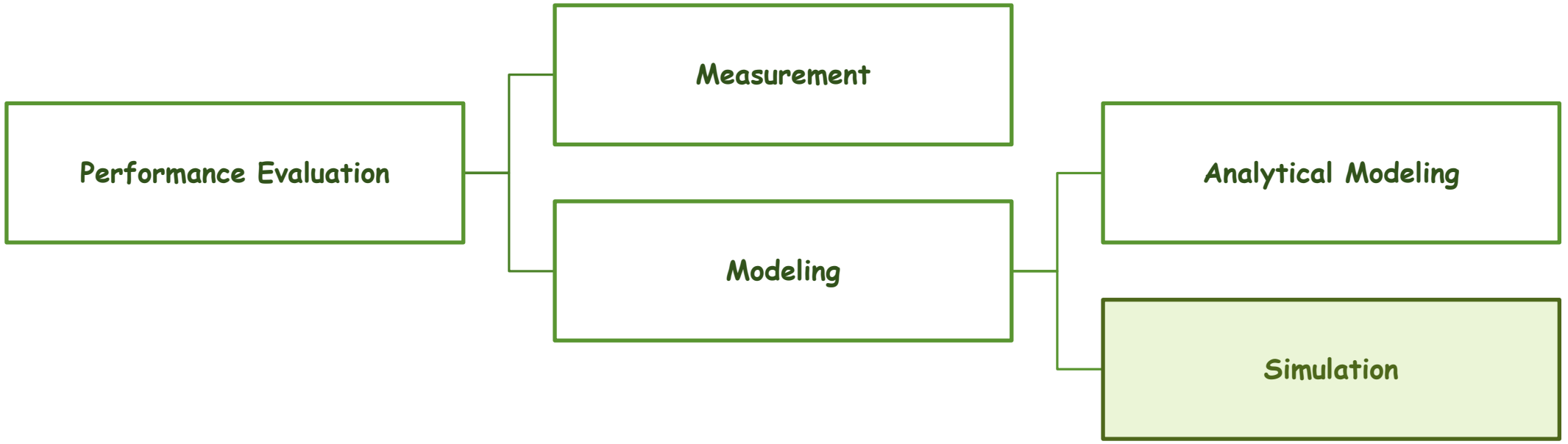
Network Simulation

CPSC 441 - TUTORIAL 8

WINTER 2020

Performance Evaluation



When to Use Simulation

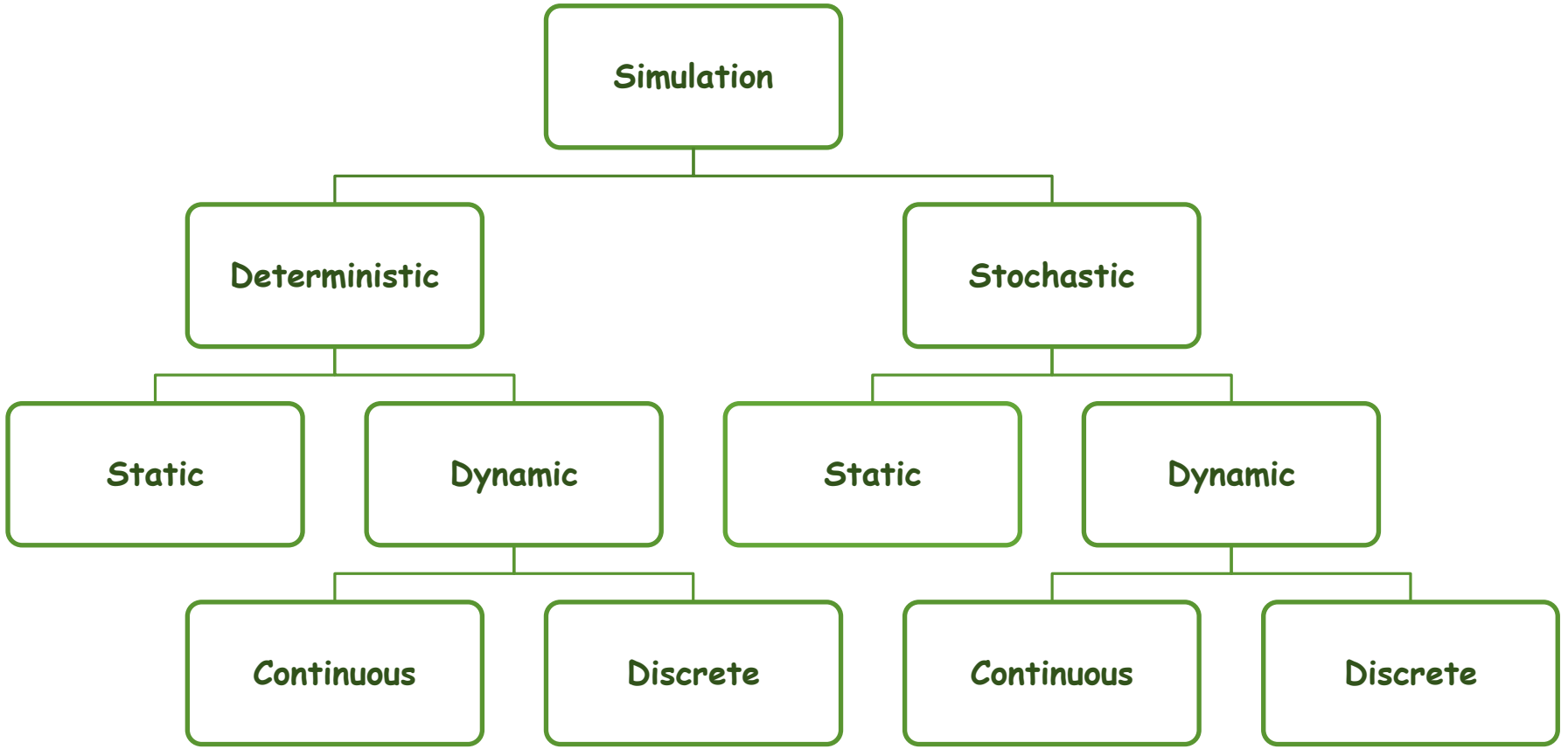
The problem cannot be solved analytically (or analytical model is too complex)

