CPE 345: Modeling and Simulation

Lecture 4

Today's topics

- Short review of simulation basics
- Basic steps for OMNET++ simulation implementation
- Statistical Models in Simulations (textbook chapter 5)

Short review on simulation basics

- Steps to implement a simulation (less formal description)
  - Find out what is the problem that you want to simulate
    - What will you study
    - Objectives – *questions to be answered by your simulation*
    - Overall approach – *alternative solutions/algorithms to be studied*
  - Come up with a system model
    - Important system components (entities)
    - State description for the system (what variables really describe the system)
    - Events (what causes state changes in the system)
    - Actions implemented when an event takes place
  - Implement the simulation (actual program, e.g. OMNET++)
  - Verify and validate
    - Verify the logic of your program
    - Validate the model (does the model make sense?)
  - Collect and analyze results

Handling events

- Managing events
  - Scheduling new events, canceling some events and maintaining a list of chronological ordered events – **Future Event List (FEL)**
  - **FEL contains entries: (type of event, time the event will occur)**
- Simulation focus is on events
  - When an event takes place, some actions are taken, system state changes, FEL needs to be updated: **Event-scheduling/time advance algorithm**
  - **Event-scheduling/time advance algorithm:**
    - **Remove imminent event from FEL, and all other events that occur simultaneously**
    - **Handle these events (e.g. change system state, schedule new events, cancel some events, etc.)**

Simulation basics for OMNET++

- Identify your network model
  - Modules ←→ entities
  - What type of events ? → messages generate events
    - Need **connections** between modules that might exchange messages
    - FEL management – handled automatically by OMNET++
    - Still need to schedule/cancel events – library functions **(Read the manual!)**
      - Send messages to schedule event;
        - » message from generator to fifo queue generates an arrival event
        - » message from fifo to sink generates departure event
- Write network description using NED language
  - The network: compound module
  - Each module in the network (e.g. generator, fifo, sink) should be declared as a submodule
  - Connections, parameters, also declared
  - Each submodule must have a declaration and an implementation in a related C++ file (maybe also an associated header file) with the same name.
- Declare each simple module (submodule), and implement it in a C++ file with the same name

Simulation basics for OMNET++ - Cont.

- The simple module code – implements the action of the module
  - The end of an action generates (schedules) another event. E.g.
    - Generator: action wait for inter-arrival time. At end of wait, send message (generates arrival event)
    - FIFO:
      - when arrival received – action according to flowchart (put in queue or serve)
        - » At the end of service (departure event), message sent to sink
- Coding a simple module
  - Process – interaction approach
    - Implement two functions: activity() and finish()
  - Event handling approach
    - Implement 2-3 functions: initialize(), handleMessage() and finish() (if statistics are collected)
    - Use finite state machine formulation (FSM)

FSM implementation for OMNET++

- Module is implemented as FSM
  - It is characterized by a collection of states (steady states and transient states)
  - Each event may change the state of the module
    - Between two events, the module is always in steady state
    - When an event occurs: steady state -> transient state -> steady state
  - In OMNET++ , you can write code for both entering and leaving the state
  - Entry code should not modify the state. State changes (transitions) must be put into the exit code.
  - Example: Generator module in fifo2 example is implemented as a FSM

```
cFSM fsm;
enum {
  INIT = 0,
  SLEEP = FSM_Steady(1),
  ACTIVE = FSM_Steady(2),
  SEND = FSM_Transient(1),
};
```

FSM implementation

```
FSM_Switch(fsm)
    {
      case FSM_Exit(INIT):
        // …
      break;
      case FSM_Enter(SLEEP):
        // …
      break;
     case FSM_Exit(SLEEP):
       // …
     break;
     case FSM_Enter(ACTIVE):
      // …
     break;
      // …
    };
```

State transitions: FSM_Goto(fsm, newState);

```
case FSM_Exit(SEND):
        // …
     FSM_Goto(fsm,ACTIVE);
break;
```

