CS 925
Lecture 3
Queues in Networks

Tuesday, January 30, 2024

Network Performance

Performance Modeling and Estimation:

- Why: to build a better performing and cheaper systems
- How:
  - build and observe
  - make a projection
  - simulation
  - analytical model
- Accuracy vs feasibility

Probability Recap

- Probability
  - definitions & conditional probability
- Random variable
  - discrete & continuous
- Characteristics of random variables
  - cumulative distribution function (CDF) & probability density function (PDF)
  - mean / expected value
  - variance / standard deviation

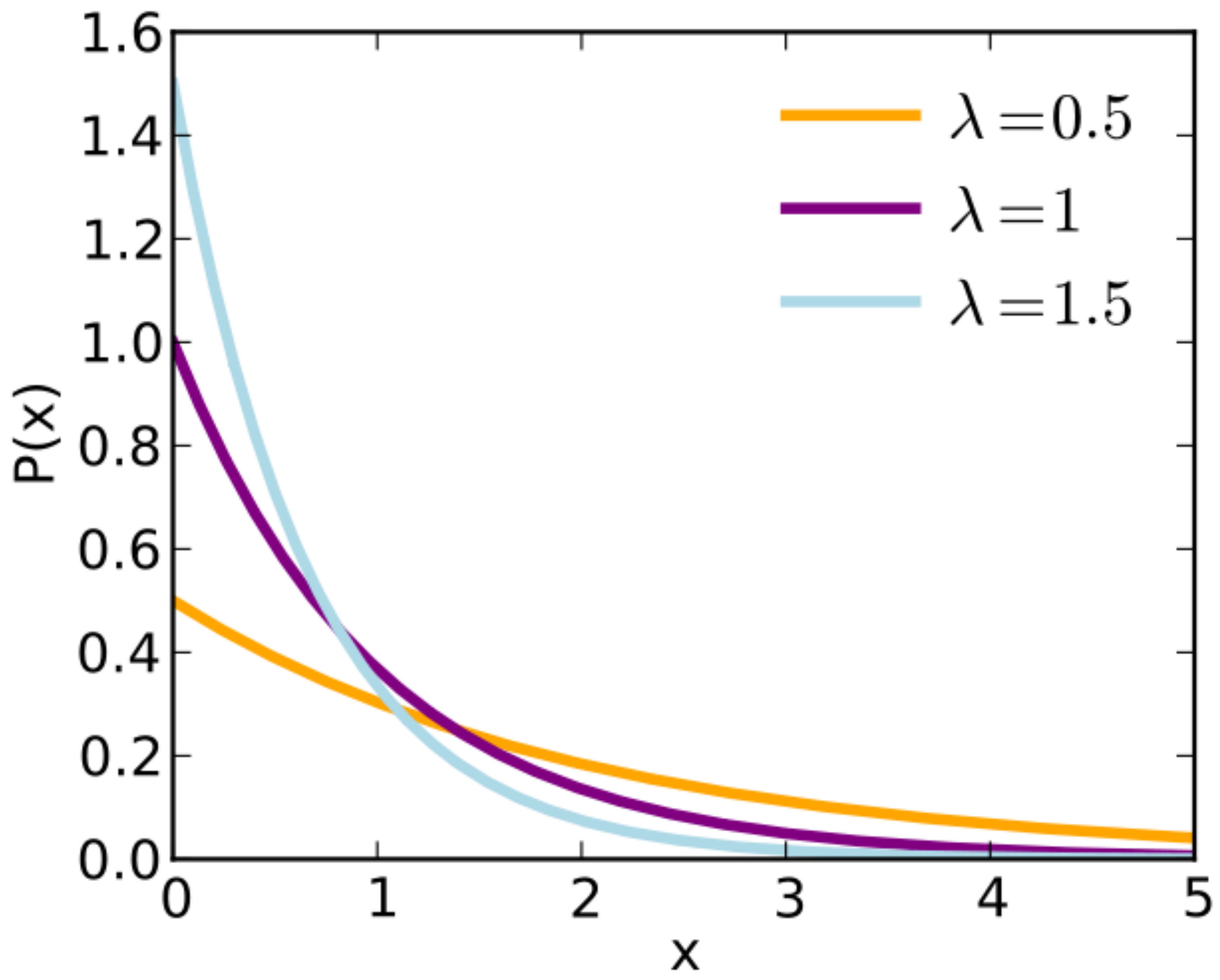
Standard Probability Distributions

- Exponential distribution (continuous)
  - inter-arrival times
- Poisson distribution (discrete)
  - counts within an interval
- Normal (Gaussian) distribution (continuous)
  - latency (?)
- Binomial distribution
  - number of busy ports