Direct experiment cannot be done

Building the real system is expensive

Assignment 3!

Simulation Model Taxonomy



Development Process

Determine Goals	What are you trying to figure out?
Conceptual Model	What is the basic concept of the system?
Specification Model	How can the actions of the system be defined as finite actions/values?
Computational Model	How can those actions/values be represented in code?
Verification	Does the model make sense? Does it work the way it's suppose to?
Validation	Does the model answer the right question?

Discrete Event Simulation

Event

- Causes simulation state to change
- Data structure for storing/communicating event information
- Should include Event Time (needed for sorting the event list and/or advancing current time of simulation) and Event Type (needed to determine how to handle event)
- Other event variables…

Event List

- Data structure for storing simulation Events
- Need to be able to sort events in time order

Basic Questions

How many events are there? How many types of events?

- Arrivals?
- Departures?
- Is the maximum number of events fixed?

How to handle data structure? (insertion and deletion)

- Array
- Linked list (single/double)
- Binary tree
- Queue

What information do you need?

Basic Setup

1. Read first event from event list
2. Update any simulation state variables impacted by event
3. Update any statistic variables impacted by event
4. Schedule any new events that follow from current event (ex: Departure event usually follows Arrival event)
5. Read next event…

Main Loop Pseudocode

```
1. while (currentEvent != STOP){
2.      switch (currentEvent.type){
3.              case EVENT_TYPE_1:
4.                      updateStateVariables();
5.                      updateStatisticVariables();
6.                      eventList.add(new Event()); //if necessary
7.                      eventList.sort(); //by event time
8.                      break;
9.              case EVENT_TYPE_2:
10.             …
11.             case EVENT_TYPE_N:
12.     }
13.     currentEvent = eventList.getNext();
14.}
```

