ZENITH STUDY HUB

Modeling and Simulation

UNIT I | PRINCIPLES - FINAL QA REVIEW

Probability, modeling, simulation studies, abstraction, system boundaries and engineering decision-making

Syllabus-first explanations, original figures and reproducible numericals

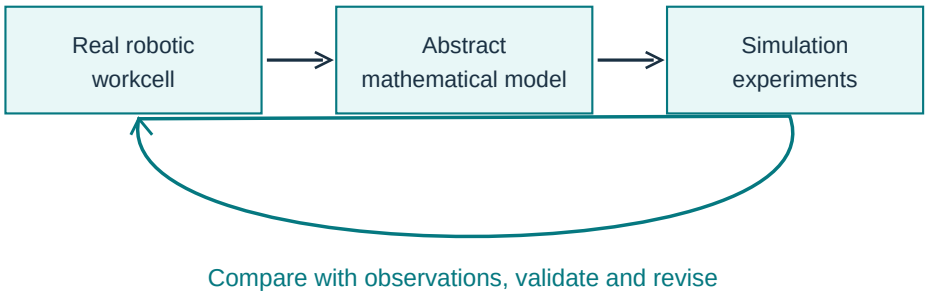

Figure 1. Original modeling-and-simulation loop: real system, explicit abstraction and repeatable experiments. Validation closes the loop; simulation results alone do not establish physical truth.

Official Unit I basis
PEC321-ROA (B), SPPU TE Robotics & Automation, 2024 Pattern, *Principles of Simulation and Modeling*, 8 hours. Topic order follows the supplied *TE Syllabus.pdf*, physical PDF p. 26. All classroom numerical datasets are illustrative, not measured factory performance.

Prepared for: Om Deshmukh | 21 November 2026 | Review candidate

Learning map: exactly what Unit I requires

Official syllabus area	Learning outcome
Probability and statistics review	Find an event probability, mean, variance, standard deviation and uncertainty.
Modeling and simulation definitions	Distinguish system, model, simulation and experiment.
Role of simulation	Explain applications where experiments on real hardware are costly or risky.
Steps in a simulation study	Follow problem definition, conceptual model, data, implementation, V&V, experiments and reporting.
System/problem formulation	Define objectives, variables, inputs, outputs, constraints and scope.
Abstraction / assumptions / boundaries	Draw the system boundary and state what is intentionally omitted.
Model-based problem solving	Compare alternatives using controlled, reproducible assumptions.
Advantages and limitations	Distinguish decision support from prediction guarantees.
Robotics and manufacturing applications	Map robot cells, conveyors, AGVs and maintenance decisions to models.

In simple language
A model is a simplified representation of a real system. A simulation is an experiment performed with that representation, often by changing time, events or uncertain inputs. The goal is not to make an impressive animation. The goal is to answer a clearly stated engineering question with evidence that is accurate enough for that use.

Study policy. Model outputs depend on assumptions. Results on a small synthetic dataset cannot be extrapolated to an industrial robot without measured input data, model validation and uncertainty analysis.

1 Probability Review for Simulation

1.1 Experiment, sample space and event

An **experiment** is a repeatable procedure with an uncertain outcome, such as whether a manufactured part fails inspection. The **sample space** Ω is the set of possible outcomes; an **event** is a subset of Ω. A probability $P(A)$ satisfies $0 \le P(A) \le 1$. For equally likely outcomes, $P(A) = |A|/|\Omega|$, but this counting formula is not valid if outcomes are unequally likely.

1.2 Conditional probability, independence and Bayes

Conditional probability measures the probability of A once B is known:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}, \qquad P(B) > 0.$$

Two events are **independent** if $P(A \cap B) = P(A)P(B)$. Independent events may still occur together; independence is *not* mutual exclusivity. Bayes' rule gives

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}.$$

Worked example: robot-tool fault alert
A fault occurs in 2% of cycles. An alert detects a real fault with probability 90%, and falsely alerts during normal operation with probability 5%. For a random alarm,

$$P(F \mid A) = \frac{0.90(0.02)}{0.90(0.02) + 0.05(0.98)} = \frac{0.018}{0.067} \approx 0.2687.$$

Although the sensor catches 90% of faults, only about 26.87% of alerts correspond to a fault under the assumed rare-fault prevalence. These are hypothetical probabilities for learning.

1.3 Discrete versus continuous random variables

A **discrete** variable counts outcomes such as number of reworks. A **continuous** variable measures values such as robot cycle time. A discrete probability mass function sums to one; a continuous probability density integrates to one. A density value itself is not a probability at one exact point.