Exponential distribution

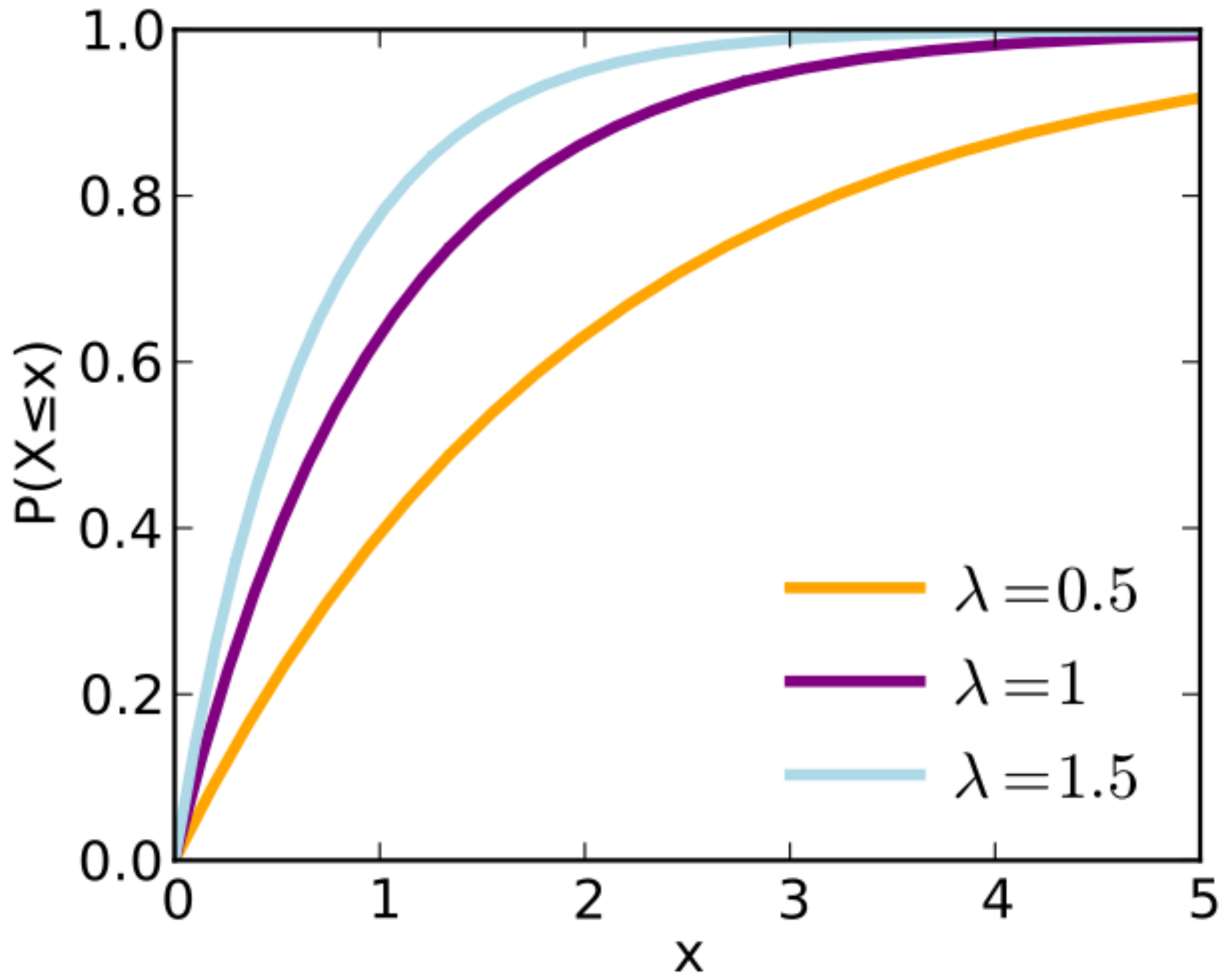
PDF:

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



CDF:

$$F(x;\lambda) = \begin{cases} 1 - e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$



