Lectures in Mathematical Modeling

Lecture 5: Monte Carlo and Discrete-Event Simulation

Kenneth Sok Kin Cheng

Revised 20 August 2026

Learning goals

By the end of the lesson, readers should be able to:

1. distinguish a **deterministic model** from a **stochastic model**;
2. map uniform random numbers to outcomes from a discrete or empirical probability distribution;
3. construct and interpret a simple **Monte Carlo estimator**;
4. explain why repeated simulation runs are needed and why simulation error decreases slowly;
5. trace a single-server queue using arrival, service-start, waiting, and departure times;
6. design a simulation experiment that includes outputs, replications, validation, sensitivity, and limitations.

1 Why simulate?

Previous lectures represented a system using equations or curves. That approach remains valuable, but some systems contain uncertain events whose exact sequence cannot be predicted: customers arrive at irregular times, service durations vary, and equipment may fail unexpectedly. A **simulation** imitates the operation of such a system and records what happens under specified assumptions.

Launch: The canteen queue **[10 min]**

At lunchtime, students arrive at one canteen counter at irregular times. Some orders take much longer than others. The school is considering opening a second counter.

In pairs, identify:

1. two quantities that vary unpredictably from student to student;
2. two quantities that describe the current *state* of the system;
3. three outputs that the school should examine before making a decision;
4. one factor that might make a simulation unrealistic.

Modeling lesson: A simulation requires more than random numbers. It needs explicit input assumptions, update rules, outputs, and a plan for checking credibility.

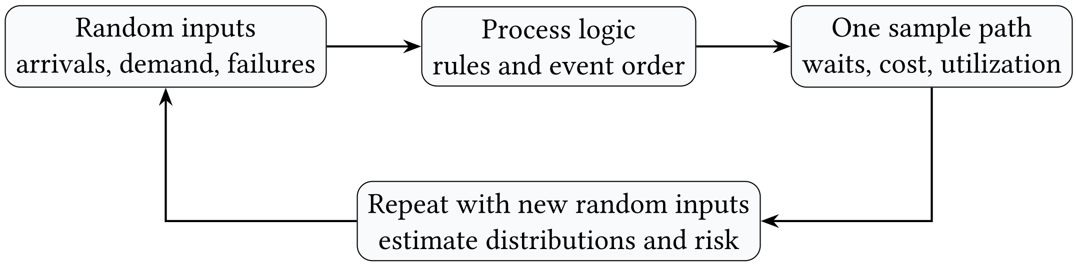

Figure 1: A simulation combines random inputs with deterministic update rules. One run gives one possible history; repeated runs reveal the range of possible outcomes.

Core simulation language

- A **state variable** describes the system at a given time, such as queue length or inventory level.
- An **event** changes the state, such as an arrival, a service completion, or a delivery.
- A **sample path** is one simulated history of the system.
- A **replication** is an independent run under the same policy and assumptions.
- A **random seed** reproduces the same pseudorandom sequence, which is useful for debugging and fair comparisons.

The computer performs deterministic calculations after the random inputs have been generated. Simulation is therefore structured experimentation, not random guessing.

2 From uniform random numbers to model inputs

A **random variable** assigns a numerical value to each possible outcome of a random experiment. To simulate a discrete random variable, partition the interval $[0, 1)$ according to cumulative probabilities and generate $U \sim \text{Uniform}(0, 1)$.

Suppose a service time S has possible values $s_1, \ldots, s_k$ with probabilities $p_1, \ldots, p_k$. Let

$$F_j = p_1 + \cdots + p_j.$$

Assign $S = s_j$ when $F_{j-1} \le U < F_j$, where $F_0 = 0$. This is the discrete **inverse transform method**.

Worked Example 1: Simulating service times

Historical observations at a snack counter suggest

service time S (min)	2	3	4	5
$P(S = s)$	0.20	0.35	0.30	0.15

The cumulative probabilities are 0.20, 0.55, 0.85, and 1.00, so

$$S = \begin{cases} 2, & 0 \le U < 0.20, \\ 3, & 0.20 \le U < 0.55, \\ 4, & 0.55 \le U < 0.85, \\ 5, & 0.85 \le U < 1. \end{cases}$$

For $U = 0.07, 0.48, 0.61, 0.93$, the simulated service times are $2, 3, 4, 5$ minutes respectively.