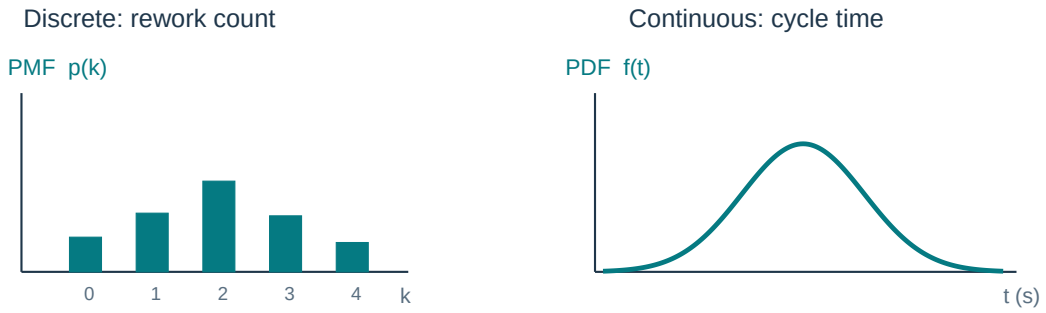

Figure 2. Original conceptual discrete and continuous distributions. Heights are illustrative, not a fitted probability law; continuous area must be normalized to represent probabilities.

1.4 Mean, variance and standard deviation

For observations $x_1, \ldots, x_n$, the sample mean is $\bar{x} = \sum_i x_i/n$. The unbiased sample variance under standard independent sampling assumptions is

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}, \qquad n > 1, \quad s = \sqrt{s^2}.$$

Population variance instead divides by N when the whole population is enumerated. Do not mix sample and population formulas without stating which dataset you have.

Sample-statistics numerical
Five independent illustrative cycle-time readings are $(9, 11, 10, 13, 7)$ seconds. Their sum is 50, so $\bar{x} = 10$ s. Squared deviations are $(1, 1, 0, 9, 9)$, total 20. Thus $s^2 = 20/(5-1) = 5$ s^2 and $s = \sqrt{5} \approx 2.236$ s.

1.5 Expectation, probabilities and uncertainty

Expected value $E[X]$ is a probability-weighted average, not a promise about an individual cycle. For a 20% chance of one rework and 80% chance of none, expected rework count per job is $0(.8) + 1(.2) = 0.2$. One hundred comparable jobs have expected rework count 20, but actual counts fluctuate. A model should report variability, not just its mean.

2 Systems, Models and Simulation

Q19 & Q21 | Probability in Simulation

Independent and dependent events (Q19)

Two events A and B are independent if P(A and B) = P(A) x P(B). Equivalently, P(B given A) = P(B) when P(A) is positive. Dependent events are those where the occurrence of one changes the conditional probability of another.

Without replacement: A box contains 3 defective and 7 acceptable parts. The probability that both of two successive draws are defective is (3/10) x (2/9) = 1/15 = 6.67%. The denominator and numerator change after the first draw.

With replacement: If the first part is returned and the box is remixed, the probability of both being defective is (3/10) x (3/10) = 9%, assuming independent draws. Mutual exclusivity and independence are not the same.

Why probability is essential in simulation (Q21)

Probability assigns a value from 0 to 1 to events and supplies models for uncertain arrivals, processing times, machine failure, repairs and rework. A stochastic simulation samples random inputs from assumed distributions and produces variable performance outcomes.

Illustration: If 100 independent production cycles each have a defect probability of 0.02, the defect count X follows a binomial model. E[X] = 100(0.02) = 2 defects and Var(X) = 100(0.02)(0.98) = 1.96 defects squared. These are model expectations, not guaranteed observed results.

Exam conclusion: State the underlying event, probability assumptions, distribution, and why repeated runs are needed to quantify uncertainty.

Source: SPPU TE (2024 Pattern) Unit I syllabus; connected M&S Unit 1 question bank and teaching slides. Illustrative examples are not measured factory data.

2.1 The real system and its interactions

A **system** is a collection of interacting components organized around a purpose. A robotic assembly cell includes parts, robot controller, gripper, sensors, conveyor, workers, routing logic and disturbances. State variables summarize its condition at a given time, such as queue length, robot availability or current part position.

2.2 A model is a purposeful simplification

A **model** represents selected properties and relationships of the system. A physical scale prototype is a physical model. A set of equations, flowcharts or software rules is a mathematical/computational model. A good model does not copy every physical detail: it includes enough fidelity to support the stated question.

2.3 A simulation runs the model through scenarios

A **simulation** evaluates the behaviour of a model under inputs, initial conditions, time evolution or random draws. It can compare policies or expose bottlenecks without moving real robots. However, an accurate simulation of an inaccurate model is still an inaccurate representation of reality.

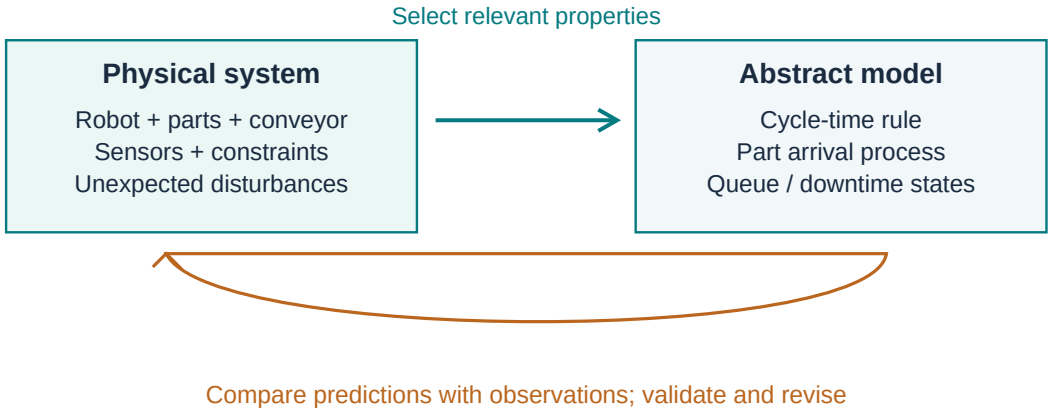

Figure 3. Original distinction between a real robotic cell and a question-specific model. Omitted real-world disturbances are potential sources of model error.

