MEM6810 Engineering Systems Modeling and Simulation
工程系统建模与仿真

Theory Analysis

Lecture 1: Introduction to Simulation

SHEN Haihui 沈海辉

Sino-US Global Logistics Institute
Shanghai Jiao Tong University

shenhaihui.github.io/teaching/mem6810f
shenhaihui@sjtu.edu.cn

Spring 2024 (full-time)

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY
董浩云智能制造与服务管理研究院
CY TUNG Institute of Intelligent Manufacturing and Service Management
（中美物流研究院）
(Sino-US Global Logistics Institute)

Contents

What is Simulation?

- *Simulation* (仿真) is the imitation of the operation of a real-world process or system over time.

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.

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?

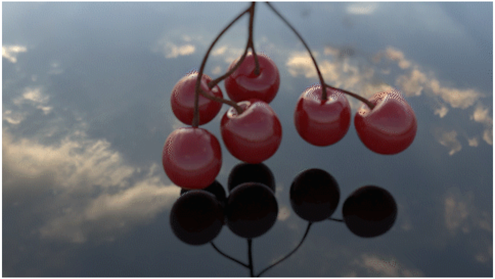


Figure: Physical Simulation of Solid-Fluid Interaction (*from* *Ruan et al. (2021)*)

What is Simulation?

Figure: Pilot Training in Boeing 787 Flat Panel Trainer (*from* Boeing)

SHANGHAI JIAO TONG UNIVERSITY

Figure: Airport Simulation (*by Vancouver Airport Services*)

[Video: https://www.youtube.com/watch?v=JuXwEbAvk2Q]

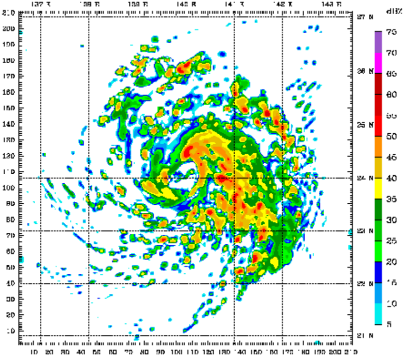


Figure: Typhoon Simulation (*image* by *Atmoz* / *CC BY 3.0*)

What is Simulation?

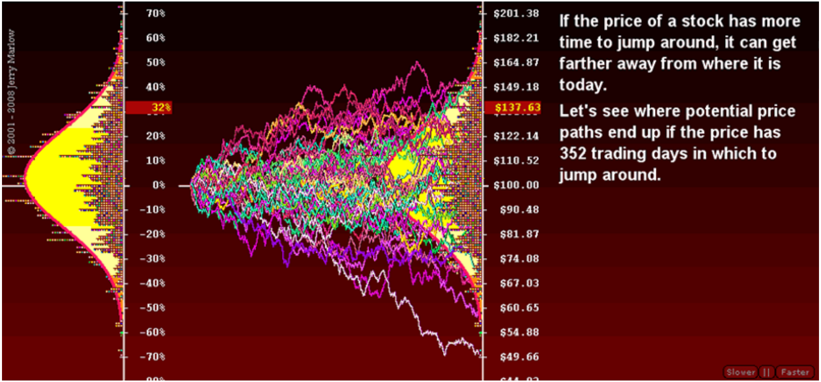


Figure: Financial Analysis

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

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*.

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*.

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*.

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.

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

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.

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

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!

- This is the focus of the course!

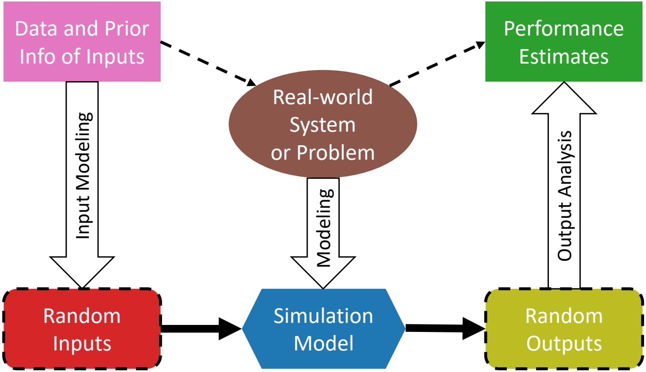


Figure: Basic Paradigm of A Simulation Study

- A model is a representation of a system or problem.

- A 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.

- 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.

- 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.

- 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.

