

# MG26018 Simulation Modeling and Analysis

## 仿真建模与分析

### Lecture 1: Introduction to Simulation

SHEN Haihui 沈海辉

Sino-US Global Logistics Institute  
Shanghai Jiao Tong University

🏠 [shenhaihui.github.io/teaching/mg26018](https://shenhaihui.github.io/teaching/mg26018)  
✉️ shenhaihui@sjtu.edu.cn

Fall 2019



上海交通大学  
SHANGHAI JIAO TONG UNIVERSITY

董浩云航运与物流研究院  
CY TUNG Institute of Maritime and Logistics  
中美物流研究院  
Sino-US Global Logistics Institute



- ① What is Simulation?
- ② Why Simulation?
- ③ How to Do Simulation?
- ④ Models
  - ▶ Definition
  - ▶ Types of Simulation Models
- ⑤ Examples
  - ▶ Estimate  $\pi$ : Buffon's Needle
  - ▶ Estimate  $\pi$ : Random Points
  - ▶ Numerical Integration
  - ▶ System Time to Failure
- ⑥ Course Outline

# What is Simulation?

- *Simulation* (仿真) is the imitation of the operation of a real-world process or system over time.
  - Done by hand or (usually) on a computer;
  - Involves the generation and observation of an artificial history of a system;
  - Draw inferences about the characteristics of the real system.
- Simulation is EVERYWHERE!

# What is Simulation?



Figure: Pilot Training in Boeing 787 Flat Panel Trainer (from [Boeing](#))

# What is Simulation?

**Figure: Airport Simulation** (*by Vancouver Airport Services*)  
[Video: <https://www.youtube.com/watch?v=JuXwEbAvk2Q>]



# What is Simulation?

Figure: Typhoon Simulation ([image](#) by [Atmoz](#) / [CC BY 3.0](#))

# What is Simulation?

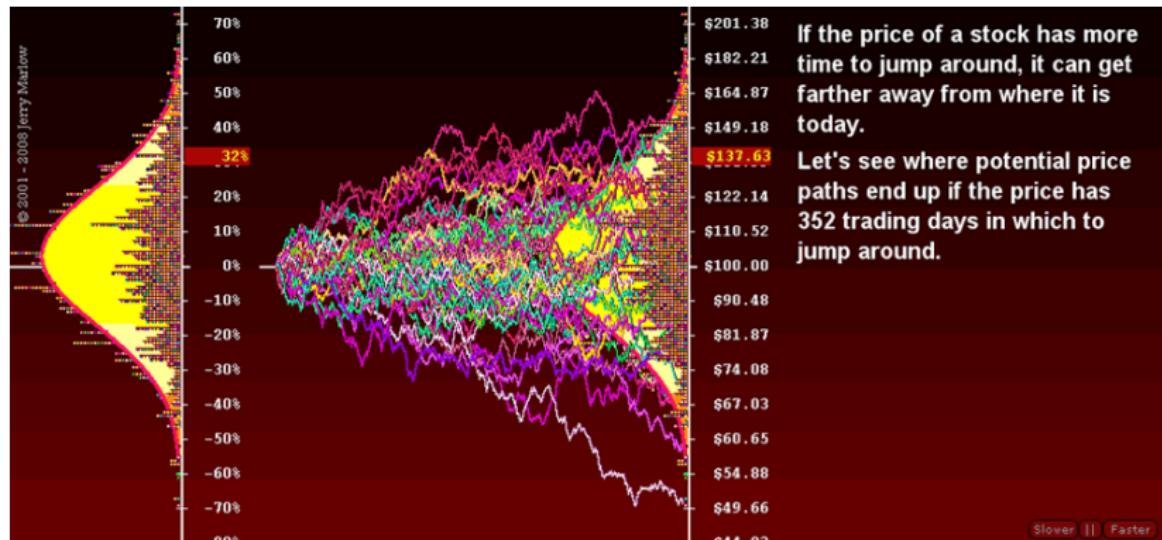


Figure: Financial Analysis

# Why Simulation?

- It is often too costly or even impossible to do physical studies in reality with the actual system.
  - May be *disruptive*, *expensive*, *dangerous*, or *rare*.
- The mathematical **model** (will be defined shortly) which can well represent the real problem, may be very *difficult* to solve.
  - You can only solve it with high *simplification*.
- With simulation technique, we can easily make change and observe the effect, while keeping high fidelity.