Model description: statistical models in simulation

- Real world is not deterministic
  - Actions for a system under study cannot be completely predicted in advance
  - In the real world, there are many causes for randomness
    - Example 1: time required to fix a broken machine – not known in advance
      - Depends on factors such as complexity of the breakdown, availability of replacement parts, etc.
    - Example 2: Driving time from Hoboken to Philadelphia
      - Depends on the traffic conditions (although you may infer that at peak hours, the time is usually much longer), the exact travel time is a random variable
  - While if the processes would be deterministic, you could come up with exact answers for the time to fix a machine (example 1), and the travel time (example 2), for the random case, there are still some ways to characterize these variables (called random variables)
    - Statistical measures: mean (statistical average), variance, probability density function

Terminology and Concepts

- Discrete random variables
  - Notation X
  - Number of possible values for X is finite or countable infinite

Example 1. X = number of jobs arriving at a shop in a given week

- possible values of X = range space of X

$R_X = \{1,2,3, \ldots\}$

- the probability that X takes the value x_i = $p(x_i) = P(X = x_i)$
  - cannot take negative values:

$$p(x_i) \geq 0, \text{ for all } i$$

- $p(x_i)$ measures the frequency with which event x_i *occurs*

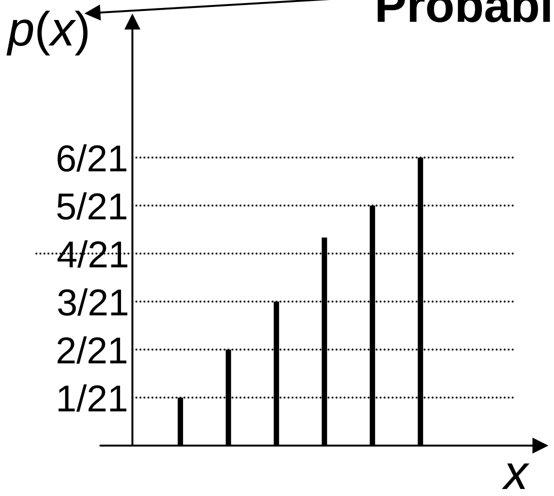
$$\sum_{i=1}^{\infty} p(x_i) = 1$$

Discrete random variables

- Example 2 in textbook. Tossing a die experiment
  – Assume the die is loaded, with the probability of one face showing up, proportional to the number of spots on the die

x_i	1	2	3	4	5	6
$p(x_i)$	1/21	2/21	3/21	4/21	5/21	6/21

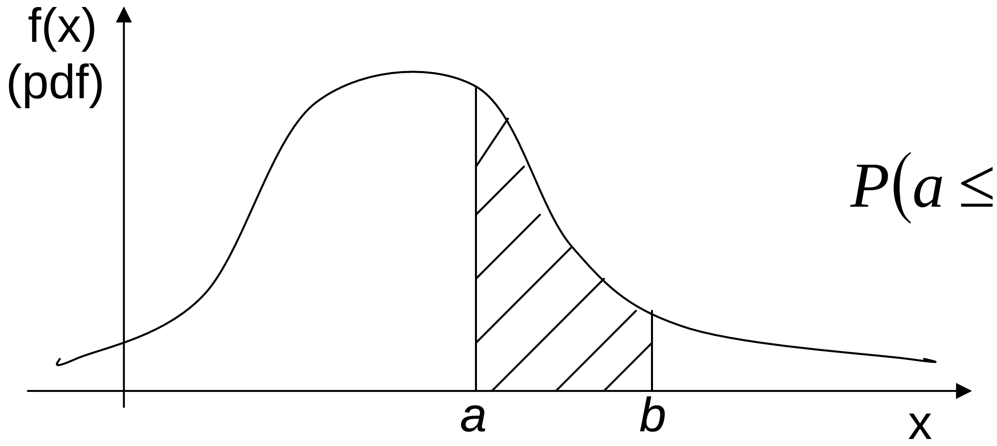
Probability mass function (*pmf*)



What would be the *pmf* for a regular die ?
- every face shows with equal probability

Continuous random variables

- If the random variable can take values in a continuous interval (or a collection of intervals) – X = continuous random variable
- Characterized by the probability density function (pdf)



$$P(a \le X \le b) = \int_a^b f(x)dx$$

Properties:

(a) $f(x) \ge 0 \quad \forall x \in R_X$

(b) $\int_{R_X} f(x) = 1 \quad \forall x \in R_X$

(c) $f(x) = 0, \; if \;\; x \notin R_X$

Example for continuous random variable

- Driving time from Hoboken to Philadelphia
  - Is this characterized by a known *pdf* ?
  - In real life, you should sample the phenomenon, try to fit it to a known distribution *f(x)* (goodness of fit tests – study later on in this course)
  - What would be some obvious measures that you would use to characterize the driving time
    - **(a) On *average* will be about 2 hours → statistical mean**
    - **(b) 90% of the time, it will take between 1h 45 min and 2 h 10 min.**
    - **(c) What is the spread (variance) from the mean driving time?**

(b) – already discussed:

$$P(105\,\min \le X \le 130\,\min) = \int_{105}^{130} f(x)dx = 0.9$$

Mean and Variance

- Mean = expected value (expectation) *E(X)*= μ = *1st moment of X*
  - Discrete case:

$$E(X) = \sum_{i \in R_X} x_i p(x_i)$$

  - Continuous case:

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx$$

  - $E(X^n)$= n^{th} *moment of X*

$$E(X^n) = \sum_{i \in R_X} x_i^n p(x_i) \quad \text{discrete}$$

$$E(X^n) = \int_{-\infty}^{\infty} x^n f(x) dx \quad \text{continuous}$$

Mean and variance - cont

- Variance – measure of the spread (variation) of possible values of X around the mean

$$\sigma^2 = \text{var}(X) = V(X) = E[(X - E(X))^2] = E(X^2) - [E(X)]^2$$

- Standard deviation

$$\sigma = \sqrt{\text{var}(X)}$$

- Mode – peak of the pdf or pmf

Cumulative Distribution Function (CDF)

- Measures the probability that X has a value less or equal to x
  - Discrete r.v.

$$F(x) = \sum_{i, x_i \le x} p(x_i)$$

  - Continuos r.v.

$$F(x) = \int_{-\infty}^{x} f(t)\,dt$$

- Properties of CDF function:

