ECE/CS 541
Computer System Analysis:
Intro to Discrete Event Simulation

Mohammad A. Noureddine
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Fall 2018

Learning Objectives

- Or what is this course about?

- At the start of the semester, you should have
  - Basic programming skills (C++, Python, etc.)
  - Basic understanding of probability theory (ECE313 or equivalent)

- At the end of the semester, you should be able to
  - Understand different system modeling approaches
    - Combinatorial methods, state-space methods, etc.
  - Understand different model analysis methods
    - Analytic/numeric methods, simulation
  - **Understand the basics of discrete event simulation**
  - Design simulation experiments and analyze their results
  - Gain hands-on experience with different modeling and analysis tools

Motivation

- The state spaces of some models can be completely explored in a reasonable amount of time
- When state spaces become too large, discrete event simulation is an attractive alternative, as it can be used to solve models with arbitrarily large state spaces

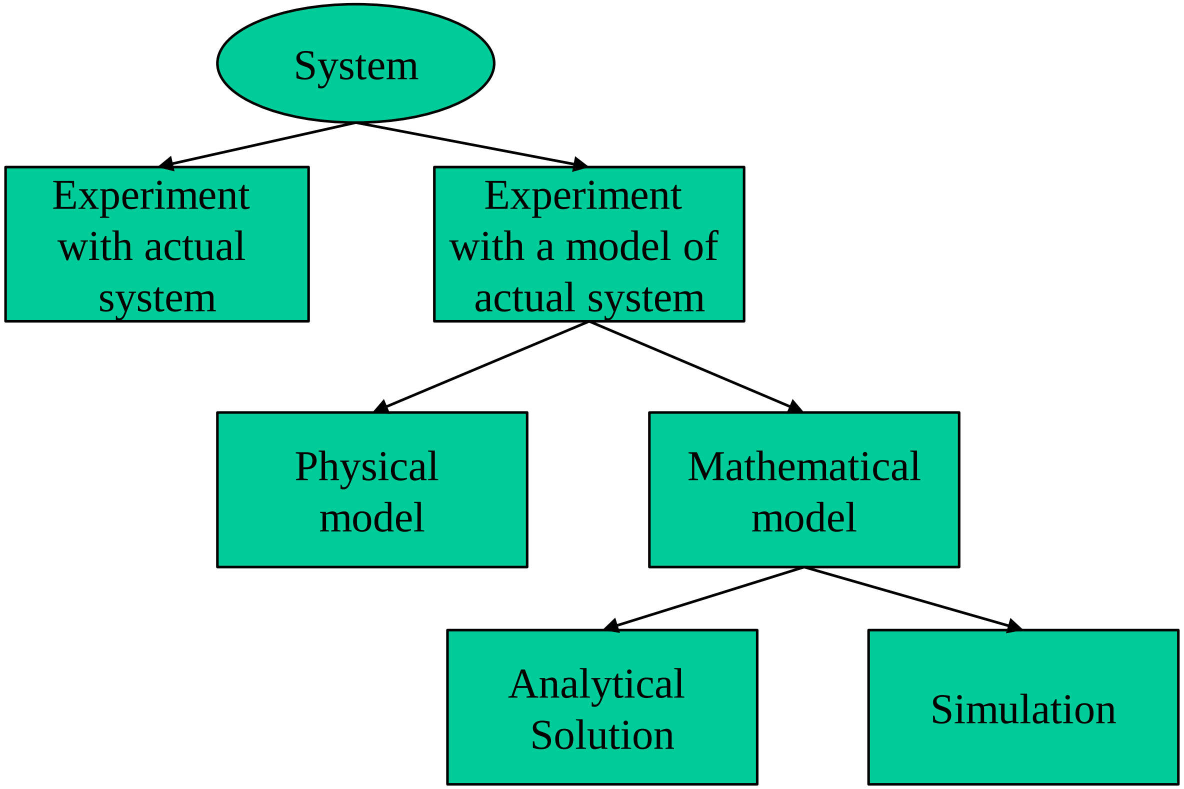


Figure based on *Simulation, Modeling & Analysis* (3/e) by Law and Kelton, 2000, p. 4, Figure 1.1

Discrete Event Simulation Basics

- Basic elements of a discrete event simulation are *variables* and *events*
- Variables include
  - Clock
  - System State Variables
  - Metrics
- When an event occurs, at least one of the variables change

Discrete Event Simulation Basics

- Basic elements of a discrete event simulation are *variables* and *events*
- Variables include
  - **Clock**
  - System State Variables
  - Metrics
- When an event occurs, at least one of the variables change

Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

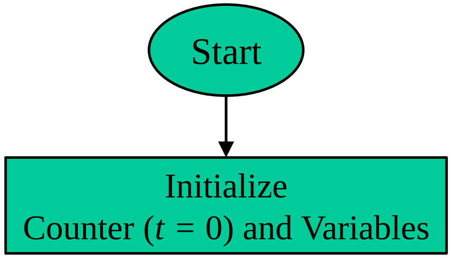
Simulation clock advanced a fixed amount of time each step

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

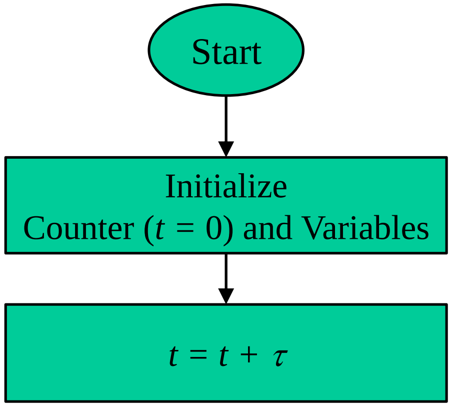
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step



