ZENITH STUDY HUB

Modeling and Simulation

UNIT II | SYSTEM TYPES

Classifications, discrete-event logic, continuous simulation, Monte Carlo, hybrid systems and engineering numericals

SPPU TE Robotics & Automation (2024 Pattern) | PEC321-ROA (B)

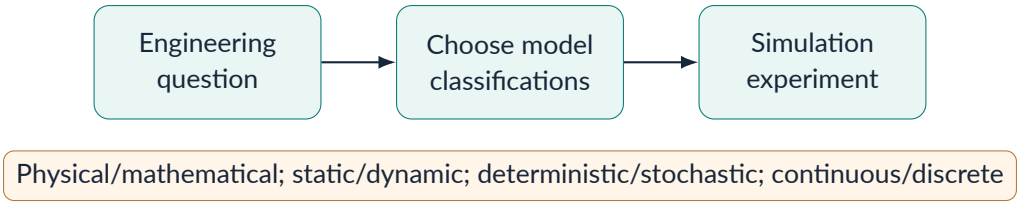

Figure 1. Original concept diagram: the problem determines the modeling technique. The categories describe different axes, so a system can belong to several at once.

Exact syllabus basis: SPPU 2024 Pattern *TE Syllabus.pdf*, PEC321-ROA (B), Unit II, physical PDF p.26. Supporting course files read from Drive: *M&S Unit 2 PPT-1.pdf* and *M&S Unit 2 Question Bank.pdf*. The customer data below come from the supplied question bank, while clearly labelled additional exercises use synthetic teaching inputs.

Author / learning edition: Om Deshmukh | 21 November 2026 | Visual and arithmetic review candidate

One-page syllabus coverage map

Exact official area	Where to learn it in this guide
Physical vs mathematical models	Classification with example and sketch
Static vs dynamic systems	Time dependence, Monte Carlo distinction
Deterministic vs stochastic systems	Inputs, reproducibility, uncertainty
Continuous vs discrete systems	State-change diagrams and equations
Discrete-event simulation	Event calendar, clock, state and queue
Continuous simulation	Differential equation and Euler example
Monte Carlo simulation	Sampling steps and inventory numerical
Hybrid simulation	Mobile-robot and battery example
Applications in inventory systems	Stock rule, lead time and cost
Applications in queueing systems	Full 20-customer worked exercises
System logic development	Arrival/departure routines, pseudocode
Event scheduling	Future-event list, timing routine, tie rule
System state variables	Queue length, busy flag and inventory state

In simple language
This unit answers three questions: **what sort of model do we need? how should the simulation clock advance?** and **how do we calculate useful outputs?** The classifications are not competing names for the same feature. A model may be mathematical, dynamic, stochastic and discrete-event at the same time.

Important notation rule. Every numerical example declares the time origin, random-digit mapping, service discipline and tie convention. When a college question does not specify a detail, an explicit assumption is better than silently inventing it.

1 Classification of Models and Simulation Systems

1.1 Physical model versus mathematical model

A **physical model** uses a real, scaled or simplified physical representation to study an engineering object, such as a 3D-printed prototype of a gripper. Its measured forces, size or motion may reveal issues not captured by a simple formula. A **mathematical model** expresses a system through equations, logic and data. A robot joint model might use

$$J\ddot{\theta} + b\dot{\theta} = \tau(t),$$

where J is rotational inertia, b is viscous damping, θ is position and τ is torque. Both are models; each omits features of the full industrial machine.

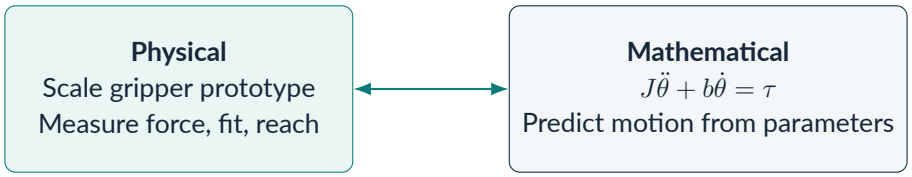

Figure 2. Original two-model comparison. A physical prototype is not automatically a digital simulation.

1.2 Static versus dynamic

A **static model** does not explicitly represent how a system evolves with time; it may calculate a snapshot or a one-period outcome. A **dynamic model** includes changes in state over time: inventory from day to day, queue length after events, or a moving robot's position.

Simple examples
Static: Sample a day's uncertain demand 1,000 times to estimate the probability of exceeding a fixed stock allowance. **Dynamic:** Follow inventory day by day, including backorders and three-day replenishment lead time. The college slides describe a static model as "also known as Monte Carlo simulation"; that is a classroom simplification, not a universal equivalence. **Monte Carlo is a sampling method that can also be used inside dynamic models.**

1.3 Deterministic versus stochastic

A deterministic model has no random inputs and gives the same result for the same initial condition and parameters. A stochastic model explicitly represents random variables (e.g., arrival times or failures), so independent runs can differ. A seeded computer generator can reproduce one stochastic sample path exactly; that does not make the underlying model deterministic.