$$\lim_{x \to -\infty} F(x) = 0$$

$$\lim_{x \to \infty} F(x) = 1$$

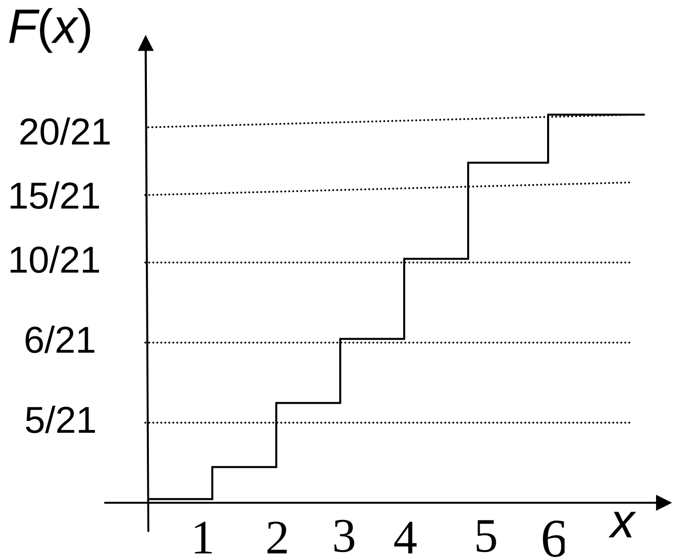
$$a < b \Rightarrow F(a) \le F(b)$$

CDF example

- Loaded die

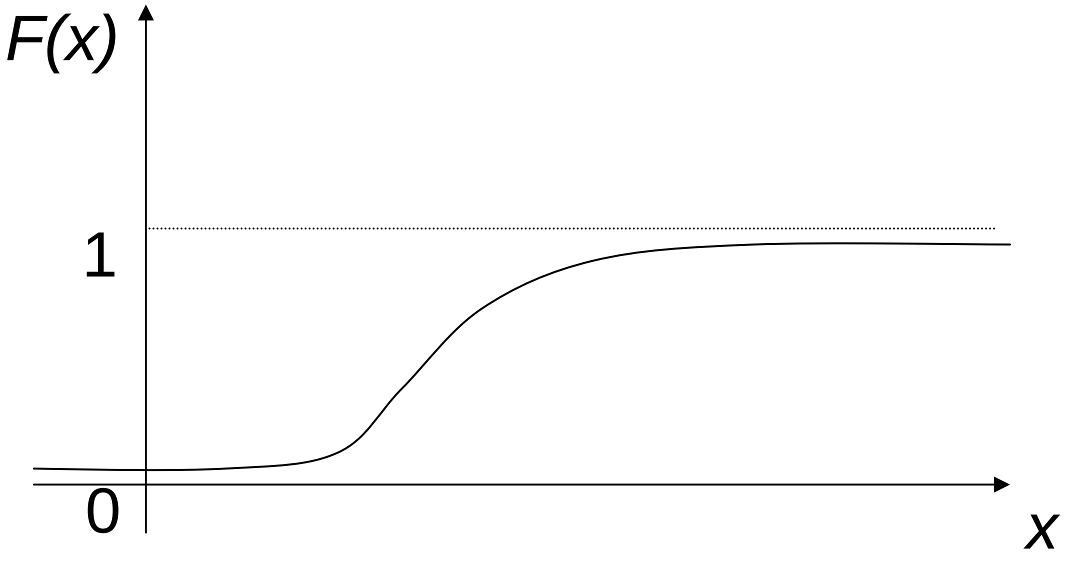
x_i	1	2	3	4	5	6
$p(x_i)$	1/21	2/21	3/21	4/21	5/21	6/21

x	$(-\infty,1)$	[1,2)	[2,3)	[3,4)	[4,5)	[5,6)	$[6, \infty)$
F(x)	0	1/21	3/21	6/21	10/21	15/21	21/21



Continuous CDF example

- Based on the three properties, a generic CDF for a continuous r.v. should look like in the figure



Discrete Distributions

- Bernoulli trials
  - Consider an experiment, consisting of n trials, which can be a success (1) or a failure (0)
    - E.g. coin flipping, receiving a bit, etc.
  - The n Bernoulli trials are called a Bernoulli process, if
    - The trials are independent

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\ldots p(x_n)$$

    - Probability of success remains constant from trial to trial
  - For one trial, the Bernoulli distribution is

$$p(x) = \begin{cases} p & x = 1 \\ 1 - p = q & x = 0 \\ 0 & ow \end{cases}$$

$$E(X) = 0 \cdot q + 1 \cdot p = p$$

$$\mathrm{var}(X) = E(X^2) - E(X)^2 =$$

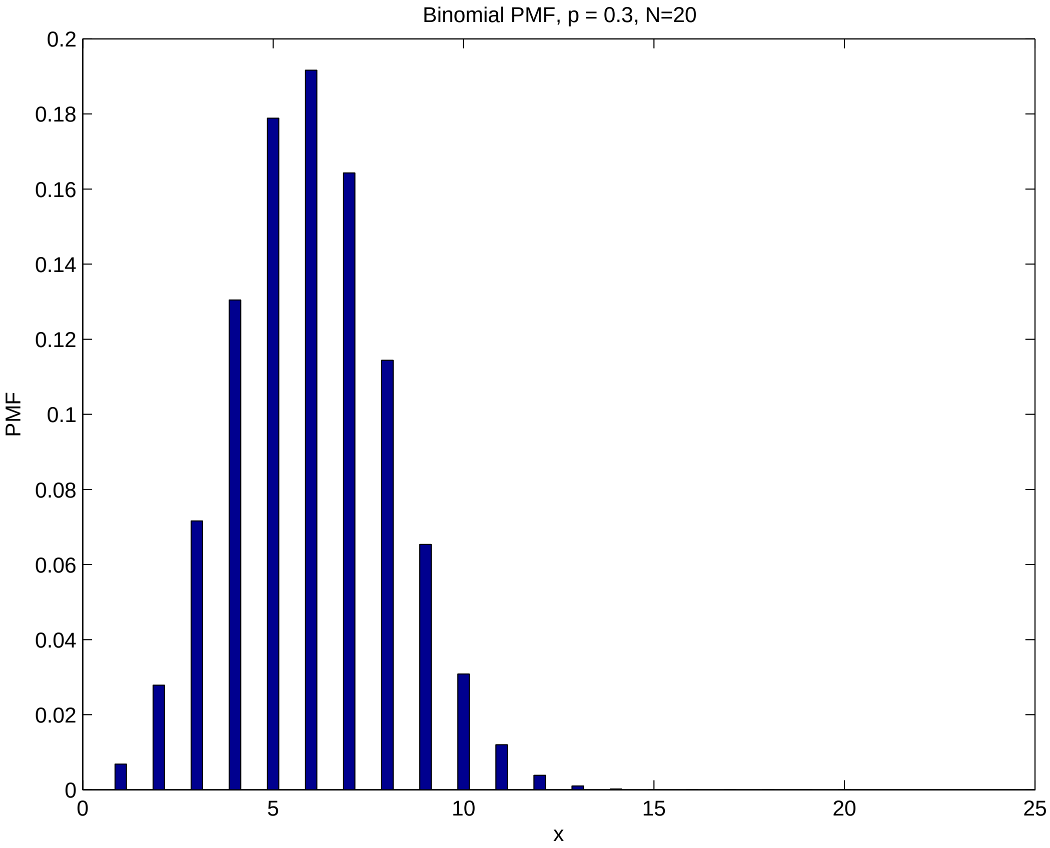
$$= \left[0^2 \cdot q + 1^2 \cdot p\right] - p^2 = p(1 - p)$$