- 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

- 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 is a scaled-down (or -up) version of the system.

- 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 is a scaled-down (or -up) version of the system.
  2. Mathematical model uses symbolic notation and mathematical equations to represent the system.

- 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 is 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.

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

- 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 is a scaled-down (or -up) version of the system.
  2. Mathematical model uses symbolic notation and mathematical equations to represent the system.

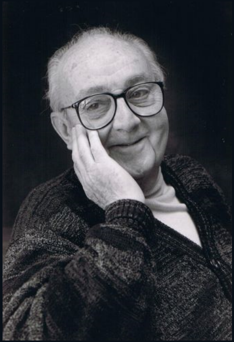
- Instead of doing physical studies with the actual system in real world, we can study the model.
  - It will be much easier, faster, cheaper, and safer!

- A 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 is a scaled-down (or -up) version of the system.
  2. Mathematical model uses symbolic notation and mathematical equations to represent the system.

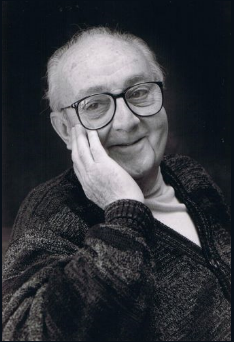
- Instead of doing physical studies with the actual system in real world, we can study the model.
  - It will be much easier, faster, cheaper, and safer!

- A simulation model is a particular type of mathematical model.



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

— *George E. P. Box*



“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).

- 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".

- 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:

- 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*.

- 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*.

- Simulation models may be classified as being *static* or *dynamic*.

1. Static: Time does not play a **natural** role.

2. Dynamic: Time does play a **natural** role.

- 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.

2. Dynamic: Time does play a **natural** role.

- 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.

2. Dynamic: Time does play a **natural** role.

Figure: Monte Carlo Casino (*photo* by *Cristian Lorini* / *CC BY-SA 3.0*)

Figure: Monte Carlo Casino (*photo* by *Cristian Lorini* / *CC BY-SA 3.0*)

Figure: Monte Carlo Casino (*photo* by *Cristian Lorini* / *CC BY-SA 3.0*)

- 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.

2. Dynamic: Time does play a **natural** role.

- 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.

- 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.

- 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*.

- Simulation models may be classified as being *deterministic* or *stochastic*.

1. Deterministic: Everything is known with **certainty**.

2. Stochastic: **Uncertainty** exists.

- 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.

- 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.

- 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*.

- Simulation models may be classified as being *discrete* or *continuous*.

1. Discrete: System states change only at **discrete** time points.

2. Continuous: System states change **continuously** over time.

- 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.

- 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.

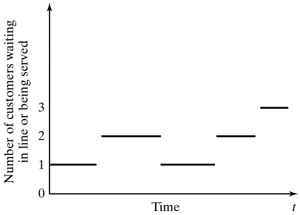


Figure: Discrete State (*from* *Banks et al. (2010)*)

- 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*).

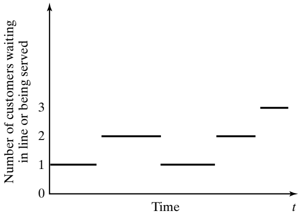


Figure: Discrete State (*from* Banks et al. (2010))

- 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*).

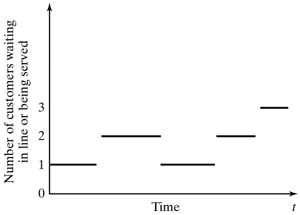


Figure: Discrete State (*from* Banks et al. (2010))

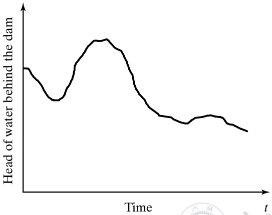


Figure: Continuous State (*from* Banks et al. (2010))

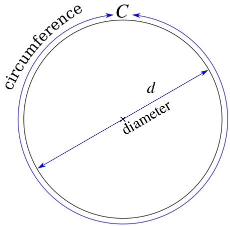
- In summary, simulation models may be classified as being *static* or *dynamic*, *deterministic* or *stochastic*, and *discrete* or *continuous*.

- 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*.

- 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.

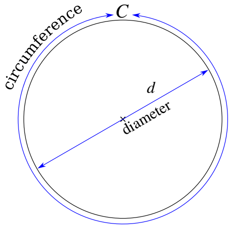
- The mathematical constant π, is originally defined as the ratio of circle's circumference to its diameter.