Robot example. A robot that always handles one part in exactly 5 seconds is deterministic as modeled. If handling time follows a measured probability distribution and breakdowns occur randomly, it is stochastic. The real world can contain uncertainty even if the engineer chooses to omit it.

1.4 Discrete versus continuous state changes

A **continuous-state model** describes quantities that vary continuously, such as temperature, voltage or position. A **discrete-state model** describes countable state changes, such as the number of parts waiting in a queue. A discrete-event model changes state at selected event times; its simulation clock may jump directly from event to event. *Discrete state* and *fixed time step* are not the same concept.

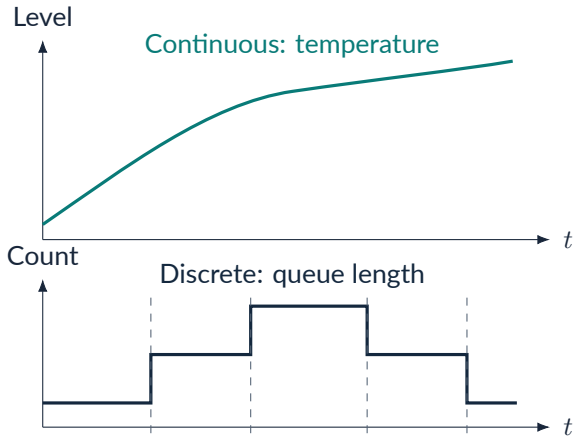

Figure 3. Original illustrative plots. Queue length remains constant between events and jumps at arrivals/departures; continuous temperature evolves smoothly in this simplified example.

1.5 Categories combine: a classification matrix

Engineering question	Useful classification
Predict joint temperature using an ODE	Mathematical, dynamic, continuous-state; possibly deterministic
Estimate defect risk with random sampling	Mathematical, often static, stochastic, Monte Carlo
Model repairable robot with random failures	Mathematical, dynamic, stochastic, discrete-event
Model AGV motion plus discrete jobs	Mathematical, dynamic, hybrid; may be stochastic

2 Discrete-Event Simulation (DES)

2.1 System state, event, activity and simulation clock

A **state variable** gives relevant information at model time t. For one service counter, a useful state is $S(t) = (Q(t), B(t))$, where $Q(t)$ is the number waiting and $B(t) = 1$ if the server is busy (otherwise 0). A job's ID, priority and arrival time are entity attributes; they should not be mistaken for the global queue length. **Events** update the state at an instant, such as an arrival, departure or repair completion. **Activities** span time, such as processing a part for four minutes. The **future-event list (FEL)** contains scheduled event notices and their timestamps.

2.2 Event scheduling: next-event advance

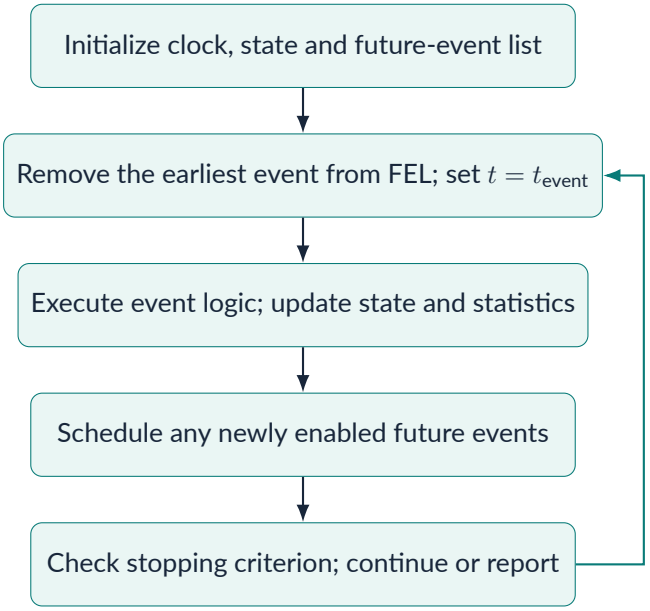

Figure 4. Original next-event executive. When the event list is empty, the model must terminate or correctly handle the declared boundary; it must not keep advancing to undefined times.

2.3 Next-event and fixed-increment advances

In the **next-event** method, the clock skips idle periods and moves to the next event's exact timestamp. In a **fixed-increment** method, it advances by Δt and checks for events. Naively rounding events to the grid can introduce timing errors. For an event at $t = 2.3$ s and $\Delta t = 1$ s, processing at $t = 3$ s is 0.7 s late. Interpolation or sub-step handling requires an explicitly implemented policy.

Event-list numerical trace

The first arrival occurs at 0 min, with service time 3 min. Second arrival at 2 min has service time 2 min; third arrival at 4 min has service time 1 min. Assume FIFO, nonpreemptive service, and departures processed before arrivals when tied.