Discrete distributions - cont

- Binomial distribution
  - The number of successes in a Bernoulli process has a binomial distribution

$$p(x) = \begin{cases} \binom{n}{x} p^x q^{n-x} & x = 1,2,..,n \\ 0 & ow \end{cases}$$

$$E(X) = np$$
$$\mathrm{var}(X) = npq$$



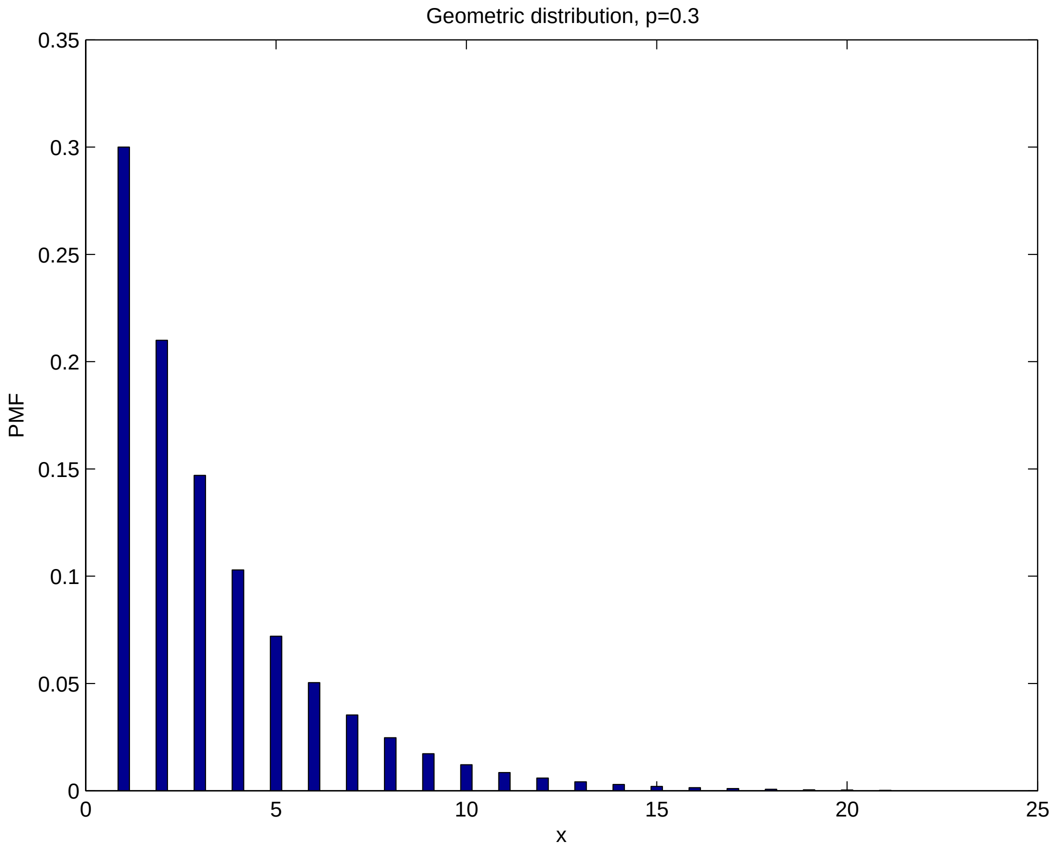
Discrete distributions - cont

- Geometric distribution
  - The number of Bernoulli trials before the first success