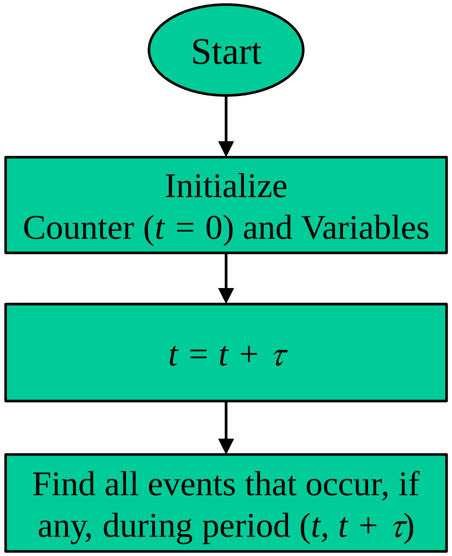
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step



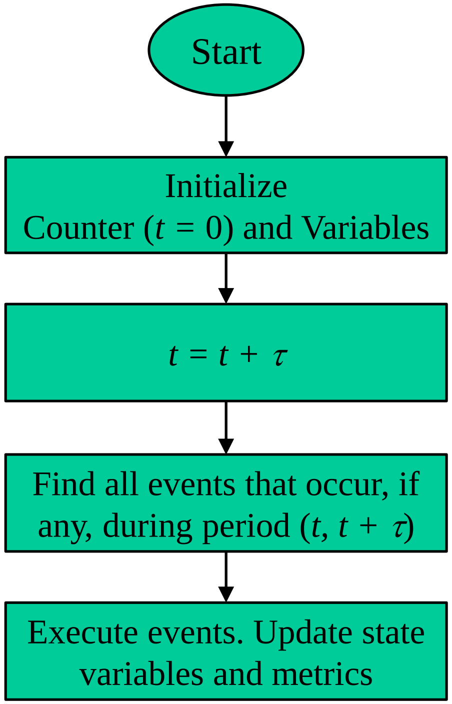
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step



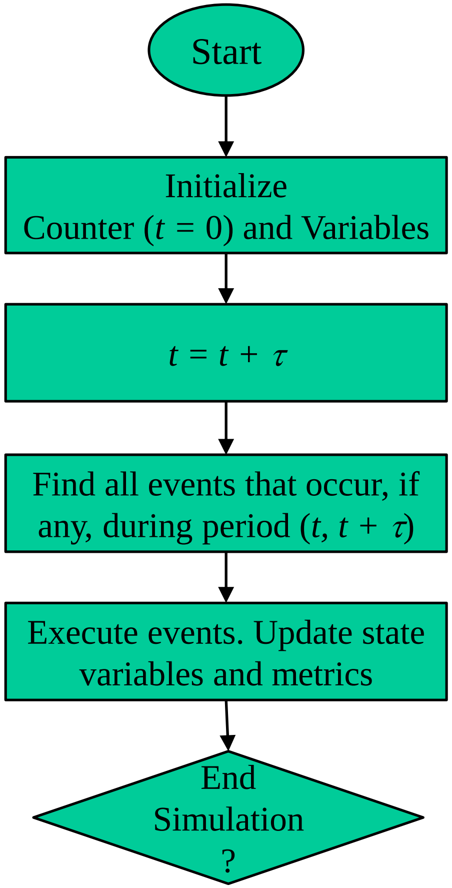
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step



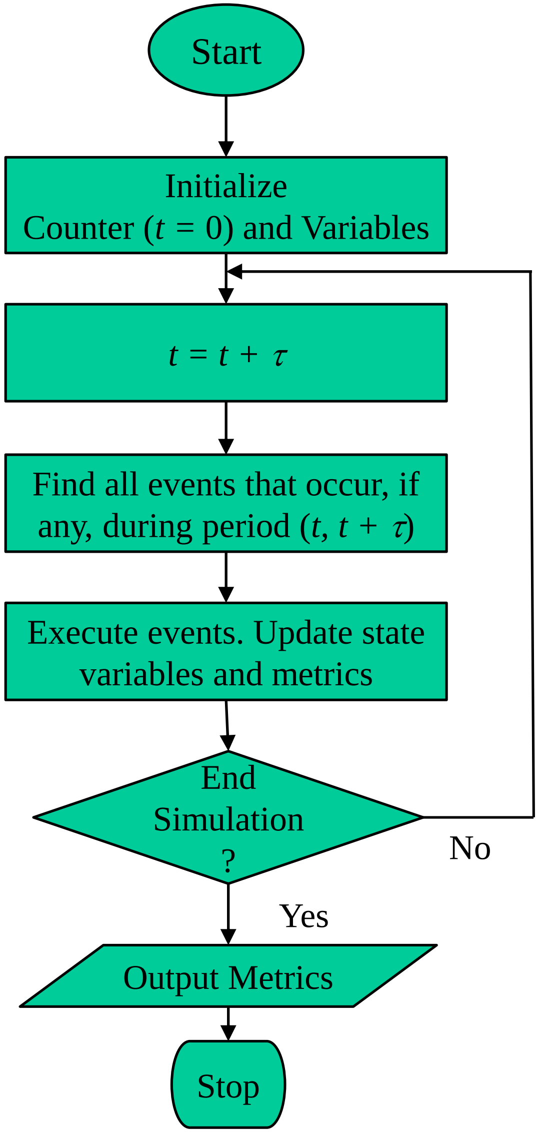
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step



