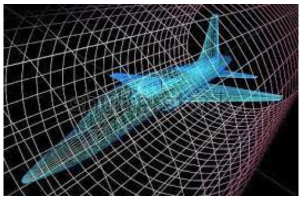
Lecture 18: Introduction to Simulation



Simulation

- Simulation ranks very high among the most widely used of operations research techniques.
- This technique involves using a computer to imitate (simulate) the operation of an entire process or system.
- For example, simulation is frequently used to perform risk analysis on financial processes.
- Simulation also is widely used to analyze stochastic systems that will continue operating indefinitely.
- For such systems, the computer randomly generates and records the occurrences of the various events that drive the system just as if it were physically operating.
- First you record the performance of the simulated operation of the system for a number of alternative designs or operating procedures.
- Then you evaluate and compare these alternatives before choosing one.

Simulating Airplane Flight

- Simulating airplane flight in a wind tunnel is standard practice when a new airplane is designed.
- Theoretically, the laws of physics could be used to obtain the same information → too complicated.
- Another alternative would be to build real airplanes with alternative designs and test them in actual flight to choose the final design → too expensive as well as unsafe.
- Therefore, after some preliminary theoretical analysis is performed to develop a *rough design,* simulating flight in a wind tunnel is a vital tool for experimenting with *specific designs.*
- After a detailed design is developed in this way, a prototype model can be built and tested in actual flight to fine-tune the final design.

The Role of Simulation in OR Studies

- Simulation plays essentially this same role in many OR studies.
- However, rather than designing an airplane, the OR team is concerned with developing a design or operating procedure for some stochastic system (a system that evolves probabilistically over time).
- A simulation model *synthesizes* the system by building it up component by component and event by event.
- Then the model *runs the simulated system to obtain statistical observations* of the performance of the system that result from various randomly generated events.

Basic Building Blocks of Simulation Models

- To prepare for simulating a complex system, a detailed simulation model needs to be formulated to describe the operation of the system and how it is to be simulated.
- A simulation model has several basic building blocks:
  1. A definition of the *state of the system* (e.g., the number of customers in a queuing system).
  2. Identify the *possible states of the system that can occur.*
  3. Identify the *possible events* (e.g., arrivals and service completions in a queuing system) that would change the state of the system.

Basic Building Blocks of Simulation Models

- Building blocks (Cont'd):
  4. A provision for a *simulation clock, located at some address in the simulation program,* that will record the passage of (simulated) time.
  5. A method for *randomly generating the events of the various kinds.*
  6. A formula for identifying *state transitions that are generated by the various kinds of* events.

When to Use Simulation?

- Simulation typically is used when the stochastic system involved is too complex to be analyzed satisfactorily by the kinds of mathematical models (e.g., queuing models).
- One of the main strengths of a mathematical model is that it abstracts the essence of the problem and reveals its underlying structure, thereby providing insight into the cause-and-effect relationships within the system.
- Therefore, if the modeler is able to construct a mathematical model that is both a reasonable idealization of the problem and amenable to solution, this approach usually is superior to simulation.
- However, many problems are too complex to permit this approach.
- Thus, simulation often provides the only practical approach to a problem.

Discrete-Event vs. Continuous Simulation

- Two broad categories of simulations are discrete-event and continuous simulations.
- A **discrete-event simulation** is one where changes in the state of the system occur instantaneously at random points in time as a result of the occurrence of *discrete events.*
- For example, in a queuing system where the state of the system is the number of customers in the system, the discrete events that change this state are the arrival of a customer and the departure of a customer due to the completion of its service.
- Most applications of simulation in practice are discrete-event simulations.

Discrete-Event vs. Continuous Simulation

- A **continuous simulation** is one where changes in the state of the system occur *continuously* over time.
- For example, if the system of interest is an airplane in flight and its state is defined as the current position of the airplane, then the state is changing continuously over time.
- Some applications of continuous simulations occur in design studies of such engineering systems.
- Continuous simulations typically require using differential equations to describe the rate of change of the state variables.
- Thus, the analysis tends to be relatively complex.

Discrete-Event vs. Continuous Simulation

- By approximating continuous changes in the state of the system by occasional discrete changes, it often is possible to use a discrete-event simulation to approximate the behavior of a continuous system.
- This tends to greatly simplify the analysis.
- This lecture focuses on discrete-event simulations.

Example 1: A Coin Flipping Game

- You are the lucky winner of a sweepstakes contest. Your prize is an all-expense-paid vacation at a major hotel in Las Vegas, including some chips for gambling in the hotel casino.