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



$$\pi = \frac{\text{circumference}}{\text{diameter}} = 3.14159\,26535\ldots$$

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



$$\pi = \frac{\text{circumference}}{\text{diameter}} = 3.14159\,26535\ldots$$

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

- The earliest written approximations of π:
  - Babylon: A clay tablet (1900–1600 BC), $\pi \approx \frac{25}{8} = 3.125$;
  - Egypt: The Rhind Papyrus (莱因德纸草书, 1650 BC, 1850 BC), $\pi \approx (\frac{16}{9})^2 = 3.160\ldots$.

- The earliest written approximations of π:
  - Babylon: A clay tablet (1900–1600 BC), $\pi \approx \frac{25}{8} = 3.125$;
  - Egypt: The Rhind Papyrus (莱因德纸草书, 1650 BC, 1850 BC), $\pi \approx (\frac{16}{9})^2 = 3.160\ldots$.

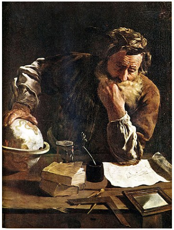


Figure: Archimedes of Syracuse (287–212 BC) (Source/Photographer)

$$\frac{223}{71} < \pi < \frac{22}{7}$$
$$\frac{223}{71} = 3.1408...$$
$$\frac{22}{7} = 3.1428...$$

- The earliest written approximations of π:
  - Babylon: A clay tablet (1900–1600 BC), $\pi \approx \frac{25}{8} = 3.125$;
  - Egypt: The Rhind Papyrus (莱因德纸草书, 1650 BC, 1850 BC), $\pi \approx (\frac{16}{9})^2 = 3.160\ldots$.

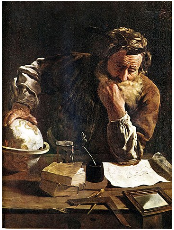


Figure: Archimedes of Syracuse (287–212 BC) (Source/Photographer)

$$\frac{223}{71} < \pi < \frac{22}{7}$$
$$\frac{223}{71} = 3.1408...$$
$$\frac{22}{7} = 3.1428...$$

Figure: Liu Hui (刘徽, 魏晋时期, 225–295 AD)

$$\pi \approx 3.1416$$

- The earliest written approximations of π:
  - Babylon: A clay tablet (1900–1600 BC), $\pi \approx \frac{25}{8} = 3.125$;
  - Egypt: The Rhind Papyrus (莱因德纸草书, 1650 BC, 1850 BC), $\pi \approx (\frac{16}{9})^2 = 3.160\ldots$.

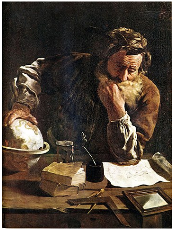


Figure: Archimedes of Syracuse (287–212 BC) (Source/Photographer)

$$\frac{223}{71} < \pi < \frac{22}{7}$$
$$\frac{223}{71} = 3.1408...$$
$$\frac{22}{7} = 3.1428...$$

Figure: Liu Hui (刘徽, 魏晋时期, 225–295 AD)

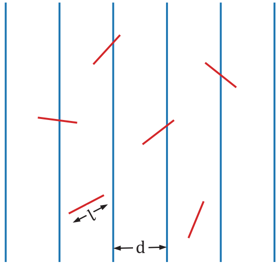
$$\pi \approx 3.1416$$

Figure: Zu Chongzhi (祖冲之, 南北朝时期, 429–500 AD) (statue image by 三猎 / CC BY 4.0)

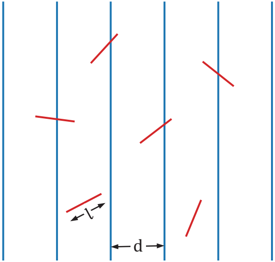
$$\pi \approx \frac{355}{113} = 3.14159\ 292...$$

- 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 π.

- 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 π.
  - 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.



- 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 π.
  - 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.



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

- 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, $\pi \approx \dfrac{2ln}{dh}$.

- 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, $\pi \approx \dfrac{2ln}{dh}$.
- Let $d = 2l$, then $\pi \approx n/h$.