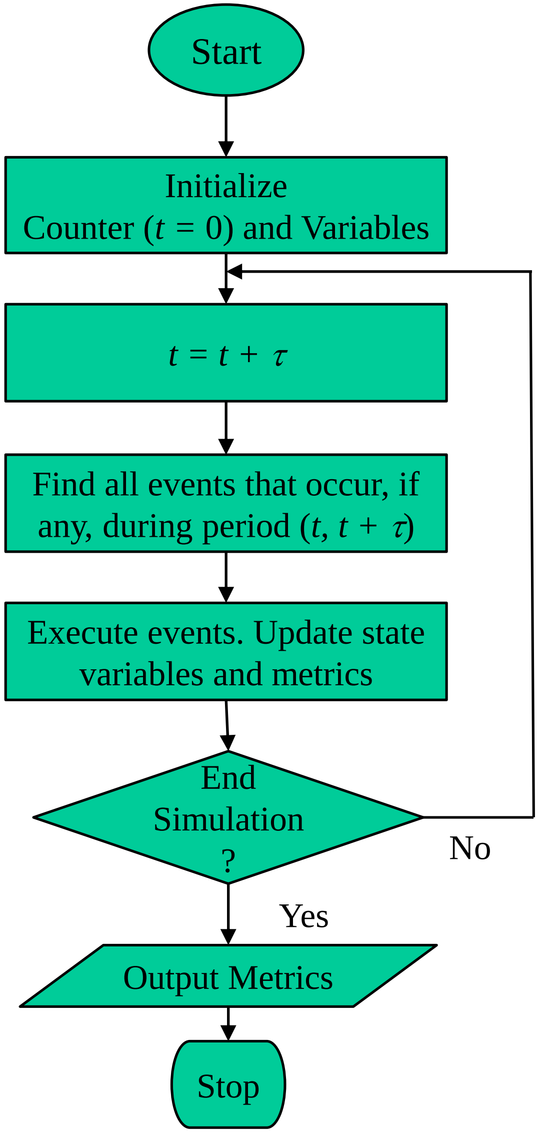
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step