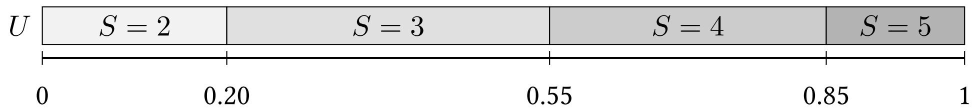

Figure 2: The length of each interval equals the probability of its outcome. A larger probability occupies more of $[0, 1)$.

Guided practice: Interarrival times **[10 min]**

Student interarrival times have the distribution

A (min)	2	3	4	5
$P(A=a)$	0.15	0.35	0.35	0.15.

1. Construct the cumulative probability table and the interval mapping.
2. Convert $U = 0.08, 0.32, 0.71, 0.94, 0.51$ into interarrival times.
3. Explain why a fixed seed is helpful when comparing one counter with two counters.

Model criticism: The input distribution is part of the model

A simulation can produce precise-looking output from a poor input model. Historical frequencies may be unrepresentative of examination days, special menus, or changing student behavior. The probabilities, dependence between variables, and time period represented by the data must all be justified.

3 Monte Carlo estimation

A **Monte Carlo simulation** estimates a quantity by repeating a random experiment. Consider points drawn uniformly from the unit square. The probability of falling inside the quarter circle $x^2 + y^2 \leq 1$ is its area, $\pi/4$. If C of n points fall inside, then

$$\frac{C}{n} \approx \frac{\pi}{4}, \qquad \boxed{\widehat{\pi} = 4\frac{C}{n}.}$$

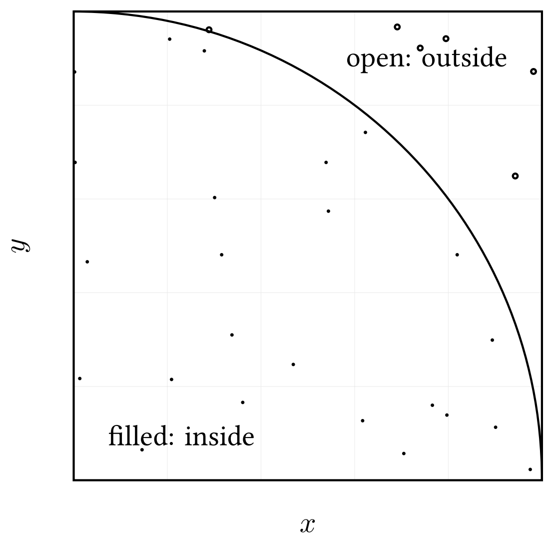

Figure 3: 30 reproducible sample points: 24 fall inside the quarter circle and 6 outside. The picture represents one sample path, not a guarantee of accuracy.

Worked Example 2: One estimate of π

In Figure 3, $C = 24$ and $n = 30$, so

$$\widehat{\pi} = 4\left(\frac{24}{30}\right) = \boxed{3.20}.$$

The estimate is plausible but rough. A different seed would produce a different count. The value 3.20 should therefore be reported together with n, the random-number procedure, and preferably results from repeated runs.

For independent samples, Monte Carlo uncertainty usually decreases at the rate

$$\text{typical error} \propto \frac{1}{\sqrt{n}}.$$

Thus reducing the typical error by a factor of 2 requires about 4 times as many samples; reducing it by a factor of 10 requires about 100 times as many.

Checkpoint: Convergence

A simulation uses $2{,}500$ independent samples. Approximately how many samples are required to reduce its typical Monte Carlo error to one-fifth of the current value? Explain using the square-root rule.

Replication and uncertainty

Increasing the number of observations within one run and increasing the number of independent replications answer different questions. A long run may estimate steady performance; several replications reveal run-to-run variability. Report at least a mean and a measure of spread, and retain the individual replication results rather than only the grand average.

4 Discrete-event simulation of a queue

In a **discrete-event simulation**, the state changes only when an event occurs. For a single-server queue, let

- A_i be the arrival time of customer i;
- S_i be the required service time;
- B_i be the service-start time;
- D_i be the departure time;
- $W_i = B_i - A_i$ be the waiting time.

The update equations are

$$\boxed{B_i = \max(A_i, D_{i-1}), \qquad D_i = B_i + S_i, \qquad W_i = B_i - A_i,}$$

with $D_0 = 0$. The maximum expresses the queue logic: service begins only after both the customer has arrived and the server is free.