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

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

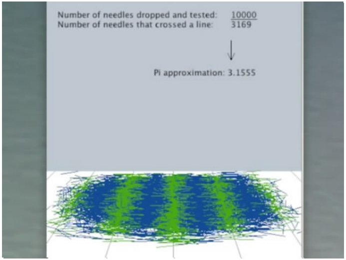


Figure: A Computer Simulation (*by Jeffrey Ventrella*)
[Video: https://www.youtube.com/watch?v=kazgQXae0Hk]

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

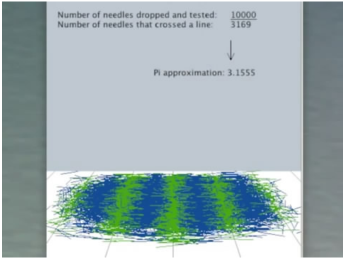


Figure: A Computer Simulation (*by Jeffrey Ventrella*)
[Video: https://www.youtube.com/watch?v=kazgQXae0Hk]

- Try it out!

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

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

- Now consider another simulation to estimate π.

- Now consider another simulation to estimate π.
  - Randomly throw n dots to a square.

- Now consider another simulation to estimate π.
  - 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.

- Now consider another simulation to estimate π.
  - 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.

- $\mathbb{P}(\text{dot in sector}) = \dfrac{\text{sector area}}{\text{square area}} = \dfrac{\pi d^2/4}{d^2} \approx \dfrac{h}{n}.$

- Now consider another simulation to estimate π.
    - 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.

- $\mathbb{P}(\text{dot in sector}) = \dfrac{\text{sector area}}{\text{square area}} = \dfrac{\pi d^2/4}{d^2} \approx \dfrac{h}{n}. \quad \Rightarrow \pi \approx \dfrac{4h}{n}.$

- Now consider another simulation to estimate π.
  - 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.

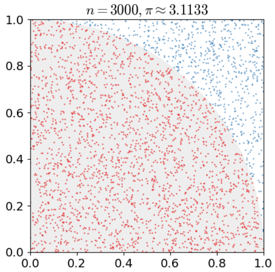


Figure: Animation (image by nicoguaro / CC BY 3.0)

- $\mathbb{P}(\text{dot in sector}) = \dfrac{\text{sector area}}{\text{square area}} = \dfrac{\pi d^2/4}{d^2} \approx \dfrac{h}{n}. \quad \Rightarrow \quad \pi \approx \dfrac{4h}{n}.$

- Now consider another simulation to estimate π.
  - 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.

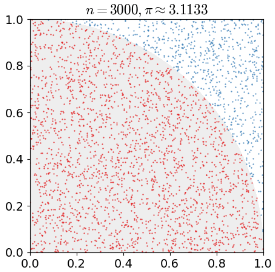
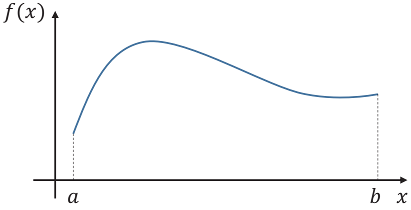


Figure: Animation (image by nicoguaro / CC BY 3.0)

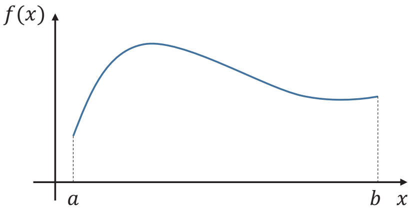
- $\mathbb{P}(\text{dot in sector}) = \dfrac{\text{sector area}}{\text{square area}} = \dfrac{\pi d^2/4}{d^2} \approx \dfrac{h}{n}. \quad \Rightarrow \quad \pi \approx \dfrac{4h}{n}.$
- Visit https://xiaoweiz.shinyapps.io/calPi for interaction.

- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.

- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.

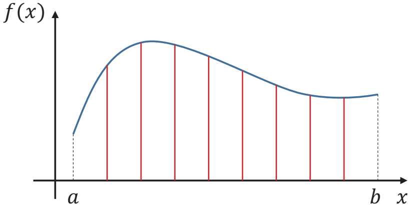


- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.



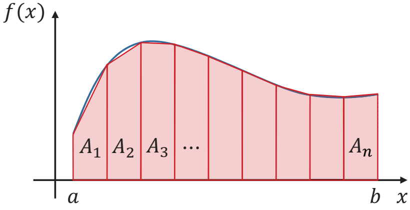
- Trapezoidal rule (梯形法):

- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.