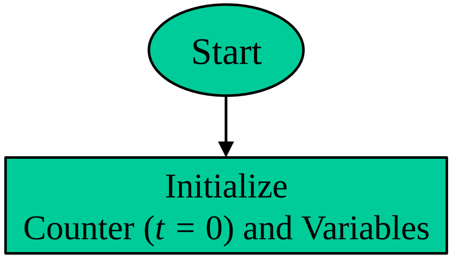


Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

Fixed-Time-Step

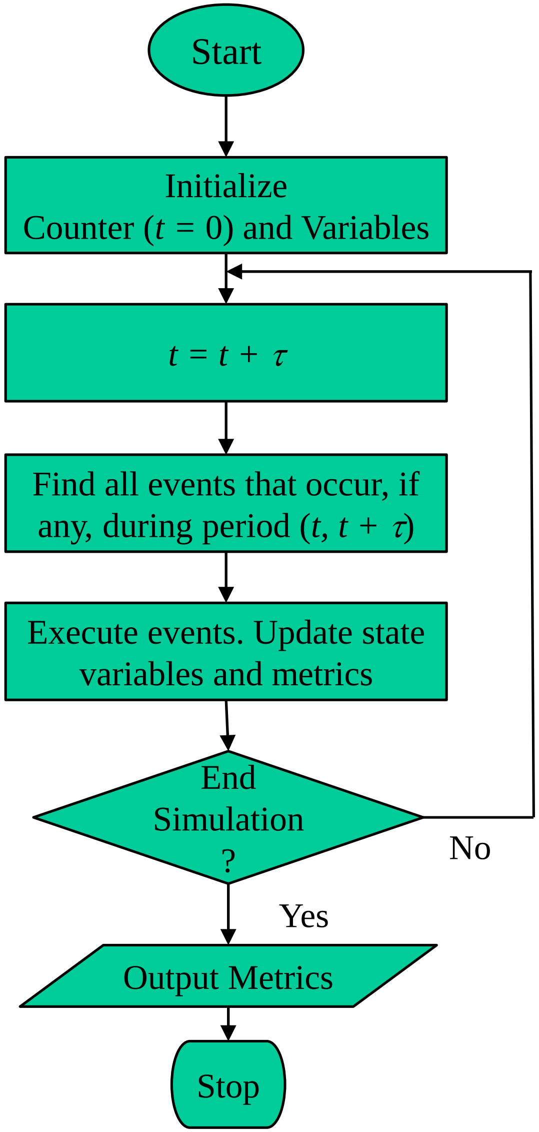


Variable-Time-Step

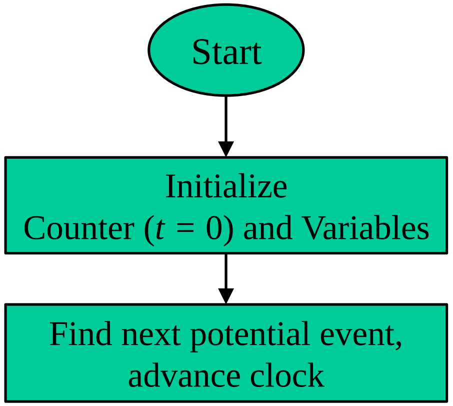


Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

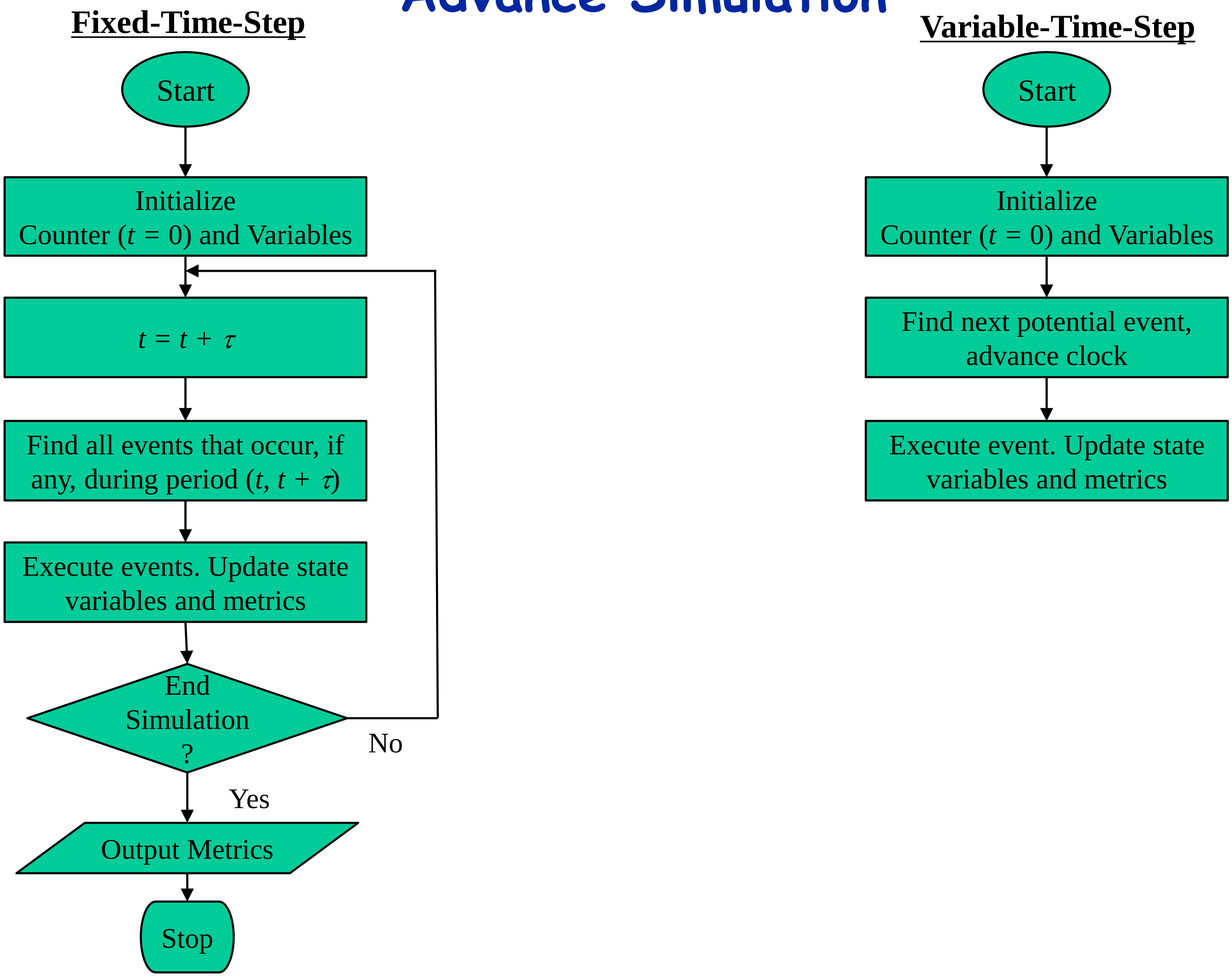