2.4 Three model representations

Conceptual: block diagrams, process rules and assumptions understandable to stakeholders.

Mathematical: equations for movement, service times, resource capacities, events or probability distributions.

Computational: implementation using code or simulation software. Verification checks that the implementation reproduces the chosen conceptual/mathematical rules.

3 Why Engineers Use Simulation

3.1 Purpose and practical benefits

Simulation is valuable when altering the real plant is expensive, dangerous, slow or disruptive. Examples: estimate queue buildup after conveyor speed changes, compare AGV fleet sizes, predict bottlenecks before purchasing a robot, evaluate downtime policies, and rehearse a controller sequence. Simulation can also support operator training and communicate system logic.

Engineering insight
Simulation is a *decision-support experiment*. It is especially useful for relative comparisons (Plan A vs Plan B) if both plans use equally credible assumptions. It cannot automatically certify safety or guarantee that a factory will reach its predicted throughput.

3.2 When an equation may be sufficient

If a single station always completes exactly one part every 8 s without downtime or blocking, its ideal capacity is $3600/8 = 450$ parts per hour. No complex simulation is needed for this trivial bound. Simulation becomes valuable when multiple interacting stations, queues, repairs, changeovers, priorities or uncertain arrivals make a simple formula inadequate.

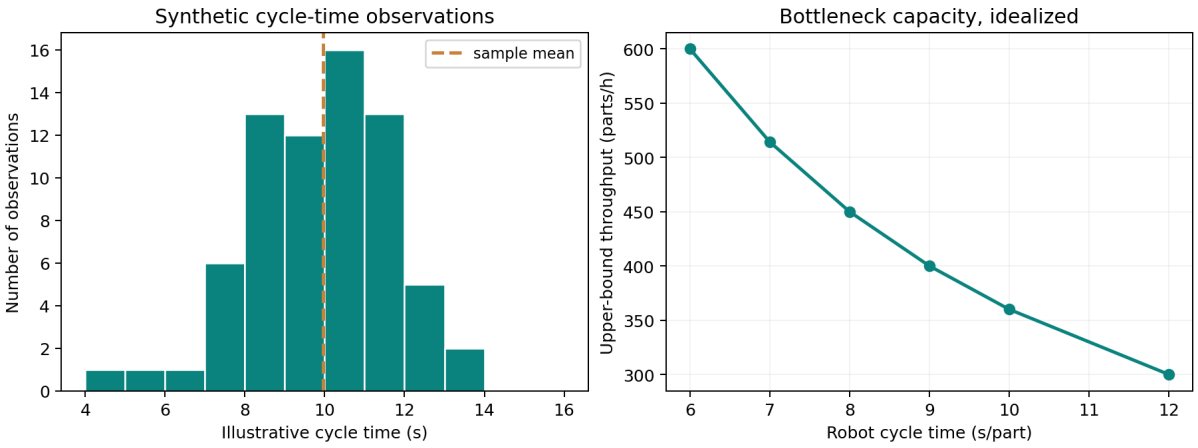

Figure 4. Original synthetic cycle-time histogram and a mathematically derived ideal throughput curve. The histogram is generated test data, not factory measurements; the throughput bound excludes interruptions.

3.3 Question-driven methodology

Before building a 3D animation, ask: Which decision must change? What measurable outcome matters? What data exist? Which assumptions could reverse the decision? An attractive animation can be a poor tool if the chosen model ignores the most influential physical constraints.

4 Steps in a Simulation Study

In simple language
A credible simulation study is a sequence of engineering decisions with feedback. It is not "install software, draw a factory, press Run."

1. **Problem formulation.** Decide what question must be answered and why.
2. **Objectives and performance measures.** Define throughput, queue time, cost, downtime, safety constraints and success criteria.
3. **Conceptual model.** Identify entities, processes, resources, events, logic, assumptions and boundaries.
4. **Input-data collection.** Obtain credible arrivals, cycle times, failure and repair data; document units and time periods.

5. **Implementation.** Convert the conceptual model into equations or executable software.
6. **Verification.** Debug the implemented rules; check extreme cases and known solutions.
7. **Validation.** Compare suitable model behaviour against the real system for the intended use.
8. **Experimental design.** Choose scenarios, replications, warm-up and statistics appropriate to the question.
9. **Results and implementation.** Analyze uncertainty, recommend a decision, communicate limits and monitor field outcomes.