$$p(x) = \begin{cases} q^{x-1}p & x = 1,2,\ldots \\ 0 & ow \end{cases}$$

$$E(X) = \frac{1}{p}$$

$$\mathrm{var}(X) = \frac{q}{p^2}$$

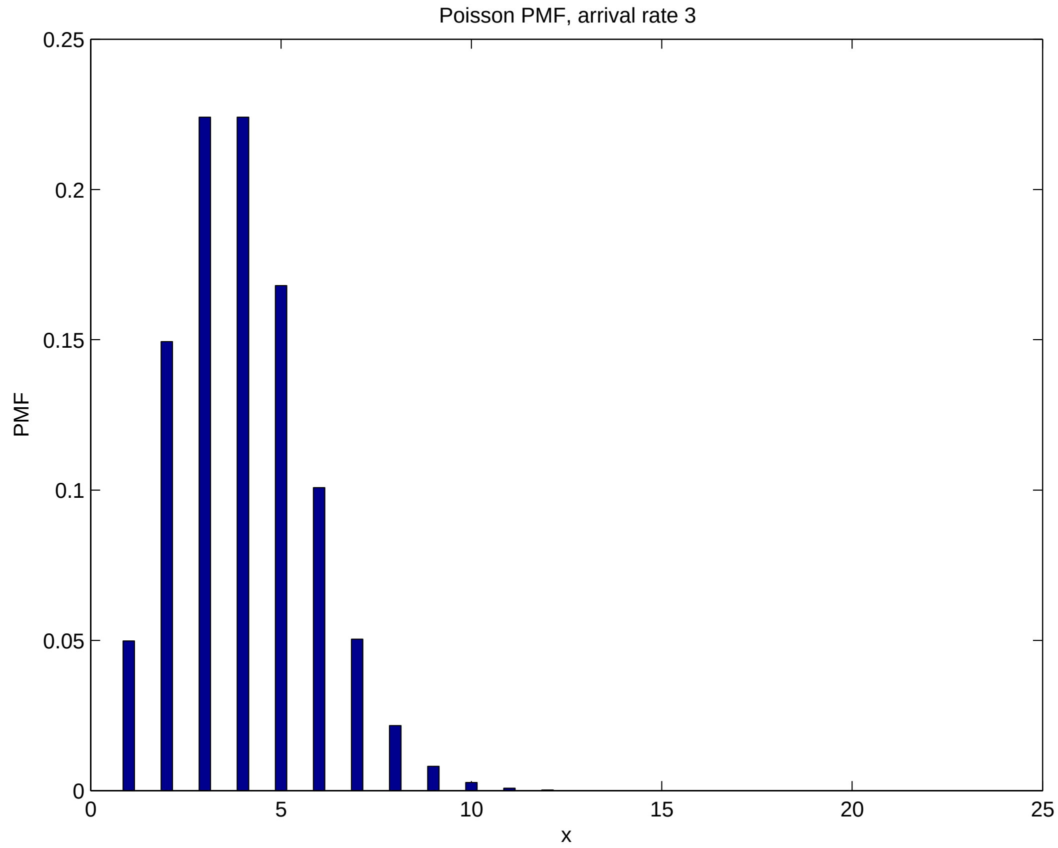


Discrete distributions - cont

- Poisson distribution
  - Very often used – good model for arrival processes

$$p(x) = \begin{cases} \dfrac{e^{-\lambda}\lambda^{x}}{x!} & x = 0,1,2,\ldots \\ 0 & ow \end{cases}$$