Fixed-Time-Step



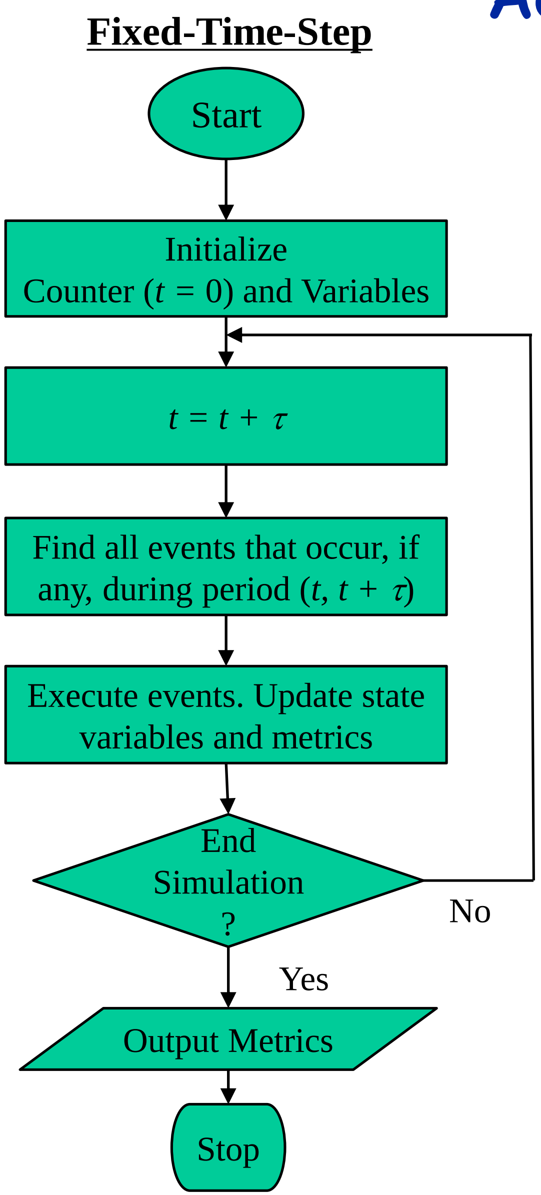
Variable-Time-Step

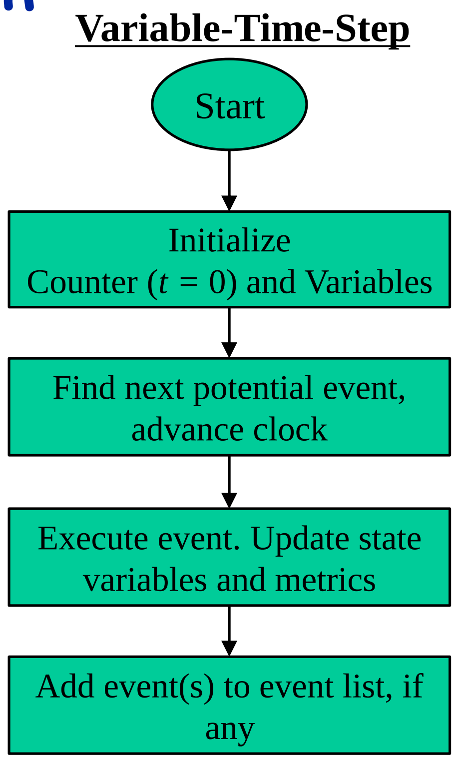


Fixed-Time-Step vs. Variable-Time-Step Advance Simulation



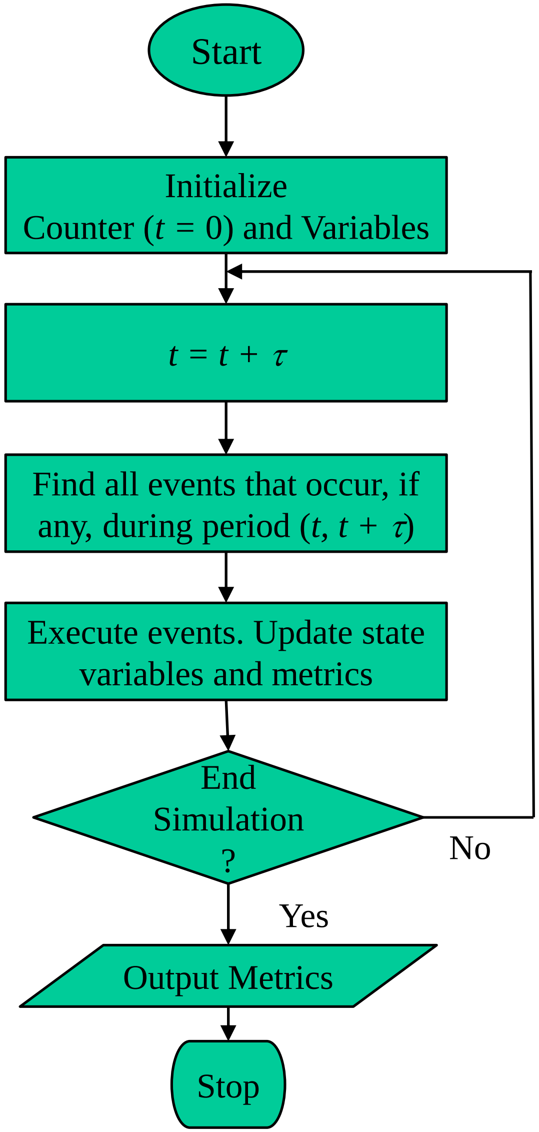
Fixed-Time-Step vs. Variable-Time-Step Advance Simulation