Example: Phone Lines

There is a call center with three phone lines to answer customer calls

Up to three callers may occupy the phone lines at once

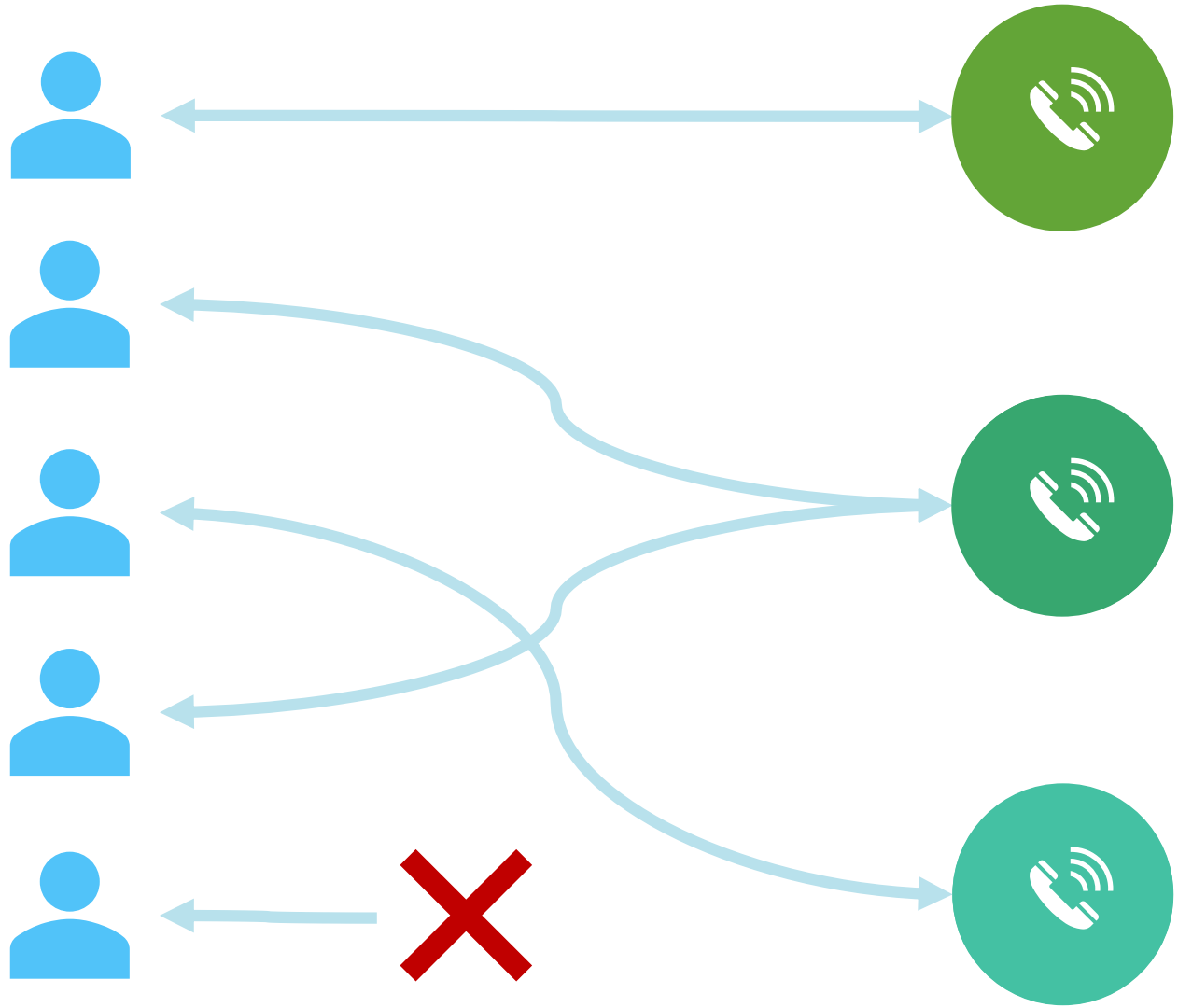
If all phone lines are busy when a customer calls, the customer's call is missed

Want to find out the percentage of calls missed

Example: Phone Lines

Conceptual Model

- First call arrives
- Second call arrives
- Third call arrives
- Second call terminates
- Forth call arrives
- Fifth call missed



Example: Phone Lines

Specification Model

Call Arrivals

- randomInterArrivalTime()

Call Duration

- randomCallDuration()

Event Types

- Call arrived
- Call ended

Event Structure

- Type
- Time

State Variables

- Time
- Number of phone lines available

Statistic Variables

- Number of calls missed/completed/answered
- Total number of calls made

Example: Phone Lines

Specification Model

Call Arrivals

- randomInterArrivalTime()

Call Duration

- randomCallDuration()

Call Arrives

- If no lines are available
  - Increment the number of missed calls
- If a line is available
  - Decrease the number of lines available
  - Increment the number of calls answered
  - Add the end of the call to the event list
- Either way:
  - Add the next call arrival to the event list

Call Ends

- Increment the number of calls completed
- Incrment the number of lines available

Example: Phone Lines

Computational Model

State Variables:

- Time (initially 0)
- Lines Available (initially 3)

Statistic Variables

- Number of Callers
- Number of Calls Answered
- Number of Calls Missed
- Number of Calls Completed

```
1.while (currentEvent != STOP){
2.   time = currentEvent.time
3.   switch (currentEvent.type){
4.        case CALL_ARRIVED:
5.             numCallers++;
6.             if (linesAvailable <= 0){
7.                  numCallsMissed++;
8.             } else {
9.                  linesAvailable--;
10.                 numCallsAnswered++;
11.                 eventList.add(new Event(
                         CALL_ENDED,
                         currentEvent.time + randomCallDuration()
                    ));
12.                 eventList.sort();
13.            }
14.            eventList.add(new Event(
                    CALL_ARRIVED,
                    currentEvent.time + randomInterArrivalTime()
               ));
15.            eventList.sort();
16.            break;
17.       case CALL_ENDED:
18.            numCallsCompleted++;
19.            linesAvailable++;
20.  }
21.  currentEvent = eventList.getNext();
22.}
```