# Why Simulation?

- Simulation can be used as both an *analysis tool* and a *design tool*.
- ① An analysis tool: To answer “**what if**” questions about the existing real-world system.
  - E.g., try alternative layout of a production line, try other staff shifts of a service center, test a financial system in some extreme situation, etc.
- ② A design tool: To study systems in the design stage, before they are built.
  - E.g., evaluate designs and operations for new transportation facilities, service organizations, manufacturing systems, etc.
- Simulation is also an important type of numerical methods.



# How to Do Simulation?

- This is the focus of the course!

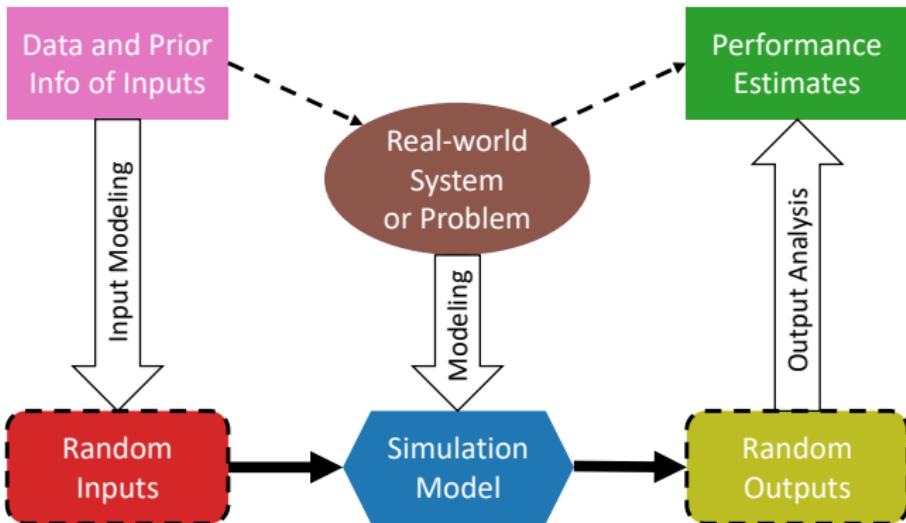


Figure: Basic Paradigm of A Simulation Study

- A **model** is a representation of a system or problem.
  - A set of **assumptions** and/or **approximations** about how the system works will often be imposed.
  - It is only necessary to consider those aspects that affect the problem under investigation.
  - However, the model should be sufficiently **detailed** to draw **valid** conclusions about the real system or problem.
  - The trade-off: **simplicity** vs. **accuracy**.
- Physical model vs. Mathematical model
  - ① **Physical model**: a scaled-down (or -up) version of the system.
  - ② **Mathematical model**: uses symbolic notation and mathematical equations to represent the system.
- Instead of doing physical studies with the actual system in real world, we can study the model.
  - It will be much easier, faster, cheaper, and safer!
- A **simulation model** is a particular type of **mathematical model**.





“ All models are wrong,  
but some are useful. ”

— George E. P. Box

George E. P. Box (1919.10 – 2013.03) was a British statistician, who worked in the areas of quality control, time-series analysis, design of experiments, and Bayesian inference. He has been called “one of the great statistical minds of the 20th century”.