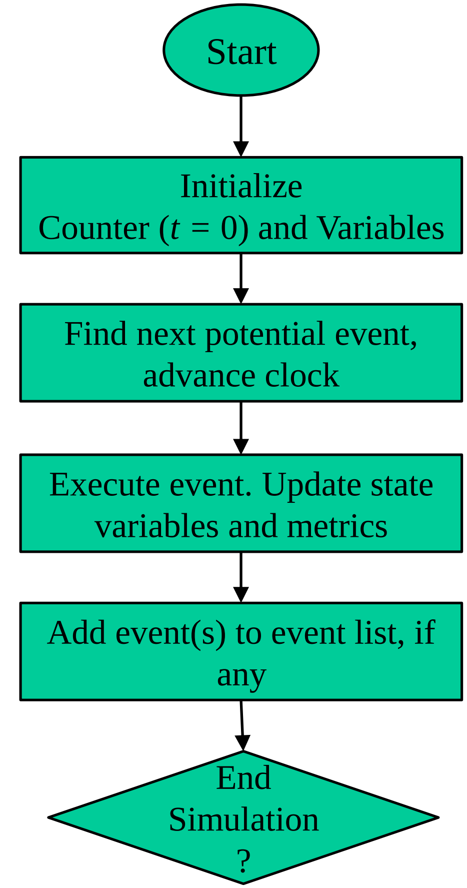


Fixed-Time-Step vs. Variable-Time-Step Advance Simulation

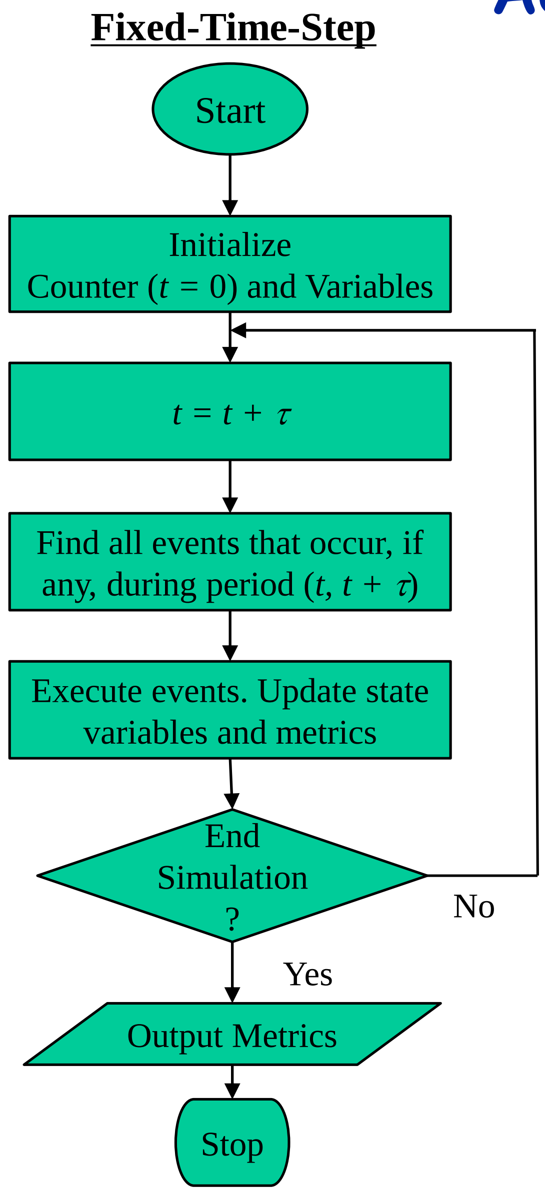
Fixed-Time-Step

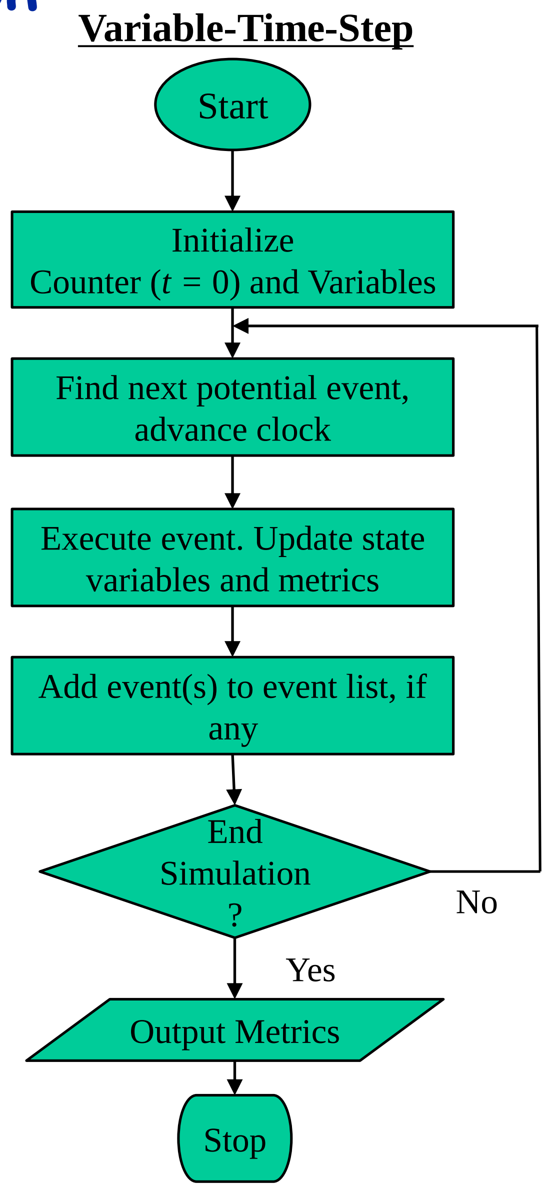


Variable-Time-Step



Fixed-Time-Step vs. Variable-Time-Step Advance Simulation





Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

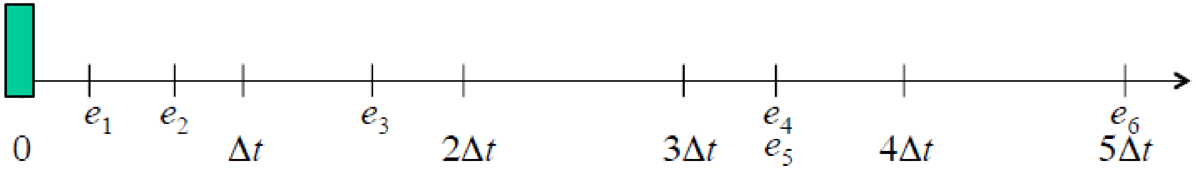
- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