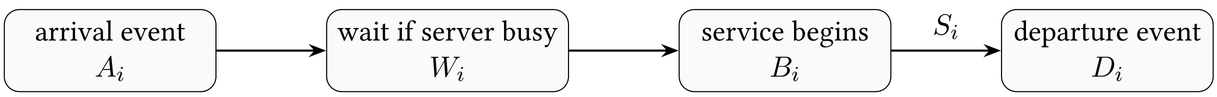

Figure 4: A queue is governed by event order. Randomness enters through arrival and service times; the update logic is deterministic.

Worked Example 3: Manual trace of one counter

Six customers have arrival times

$$A_i = (0, 3, 5, 10, 12, 16)$$

and service times

$$S_i = (4, 5, 3, 4, 2, 5)$$

in minutes. Applying the update equations gives:

Customer i	1	2	3	4	5	6
Arrival A_i	0	3	5	10	12	16
Service S_i	4	5	3	4	2	5
Start B_i	0	4	9	12	16	18
Wait W_i	0	1	4	2	4	2
Departure D_i	4	9	12	16	18	23

The average waiting time is

$$\bar{W} = \frac{0+1+4+2+4+2}{6} = \boxed{2.17 \text{ min}},$$

and the maximum waiting time is 4 minutes. These values describe only this six-customer sample path.

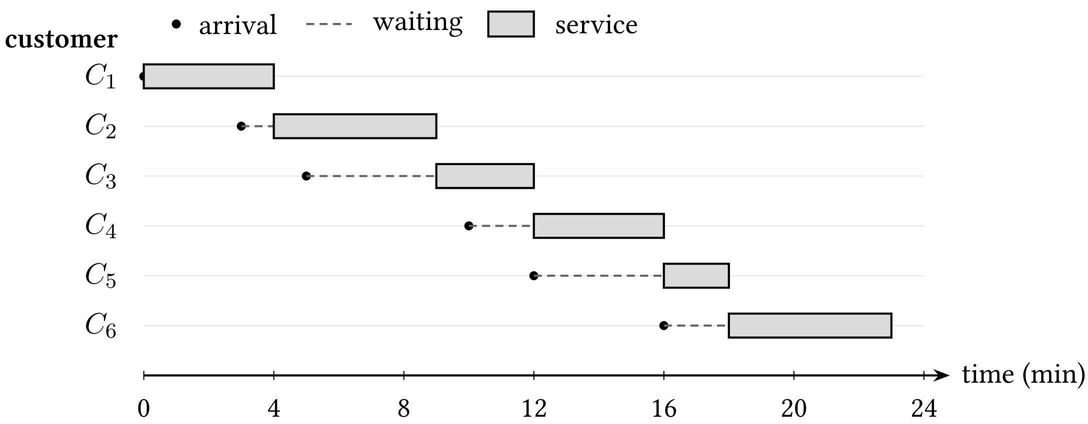

Figure 5: Customer-centred queue timeline. A filled point marks arrival, a dashed segment shows waiting, and a shaded block shows the service interval.

Policy comparison: Add a second counter **[15 min]**

Use the same arrivals and service times as Worked Example 3. Assign each arriving customer to the counter that becomes available first.

1. Complete a table containing counter number, start time, waiting time, and departure time.
2. Find the average and maximum waiting times.
3. Explain why this single sample path is insufficient evidence for permanently staffing a second counter.

5 From one run to a decision

A simulation study is an experiment on a model. Before running it, define the policy alternatives, input distributions, performance measures, simulation length, number of replications, and comparison method.

A useful preliminary check for a queue is the **traffic intensity**

$$\rho \approx \frac{\text{mean service time}}{c \times \text{mean interarrival time}},$$

where c is the number of servers. If $\rho > 1$, work arrives faster than the system can process it, so the queue tends to grow rather than settle to a stable long-run distribution.

Worked Example 4: Capacity before detailed simulation

Suppose the mean interarrival time is 3.5 minutes and the mean service time is 4.7 minutes.
For one counter,

$$\rho_1 = \frac{4.7}{1(3.5)} \approx 1.34 > 1.$$

The counter lacks sufficient average capacity. A long simulation will show increasing waits, but the key conclusion is structural: one counter cannot keep up under these assumptions.
For two counters,

$$\rho_2 = \frac{4.7}{2(3.5)} \approx 0.67 < 1.$$

A stable system is possible, although waiting-time variability still requires simulation. Capacity checks should accompany, not replace, detailed simulation.