Example:
describes packet inter-arrival times with rate λ

Source: https://en.wikipedia.org/wiki/Exponential_distribution

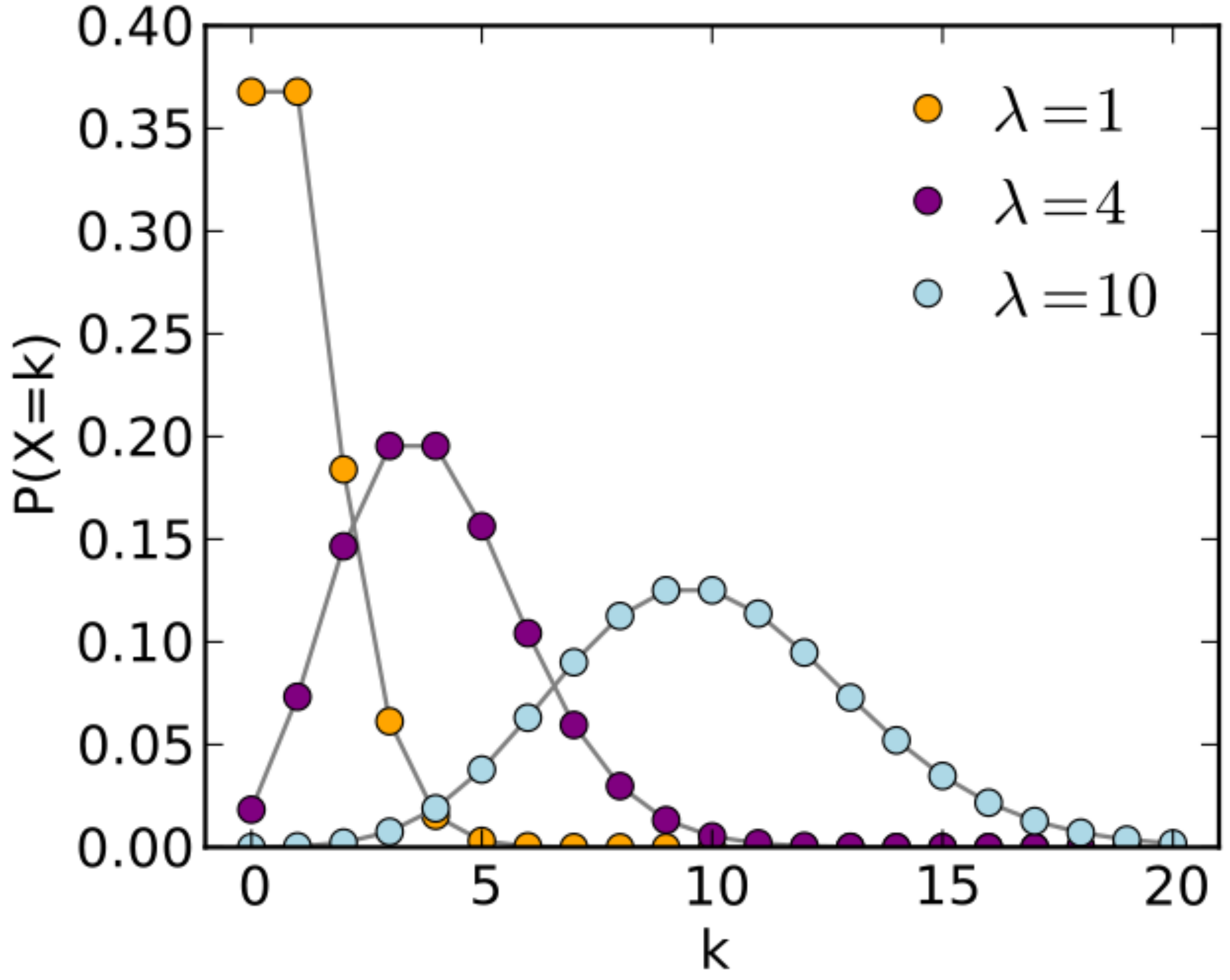
Poisson distribution

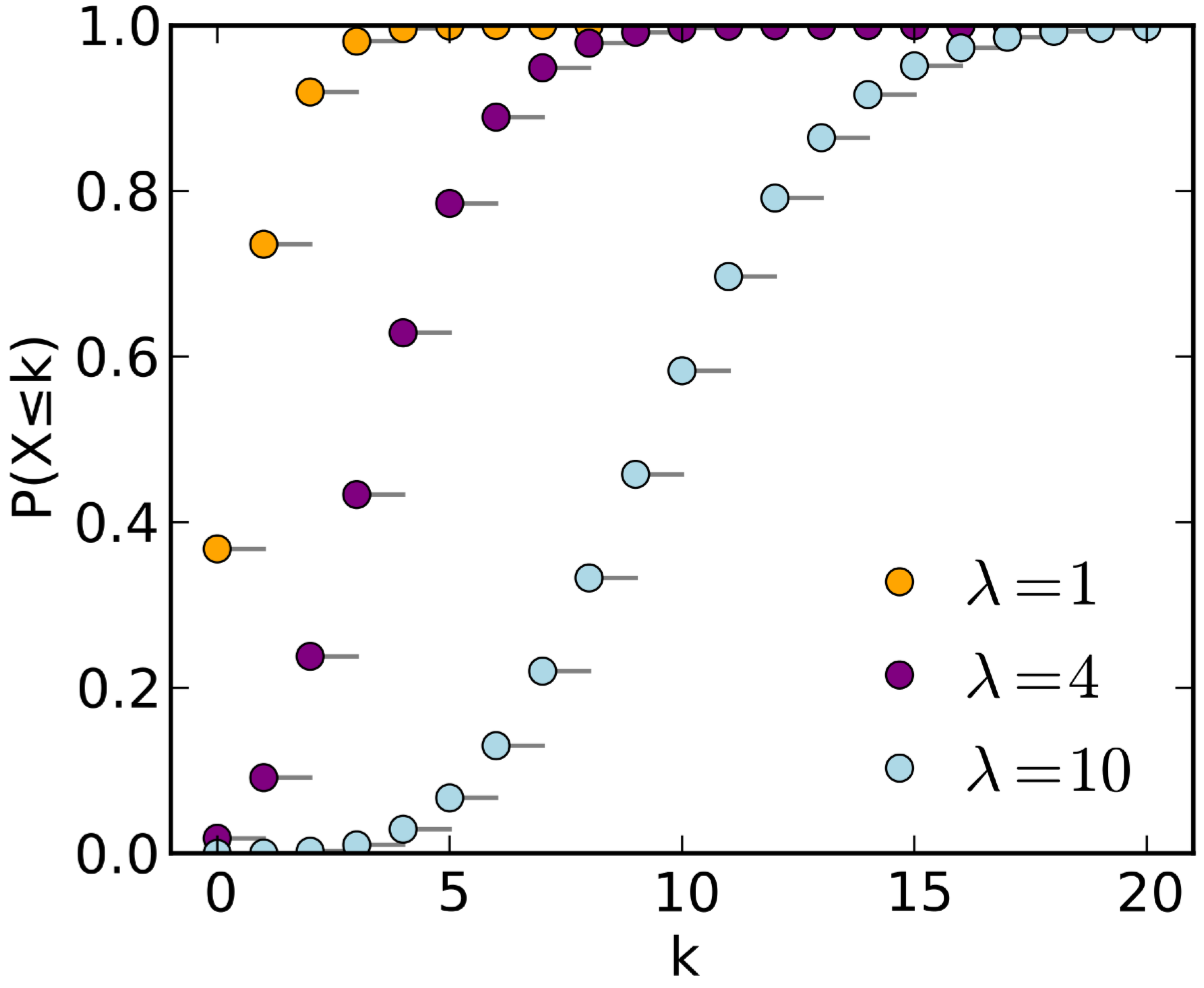
PMF:

$$P(k \text{ events in interval}) = e^{-\lambda}\frac{\lambda^k}{k!}$$

CDF:

$$F(k;\lambda) = \Pr(X \le k) = e^{-\lambda}\sum_{i=0}^{\lfloor k \rfloor}\frac{\lambda^i}{i!}$$





Example:
describes the number of packet arrivals within a time interval in a network
with exponentially distributed inter-arrival times