Time	Event	State just after event	Next action
0	Arrival 1	Busy, $Q = 0$	Schedule departure at 3
2	Arrival 2	Busy, $Q = 1$	Wait in queue
3	Departure 1	Busy, $Q = 0$	Start job 2; depart at 5
4	Arrival 3	Busy, $Q = 1$	Wait in queue
5	Departure 2	Busy, $Q = 0$	Start job 3; depart at 6
6	Departure 3	Idle, $Q = 0$	No future jobs

Waiting times: job 1 = 0, job 2 = 1, job 3 = 1 minute, so mean waiting time is $2/3$ minute.

2.4 Correct program logic: arrival and departure

```
state: current_time, server_busy, FIFO_queue
FEL: timestamped arrival and departure events
on ARRIVAL(job):
    if server_busy:
        append job to FIFO_queue
    else:
        server_busy = True
        schedule DEPARTURE(now + service_time(job))
    schedule the next ARRIVAL, if the input stream continues
on DEPARTURE(job):
    record completion and waiting/flow times
    if FIFO_queue is not empty:
        next_job = pop_front(FIFO_queue)
        schedule DEPARTURE(now + service_time(next_job))
    else:
        server_busy = False
```

Check your understanding
No service starts while the only server is busy. Every departure is scheduled after its service starts. Never assume a queue is empty just because arrival times have gaps.

3 Queueing Systems: Complete Components

3.1 Calling population, arrival process and service

A **calling population** is the set of potential customers or parts. It may be finite (a fixed fleet of tools) or effectively infinite relative to one queue (a large external demand stream). An **arrival process** specifies when jobs arrive; some arrive individually, others in batches. An **interarrival time** is the time between successive arrivals, not the service duration. A **server** is a resource that processes jobs (human clerk, robot, machine, inspection station). There may be one server or many. **System capacity** includes waiting positions and servers; if finite, full-system arrivals need a defined rule: reject, defer, block upstream or reroute.

3.2 Queue discipline

FIFO/FCFS serves in arrival order; LIFO/LCFS serves the newest job; priority serves according to a priority field; SIRO chooses a waiting job at random. Models may also include *balking* (not entering a long queue) and *reneging* (leaving after waiting). Different queue disciplines can change waiting-time distributions without changing individual service times.

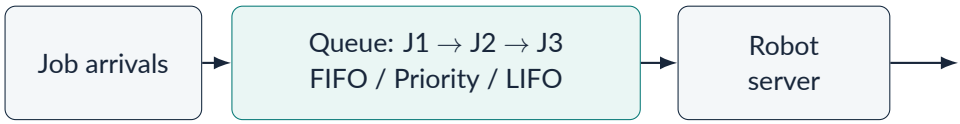

Figure 5. Original conceptual single-server queue. The discipline determines which waiting job is selected; it does not change when jobs arrived.

3.3 Time-weighted and job-weighted metrics

Mean waiting time is the average of individual job waits: $\bar{W}_q = \sum_i W_{q,i}/n$. The **time-average queue length** is the area under $Q(t)$ divided by the observation horizon: $\bar{L}_q = \frac{1}{T}\int_0^T Q(t)\,dt$. Those are different denominators and should not be swapped. Utilization for a single server over $[0, T]$ is busy time divided by T; the observation horizon and treatment of warm-up time must be stated.

4 Source Numerical A: Grocery Store, 20 Customers

The connected *M&S Unit 2 Question Bank.pdf*, Question 13, provides twenty customer records. The problem states interarrival times 1–8 minutes are equally likely and service times 1–6 minutes follow a six-value probability distribution shown below. We preserve its random digits exactly.

4.1 Step 1: convert random digits into times

For three-digit interarrival random numbers, equal probability $1/8$ assigns 125 digits to each interval:

RN range	Interarrival (min)	RN range	Interarrival (min)
000–124	1	500–624	5
125–249	2	625–749	6
250–374	3	750–874	7
375–499	4	875–999	8

For the two-digit service random numbers, the cumulative distribution gives:

Service (min)	Probability	RN range
1	0.10	00–09
2	0.20	10–29
3	0.30	30–59
4	0.25	60–84
5	0.10	85–94
6	0.05	95–99

4.2 Step 2: define the first arrival and recurrence

The question's first arrival has no interarrival digit. **Assumption: customer 1 arrives at** $t = 0$**.** All later arrival times follow $A_i = A_{i-1} + I_i$. For FIFO non-preemptive service, start $B_i = \max(A_i, D_{i-1})$, departure $D_i = B_i + S_i$, waiting $W_i = B_i - A_i$, and idle gap $G_i = \max(0, A_i - D_{i-1})$. Set $D_0 = 0$.

Hand trace: customers 2–4
Customer 2: RN 913 maps to 8-minute interarrival; $A_2 = 8$. RN 10 maps to 2-minute service; start 8, depart 10, wait 0.
Customer 3: RN 627 maps to 6-minute interarrival; $A_3 = 14$, service RN 74 maps to 4 minutes; start 14, depart 18.
Customer 4: RN 015 maps to 1-minute interarrival; $A_4 = 15$ while customer 3 is busy; starts at 18, finishes at 21, waits 3 minutes.