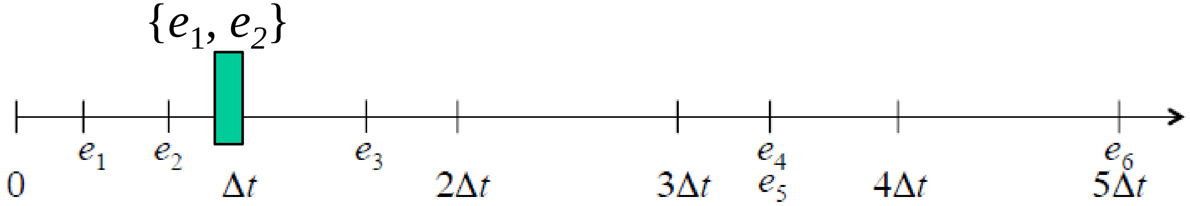


Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

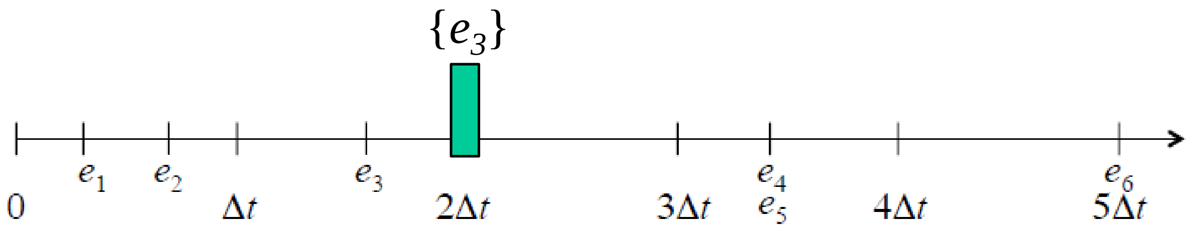


Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

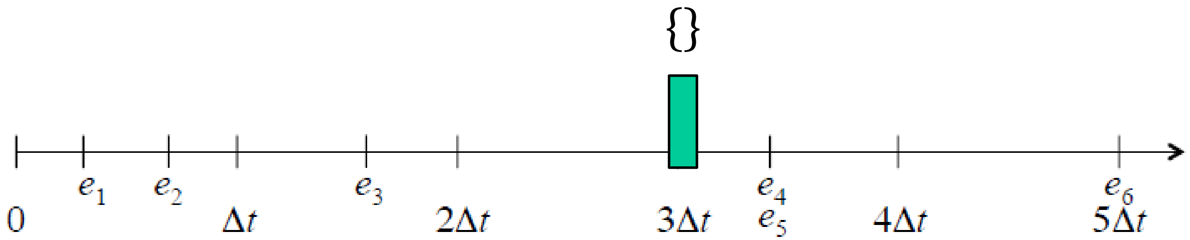


Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

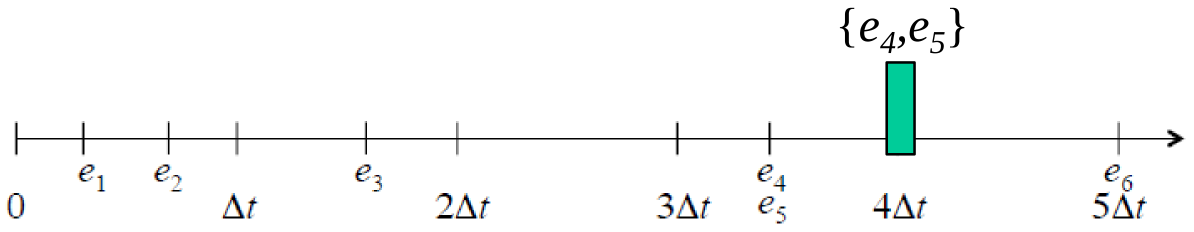


Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

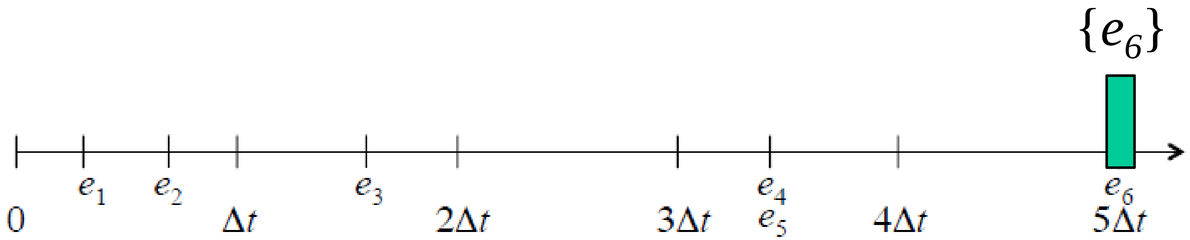


Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

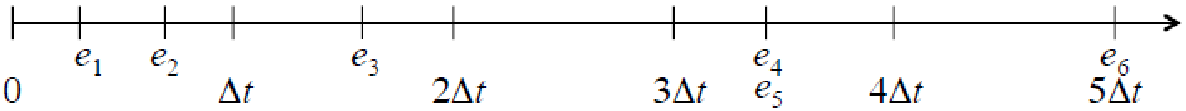
Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) ⇒ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

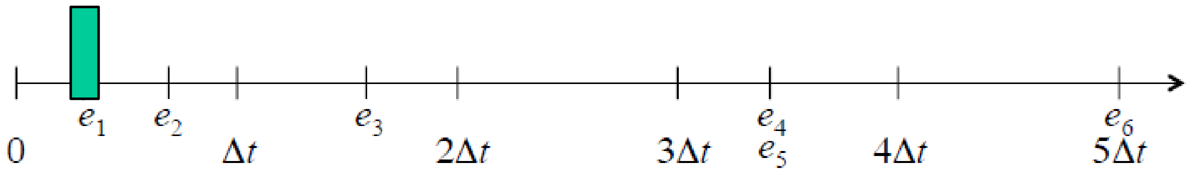
Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) ⇒ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

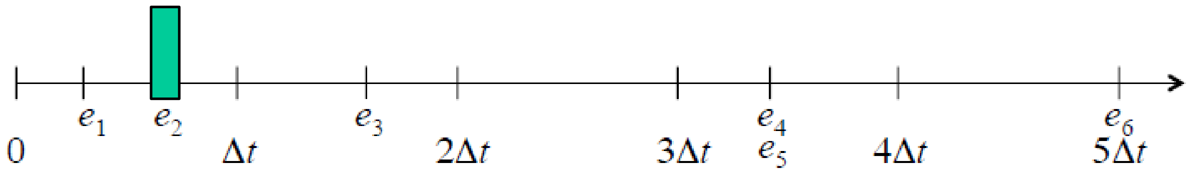
Simulation clock advanced a fixed amount of time each step

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) $\Rightarrow$ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

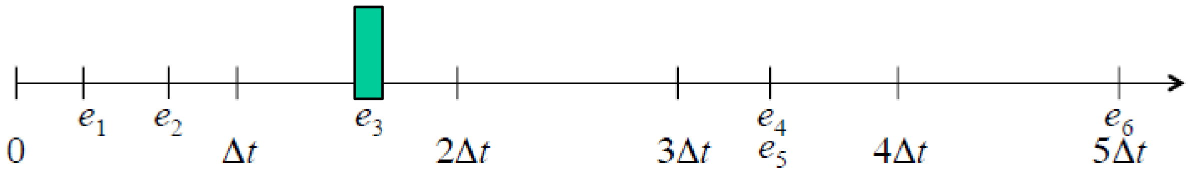
Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) ⇒ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