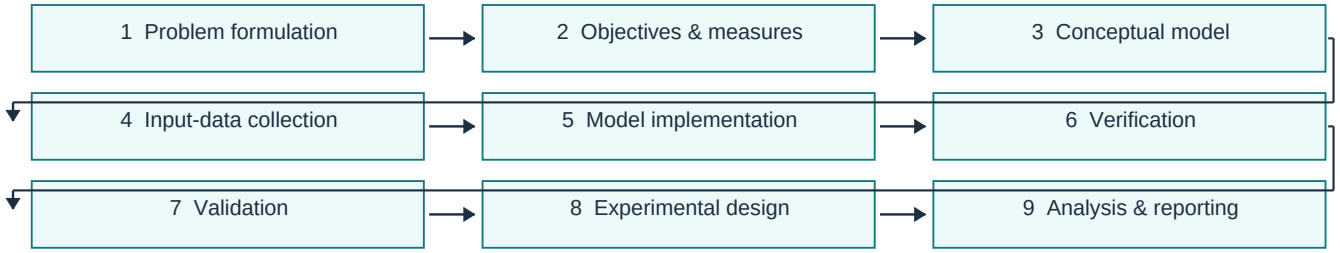

Figure 5. Original nine-stage simulation-study workflow. Verification and validation act as explicit gates; results can trigger conceptual-model revision.

4.1 Verification is not validation

Verification: does the model/code implement the conceptual specification correctly? For example, a service station with 8 s constant cycle time should output 8 s when given one job.

Validation: is the implemented model sufficiently representative of the real cell for its stated intended use? For example, do predicted queue lengths agree with observed queues at comparable demand and staffing?

You can verify a model perfectly and still fail validation because an assumption was unrealistic.

Credibility note. Comparing a few points to recorded data does not grant universal validity. Document the operating range and the particular decisions for which the model is acceptable.

5 Defining the System and the Problem

5.1 Write a decision question, not just a topic

Weak problem statement: “Simulate our robot line.” Strong statement: “Should we add a second robot to keep average part waiting time below 20 s at 70 arrivals/hour, subject to a maximum budget and safety constraints?” The second version defines controllable decision variables, a demand scenario and measurable success criteria.

5.2 Inputs, state variables, decisions and outputs

Model category	Robotic assembly example
Input parameters	Arrival rate, cycle-time distribution, repair time, batch size.
State variables	Queue length, robot busy/idle flag, number of defective parts.
Decision variables	Robot count, job priority rule, buffer capacity, AGV dispatch policy.
Outputs / metrics	Parts per hour, mean waiting time, robot utilization, WIP, cost.
Constraints	Floor space, shift length, safety clearance, loading capacity.

5.3 Define the model’s system boundary

The **boundary** is the line between modeled and external influences. A study of robot bottlenecks may include conveyor, robot, queue buffer and downtime, but treat supplier delivery schedules as externally specified inputs. However, if supplier delays dominate the backlog, excluding them makes the model incomplete for its purpose.

Q1 | Measures of Simulation Performance

Definition and engineering use

Simulation imitates a system or process through an explicitly defined model to study its behavior. A performance measure is a numerical output used to judge a system or compare alternatives against an engineering objective.

Measure	Definition / formula	Units
Throughput	Completed jobs / observation duration	parts/hour
Resource utilization	Busy time / available time	fraction or percent
Mean waiting time	Sum of job waiting times / number of observed jobs	seconds
Time-average queue length	Area under Q(t) / total observation time	jobs
Mean work in process	Time-average number of unfinished items within the boundary	jobs

Solved example

A robot cell completes 90 parts during a 30-minute observation period. The throughput is 90 / 0.5 = **180 parts/hour**. If the robot is busy for 24 of the 30 available minutes, utilization is 24/30 = **80%**.

Reporting rule: Declare whether downtime is excluded from available time, whether initial warm-up data were discarded, and whether statistics are averaged over jobs or over time.

Exam conclusion

Choose performance measures based on the purpose: throughput for production capacity, waiting and queue length for congestion, utilization for loading, and cost or defect rate for engineering trade-offs.

Source: SPPU TE (2024 Pattern) Unit I syllabus; connected M&S Unit 1 question bank and teaching slides. Illustrative examples are not measured factory data.

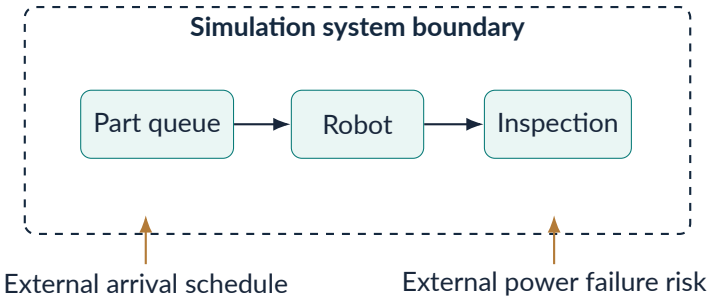

Figure 6. Original example of a system boundary. Arrival demand and power risks are supplied as external processes here; changing the study question may require bringing them inside.