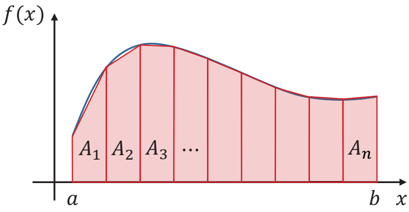
- Trapezoidal rule (梯形法):
  1. Divide the area into n parts.

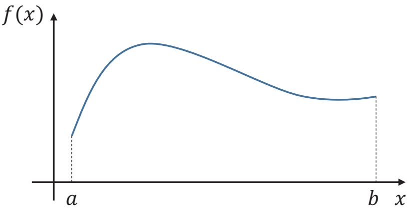
- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.



- Trapezoidal rule (梯形法):
  1. Divide the area into n parts.
  2. $\int_a^b f(x)\mathrm{d}x \approx A_1 + A_2 + \cdots + A_n$.

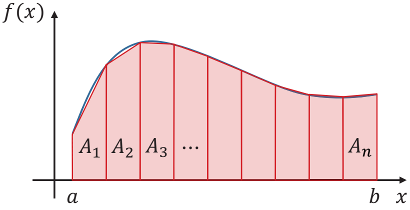
- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.

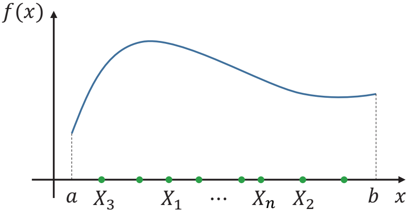




- Trapezoidal rule (梯形法) (*left fig*):
  1. Divide the area into n parts.
  2. $\int_a^b f(x)\mathrm{d}x \approx A_1 + A_2 + \cdots + A_n$.
- Monte Carlo method (*right fig*):

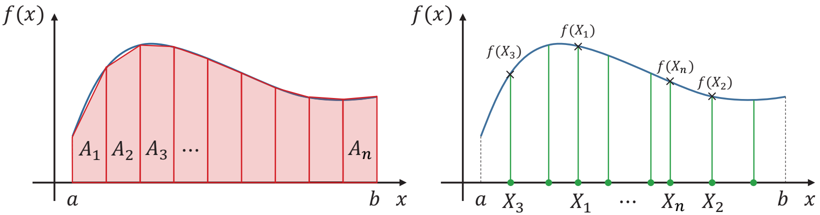
- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.





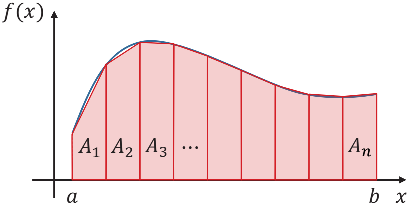
- Trapezoidal rule (梯形法) (*left fig*):
  1. Divide the area into n parts.
  2. $\int_a^b f(x)\mathrm{d}x \approx A_1 + A_2 + \cdots + A_n$.
- Monte Carlo method (*right fig*):
  1. Randomly sample n points on $[a, b]$ from $\text{uniform}(a, b)$.

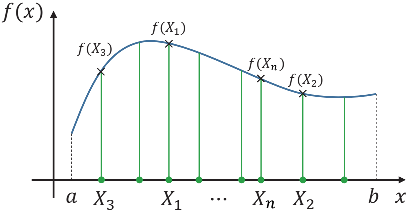
- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.



- Trapezoidal rule (梯形法) (*left fig*):
  1. Divide the area into n parts.
  2. $\int_a^b f(x)\mathrm{d}x \approx A_1 + A_2 + \cdots + A_n$.
- Monte Carlo method (*right fig*):
  1. Randomly sample n points on $[a, b]$ from $\mathrm{uniform}(a, b)$.
  2. $\int_a^b f(x)\mathrm{d}x \approx \frac{b-a}{n}\left[f(X_1) + f(X_2) + \cdots + f(X_n)\right]$.

- Consider a numerical integration (数值积分) $\int_a^b f(x)\mathrm{d}x$.