Model criticism: Do not mistake a simulated average for a universal fact

A result such as "average wait $=$ 2.17 minutes" is conditional on the input distributions, queue discipline, staffing policy, operating period, and random sample. A credible report should provide:

1. the assumptions and data source for each random input;
2. the number and length of replications;
3. the mean together with variability or percentiles;
4. sensitivity to uncertain assumptions;
5. verification of the algorithm and validation against real observations.

Simulation study design **[10 min]**

The school wants to compare one counter, two counters, and one counter with a shorter menu.
Write a short experiment plan specifying:

1. the random inputs and how they would be estimated;
2. at least four outputs;
3. the simulation horizon and number of replications;
4. one fair comparison technique using common random seeds;
5. one validation check and one sensitivity test.

Verification and validation

Verification asks, "Did we implement the model correctly?" Trace small cases by hand, check conservation rules, and reproduce results with a fixed seed.
Validation asks, "Does this model represent the real system adequately for its purpose?" Compare simulated and observed queue lengths, waits, utilization, and arrival patterns. A program can be verified but still model the wrong system.

Transient warm-up and replication-level uncertainty

A long-run simulation started from an artificial empty state usually has a **transient**: early observations reflect initialization rather than steady operation. For a steady-state question, choose and justify a warm-up (or burn-in) period b, discard observations before b, and check whether conclusions are stable under nearby choices of b. Do not discard the beginning of a finite-horizon study—for example, one actual lunch period starting with an empty queue—because that initial condition is part of the system being modeled.
For R independent replications, first compute one output Y_r per replication. With replication mean $\bar{Y}$ and sample standard deviation s_Y, an approximate two-sided $100(1-\alpha)\%$ confidence interval is

$$\bar{Y} \pm t_{1-\alpha/2,\,R-1}\frac{s_Y}{\sqrt{R}}.$$

The replication, not each correlated customer or time step within it, is the unit of uncertainty. State the initialization, warm-up rule, run length, number of replications, and interval method together.

6 Lesson summary

Six ideas to retain

1. Simulation combines random inputs with deterministic process rules.
2. Uniform random numbers can be mapped to empirical outcomes through cumulative probabilities.
3. One Monte Carlo run is one possible outcome, not the answer.
4. Typical Monte Carlo error decreases only as $1/\sqrt{n}$.
5. In a queue, event order is captured by $B_i = \max(A_i, D_{i-1})$.
6. Decisions require replications, uncertainty summaries, verification, validation, and sensitivity analysis.

Checkpoint: Exit ticket **[2 min]**

Complete both statements:
"A fixed random seed is useful because ______________________________."
"A second seed or independent replication is necessary because ______________________________."

7 Practice problems

Core practice

1. A repair time T has distribution

T (h)	1	2	3	5
$P(T=t)$	0.10	0.40	0.35	0.15.

Construct the interval mapping and convert $U = 0.04, 0.36, 0.71, 0.92$ into repair times.

2. A quarter-circle simulation produces $C = 7{,}824$ inside points out of $n = 10{,}000$.
   (a) Estimate π.
   (b) If the sample size is multiplied by 25, by what factor should the typical Monte Carlo error change?
   (c) Explain why the actual error need not decrease in every individual run.

3. For a single-server queue, arrivals occur at

$$A = (0, 2, 6, 7, 11),$$

and service times are

$$S = (3, 5, 2, 4, 3).$$

Construct a complete table of start times, waits, and departures. Find the average wait, maximum wait, and average time in the system.

4. The mean interarrival time at a help desk is 6 minutes and the mean service time is 8 minutes.
   (a) Calculate ρ for one server and two servers.
   (b) What can be concluded before simulation?
   (c) What important information is still missing?

5. Explain why each statement is inadequate:
   (a) "The simulation was run once and gave an average wait of 4.2 minutes."
   (b) "The program has no coding errors, so the model is valid."
   (c) "The histogram fits last month's data, so it will remain correct all year."
   (d) "The two-counter system has a lower mean wait, so it is automatically the best policy."

6. A team compares two staffing policies using exactly the same sequence of arrivals and service requirements. Explain why this **common random numbers** design can make the comparison clearer. Why should the team still use several seeds?