6 Abstraction, Fidelity and Assumptions

6.1 Choose only the necessary details

An **abstraction** deliberately leaves out detail to make a model useful. A high-level cell model might use a distribution for robot service time. A middle-level model includes gripper change and travel path. A low-level model includes motor dynamics and torque limits. These models answer different questions; more geometric detail is not automatically better.

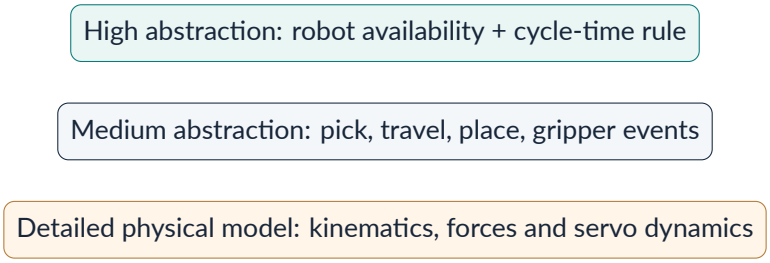

Figure 7. Original abstraction ladder. Fidelity should match the decision question and evidence available, not the modelling software's graphics capabilities.

6.2 An assumptions register

Assumption	Risk if violated
Every part available at time zero	May understate arrival-driven waiting and starvation.
Cycle time fixed at 8 s	May omit variation, rework and changeovers.
No failures or downtime	May overstate achievable throughput; utilization depends on demand and its time denominator.
Unlimited buffers	May hide blocking and floor-space constraints.
Operator available when needed	May overestimate production on short-staffed shifts.

What-if calculation: ideal robot capacity
Robot time is 8 s/part, inspection 3 s/part and the conveyor segment 2 s/part. In an independently pipelined line without blocking and other losses, the *capacity upper bound* is $3600/\max(8,3,2) = 450$ parts/hour. Adding a faster camera does not raise that bound, because the robot remains the bottleneck. This is not a prediction of actual shift output.

6.3 Sensitivity analysis: which assumption matters?

Change one uncertain factor while holding the others at a documented baseline. Robot cycle times 6, 8 and 10 s give ideal upper bounds 600, 450 and 360 parts/hour. A bound that shifts strongly under small plausible time changes indicates the need for better robot-time measurements. Simultaneous changes may interact; one-at-a-time sensitivity cannot reveal every interaction.

7 Model-Based Problem Solving

7.1 A structured comparison of alternatives

Define current state, candidate scenarios and metrics. For example: A = one robot, B = two robots with shared buffer, C = one robot plus an improved gripper. Use identical demand assumptions, comparable run lengths and matched

uncertainty reporting. Then check whether differences exceed output variability and are meaningful in cost and safety.

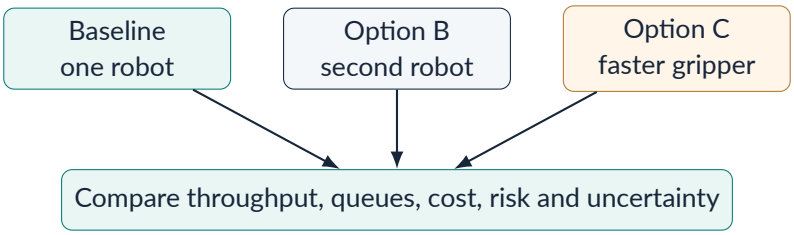

Figure 8. Original alternatives-comparison concept. Each candidate should be evaluated under matched demand, failure and staffing assumptions.

7.2 Small deterministic timing example

Three consecutive single-server robot jobs arrive at times 0, 3 and 7 s. Each service time is exactly 5 s, and service is FIFO. Start times are $0, 5, 10$ s, completion times $5, 10, 15$ s, and waits $0, 2, 3$ s. Mean waiting time is $(0 + 2 + 3)/3 = 5/3$ s. The machine is busy continuously from 0 to 15 s, giving 100% utilization over that *specific finite interval*; this does not imply permanent stability at any arrival rate.

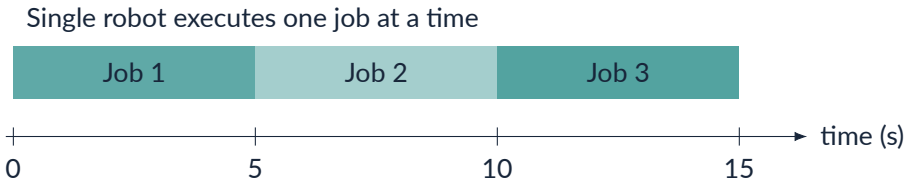

Figure 9. Original FIFO timing chart for three deterministic 5 s jobs. Arrival times differ from service start times because jobs wait in a queue.

Important scope. More detailed discrete-event logic, random number generation and output confidence intervals are developed in later units. This Unit I example illustrates the problem-formulation principles without claiming to be a full validated factory simulation.