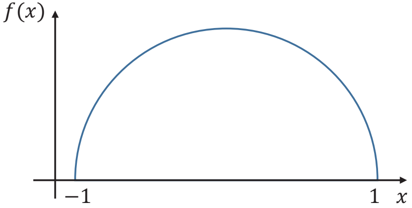




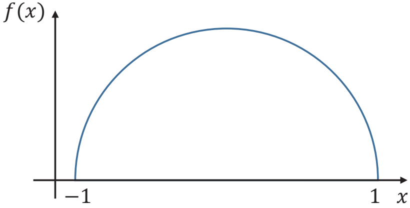
- Trapezoidal rule (梯形法) (*left fig*):
  1. Divide the area into n parts.
  2. $\int_a^b f(x)\mathrm{d}x \approx A_1 + A_2 + \cdots + A_n$.
- Monte Carlo method (*right fig*):
  1. Randomly sample n points on $[a, b]$ from uniform(a, b).
  2. $\int_a^b f(x)\mathrm{d}x \approx \frac{b-a}{n}[f(X_1) + f(X_2) + \cdots + f(X_n)]$.
- Monte Carlo method will be much more **efficient** when the dimension is high! (E.g., $\int_{[a,b]^d} f(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$ for large d.)

- Recall the numerical integration problem $\int_a^b f(x)\mathrm{d}x$.

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

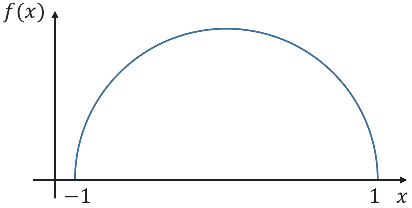


- Recall the numerical integration problem $\int_a^b f(x)\mathrm{d}x$.
- Let $f(x) = \sqrt{1-x^2}$, $a=-1$, $b=1$.



- Then, $\int_{-1}^{1} \sqrt{1-x^2}\mathrm{d}x = \pi/2$.

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



- Then, $\int_{-1}^{1} \sqrt{1-x^2}\mathrm{d}x = \pi/2$.
- So we have another way to estimate π 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 components are failed.
  - Suppose the time to next component failure is random (when there is at least one functional components), which follows a known distribution, and we know how to generate it.
  - To make it simple, suppose the time to next failure is equally likely 1, 2, 3, 4, 5 or 6 days (no memory).
  - Repair takes exactly 2.5 days (only one at a time).

- There is a system:
  - Two components work as active and spare, so the system fails if both components are failed.
  - Suppose the time to next component failure is random (when there is at least one functional components), which follows a known distribution, and we know how to generate it.
  - To make it simple, suppose the time to next failure is equally likely 1, 2, 3, 4, 5 or 6 days (no memory).
  - Repair takes exactly 2.5 days (only one at a time).

- What can we say about the time to failure for this system?

- There is a system:
  - Two components work as active and spare, so the system fails if both components are failed.
  - Suppose the time to next component failure is random (when there is at least one functional components), which follows a known distribution, and we know how to generate it.
  - To make it simple, suppose the time to next failure is equally likely 1, 2, 3, 4, 5 or 6 days (no memory).
  - 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!
  - Let the system **state** denote 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		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1		$5 + 2.5 = 7.5$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0+\mathbf{5}=5$	∞
5	1	$5+\mathbf{3}=8$	$5+2.5=7.5$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2		∞

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1		$8 + 2.5 = 10.5$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0+\mathbf{5}=5$	∞
5	1	$5+\mathbf{3}=8$	$5+2.5=7.5$
7.5	2	8	∞
8	1	$8+\mathbf{6}=14$	$8+2.5=10.5$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2		∞

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1		$14 + 2.5 = 16.5$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	$\mathbf{0}$		

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	$\mathbf{0}$		16.5

Clock	System State	Event Calendar: Next Failure	Event Calendar: Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	$\mathbf{0}$	∞	16.5

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	**0**	∞	16.5

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	$\mathbf{0}$	∞	16.5

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

$$\frac{1}{15-0}\left[2(5-0) + 1(7.5-5) + 2(8-7.5) + 1(10.5-8) + 2(14-10.5) + 1(15-14)\right] = \frac{24}{15}$$

		Event Calendar	
Clock	System State	Next Failure	Next Repair
0	2	$0 + \mathbf{5} = 5$	∞
5	1	$5 + \mathbf{3} = 8$	$5 + 2.5 = 7.5$
7.5	2	8	∞
8	1	$8 + \mathbf{6} = 14$	$8 + 2.5 = 10.5$
10.5	2	14	∞
14	1	$14 + \mathbf{1} = 15$	$14 + 2.5 = 16.5$
15	**0**	∞	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
- Elements of Probability and Statistics
- Queueing Models
- Random Variate Generation
- Input Modeling
- Verification and Validation of Simulation Models
- Output Analysis I: Single Model
- Simulation in Excel and FlexSim
- Output Analysis II: Comparison
- Output Analysis III: Optimization