4.3 Step 3: reproduce all 20 records

The next four tables retain all customer random digits. First read the input/time-mapping columns, then follow the start/end/wait recurrence. *Service RN values are written as two digits and interarrival RN values as three digits to preserve leading zeros.*

#	IA RN	IA	Arr	S RN	S	Start	End	Wait
1	—	0	0	84	4	0	4	0
2	913	8	8	10	2	8	10	0
3	627	6	14	74	4	14	18	0
4	015	1	15	53	3	18	21	3
5	948	8	23	17	2	23	25	0
6	309	3	26	79	4	26	30	0
7	922	8	34	91	5	34	39	0
8	753	7	41	67	4	41	45	0
9	235	2	43	89	5	45	50	2
10	302	3	46	38	3	50	53	4
11	109	1	47	32	3	53	56	6
12	093	1	48	94	5	56	61	8
13	607	5	53	79	4	61	65	8
14	738	6	59	05	1	65	66	6
15	359	3	62	79	4	66	70	4
16	888	8	70	84	4	70	74	0
17	106	1	71	52	3	74	77	3
18	212	2	73	55	3	77	80	4
19	493	4	77	30	3	80	83	3
20	535	5	82	50	3	83	86	1

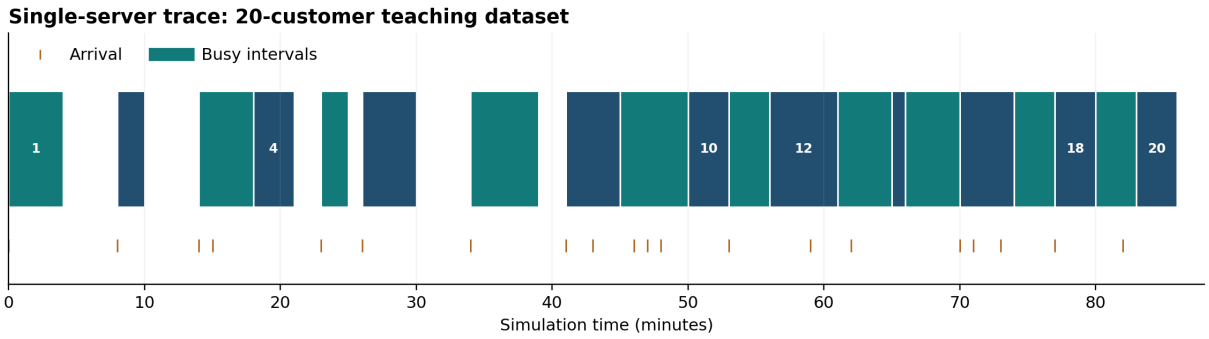

Figure 6. Original timeline computed from all 20 question-bank customers. Orange marks are arrival times; the single service resource is occupied only during coloured busy intervals.

Recomputed answer for the chosen first-arrival convention
Last departure $T = 86$ min. Total service/busy time $= 69$ min. Total idle time $= 17$ min. Sum of waiting times $= 52$ min. Mean waiting $= 52/20 =$ **2.60** min. Observed busy fraction $= 69/86 \approx$ **80.23%**; idle fraction $= 17/86 \approx$ **19.77%**. These figures are exact for the stated mapping and sample path, not a long-run expectation.

Check your understanding
Changing the first-arrival convention changes the observation horizon and can change the idle proportion. If the instructor specifies a different first-customer origin, recompute all timestamps consistently.

5 Source Numerical B: Garment Counter, 20 Customers

Question 14 provides a different discrete distribution and all twenty random digit pairs. The interarrival probabilities for 1–5 min are $35, 25, 20, 12, 8\%$; the service times are $1, 1.5, 2, 2.5, 3$ min with probabilities $20, 35, 25, 15, 5\%$.

5.1 Random digit mapping and timing convention

IA time	Prob.	RN	Service	Prob.	RN
1	.35	00–34	1.0	.20	00–19
2	.25	35–59	1.5	.35	20–54
3	.20	60–79	2.0	.25	55–79
4	.12	80–91	2.5	.15	80–94
5	.08	92–99	3.0	.05	95–99

Question 14 includes a first-customer interarrival digit (48), unlike Question 13. We begin at $t = 0$ and sample the first interval: customer 1 arrives at 2 min, receives 1.5 min service (RN 22) and departs at 3.5 min. Customer 2 arrives at 4 min and so the clerk is idle for another 0.5 min. The initial idle interval must be counted when reporting utilization from $t = 0$.