Q3 & Q6 | Suitability and Challenges

When simulation is not appropriate (Q3)

Simulation may not add value when (1) a simple analytical solution already answers the question, (2) direct experimentation is cheaper and safe, (3) defensible inputs are unavailable, (4) the study costs more than expected benefit, (5) time and skilled staff are insufficient, or (6) objectives and system boundaries cannot be meaningfully defined.

Example: An uninterrupted station producing exactly one part every 8 seconds has an ideal capacity of 3600/8 = **450 parts/hour**. A complex simulation is unnecessary for this simple theoretical bound; introduce simulation when queues, failures, blocking or variability matter.

Challenges in a simulation study (Q6)

Challenge	Mitigation
Poor or biased input data	Record conditions and missing values; validate distributions
Incorrect boundary or assumptions	Document exclusions and perform sensitivity tests
Programming errors	Trace small cases and check event conservation
Weak model validity	Compare outputs to independent real-system observations
Stochastic uncertainty	Use replications and uncertainty intervals
High study cost	Begin with a focused model and decision metric
Misread results	State units, limitations and operating range

Exam conclusion: Simulation supports decisions, but it does not automatically establish an optimum, safety, or predictive validity.

Source: SPPU TE (2024 Pattern) Unit I syllabus; connected M&S Unit 1 question bank and teaching slides. Illustrative examples are not measured factory data.

8 Advantages, Disadvantages and Limits

Useful strengths	Important limitations
Test alternatives without modifying real hardware.	Wrong input distributions give misleading output.
Visualize interacting queues and resources.	Detailed models cost time and require expert validation.
Find bottlenecks and policy interactions.	Simulation is not an optimization algorithm by itself.
Estimate variability and uncertainty.	Results only apply within the model's stated validity domain.
Communicate a process to engineers.	Strong-looking animations can hide modeling errors.
Re-run reproducible "what-if" scenarios.	Rare accidents and human behaviour may be under-modelled.

In simple language
A simulation can compare alternatives under its assumptions. It cannot prove that a physical factory will perform as simulated, or make untested robot motion safe. Better input data and validation are more valuable than decorative graphics.

9 Applications in Robotics, Automation and Manufacturing

9.1 Robotic assembly and welding

Model robot cycle phases, gripper/tool changes, part arrival patterns, fixture availability and safety-gated motions. Output measures might include parts/hour, robot idle proportion and queue waiting time. A sequence that is safe in an abstract task model still needs physical collision checks and cell-level safety engineering.

9.2 Conveyors and material handling

Model belt speed, accumulation buffers, loading/unloading rates, transfer delays, jams and sorting rules. The effect of a downstream stoppage can propagate upstream through finite buffer capacities.

9.3 AGVs and mobile robots

Model dispatch rules, travel-time distributions, battery charging, aisle capacity and interaction with pedestrian areas. A simulation can compare fleet sizes, but path collision avoidance and safety still require suitable robotics engineering.

9.4 Maintenance, reliability and flexible manufacturing

Combine repair and failure assumptions with production logic to explore scheduled maintenance, spare parts, changeovers and rework. Performance should be assessed under varied operating conditions instead of one idealized run.

Example question for an industrial decision
"At an observed demand of 70 parts/hour with a mix of part types, how many robots and buffers are needed to keep 95th-percentile queue waiting time under 30 seconds?" This specifies an outcome, but needs actual arrival, service and failure data before a trustworthy numerical answer can be produced.

10 System Components: Entities, Attributes, Activities and Events

10.1 What is inside a system?

The connected *M&S Unit 1 Question Bank* explicitly asks for the system components, not only for a high-level boundary. These terms should not be confused:

- **Entity**: a distinguishable item that flows through, or participates in, the system: a job, part, AGV or inspection request.
- **Attribute**: data belonging to an entity: part type, arrival time, priority, se-

lected process route or required tool.

- **Activity**: an interval with duration, such as the robot handling a part for five seconds.
- **Event**: an occurrence at a particular model time that can change the state, such as job arrival, operation completion or machine failure.
- **System state**: the minimal set of variables describing the modeled system at a specified time: queue length, robot status, buffer contents and next event times.

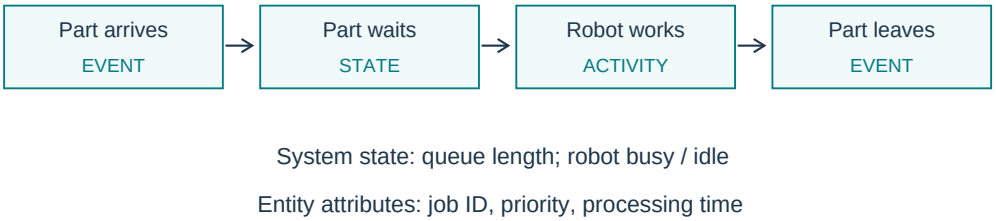

Figure 10. Original element-by-element model of a robot service station. Entities have attributes; events update state; activities occupy elapsed time.

Example: event versus activity in a robot
If a welding robot starts work at time 12 seconds and finishes at 19 seconds, the *start* and *completion* are events; the seven-second welding operation is an activity. During that activity, the state *robotBusy* remains true. A job's arrival time is an attribute of the job, not a system-wide state variable.

