# NETWORK **SIMULATION**

CPSC 441 - Tutorial 12

Winter 2018

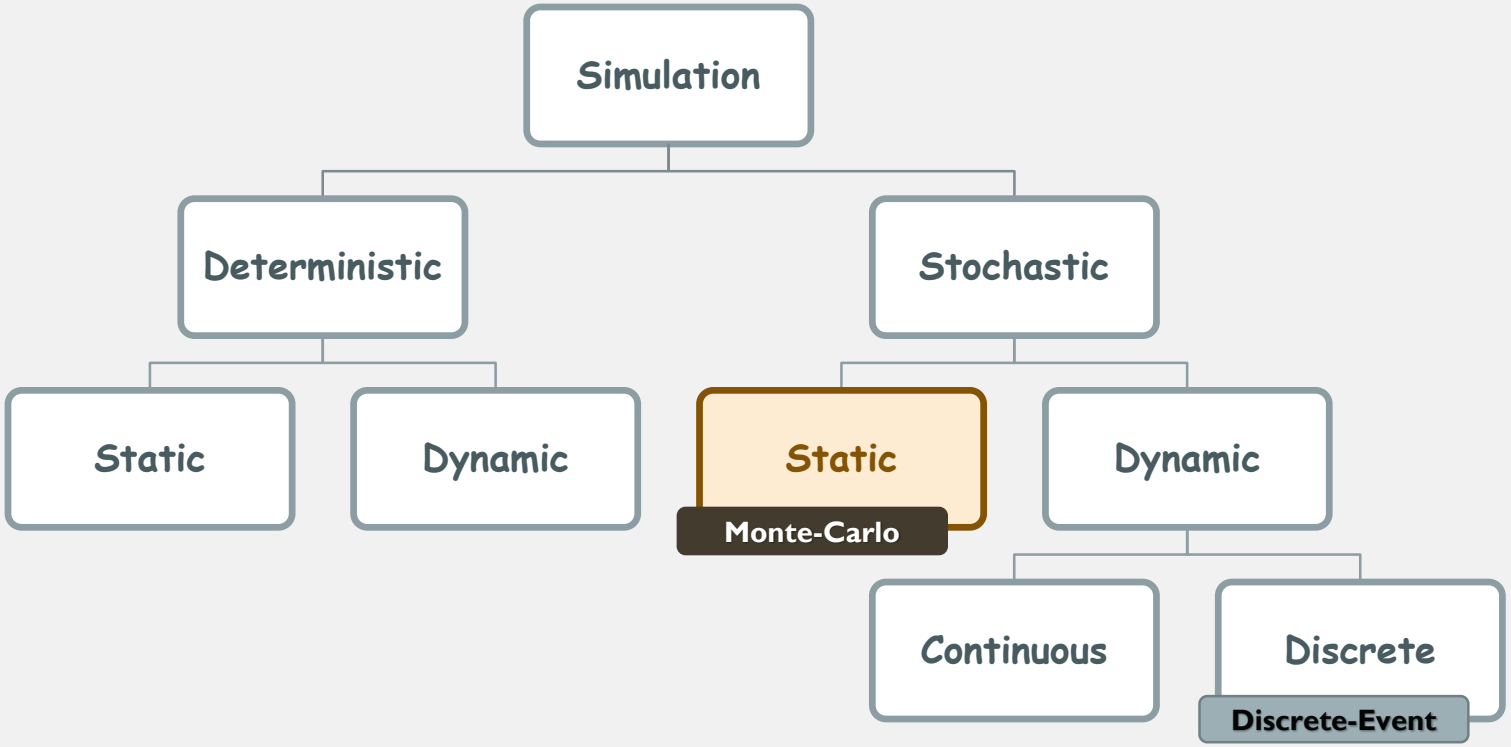
UNIVERSITY OF CALGARY

# PERFORMANCE **EVALUATION**

# **WHEN** TO USE SIMULATION

- The problem cannot be solve analytically (or analytical model is too complex)
  - Assignment 4!
- Direct experiment cannot be done
  - Network protocol scalability test
- Building the real system is expensive
  - Reservoir simulation

# SIMULATION MODEL **TAXONOMY**



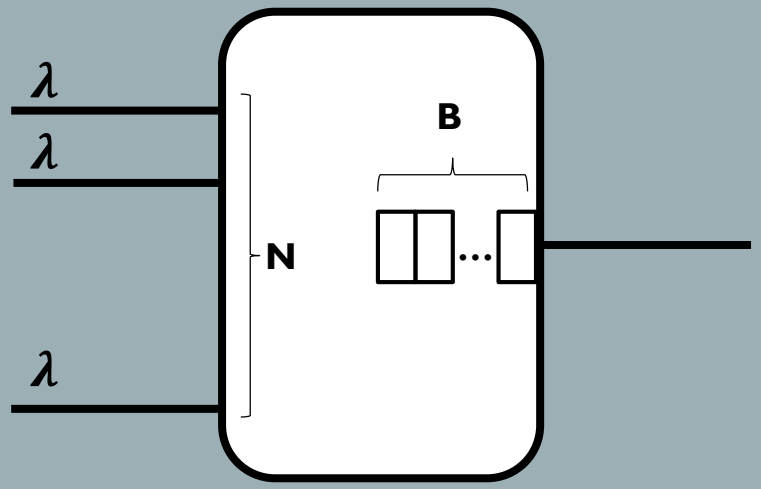

UNIVERSITY OF CALGARY

# MONTE CARLO SIMULATION

- Static Simulation (no time dependency)
- Can be used for:
  - Modeling a probabilistic phenomenon – Card games
  - Evaluating non-probabilistic expressions using probabilistic methods – Estimating $\pi$
  - Estimating quantities that are "hard" to determine analytically or experimentally – Assignment 4

# SWITCH EXAMPLE

- A slotted-time switch
  - N input links
  - One output link
  - Finite buffer (B) at the output link
  - Packet Arrival rate at each link is $\lambda$




- **Question:**
  - What is the packet-loss rate at output buffer for N= 4 and B=2 and $\lambda = 0.5$?
  - What is the effect of changing N on the packet-loss rate?
  - What is the effect of changing B on the packet-loss rate?

- **Answer:**
  - **Simulation!**

# COMMON **MISTAKES** IN SIMULATION

- Poor Random Number Generation
  - Seed control
  - Range control
- Inappropriate level of detail
  - More detail → More time → More bugs
  - More parameters != More accuracy
- Improper initialization
- Run-Length issues
  - Too short → Invalid output
  - Too long → Unnecessary cost

# RANDOM NUMBER **GENERATION**

- Controlling seed
- Generating numbers within the desired range

```
1.  int seed = 12345;
2.  int randint = 0;
3.  int max_index = 10;
4.  srandom(seed);
5.  while(randint == 0)
6.  {
7.    randint = random();
8.    randint = randint % (max_index + 1)
9.  }
```

- `max_index`: Upper bound of the desired range.
- This code is for generates a uniform random number between **1** to **max_index**.

# ASSIGNMENT 4

- Calculate success-rate for:
  - Specific k
  - Specific sending-stations
  - Specific starting-level

1

- Repeat **1** for:
  - Specific k
  - **100 random** cases for sending-stations
- Calculate average of success-rate for 100 cases

2

- Repeat **2** for:
  - **Different values** for starting-level i = 0, 2, 4, 6, 8, 10

3

- Repeat **3** for:
  - **Different values** of k = 1, 2 , 4, 8, 16, …, 1024

# REFERENCES

- http://pages.cpsc.ucalgary.ca/~carey/CPSC531/slides.html