Source: https://en.wikipedia.org/wiki/Poisson_distribution

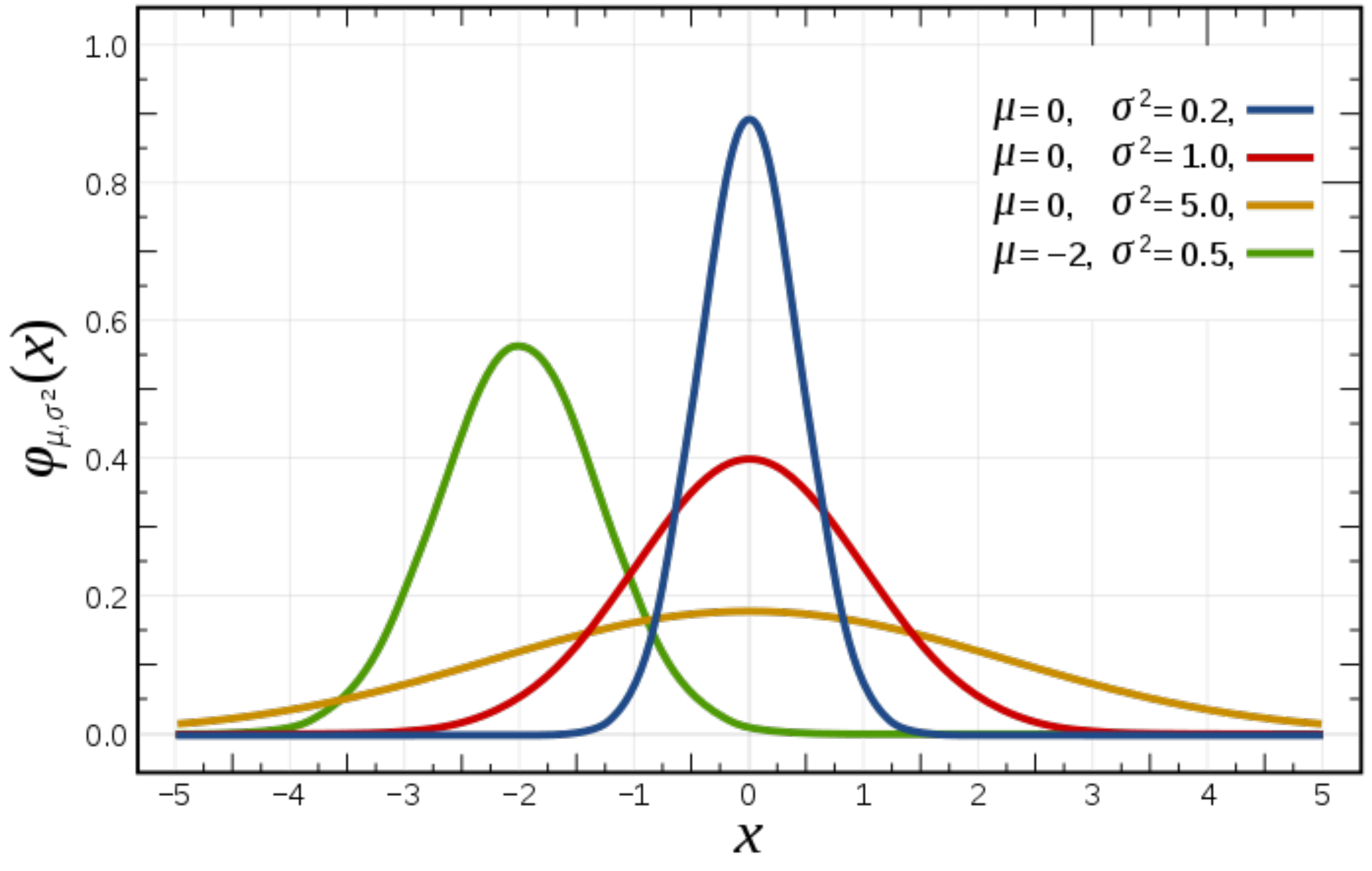
Normal distribution

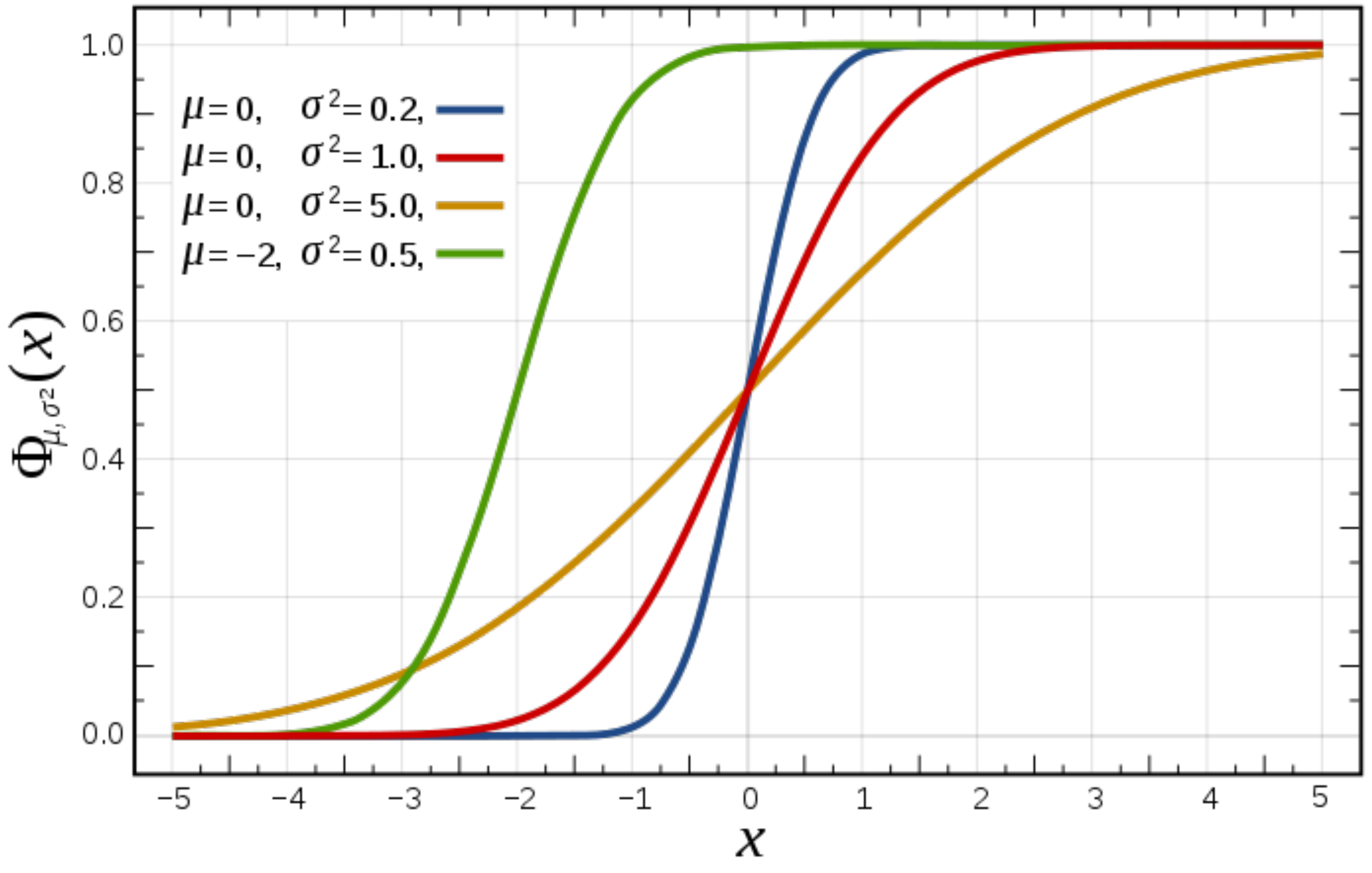
PDF:

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

CDF:

$$F(x) = \Phi\left(\frac{x-\mu}{\sigma}\right) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right]$$





... averages of samples of observations of independent random variables of independent distributions converge to normal distribution (Central Limit Theorem)

Source https://en.wikipedia.org/wiki/Normal_distribution

Binomial distribution

PMF:

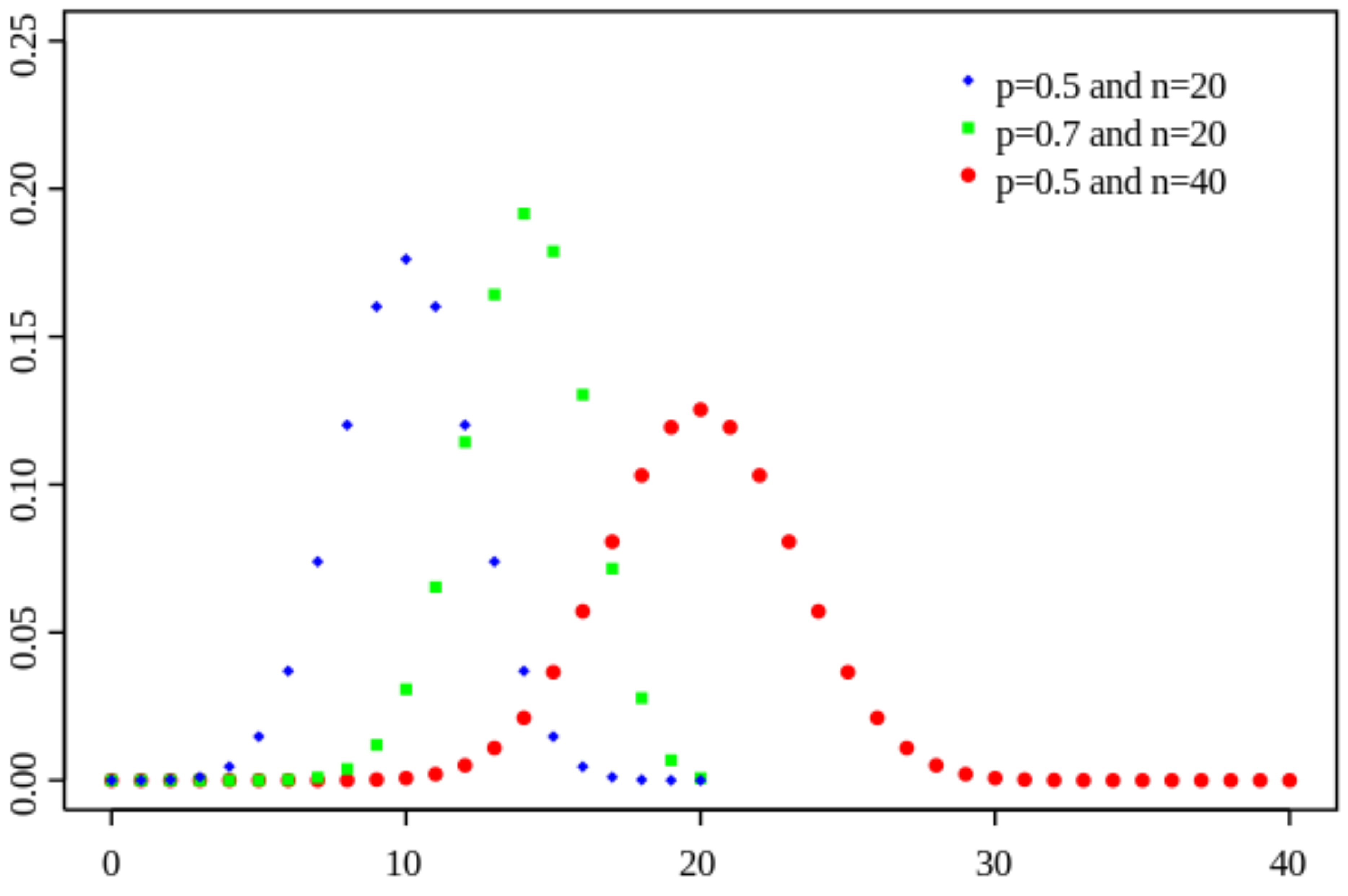
$$f(k, n, p) = \Pr(k; n, p) =$$

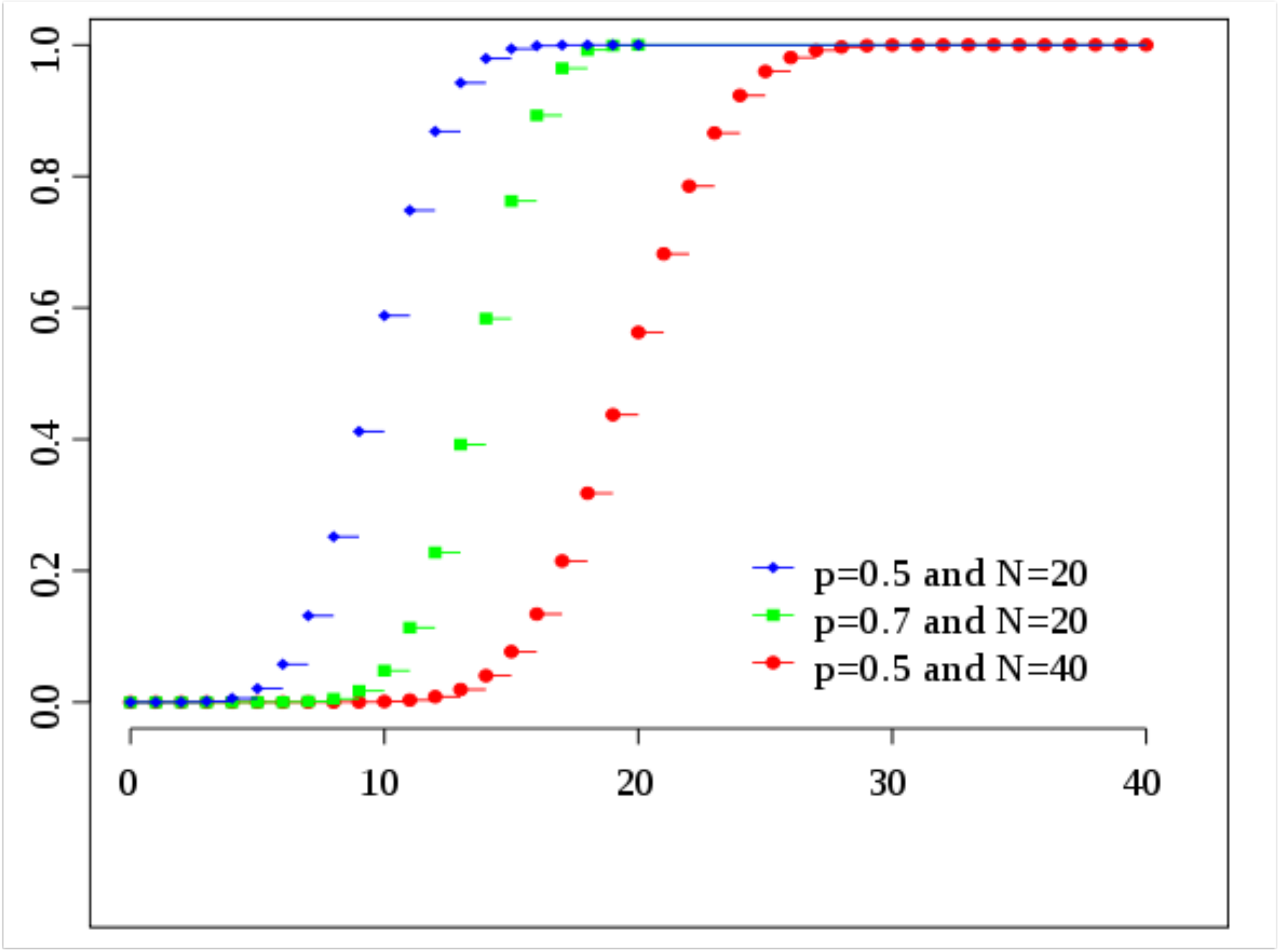
$$\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