10.2 System environment, endogenous and exogenous variables

The **environment** consists of influences outside the modeled system boundary. An **endogenous variable** is determined by processes and rules within that boundary, such as queue length or utilization. An **exogenous input** is supplied from outside it, such as incoming demand or a recorded shift schedule. These labels depend on where the boundary is drawn. If the model expands to include suppliers, a former exogenous delivery delay may become endogenous.

In simple language
Endogenous: the model calculates it. **Exogenous:** the model receives it from outside. A variable can change category when the model boundary changes.

10.3 Open and closed systems — a modeling convention

An **open system** exchanges material, energy or information with its environment. A robot cell receiving incoming parts and exporting finished products is open. In a **closed system**, selected exchanges across the defined boundary are absent or neglected. For example, a finite AGV fleet circulating within a fixed work area can be treated as a closed population in a queueing model, although the real equipment still consumes electricity and exchanges information. Do not confuse a closed *queueing population* with a physically isolated system.

Boundary question
A factory may be an open material-flow system, yet the number of robots can be fixed inside its simulation boundary. "Open" and "closed" always require a named boundary and a specific type of exchange.

11 Descriptive Statistics: Median, Mode and Outliers

Alongside mean, variance and standard deviation, the connected Unit I question bank asks about **median** and **mode**. The **median** is the middle sorted observation (or the mean of the two middle observations for an even count). The **mode** is the most frequent observation; a dataset can have more than one mode or no unique mode.

Worked example: cycle-time readings
For five cycle times $7, 9, 10, 11, 13$ s, the mean is 10 s and the median is 10 s. All five values occur once, so there is no *unique* mode. Replace 13 s with an unusual 40 s reading: the new mean is $77/5 = 15.4$ s, but the median stays 10 s. The median is more resistant to an extreme observation, although that observation still requires investigation, not automatic removal.

Check your understanding
If a report gives only average wait time, ask whether rare very long waits might be hiding in the tail. In production systems, percentiles and the full observed distribution can matter as much as the mean.

12 Model Logic and Simulation Time Advance — Exam Preview

The question bank includes discrete-event program organization and a comparison of clock-update policies. These concepts bridge Unit I system definitions with the official Unit II methodologies.

12.1 Future-event list and simulation clock

A **future-event list (FEL)** stores event notices with scheduled timestamps. A **timing routine** selects the next event, advances the simulation clock, and invokes an event routine. The event updates the system state, collects statistics and may schedule new events. A **termination condition** ends the experiment at a stated horizon or event count.

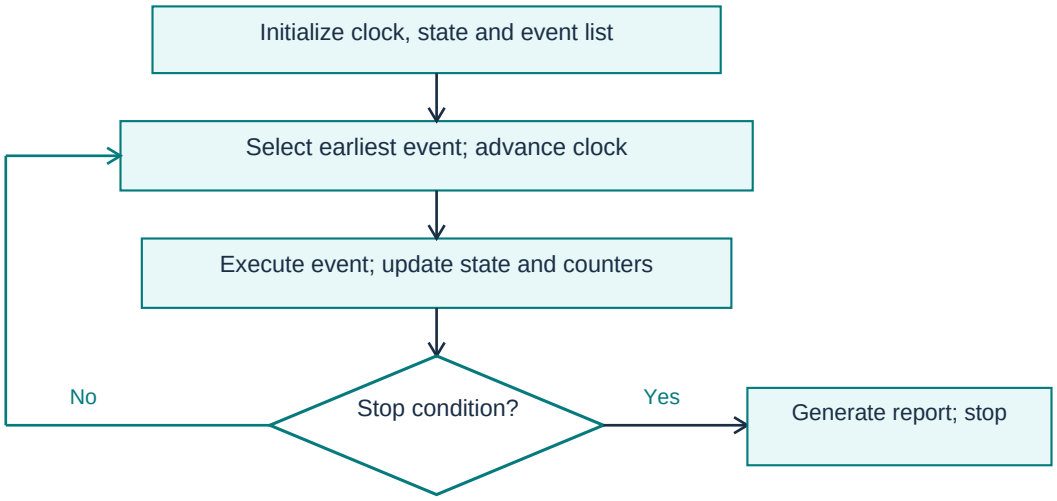

Figure 11. Original conceptual organization of a next-event simulation program. The loop stops when the declared termination condition is satisfied.

12.2 Next-event versus fixed-increment advancement

With **next-event advance**, time jumps to the earliest scheduled event. With **fixed-increment advance**, time moves by a fixed step Δt and the program checks whether events have occurred. An uncorrected fixed-step implementation can shift event times to grid points; that approximation changes waiting-time results unless the model handles events between ticks accurately.

	Next-event	Fixed increment
Clock	Jumps to next event timestamp	Steps by fixed Δt
Idle intervals	Skipped computationally	Still stepped through
Typical use	Queues, jobs, failures, operations	Regularly sampled dynamics or periodic controls
Care required	Deterministic tie handling	Step size, missed events and discretization error

Q15 & Q16 | DES Program Organization

Essential components of a DES model (Q15)

A discrete-event simulation changes its state at event timestamps. Its core data are the **system state** (e.g., queue length and robot busy/idle), **simulation clock**, **future-event list**, and **statistical counters**. Common program modules include an initialization routine, timing routine, event routines, random-variate generators, main program and report generator.