  Upon entering the casino, you find that, in addition to the usual games (blackjack, roulette, etc.), they are offering an interesting new game with the following rules.
- **Rules of the Game.**
  1. **Each play of the game involves repeatedly flipping an unbiased coin until the *difference*** between the number of heads tossed and the number of tails is 3.
  2. **If you decide to play the game, you are required to pay $1 for each flip of the coin.** You are not allowed to quit during a play of the game.
  3. **You receive $8 at the end of each play of the game.**

Example 1: A Coin Flipping Game

- Thus, you win money if the number of flips required is fewer than 8, but you lose money if more than 8 flips are required.
- Here are some examples (where H denotes a head and T a tail).

HHH	3 flips.	You win $5
THTTT	5 flips.	You win $3
THHTHTHTTTT	11 flips.	You lose $3

- How would you decide whether to play this game?

Example 2: An *M*/*M*/1 Queuing System

- Consider the *M*/*M*/1 queuing theory model (Poisson input, exponential service times, and single server).
- Although this model already has been solved analytically, it will be instructive to consider how to study it by using simulation.
- *To be specific, suppose that the values of the arrival rate and service rate are*

  $\lambda = 3$ per hour, $\mu = 5$ per hour.
- To summarize the physical operation of the system, arriving customers enter the queue, eventually are served, and then leave.
- Thus, it is necessary for the simulation model to describe and synchronize the arrival of customers and the serving of customers.

Example 2: An *M*/*M*/1 Queuing System

- Starting at time 0, the simulation clock records the amount of (simulated) time *t* that has transpired so far during the simulation run.
- The information about the queuing system that defines its current status, i.e., the state of the system, is

$$N(t) = \text{number of customers in system at time } t.$$

- The state transition formula is

$$\text{Reset } N(t) = \begin{cases} N(t) + 1 & \text{if arrival occurs at time } t \\ N(t) - 1 & \text{if service completion occurs at time } t. \end{cases}$$

Time Advance Methods

- There are two basic methods used for advancing the simulation clock and recording the operation of the system.
- **Time advance methods:**
  1. **Fixed-time incrementing**
  2. **Next-event incrementing**

Time Advance Methods

- With the **fixed-time incrementing time advance method, the following two-step procedure** is used repeatedly.

Summary of Fixed-Time Incrementing.

1. *Advance time* by a small *fixed amount.*
2. *Update the system* by determining what events occurred during the elapsed time interval and what the resulting state of the system is. Also record desired information about the performance of the system.

Time Advance Methods

- **Next-event incrementing differs from fixed-time incrementing in that the simulation** clock is incremented by a variable amount rather than by a fixed amount each time.
- This variable amount is the time from the event that has just occurred until the next event of any kind occurs; i.e., the clock jumps from event to event.

Summary of Next-Event Incrementing.

1. *Advance time* to the time of the *next event* of any kind.

2. *Update the system* by determining its new state that results from this event and by randomly generating the time until the next occurrence of any event type that can occur from this state (if not previously generated). Also record desired information about the performance of the system.

Example 2: An *M/M/*1 Queuing System

- **Fixed-Time Incrementing**
- The only two possible events during such an interval that need to be investigated are the arrival of one customer and the service completion for one customer. Each of these events has a known probability.
- To illustrate, let us use 0.1 hour (6 minutes) as the small fixed amount by which the clock is advanced each time.
- Because both interarrival times and service times have an exponential distribution, the probability P_A that a time interval of 0.1 hour will include an *arrival* is

$$P_A = 1 - e^{-3/10} = 0.259$$

Example 2: An *M/M/*1 Queuing System

- **Fixed-Time Incrementing**
- The probability P_D that it will include a departure (service completion), given that a customer was being served at the beginning of the interval, is

$$P_D = 1 - e^{-5/10} = 0.393$$

$$r_A < 0.259 \Rightarrow \text{arrival occurred,}$$
$$r_A \geq 0.259 \Rightarrow \text{arrival did not occur.}$$

$$r_D < 0.393 \Rightarrow \text{departure occurred,}$$
$$r_D \geq 0.393 \Rightarrow \text{departure did not occur}$$

Example 2: An *M/M/*1 Queuing System

TABLE 22.1 Fixed-time incrementing applied to Example 2

t, time (min)	$N(t)$	r_A	Arrival in Interval?	r_D	Departure in Interval?
0	0				
6	1	0.096	Yes	—	
12	1	0.569	No	0.665	No
18	1	0.764	No	0.842	No
24	0	0.492	No	0.224	Yes
30	0	0.950	No	—	
36	0	0.610	No	—	
42	1	0.145	Yes	—	
48	1	0.484	No	0.552	No
54	1	0.350	No	0.590	No
60	0	0.430	No	0.041	Yes

$$\text{Est}\{W\} = \frac{3+3}{2}(0.1 \text{ hour}) = 0.3 \text{ hour.} \qquad W = 1/(\mu - \lambda) = 0.5$$