- When a mathematical model is simple enough, we can **solve** it
  - *analytically*, with mathematical tools like algebra, calculus, probability theory.
  - *numerically*, with computational procedures (e.g., solving a quintic equation).
- But not all mathematical models can be “**solved**”.
- In simulation, the mathematical models (more specifically, simulation models) are **run** rather than **solved**:
  - Artificial history of the system is *generated* from the model assumptions;
  - Observations of system status are *collected* for analysis;
  - System performance measures are *estimated*.
- Essentially, running simulation is still one type of numerical methods.
  - Real-world simulation models can be large, and such runs are usually conducted with the aid of a computer.

- Simulation models may be classified as being *static* or *dynamic*.
- ① Static: Time does not play a **natural** role.
- Example 1 – Finance: evaluate portfolio return and risk.
  - Example 2 – Project Management: evaluate projects payoff in different scenarios.
  - Sometimes called **Monte Carlo (蒙特卡洛) simulation**.
  - Often used in the complex numerical calculation in financial engineering (金融工程), computational physics, etc.
- ② Dynamic: Time does play a **natural** role.
- Example 1 – Logistics Management: evaluate the efficiency of a terminal.
  - Example 2 – Service Management: evaluate waiting time of customers under different staff shifts.
  - Often used to simulate the logistics/transportation/service systems, whose status naturally changes over time.

- Simulation models may be classified as being *deterministic* or *stochastic*.
- ① Deterministic: Everything is known with **certainty**.
- E.g., patients arrive at a hospital precisely on schedule, the service time is precisely fixed, the transfer among different units is pre-determined.
- ② Stochastic: **Uncertainty** exists.
- E.g., arrival times and service times of patients have random variations, the transfer is random.
  - Used much more often (uncertainty is more or less involved in a real-world system).

- Simulation models may be classified as being *discrete* or *continuous*.
- ① Discrete: System states change only at **discrete** time points.
- E.g., the number of customers in the bank, changes only when a customer arrives or leaves after service (*left fig*).
- ② Continuous: System states change **continuously** over time.
- E.g., the head of water (水位) behind a dam changes continuously during a period of time (*right fig*).

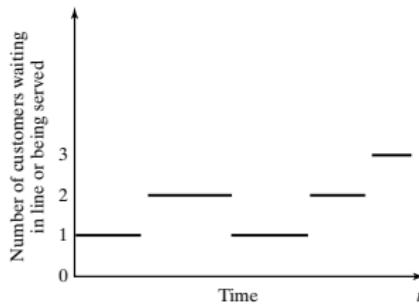


Figure: Discrete State (from Banks et al. (2010))

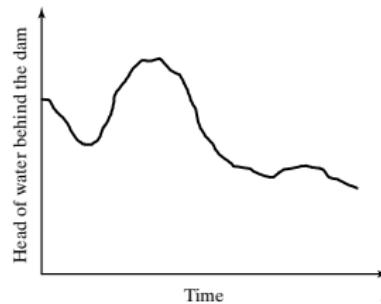
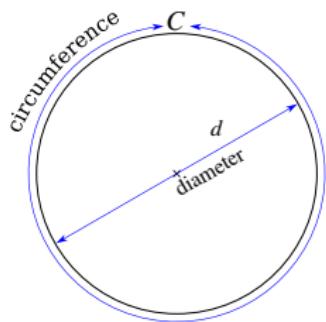


Figure: Continuous State (from Banks et al. (2010))

- In summary, simulation models may be classified as being *static* or *dynamic*, *deterministic* or *stochastic*, and *discrete* or *continuous*.
- For most operational decision-making problems, the suitable simulation models are *dynamic*, *stochastic* and *discrete*.
  - The simulation is called **Discrete-Event System Simulation** (离散事件系统仿真).
  - It is the main **focus** of this course.