Execution sequence (Q16)

Step	Action
1. Initialize	Set clock, state, counters, and initial scheduled events
2. Select event	Find earliest event in the future-event list and advance clock
3. Process event	Update time-weighted counters, system state and scheduled events
4. Check termination	If stopping condition not met, process another event
5. Report	Calculate metrics, uncertainty and model limitations

Single-robot queue: deterministic event trace

Time	Event	Queue	Robot state
0 s	Job 1 arrives, begins service	0	Busy until 5 s
3 s	Job 2 arrives	1	Busy
5 s	Job 1 leaves, Job 2 begins	0	Busy until 10 s
7 s	Job 3 arrives	1	Busy
10 s	Job 2 leaves, Job 3 begins	0	Busy until 15 s
15 s	Job 3 leaves	0	Idle

Waiting times: (0, 2, 3) seconds; mean = 5/3 seconds. Apply a documented same-timestamp event priority and accumulate queue-area statistics **before** changing the state.

Source: SPPU TE (2024 Pattern) Unit I syllabus; connected M&S Unit 1 question bank and teaching slides. Illustrative examples are not measured factory data.

Timing example
Suppose events occur at $t = 1.3, 2.2, 4.8$ s. Next-event advancement processes exactly 1.3, 2.2 and 4.8 s. A naive fixed 1 s clock checking only at 1, 2, 3, 4 and 5 s would process them late, at 2, 3 and 5 s, respectively. A properly implemented fixed-step method can interpolate or otherwise handle sub-step events, but must document its accuracy.

Check your understanding
Tie handling: If an arrival and departure occur at precisely the same simulation time, define a consistent priority or batch rule. Differing tie policies can change observed queues even when all event timestamps match.

13 Solved Unit I Revision Questions

1. Define system, model and simulation. Give one robotic example.
2. Explain why the simulation boundary changes when the engineering decision changes.
3. Distinguish verification from validation with a robot-cell example.
4. A binary defect flag occurs in 2 of 100 parts. Estimate sample defect proportion. **Answer:** 0.02, subject to sampling uncertainty.
5. For readings $9, 11, 10, 13, 7$, find mean and unbiased sample variance. **Answer:** 10 s, 5 s^2.
6. Compute $P(F \mid A)$ when $P(F) = .02$, $P(A \mid F) = .9$, $P(A \mid \neg F) = .05$. **Answer:** $0.018/0.067 \approx 0.2687$.
7. Find the ideal capacity of one robot taking 8 s/part. **Answer:** 450 parts/hour without losses.
8. List six steps in a simulation study. **Answer:** problem/objectives, conceptual model, data, implement, verification/validation, experiments/report.
9. Explain abstraction and give one detail a throughput-only model may omit.
10. For FIFO arrivals $(0, 3, 7)$ s and service time 5 s, find waits. **Answer:** $(0, 2, 3)$ s; mean $5/3$ s.

11. Why might two robots not double real throughput? **Answer:** arrivals, conveyor, fixtures, operator availability and shared resources can bottleneck.
12. Give three advantages and three limitations of simulation in manufacturing.

Frequent exam mistakes
Treating model and simulation as synonyms; confusing an illustrative example with a validated forecast; using the population-variance denominator for a sample; assuming independence merely because two events differ; hiding assumptions; and claiming simulation automatically guarantees an optimum or a safe robot deployment.

Sources and Review Boundaries

Official syllabus: Supplied SPPU *TE Syllabus.pdf*, physical PDF p. 26, course PEC321-ROA (B), Unit I, *Principles of Simulation and Modeling*.

Connected course materials actually reviewed: *M&S Unit 1 PPT-1.pdf* (definitions, appropriate uses, simulation steps and system logic) and *M&S Unit 1 Question Bank.pdf* (entities, environment, open/-closed, time advance, probability and descriptive statistics). Both are files in the user's *ENDSEM NOV 2026* Drive folder. Some listed question-bank topics anticipate later units.

Technical references:

1. NASA, *Modeling & Simulation definitions and credibility*: https://public.ksc.nasa.gov/mns/ms-definitions/
2. NASA-STD-7009, *Standard for Models and Simulations*: https://standards.nasa.gov/standard/nasa/nasa-std-7009
3. NIST, *Simulation model methodology and credibility guidance*: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=910389
4. INFORMS Winter Simulation Conference, *Verification and Validation of Simulation Models*: https://informs-sim.org/wsc20papers/002.pdf
5. Law, *Simulation Modeling and Analysis*; Banks et al., *Discrete-Event System Simulation*; textbook titles listed in the supplied official syllabus. They are not asserted to be consulted page-by-page here.

Figure provenance and limits: All figures and synthetic datasets are original. No external photos, live factory data or independent academic sign-off were used; numerical checks are deterministic.