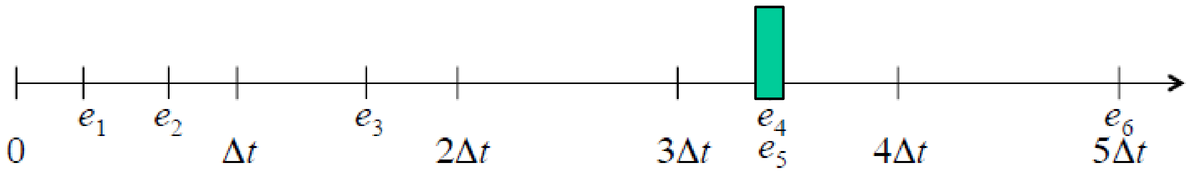
Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) ⇒ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled



Variable-Time-Step vs. Fixed-Time-Step Advance Simulation

Fixed-Time-Step

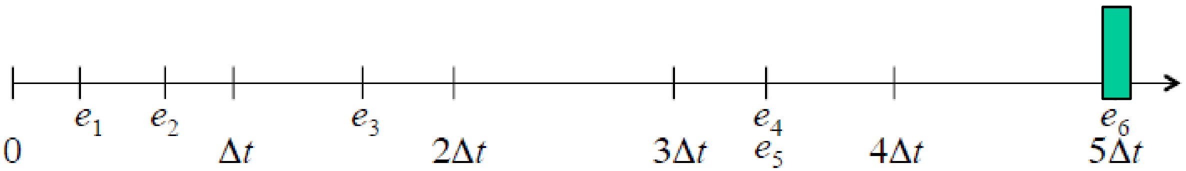
Simulation clock advanced a fixed amount of time each step

- Rules must be given to determine the ordering of events that occur in each interval of time
- Good for models where most events happen at fixed intervals
- Can be inefficient if events occur in a bursty manner, relative to time-step used
- No future event list needs to be maintained

Variable-Time-Step

Simulation clock advanced a variable amount of time each step, to time of the next event

- If all event times are exponentially distributed, the next event to complete and time of next event found using the equation for the minimum of *n* exponentials (memoryless) ⇒ no future event list is needed
- Otherwise, future event list is needed
- Skips over periods of inactivity, and bursty occurrence of events are efficiently handled

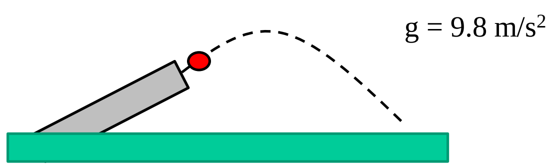


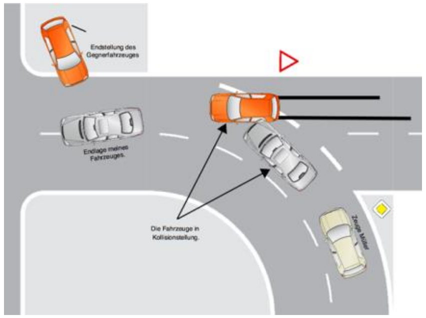
Discrete Event Simulation Basics

- Basic elements of a discrete event simulation are *variables* and *events*
- Variables include
  - Clock
  - **System State Variables**
  - Metrics
- When an event occurs, at least one of the variables change

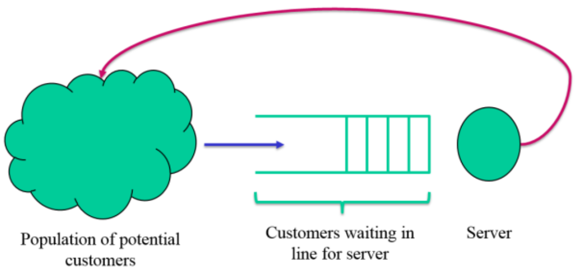
Types of Simulation

- Continuous-state simulation is applicable to systems where the notion of state is continuous and typically involves solving (numerically) systems of differential equations.





- Discrete-event simulation is applicable to systems in which the state of the system changes at discrete instants of time, with a finite number of changes occurring in any finite interval of time.



Discrete Event Simulation Basics

- Basic elements of a discrete event simulation are *variables* and *events*
- Variables include
  - Clock
  - System State Variables
  - **Metrics**
- When an event occurs, at least one of the variables change

Metrics

- The purpose of a simulation is to calculate metrics
- Two types of discrete-event simulation exist, depending on what type of metrics are to be calculated
  - *Terminating*: Measures to be estimated are measured at fixed instants of time or intervals of time of finite length.
    - This may also include random but finite (in some sense) times, such as time to failure
    - Terminating simulation can end after set amount of time or after some event occurs
  - *Steady-state*: Measures to be estimated depend on instants of time or intervals whose starting points are taken to be $t \rightarrow \infty$

Metrics

- The purpose of a simulation is to calculate metrics
- Metrics can be:
  - *Instant-of-time*: at a fixed t, or in steady-state
    - Examples: Number in queue at time t, number jobs processed at time t
  - *Interval-of-time*: for a fixed interval, or in steady-state
    - Examples: Number of jobs processed between time t and $t + \tau$
  - *Time-averaged interval-of-time*: for fixed interval, or in steady-state
    - Examples: Fraction of time servers are busy, fraction of time more than k jobs in system

Estimators of these measures include:

- Mean
- Variance
- Interval: Probability that the measure lies in some interval [x, y]

Different Types of Processes and Measures Require Different Statistical Techniques

- Transient measures (terminating simulation):
  - Multiple trajectories are generated by running basic simulation loop multiple times using different random number streams. Called *Independent Replications*.
  - Each trajectory used to generate one observation of each measure.

- Steady-State measures (steady-state simulation):
  - Initial transient must be discarded before observations are collected.
  - If the system is ergodic (irreducible, recurrent non-null, aperiodic), a single long trajectory can be used to generate multiple observations of each measure.
  - For all other systems, multiple trajectories are needed.

- **Rare event simulation** – issue can sometimes be addressed with variance reduction techniques

Next class!

- How to generate random number streams for use in the simulation?