Assignment 3

Network Topology

- End points
- Propagation delay
- Maximum load

Trace File

- Call start time
- Call origin
- Call destination
- Call duration

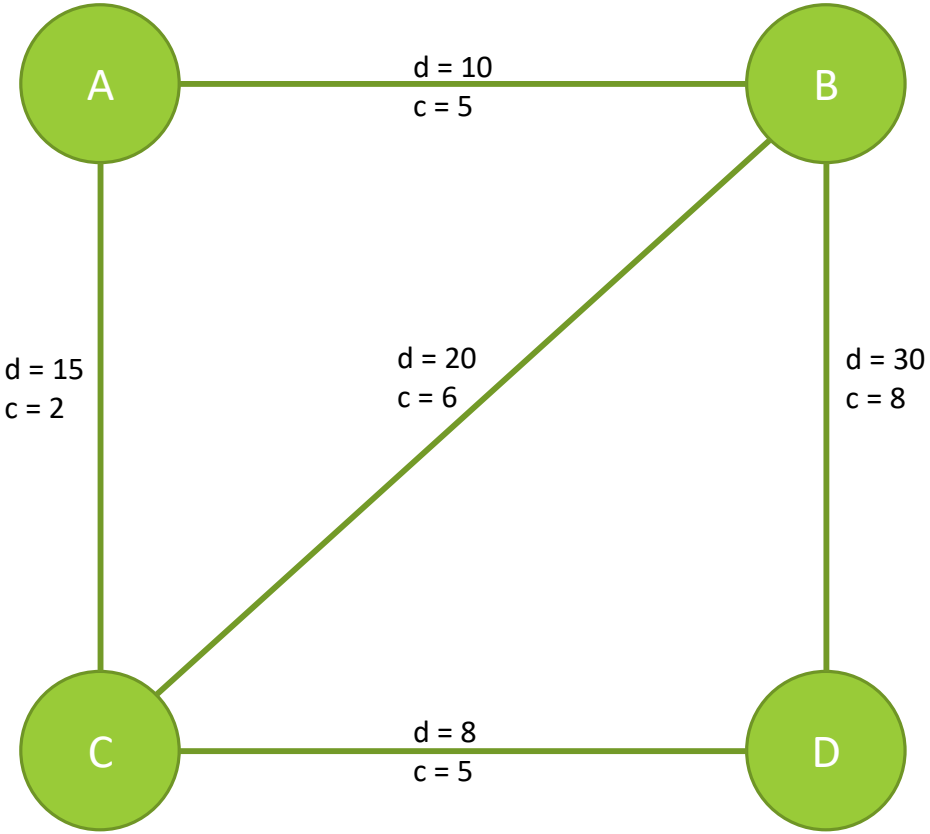
Routing Algorithms

- Shortest Hop Path First (SHPF)
- Shortest Delay Path First (SDPF)
- Least Loaded Path (LLP)
- Maximum Free Circuits (MFC)

Assignment 3

Example: Call from A to D

- ABD
  - # Hops = 2
  - Propagation Delay = 40
  - Free Circuits = 5
- ACD
  - # Hops = 2
  - Propagation Delay = 23
  - Free Circuits = 2
- ABCD
  - # Hops = 3
  - Propagation Delay = 38
  - Free Circuits = 5
- ACBD
  - # Hops = 3
  - Propagation Delay = 65
  - Free Circuits = 2



Assignment 3

Events:

- Call arrivals (Read from trace file)
- Call termination (Scheduled according to call length from trace file)

Statistic Variables:

- Total number of calls read from the workload file
- Number/percentage of successfully routed calls
- Number/percentage of blocked calls
- Average number of hops (links) per successfully routed call
- Average end-to-end propagation delay per successfully routed call

*Similar to “Phone Lines” example

Assignment 3

Event Handling:

- Have to use routing algorithm to connect two endpoints according to provided graph topology

State Variables:

- Have to track which links are busy and which links are free

References

https://pages.cpsc.ucalgary.ca/~carey/CPSC531/slides/CPSC531-DES.pdf

https://pages.cpsc.ucalgary.ca/~carey/CPSC531/slides/CPSC531-Events.pdf

https://pages.cpsc.ucalgary.ca/~carey/CPSC441/archive/W2018/tutorials/Simulation.pdf