$$E(X) = \text{var}(X) = \lambda$$



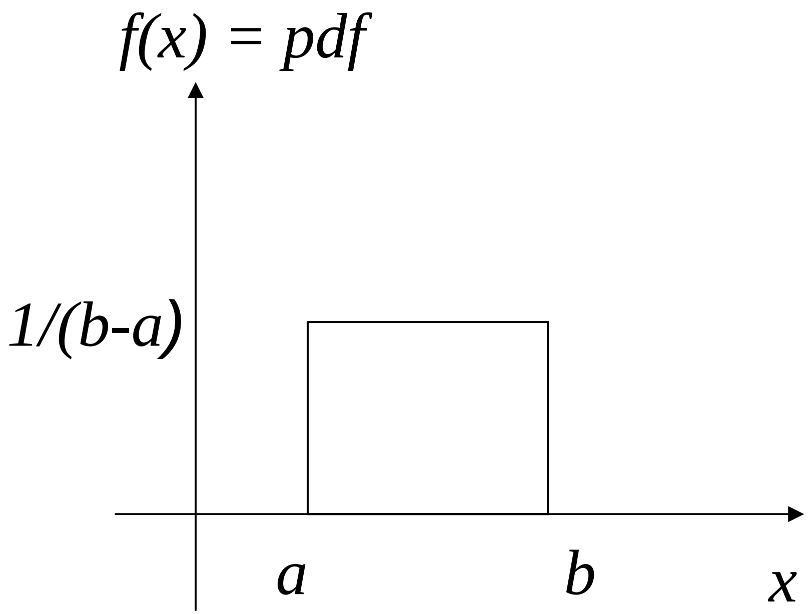
Continuous Distributions

- Uniform distribution
  - Very easy to generate (recall rand() function), is used for generating other types of r.v.s

$$f(x) = \begin{cases} \frac{1}{b-a} & a \le x \le b \\ 0 & ow \end{cases}$$

$$E(X) = \frac{a+b}{2}$$

$$\text{var}(X) = \frac{(b-a)^2}{12}$$



Continuous Distributions – Cont.

- Exponential distribution
  - Used to model inter-arrival times and service times for queues
  - Has long tail – useful for modeling component lifetime, e.g. life of a light bulb

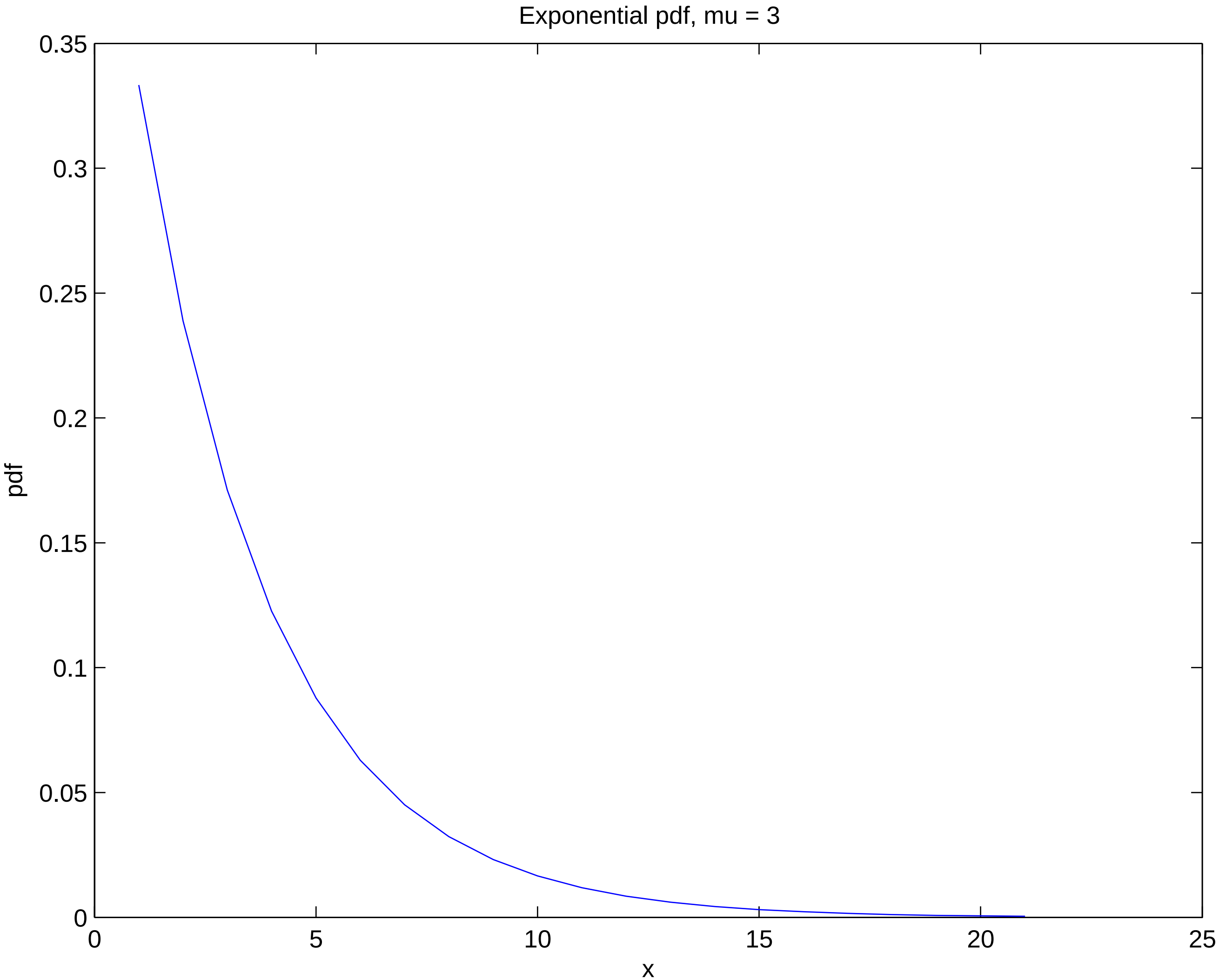
$$f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & ow \end{cases}$$