#	IA RN	IA	Arr	S RN	S	Start	End	Wait
1	48	2	2	22	1.5	2	3.5	0
2	51	2	4	62	2	4	6	0
3	06	1	5	25	1.5	6	7.5	1
4	22	1	6	31	1.5	7.5	9	1.5
5	79	3	9	23	1.5	9	10.5	0
6	56	2	11	07	1	11	12	0
7	06	1	12	93	2.5	12	14.5	0
8	91	4	16	44	1.5	16	17.5	0
9	51	2	18	12	1	18	19	0
10	13	1	19	26	1.5	19	20.5	0
11	65	3	22	93	2.5	22	24.5	0
12	59	2	24	01	1	24.5	25.5	0.5
13	51	2	26	17	1	26	27	0
14	50	2	28	49	1.5	28	29.5	0
15	13	1	29	58	2	29.5	31.5	0.5
16	94	5	34	98	3	34	37	0
17	57	2	36	61	2	37	39	1
18	26	1	37	41	1.5	39	40.5	2
19	78	3	40	13	1	40.5	41.5	0.5
20	33	1	41	59	2	41.5	43.5	0.5

Recomputed garment-counter answer
From time $t = 0$ to the final departure at $t = 43.5$ min, busy time $= 33.0$ min, idle time $= 10.5$ min, so observed idle fraction $= 10.5/43.5 \approx$ **24.14%** and utilization $\approx$ **75.86%**. Total waiting across 20 customers $= 7.5$ min, mean wait $= 0.375$ min. Including the initial two idle minutes is appropriate because the observation window begins at $t = 0$.

6 Monte Carlo Simulation: Meaning and Steps

Monte Carlo simulation uses repeated samples of uncertain quantities to estimate the distribution or expected value of an output. Its basic stages are: define a model and response, specify input probability laws, map random values into physical quantities, evaluate the response many times, and summarize

the estimates and uncertainty.

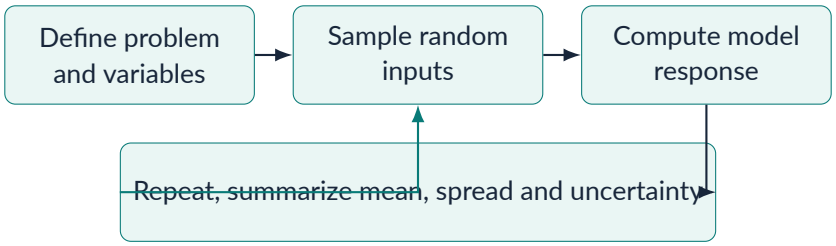

Figure 7. Original Monte Carlo workflow. A short classroom sample path is a teaching demonstration, not statistical proof that results have converged.

6.1 Probability ranges and estimates

For a uniformly distributed two-digit RN 00–99, a probability of 0.30 occupies thirty distinct codes. The ranges must be exhaustive, non-overlapping and correctly normalized. With ten independent simulated days, the empirical mean may differ from the true distribution mean; more replications generally reduce Monte Carlo sampling error but do not fix a wrong distribution or model.

Sampling without time versus with time
Static sampling: draw one day's demand to estimate $P(D > 5)$ with no inventory carryover. **Dynamic inventory:** draw demand every day, subtract from on-hand stock, receive pending orders, place new orders and maintain the inventory state. Both may use Monte Carlo samples, but only the second explicitly models history.

7 Source Numerical C: Ten-Day Inventory Rule

Question 15 asks about an initial inventory of 20 units, discrete daily demands 3–7, a three-day replenishment lead time, and an order of 15 when the inventory position falls below 15. It provides the random digits $0, 9, 1, 1, 5, 1, 8, 6, 3, 5, 7, 1, 2, 9$ and asks for a ten-day simulation: only the first ten digits are needed to sample those ten days' demands.

7.1 State update and explicit assumptions

The demand probability distribution is $(.1, .2, .3, .3, .1)$ for demands $(3, 4, 5, 6, 7)$ units respectively. The single-digit mapping is `0` $\mapsto$ 3, `1--2` $\mapsto$ 4, `3--5` $\mapsto$ 5, `6--8` $\mapsto$ 6, `9` $\mapsto$ 7. **Simulation convention:** pending deliveries arrive at the *start* of their due day; that day's demand is then met from stock; any shortage is a lost sale (recorded separately); at the *end* of the day, place an order of 15 if on-hand + outstanding < 15; a new order arrives at the start of day $d + 3$. Holding cost is charged on end-of-day on-hand inventory. The source question does not specify all these within-day rules, so this answer is explicitly convention-dependent.

Day	RN	Dem.	Start	Recv.	Sold	Lost	End	Order
1	0	3	20	0	3	0	17	0
2	9	7	17	0	7	0	10	15
3	1	4	10	0	4	0	6	0
4	1	4	6	0	4	0	2	0
5	5	5	2	15	5	0	12	15
6	1	4	12	0	4	0	8	0
7	8	6	8	0	6	0	2	0
8	6	6	2	15	6	0	11	15
9	3	5	11	0	5	0	6	0
10	5	5	6	0	5	0	1	0

