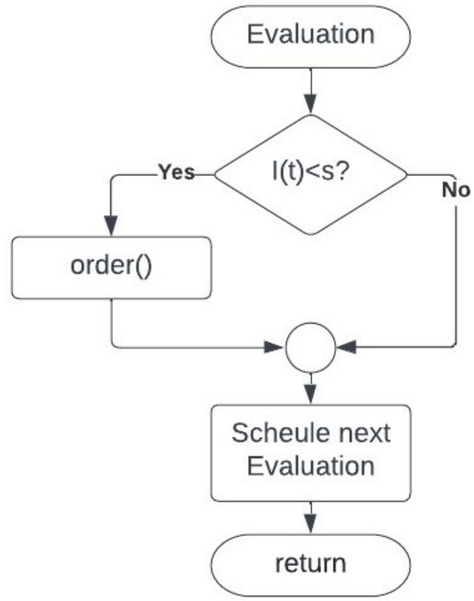
$$\bar{I^-}(t) = \frac{\int_0^n I^-(t)dt}{n}$$

I. Output Equations

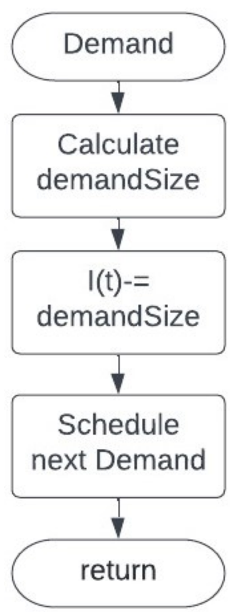
$$\text{Average holding cost/month} = h\bar{I^+}(t) \text{Average backlog cost/month} = \pi \bar{I^-}(t)$$

J. Event Routines

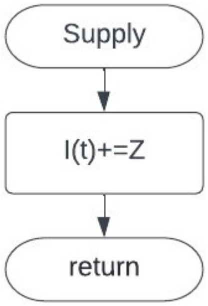
1) Evaluation Event: The Evaluation event does not affect the system state directly. It evaluates whether an order is to be placed. If the Inventory level is lower than the order threshold at the beginning of the month, Evaluation event triggers an order placement.



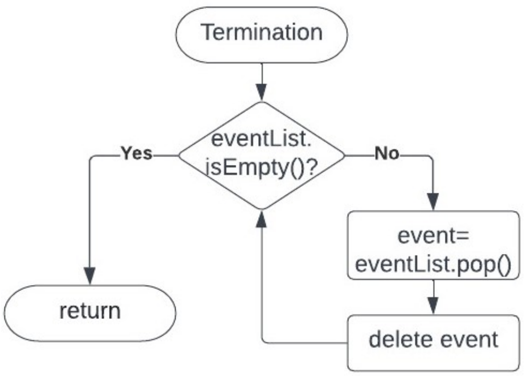
2) Demand Event: The Demand event decreases the Inventory level by the demand amount which is determined by a discrete random variable. The demand interval time is random and a demand may make the Inventory level negative.



3) Supply Event: Supply event represents the arrival of the placed order. It clears the backlog and increases the Inventory level by the supply amount. The arrival takes random time after an order is placed.

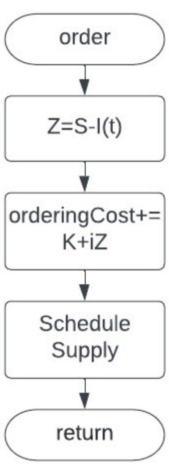


4) Termination Event: When triggered, this event deletes all the other events from the simulation without giving them any chance to call the handle function.

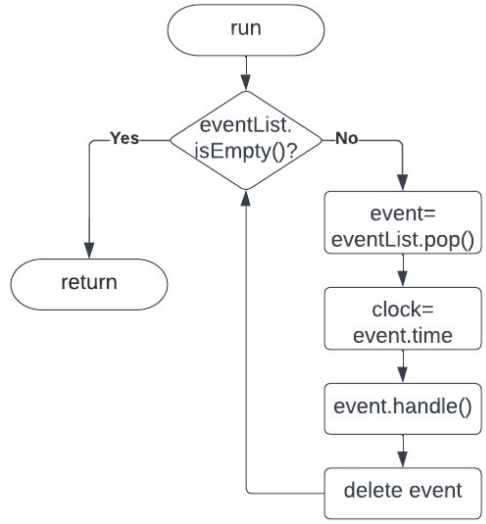


II. Simulation Program Description

The program consists of three major classes: Simulator, Inventory and Event. Minor classes include a min Heap.
The event class is abstract. Its child classes represent different types of event. Each of those classes have a handler function which manipulates the state of the Inventory.
The Inventory class consists of the system variables, and the statistical variables. Handling an event involves updating the statistical variables and creating changes to the system variables. There is a utility function in the Inventory class that performs the task of ordering items.