CDF:

$$F(k; n, p) = \Pr(X \le k) =$$

$$\sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}$$



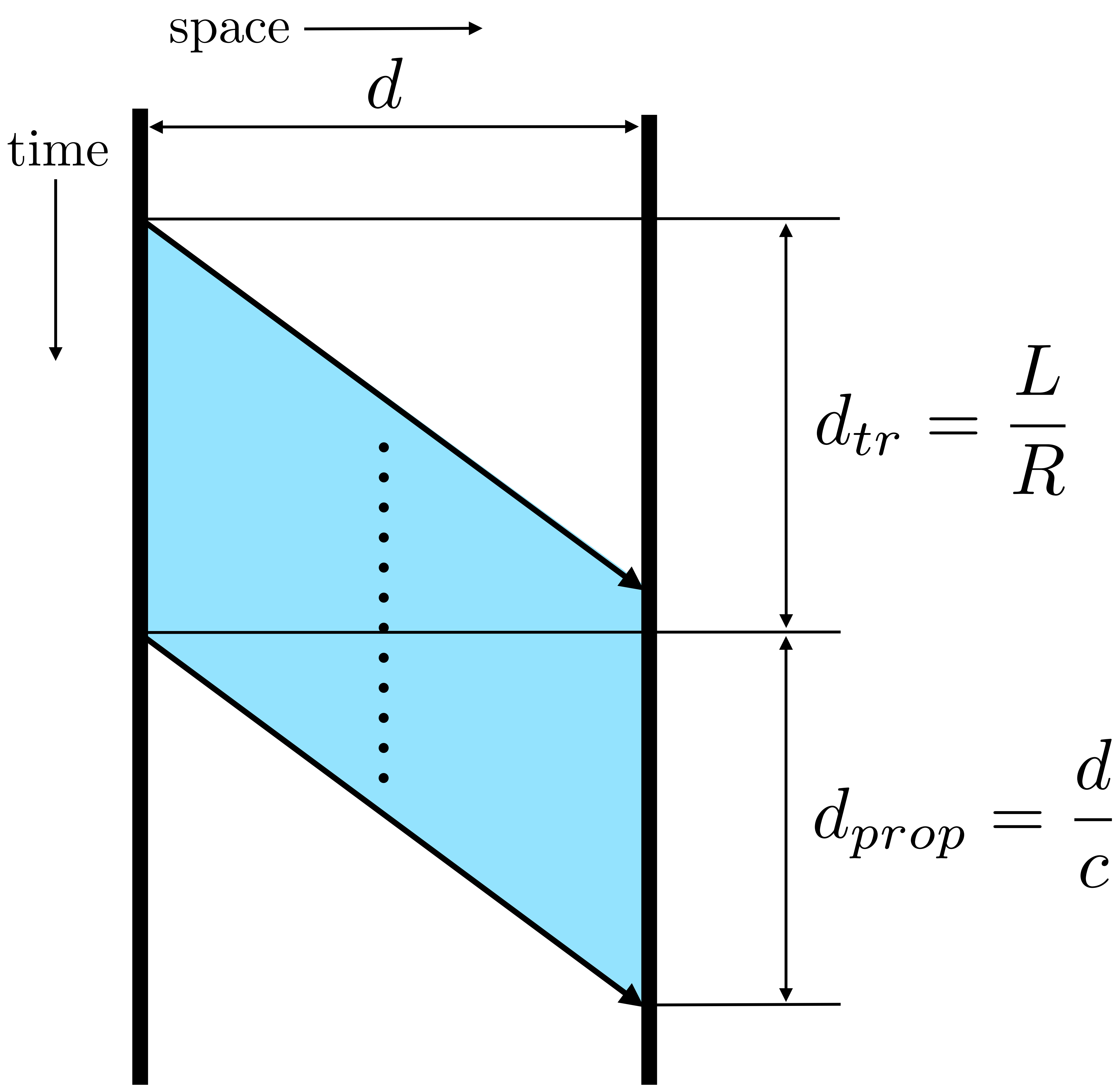


Example:

n ports, *p* probability of a packet present at a port, *k* number of ports with a packet