$$E(X) = \frac{1}{\lambda}$$

$$\operatorname{var}(X) = \frac{1}{\lambda^2}$$

λ is a rate: e.g. arrival rate, service rate, failure rate, etc…

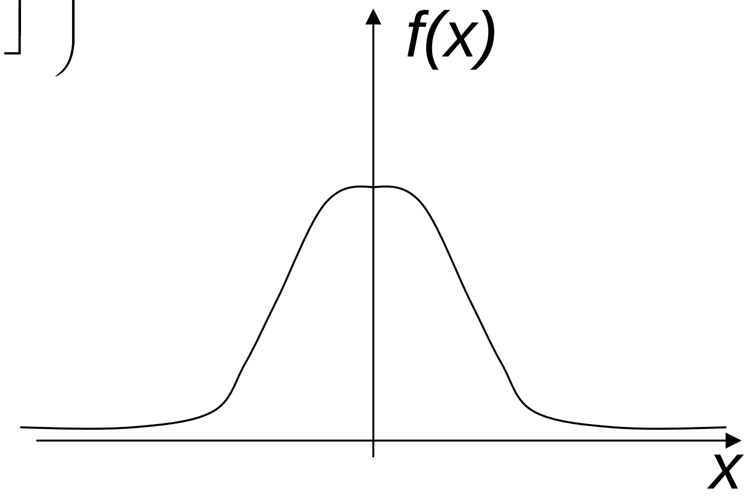
Exponential distribution



Continuous Distributions – Cont.

- Normal distribution (Gaussian distribution)
  - Widely used: e.g. model of thermal noise in circuits, communications
  - Mean μ, variance σ^2

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{1}{2}\left[\frac{x-\mu}{\sigma}\right]^2\right)}$$



  - Mode and mean are equal

$$F(x) = P(X \leq x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$$

$$\Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \quad \text{- tabulated}$$

First phase project – due next week

- Choose your simulation topic
- Formulate a simulation plan and present a proposal (1-2 pages)
  - Project title
  - What will you study
  - Objectives – *questions to be answered by your simulation*
  - Overall approach – *alternative solutions/algorithms to be studied*
  - Data gathering (observe physical system, or use references for modeling)
  - Project group members (3-5 members)

Notes: the topic must deal with dynamic, stochastic, discrete event problem
please select simple example that you can model with a few entities
(few modules in OMNET++)

Steps in a simulation study