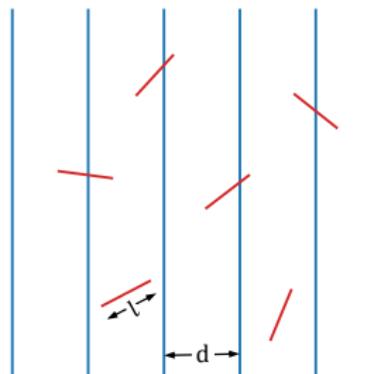
- The mathematical constant  $\pi$ , is originally defined as the ratio of circle's circumference to its diameter.



$$\pi = \frac{\text{circumference}}{\text{diameter}} = 3.14159\dots$$

- It was considered as a quite difficult problem in the history of mankind to find the value of  $\pi$ .

- Buffon's Needle (布丰投针)
  - Buffon, a French mathematician, in 1733 (1777) did a static simulation (by hand), which can be used to estimate  $\pi$ .
  - Drop a needle of length  $l$  onto the floor with parallel lines  $d$  apart, where  $l < d$ .
  - Suppose the needle is *equally likely* to fall anywhere.

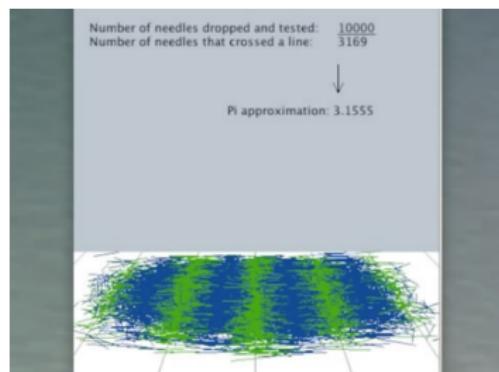


- $\mathbb{P}(\text{needle crosses a line}) = \frac{2l}{\pi d}.$

- If Buffon repeats the experiment for  $n$  times (i.e., drops  $n$  needles), and let  $h$  denote the number of needles crossing a line, then,

$$\mathbb{P}(\text{needle crosses a line}) = \frac{2l}{\pi d} \approx \frac{h}{n}.$$

- So,  $\pi \approx \frac{2ln}{dh}$ .
- Let  $d = 2l$ , then  $\pi \approx n/h$ .
- The approximation gets more and more accurate when  $n$  increases.



- Try it out!

<https://mste.illinois.edu/activity/buffon>

<http://datagenetics.com/blog/may42015/index.html>

Figure: A Computer Simulation (by Jeffrey Ventrella)

[Video: <https://www.youtube.com/watch?v=kazgQXaeOHk>]

- Now consider another simulation to estimate  $\pi$ .
  - Randomly throw  $n$  dots to a square.
  - Suppose the dots are *equally likely* to fall anywhere inside the square.
  - Let  $h$  denote the number of dots in the circular sector.

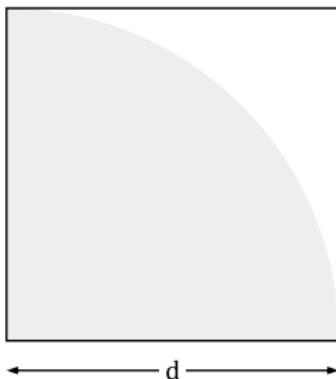
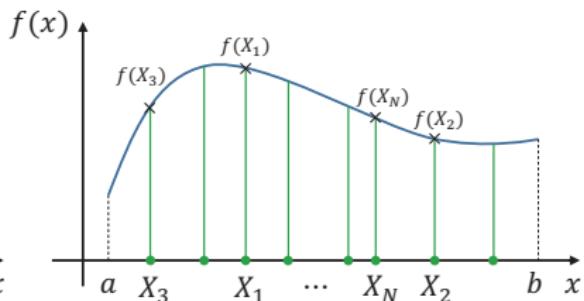
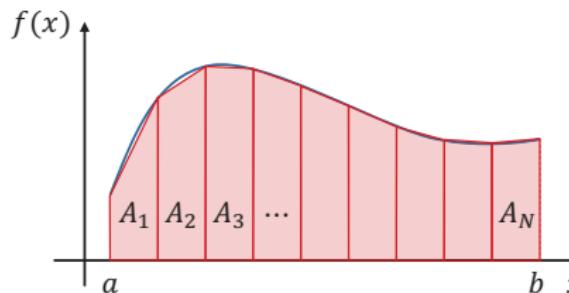