Source: https://en.wikipedia.org/wiki/Binomial_distribution

Time-Space Diagram



d_{tr} - time to transmit

d_{prop} - propagation time

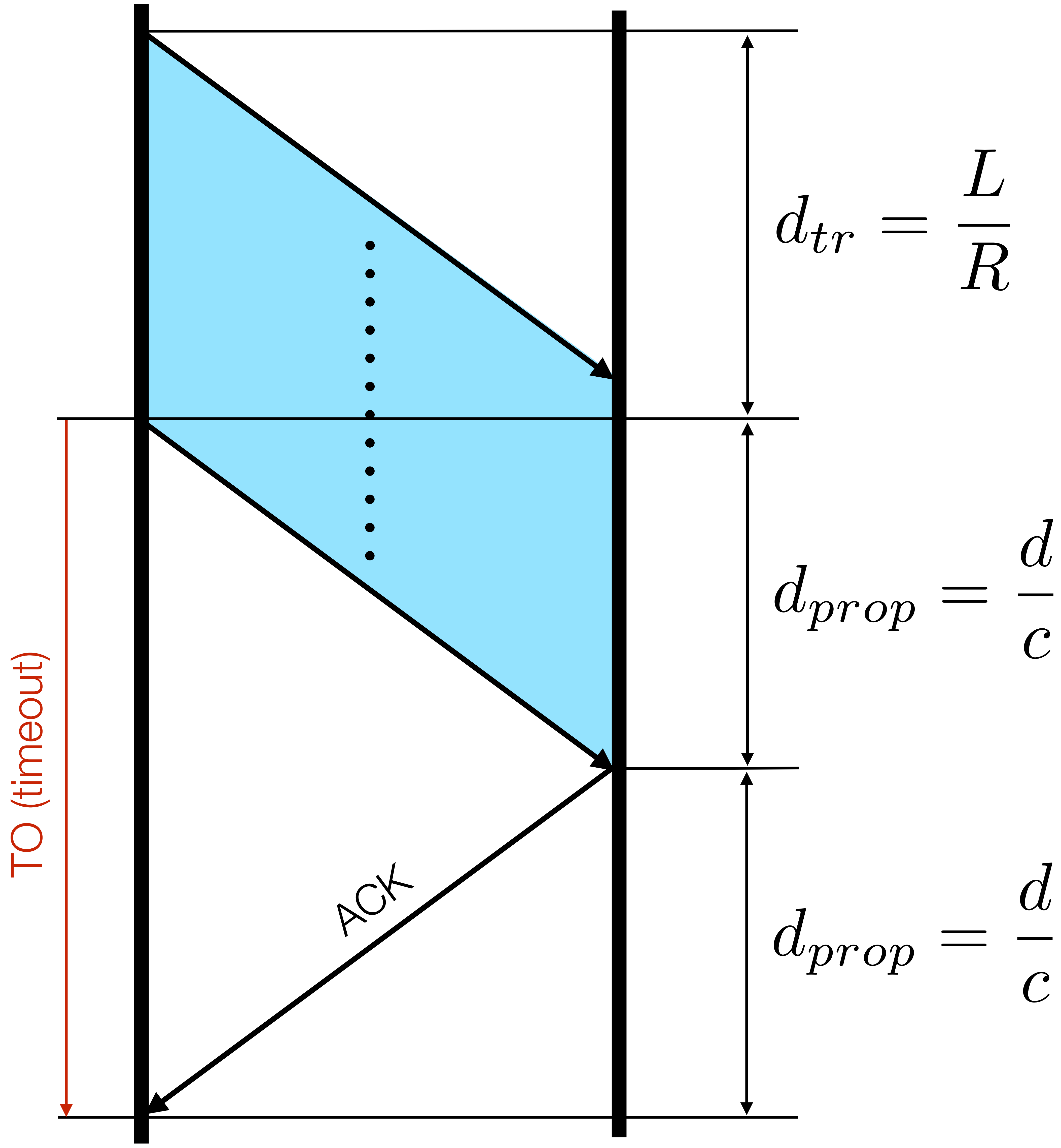
L - packet length

R - transmission rate

d - distance

c - propagation speed

Stop and Wait Protocol



d_{tr} - time to transmit

d_{prop} - propagation time

L - packet length

R - transmission rate

d - distance

c - propagation speed

Assuming:

- ACK is infinitely small
- "tight" $TO = 2d_{prop}$

Throughput and Efficiency

- Throughput (measured in packets per second)
  - maximum (theoretical) throughput
    - $T_{max} = \frac{1}{d_{tr}}$
  - actual throughput of stop and wait protocol (no loss)
    - $T_{act} = \frac{1}{d_{tr} + 2d_{prop}}$
- Efficiency (no loss scenario)
  - ratio of maximum vs actual throughput
    - $E = \frac{T_{act}}{T_{max}} = \frac{\frac{1}{d_{tr} + 2d_{prop}}}{\frac{1}{d_{tr}}} = \frac{d_{tr}}{d_{tr} + 2d_{prop}}$

Efficiency under loss

- First, let's assume that the first transmission fails but the retransmission succeeds:
  - the packet is retransmitted as soon as the ACK fails to arrive (not realistic, but helps to keep the math simple)

    - $$E = \frac{T_{act}}{T_{max}} = \frac{\frac{1}{2(d_{tr} + 2d_{prop})}}{\frac{1}{d_{tr}}} = \frac{1}{2} \cdot \frac{d_{tr}}{d_{tr} + 2d_{prop}}$$

Efficiency under loss

- Assuming the first $N - 1$ transmissions fail but the N^{th} one succeeds:
  - the packets are retransmitted as soon as the ACK fails to arrive
    - $E = \frac{T_{act}}{T_{max}} = \frac{\frac{1}{N(d_{tr} + 2d_{prop})}}{\frac{1}{d_{tr}}} = \frac{1}{N} \cdot \frac{d_{tr}}{d_{tr} + 2d_{prop}}$
- Same result if it takes on average N transmissions to deliver a packet
  - How to find N?

Efficiency under loss

- Assuming that a packet transmission fails with probability p:

# of Tx's	probability P_k	total time
1	$1-p$	$d_{tr} + 2d_{prop}$
2	$p(1-p)$	$2(d_{tr} + 2d_{prop})$
3	$p^2(1-p)$	$3(d_{tr} + 2d_{prop})$
$\vdots$	$\vdots$	$\vdots$
k	$p^{k-1}(1-p)$	$k(d_{tr} + 2d_{prop})$
$\vdots$	$\vdots$	$\vdots$