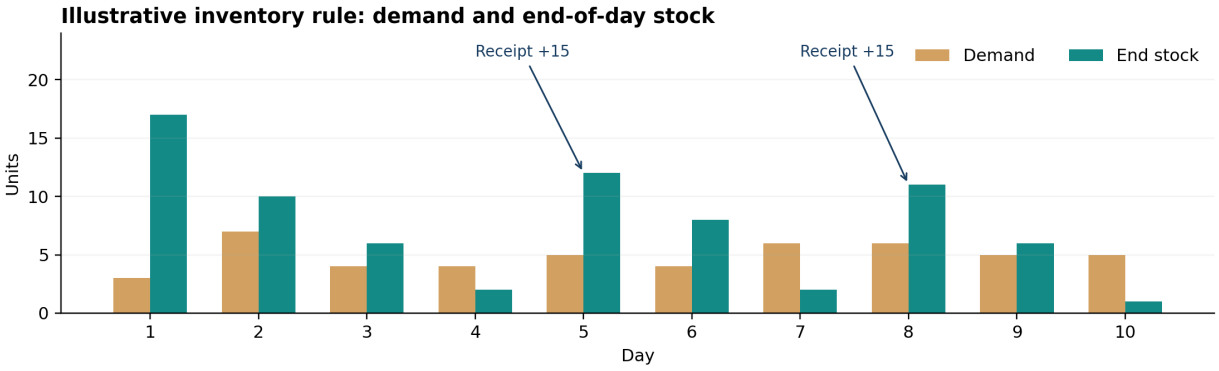

Figure 8. Original plot from the ten-day source-digit inventory calculation. Replenishment of 15 units arrives at the start of days 5 and 8. The order placed on day 8 is still outstanding after day 10.

Result under the declared convention
Orders are placed on days 2, 5 and 8 (each 15 units). End-of-day stock sums to 75 unit-days. Holding cost at Rs 0.20 per unit per day = **Rs 15**; three orders at Rs 5 each = **Rs 15**; **holding + ordering cost = Rs 30** for the ten-day horizon. The calculation excludes shortage penalties, purchasing cost and end-of-horizon accounting for an outstanding order. Lost sales in this path = 0.

Check your understanding
The task's wording may admit alternate order-arrival and holding-cost conventions. Any different timing assumption must be written down; otherwise two apparently plausible "answers" may not be comparable.

8 Source Numerical D: Fifteen-Week Inventory Control

The connected question bank's Question 16 gives an initial inventory of 10 units, reorder point 6, order quantity 12, weekly demand values 0–3, uncertain lead time 2–4 weeks, order cost Rs 100, carrying cost Rs 4 per closing stock unit per week and shortage cost Rs 60 per unit per week. It supplies 15 two-digit demand random numbers and four lead-time random numbers. The exact timing convention matters.

8.1 Random-digit mapping and week-end order rule

Demand	Prob.	RN	Lead (wk)	Prob.	RN
0	.10	00–09	2	.20	00–19
1	.45	10–54	3	.65	20–84
2	.30	55–84	4	.15	85–99
3	.15	85–99	—	—	—

Declared convention: deliver pending stock at the start of its due week, then satisfy that week's sampled demand. Any demand beyond available stock is a lost sale charged one week's shortage cost; it is *not* carried as a backorder. After demand, if on-hand stock is *strictly below 6* and no order is pending, order

12 units. An order made in week w with lead time L arrives at the start of week $w + L$. Charge holding cost on end-of-week stock. Because the assignment does not fully specify these details, other legitimate conventions can produce a different total.

8.2 Reproduce all 15 weekly events

The random demand codes are `49, 67, 06, 30, 95, 01, 10, 70` followed by `80, 66, 69, 76, 86, 56, 84`. The first two lead-time codes 84 and 79 each map to three weeks. Only two orders are triggered in this simulated path; the remaining lead-time codes are unused.

Wk	RN	Dem.	Start	Recv.	Sold	Lost	End	Order
1	49	1	10	0	1	0	9	0
2	67	2	9	0	2	0	7	0
3	06	0	7	0	0	0	7	0
4	30	1	7	0	1	0	6	0
5	95	3	6	0	3	0	3	12
6	01	0	3	0	0	0	3	0
7	10	1	3	0	1	0	2	0
8	70	2	2	12	2	0	12	0
9	80	2	12	0	2	0	10	0
10	66	2	10	0	2	0	8	0
11	69	2	8	0	2	0	6	0
12	76	2	6	0	2	0	4	12
13	86	3	4	0	3	0	1	0
14	56	2	1	0	1	1	0	0
15	84	2	0	12	2	0	10	0

Step-by-step explanation of two key weeks
Week 5 begins with 6 units. The demand code 95 gives 3 units sold, leaving 3. Because $3 < 6$ and there is no pending order, a 12-unit order is placed; lead-time code 84 gives 3 weeks, so it arrives in week 8. Week 14 begins with one unit and has demand 2; the second sale cannot be met under the declared lost-sales convention, so one unit is lost. The second order, placed at the end of week 12, arrives at the start of week 15.