Finally, we have the Simulator class which mainly comprises of the event list and system clock. The event list is a min heap where the upcoming events are stored. The Simulator has a run function which pops an event from the event list and handles it. In every occurrence of a new event, the clock is updated.



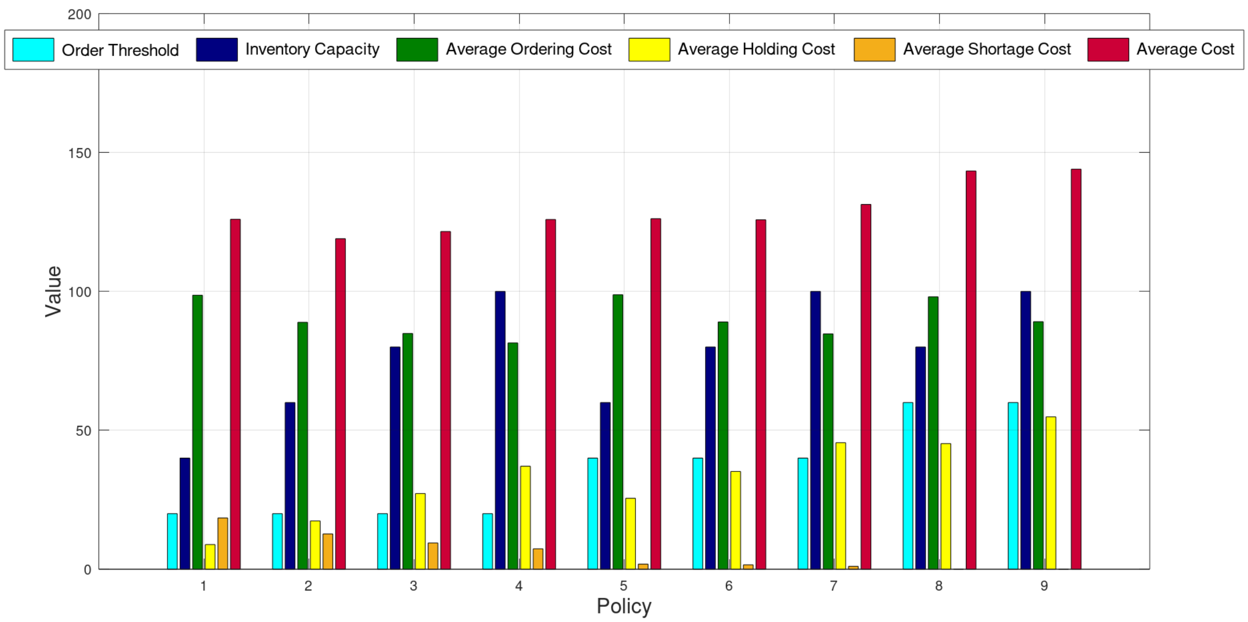
The full source code of the simulation program can be found here: https://github.com/rafio-iut/Simulation-and-Modeling-Lab/tree/main/Inventory%20System

III. Result

TABLE II
Inventory System Simulation Results

Policy	Order Threshold	Inventory Capacity	Average Ordering Cost	Average Holding Cost	Average Shortage Cost	Average Cost
1	20	40	98.6344	8.88142	18.4599	125.976
2	20	60	88.8767	17.3914	12.7115	118.98
3	20	80	84.8656	27.2403	9.46335	121.569
4	20	100	81.4678	37.0715	7.35383	125.893
5	40	60	98.8047	25.5323	1.84258	126.18
6	40	80	89.0067	35.188	1.59473	125.789
7	40	100	84.7181	45.5521	1.03715	131.307
8	60	80	98.0631	45.2199	0.0778331	143.361
9	60	100	89.1067	54.8553	0.0563413	144.018

The simulation results are taken by running the simulation 30 times for each policy. A bar graph is generated for visualization.



IV. Conclusion

From the simulation, we can come to the decision that policy 2 (s=20, S=60) is the best policy in terms of minimum average cost.