Figure: Animation ([image](#) by [nicoguardo](#)) / [CC BY 3.0](#)

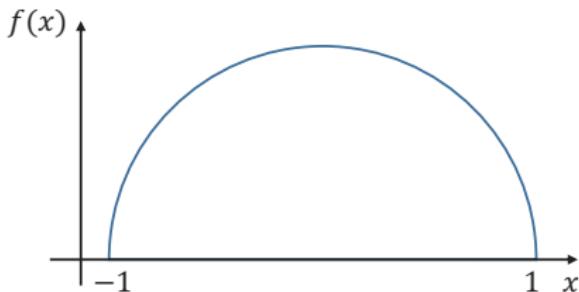
- $\mathbb{P}(\text{dot in sector}) = \frac{\text{sector area}}{\text{square area}} = \frac{\pi d^2/4}{d^2} \approx \frac{h}{n} \Rightarrow \pi \approx \frac{4h}{n}$ .
- Visit <https://xiaoweiz.shinyapps.io/calPi> for interaction.

- Consider a numerical integration (数值积分)  $\int_a^b f(x)dx$ .



- Trapezoidal rule (梯形法) (*left fig*)
  - Divide the area into  $N$  parts.
  - $\int_a^b f(x)dx \approx A_1 + A_2 + \dots + A_N$ .
- Monte Carlo method (*right fig*)
  - Randomly sample  $N$  points on  $[a, b]$  from Uniform $[a, b]$ .
  - $\int_a^b f(x)dx \approx \frac{b-a}{N} [f(X_1) + f(X_2) + \dots + f(X_N)]$ .
- Monte Carlo method will be much more “**efficient**” when the dimension of  $x$  is high! (E.g.,  $\int_{[a,b]^d} f(\mathbf{x})d\mathbf{x}$  for large  $d$ .)

- Recall the numerical integration problem  $\int_a^b f(x)dx$ .
- Let  $f(x) = \sqrt{1 - x^2}$ ,  $a = -1$ ,  $b = 1$ .



- Then,  $\int_{-1}^1 \sqrt{1 - x^2} dx = \pi/2$ .
- So we have another way to estimate  $\pi$  using Monte Carlo simulation (provided we know how to compute square root).

- There is a system:
  - Two components work as active and spare, so the system fails if both are failed.
  - The lifetime of a component is equally likely 1, 2, 3, 4, 5 or 6 days.
  - Repair takes exactly 2.5 days (only one at a time).
- What can we say about the time to failure for this system?
- Let's run a simulation by hand!
  - The **state** of the system is the number of functional components.
  - The **events** are the failure of a component and the completion of repair.

Event Calendar			
Clock	System State	Next Failure	Next Repair
0	2	$0 + \textcolor{red}{5} = 5$	$\infty$
5	1	$5 + \textcolor{red}{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	$\infty$
8	1	$8 + \textcolor{red}{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	$\infty$
14	1	$14 + \textcolor{red}{1} = 15$	$14 + 2.5 = 16.5$
15	<b>0</b>	$\infty$	16.5

- We can observe:
  - Time to failure = 15
  - Average number of functional components =  

$$\frac{1}{15-0} [2(5-0) + 1(7.5-5) + 2(8-7.5) + 1(10.5-8) + 2(14-10.5) + 1(15-14)] = \frac{24}{15}$$
- Some questions:
  - How to deal with the randomness?
  - How to generate the time interval of component failure?

# Course Outline

- Introduction to Simulation
- Queueing Models
- Random Variate Generation
- Input Modeling
- Verification and Validation of Simulation Models
- Output Analysis I: Single Model
- Simulation in Excel and Arena
- Output Analysis II: Comparison and Optimization