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

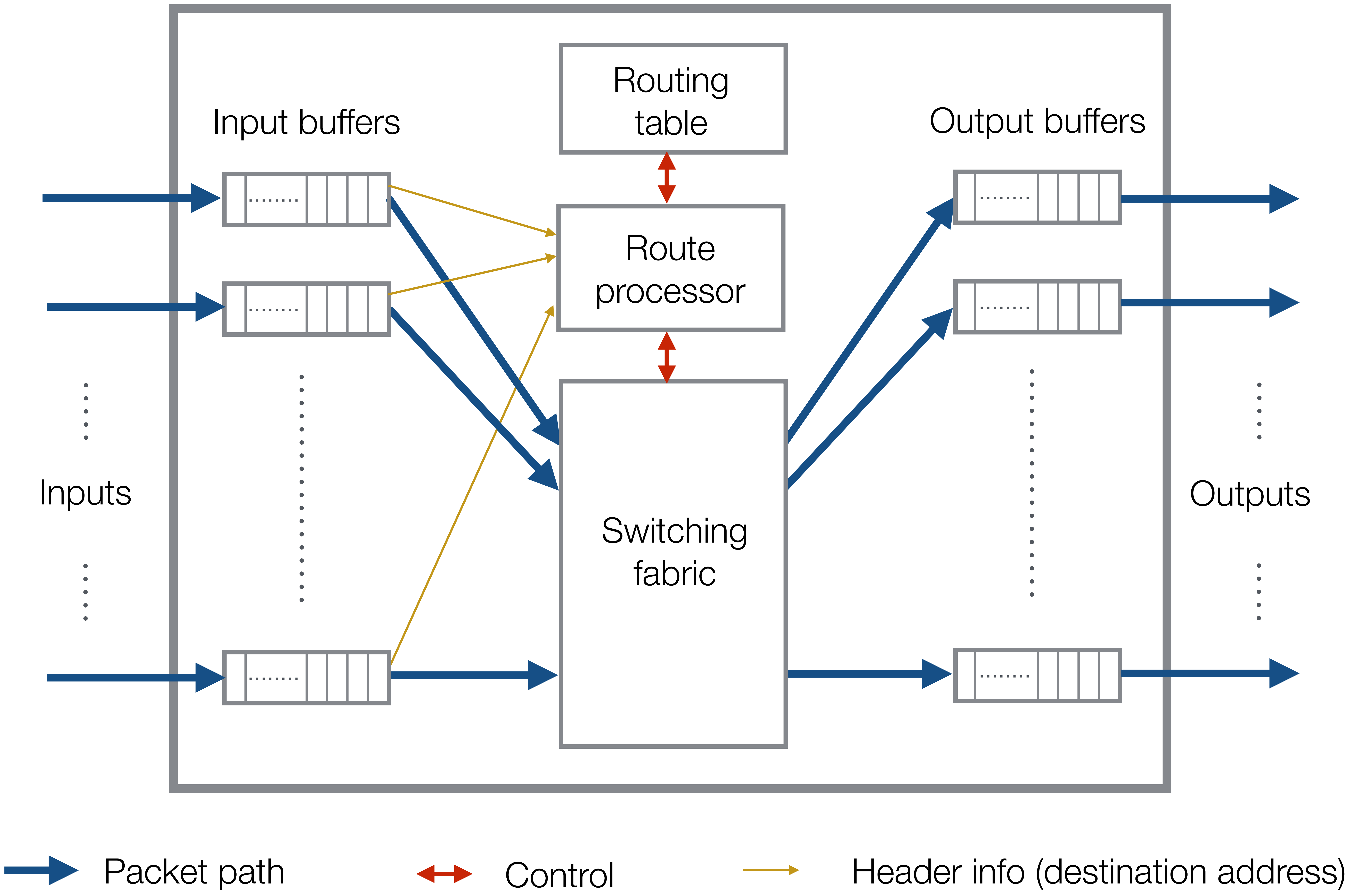
Efficiency under loss

- Finding N, the expected number of transmissions:
  - Recall, an expected value of a random variable X
    - $\mathrm{E}[X] = \sum_{i=1}^{n} x_i P_i = x_1 P_1 + x_2 P_2 + \cdots + x_n P_n$
  - so the expected number of transmissions N can be calculated as
    - $N = \sum_{i=1}^{\infty} i P_i = \sum_{i=1}^{\infty} i p^{i-1} (1-p) = \cdots = \frac{1}{1-p}$
  - and the efficiency of a *stop and wait* protocol E is
    - $E = (1-p) \cdot \frac{d_{tr}}{d_{tr} + 2d_{prop}}$

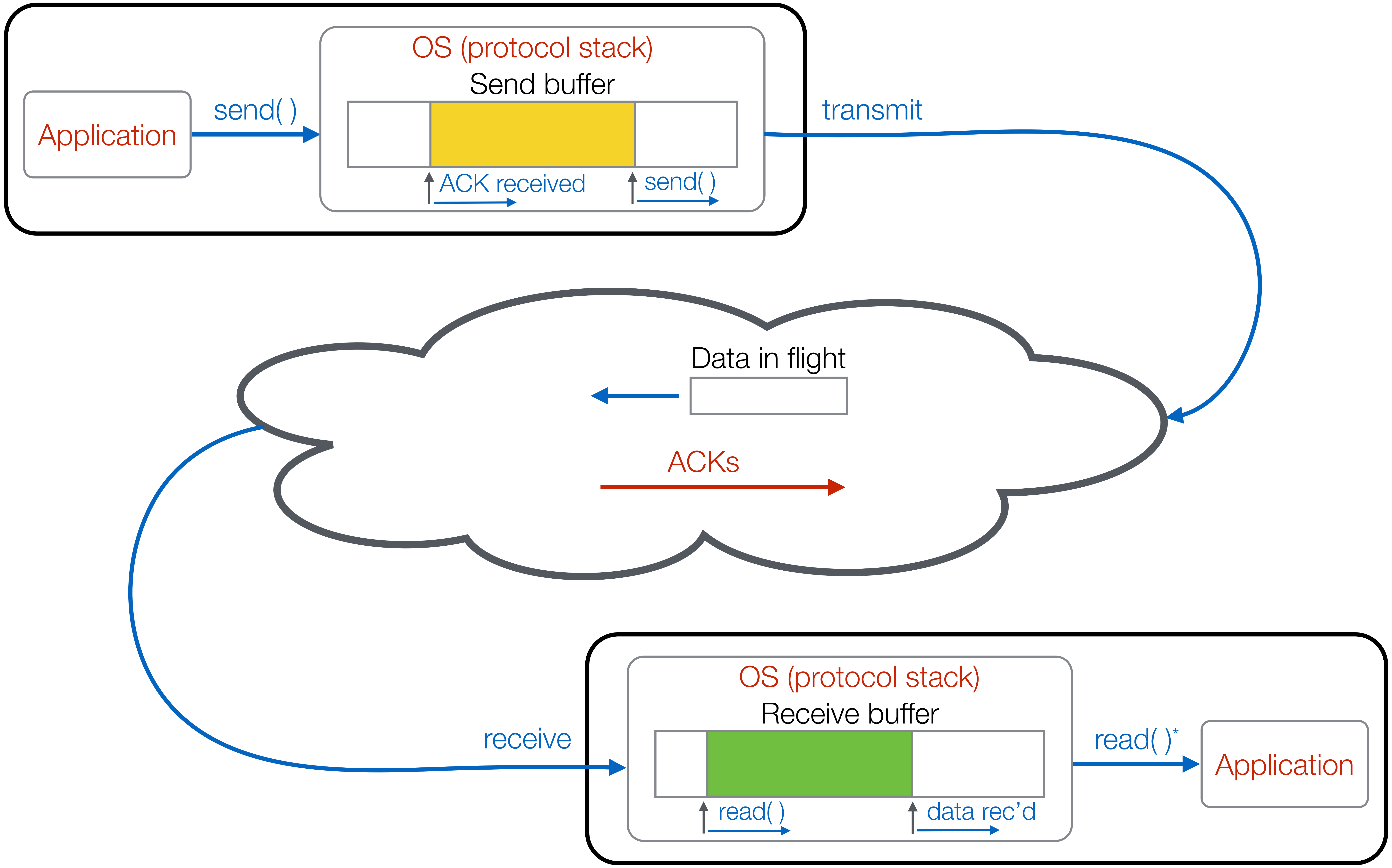
Queues are everywhere

- Applications
  - buffers/queues, APIs, threads
- Operating system kernel (protocol stack)
- Network interface cards
- Routers and switches
- … even in peripheral devices, such as storage

Anatomy of a router/switch



TCP buffering and data flow



(*) many APIs call the read() operation "receive" (eg: recv()), read is used here to avoid confusion with receiving data on an interface