Recomputed conditional result
End-of-week on-hand inventory totals 88 unit-weeks, so carrying cost is $88 \times$ Rs $4 =$ **Rs** 352. Two orders incur $2 \times$ Rs $100 =$ **Rs** 200. One lost unit incurs Rs 60. Thus holding + ordering + one-week shortage cost is **Rs 612**, under the stated start-of-week receipt and lost-sales convention. This is one reproducible interpretation of the source question, not an unqualified answer key.

Check your understanding
If the instructor uses backorders rather than lost sales, or places orders at the start rather than end of a week, recalculate the stock trajectory and shortage-cost duration. Do not silently mix conventions.

9 Continuous Simulation: Differential Equation Example

Continuous simulation describes variables evolving over continuous model time, often with ordinary differential equations (ODEs). Consider a simplified first-order thermal response for a motor housing:

$$\frac{dT}{dt} = -k(T - T_a), \qquad k > 0,$$

where T_a is ambient temperature and k has units s^{-1}. For constant T_a, the analytic solution is $T(t) = T_a + (T(0) - T_a)e^{-kt}$. An explicit Euler numerical method uses $T_{n+1} = T_n - hk(T_n - T_a)$ with step h seconds.

Euler steps compared with the exact equation
Use $T_a = 25°\text{C}$, $T_0 = 65°\text{C}$, $k = 0.1\ \text{s}^{-1}$ and $h = 2$ s.
Step 1 at $t = 2$: $T_1 = 65 - 2(0.1)(65 - 25) = 57°\text{C}$.
Step 2 at $t = 4$: $T_2 = 57 - 2(0.1)(57 - 25) = 50.6°\text{C}$.
Step 3 at $t = 6$: $T_3 = 50.6 - 2(0.1)(50.6 - 25) = 45.48°\text{C}$.
Exact at $t = 6$: $25 + 40e^{-0.6} \approx 46.95°\text{C}$, a difference of about $1.47°\text{C}$. Smaller appropriate steps generally improve Euler accuracy, subject to numerical-stability limitations.

Important distinction. Numerical integration time steps are a computational choice. They do not by themselves make a physically continuous state a discrete-event state. One model can use continuous differential equations and receive discrete control events.

10 Hybrid Models: Continuous Dynamics and Discrete Events

A **hybrid simulation** combines continuous evolution with discrete modes or events. An AGV's location and battery charge evolve continuously in the driving mode; job assignment, docking, charger connection and emergency stops are discrete events that change the mode or parameters.

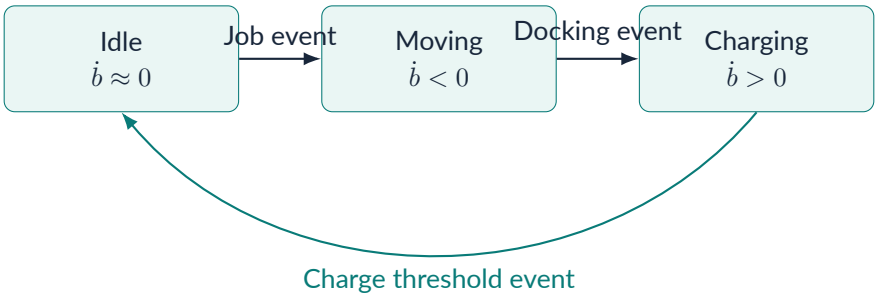

Figure 9. Original simplified hybrid AGV modes. In each mode battery charge b can evolve continuously while assignments and charger transitions are discrete events.

10.1 Which methodology for which problem?

Method	Typical engineering purpose
Discrete-event	Robot jobs, queues, shared resources, downtime and production flow
Continuous	Motor temperature, fluid level, joint state and process dynamics
Monte Carlo	Propagate random inputs into demand, risk or performance estimates
Hybrid	Couple AGV trajectories or energy with job events and switching logic

In simple language
A hybrid model is not automatically more accurate. Use it when the decision really depends on the interaction of continuous physical quantities and discrete scheduling or state transitions.

11 Verification, Validation and Common Source Pitfalls

A correct queue simulator should reproduce a short hand-calculated path, conserve time (busy + idle = horizon for one continuously observed server) and never serve two jobs at the same instant with one nonparallel server. A correct inventory simulator should reconcile beginning stock + receipts = sales + closing stock; for lost sales, demand = sales + unmet demand. These are **verification checks**. Comparing performance to genuine store or factory observations for intended use is **validation**.

Claims to avoid
"Static model = Monte Carlo" without qualification; "a discrete-state system always has a fixed clock step"; "a 20-job simulation gives the long-run expected utilization"; "queue waiting time equals total time in system"; "an event occupies time"; "Monte Carlo is a guaranteed optimizer"; "a digital twin or visual animation proves physical validity".

12 Exam Questions and Short Model Answers

1. **Physical versus mathematical model:** tangible scale/prototype representation versus equations, logical rules or data descriptions.
2. **Static versus dynamic:** single snapshot/outcome without state evolution versus a system evolving over time.
3. **Deterministic versus stochastic:** fixed response under fixed conditions versus response influenced by random variables.