Practice check and hints

1. The intervals are $[0, 0.10)$, $[0.10, 0.50)$, $[0.50, 0.85)$, and $[0.85, 1)$; the outputs are $(1, 2, 3, 5)$ h.

2. $\widehat{\pi} = 3.1296$. Multiplying n by 25 divides typical error by 5, although sampling variation need not improve monotonically in individual runs.

3. (B_i, W_i, D_i) equals $(0, 0, 3)$, $(3, 1, 8)$, $(8, 2, 10)$, $(10, 3, 14)$, $(14, 3, 17)$. Mean wait $= 1.8$ min, maximum wait $= 3$ min, and mean time in system $= 5.2$ min.

4. $\rho_1 = 8/6 \approx 1.33$ and $\rho_2 = 8/12 \approx 0.67$. One server lacks average capacity; two may be stable, but input distributions, dependence, queue discipline, and time variation remain relevant.
5. The four statements omit, respectively, replication uncertainty; validation; temporal stability; and cost, fairness, or other objectives.
6. Common random numbers reduce noise in paired policy differences; multiple independent seeds are still needed to estimate uncertainty and avoid a seed-specific conclusion.

Modeling challenge

Prepare a one-page simulation brief for one of the following systems:

1. a school canteen queue;
2. bicycle availability at a sharing station;
3. daily inventory of a popular stationery item;
4. patient arrivals at a small clinic.

Include: purpose, state variables, events, random inputs, input-data plan, process logic, outputs, policy alternatives, replication plan, validation, sensitivity, and one limitation. A complete computer program is not required.

A Optional algorithm: Single-server queue

This appendix is not part of the two-hour core lesson. It translates the event equations into pseudocode.

Queue simulation pseudocode

1. Set previous departure $D \leftarrow 0$ and initialize output lists.
2. For customer $i = 1, \dots, n$:
   (a) generate or read interarrival and service times;
   (b) update the arrival time A_i;
   (c) set $B_i \leftarrow \max(A_i, D)$;
   (d) set $W_i \leftarrow B_i - A_i$;
   (e) set $D_i \leftarrow B_i + S_i$ and update $D \leftarrow D_i$;
   (f) record waiting time, time in system, queue length, and server state.
3. Compute summary statistics for the replication.
4. Repeat with independent seeds and summarize across replications.

B Optional note: Standard error for the quarter-circle estimator

Let I_j equal 1 when sample point j lies inside the quarter circle and 0 otherwise. Then I_j is Bernoulli with $p = \pi/4$, and

$$\widehat{\pi} = 4\bar{I}.$$

Therefore

$$\mathrm{SE}(\widehat{\pi}) = 4\sqrt{\frac{p(1-p)}{n}} \approx \frac{1.64}{\sqrt{n}}.$$

This is a standard deviation of the estimator, not a guaranteed error bound. For example, increasing n from 100 to $10{,}000$ reduces the standard error by a factor of 10.

Model audit card

Audit before trusting or communicating a model

Purpose	What question or decision does the model support?	**Data**	Which inputs are observed, derived, or simulated?
Assumptions	Which simplifications may change the answer?	**Units**	Are dimensions and scales consistent?
Verification	Were equations, code, constraints, and limiting cases checked?	**Validation**	Is performance tested against independent evidence?
Uncertainty	How do measurement, parameter, solver, and structural errors affect conclusions?	**Ethics**	Who benefits, who bears risk, and what fairness criterion matters?
Reproducibility	Are data provenance, versions, seeds, and transformations recorded?	**Communication**	Are claims conditional, traceable, and decision-relevant?

Further reading

- Discrete-event simulation documentation (SimPy)
- Engineering Statistics Handbook (NIST / SEMATECH)

Sources, provenance, and licence

These notes follow the iterative modeling and assessment perspective in the SIAM/COMAP GAIMME report. Competition-specific remarks are a snapshot checked on 20 August 2026 against the official COMAP HiMCM instructions; readers should verify current rules before submission. Unless a source is explicitly identified, classroom datasets and numerical scenarios are synthetic or simulated teaching examples and are not empirical claims about a real institution. Third-party references retain their own terms.

Student response guide and checking criteria

For closed calculations, show the definition, substitution, units, and at least one independent check; the worked examples in this unit are method hints, but you must complete the arithmetic yourself. Open modeling responses are checked on five dimensions: a clear purpose and output; stated data and provenance; defensible assumptions and scope; traceable computational verification and model validation; and alignment among uncertainty, limitations, and the recommendation. Support each dimension with concrete evidence rather than writing only “completed.”