4. **Discrete versus continuous:** jumps in counts at events versus continuous-valued change, e.g., $T(t)$.
5. **What is DES?** Tracking state changes at discrete scheduled event times with an explicit simulation clock and event list.
6. **State variables for a single-server queue?** Clock, busy flag, queue length, waiting job data and next scheduled events as needed.
7. **Queue discipline?** FIFO, LIFO, priority and SIRO select which job is served next.
8. **Monte Carlo?** Repeated random sampling and model evaluation to estimate output statistics.
9. **Why is hybrid modeling needed?** Continuous AGV movement or energy combined with discrete job assignments and docking.
10. **Grocery-source Q13:** $T = 86$ min, busy 69, idle 17, mean wait 2.6 min under first arrival $t = 0$.
11. **Garment-source Q14:** $T = 43.5$, busy 33, idle 10.5 min from $t = 0$; idle 24.14%.
12. **Inventory-source Q15:** Ten-day conditional result under stated ordering rule: Rs 15 holding and Rs 15 ordering cost.
13. **Inventory-source Q16:** Fifteen-week conditional result: Rs 352 holding, Rs 200 ordering and Rs 60 lost-sale shortage cost; total Rs 612.
14. **Explain event scheduling:** Initialize FEL, advance to earliest event, execute state changes, schedule next notices, collect statistics and terminate.
15. **Difference between next-event and fixed increment?** Event jumps versus clock steps; fixed increments may introduce event-time approximation if implemented naively.

Evidence Ledger, Attribution and Scope

Official content map: SPPU TE Robotics & Automation 2024 Pattern, *TE Syllabus.pdf*, course PEC321-ROA (B), Unit II, physical PDF p.26. This is the authoritative topic list; source textbook chapter/page numbers are not invented.
Connected Drive course material directly read: *M&S Unit 2 PPT-1.pdf* (Yogesh Ramesh Chandwade lecture: model types, queueing, Monte Carlo and inventory examples) and *M&S Unit 2 Question Bank.pdf* (Questions 1–16). The grocery 20-customer source digits, garment 20-customer source

digits and ten-day inventory rule were transcribed from this question bank and recomputed independently. The within-day inventory accounting and first-arrival conventions are stated explicitly.
Independent technical background: Maria, *Introduction to Modeling and Simulation*, Proceedings of the 1997 Winter Simulation Conference, DOI: https://doi.org/10.1145/268437.268440; *SimPy documentation*: https://simpy.readthedocs.io/en/stable/; Rossetti, *Simulation Modeling and Arena*, open educational book: https://rossetti.github.io/RossettiArenaBook/; NPTEL, *Fixed-increment time advance*: https://archive.nptel.ac.in/content/storage2/courses/117103017/module17/lec1/4.html.
Figure provenance: All charts and diagrams embedded here are original, generated for these notes from explicit synthetic/question-bank data. No third-party diagram or photo reproduced. A suitable CC0 manufacturing robot photo was located on Wikimedia Commons but could not be downloaded into the build environment and is not represented as embedded.
Limitations: Numerical tests and page geometry do not establish full independent academic proofreading or industrial-system validation. Results reflect declared mappings, input data and simulation conventions; they are not production measurements.

Reproducibility Checklist and Interpretation Notes

1. Preserve the random-number conventions. A uniformly mapped random digit is a symbol for a probability interval, not a physical measurement. For example, two-digit integers 00–99 should be mapped by consecutive non-overlapping ranges whose lengths equal probability percentages. Identify whether the first arrival happens at time zero or after a sampled interval.

2. Recalculate queue paths from the same four recurrence rules:

$$A_i = A_{i-1} + IA_i, \quad B_i = \max(A_i, D_{i-1}), \quad D_i = B_i + S_i, \quad W_i = B_i - A_i.$$

Here A is arrival, B service start, D departure, S service duration, IA interarrival and W waiting time. The idle gap before job i is $\max(0, A_i - D_{i-1})$. For the declared observed window, busy time plus idle time must equal the elapsed time.

3. Recalculate inventory by separate conservation laws. In a lost-sales model, beginning stock + receipts = units sold + ending stock, while demand = units sold + lost units. In a backorder model, inventory-position accounting must include backlog rather than silently charging only a single lost-sale penalty. State whether receipts occur at the beginning or end of the period.

4. Evaluate sample outputs with restraint. The 20-customer traces are finite simulated paths. Their observed utilization and waiting time are not estimates with a stated confidence interval unless a proper replication and analysis design is supplied.

5. Check the numerical examples against their data and assumptions. The grocery queue finishes at $t = 86$ min; the garment queue at $t = 43.5$ min. Inventory Q15 costs Rs 30 for the declared ten-day bookkeeping. Inventory Q16 costs Rs 612 for its declared lost-sales accounting. Changing an input or a convention requires recomputing the complete trace, not changing only its reported total.

6. Validate externally. Matching an instructor's random digits verifies a calculation path; it does not demonstrate realistic factory behaviour.