3.2 Poisson Counting Process

Now we want to consider a continuous-time counting process. The time variable t runs continuously through an interval, and thus, $X(t)$ changes at infinitely many moments. We can obtain a continuous-time process as a limit of some discrete-time process whose frame size (time between trials) Δ approaches 0 (thus allowing more frames during any fixed period of time). We will let

$$\Delta \to 0, \text{ as } n \to \infty,$$

while keeping the arrival rate $\lambda = \text{const}$.

Think of movies, i.e. of video camera exposures. Although all motions on the screen seem continuous, we realize that an infinite amount of information could not be stored by any video device. Instead, what we see is a discrete sequence of exposures that run so fast that each motion seems continuous and smooth. Early-age video cameras shot exposures rather slowly; the interval Δ between successive shots was pretty long ($\sim 0.2 - 0.5$ sec). As a result, the quality of recorded video was rather low. Movies were "too discrete". Modern camcorders can shoot more than 200 exposures per second attaining $\Delta \leq 0.005$. With such a small Δ, the resulting movie seems perfectly continuous. A shorter frame Δ results in a "more continuous" process (see Figure 1).

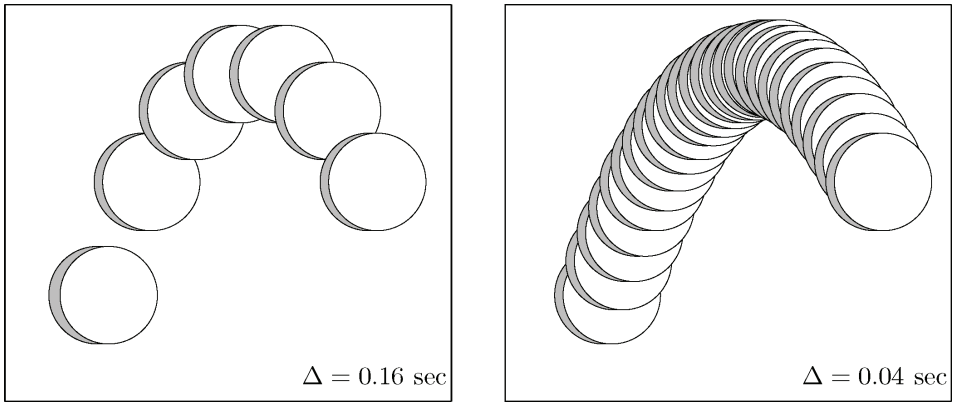

Fig. 1: Reducing the frame size Δ

So, let us take the limiting case of a Binomial counting process as $\Delta \to 0$.

Consider a Binomial counting process that counts arrivals occurring at a rate of λ / time unit. $X(t)$ denotes the number of arrivals occurring during time t.

– The *arrival rate* λ remains constant. Arrivals occur at the same rate, regardless of our choice of frame Δ.

– The *number of frames* during time t, $n = \dfrac{t}{\Delta} \to \infty$, as $\Delta \to 0$.

– The *probability of an arrival* during each frame, $p = \lambda \cdot \Delta \to 0$, as $\Delta \to 0$.

– $X(t)$, the *number of arrivals* during time t has a $B(n, p)$ distribution with expectation

$$E\big(X(t)\big) \;=\; np \;=\; \frac{t}{\Delta}p \;=\; \frac{p}{\Delta}t \;=\; \lambda t.$$

The pdf of $X(t)$ is

$$X(t) \begin{pmatrix} k \\ C_n^k p^k (1-p)^{n-k} \end{pmatrix}_{k=\overline{0,n}}.$$

So, as $n \to \infty$, the values will be $k = 0, 1, \ldots$, all the way to ∞. What about the corresponding probability $P(X = k) = C_n^k p^k (1-p)^{n-k}$? Let us see what this becomes.

$$\begin{aligned} P(X=k) &= \frac{n(n-1)\ldots(n-k+1)}{k!} p^k (1-p)^{n-k} \\ &= \frac{n(n-1)\ldots(n-k+1)}{k!} \left(\frac{\lambda t}{n}\right)^k \left(1 - \frac{\lambda t}{n}\right)^{n-k} \\ &= \frac{(\lambda t)^k}{k!} \cdot \frac{n(n-1)\ldots(n-k+1)}{n^k} \left(1 - \frac{\lambda t}{n}\right)^{-k} \left(1 - \frac{\lambda t}{n}\right)^{n} \\ &\overset{n\to\infty}{\longrightarrow} \frac{(\lambda t)^k}{k!} \cdot 1 \cdot 1 \cdot e^{-\lambda t}. \end{aligned}$$

So, the limiting pdf is

$$X(t) \begin{pmatrix} k \\ \dfrac{(\lambda t)^k}{k!} e^{-\lambda t} \end{pmatrix}_{k=0,1,\ldots}, \tag{3.1}$$

which means $X(t)$ has a Poisson $\mathcal{P}(\lambda t)$ distribution. This is a **Poisson counting process**.

Let us analyze what happens to the other characteristics.

Recall that the interarrival time $T = \Delta Y$, where Y has $SGeo(p)$ pdf. For its cdf, we have

$$\begin{aligned} F_T(t) &= P(T \le t) = P(\Delta Y \le n\Delta) \\ &= P(Y \le n) = F_Y(n) \\ &= 1-(1-p)^n = 1-\left(1-\frac{\lambda t}{n}\right)^n \\ &\stackrel{n\to\infty}{\longrightarrow} 1-e^{-\lambda t}. \end{aligned}$$

Hence,

$$f_T(t) = F_T'(t) = \lambda e^{-\lambda t},\ t>0, \tag{3.2}$$

so T has an $Exp(\lambda)$ pdf. Then its expectation and variance are given by

$$E(T) = \frac{1}{\lambda},\ V(T) = \frac{1}{\lambda^2}. \tag{3.3}$$

Furthermore, the time T_k of the k-th arrival is the sum of k independent $Exp(\lambda)$ interarrival times, so has $Gamma(k, 1/\lambda)$ distribution, with

$$E(T_k) = k\frac{1}{\lambda},\ V(T_k) = k\frac{1}{\lambda^2}. \tag{3.4}$$

Remark 3.1. From here, we immediately find the already-known Gamma-Poisson formula: if we want the k-th arrival to be *before or at* time t, then that means the number of arrivals by time t (X, having a Poisson $\mathcal{P}(\lambda t)$ distribution) should be *at least* k.

$$\begin{aligned} P(T_k \le t) &= P(X \ge k), \text{ or, equivalently,} \\ P(T_k > t) &= P(X < k). \end{aligned} \tag{3.5}$$

A sample path of some Poisson process is shown in Figure 2.

Example 3.2. The number of hits to a certain web site follows a Poisson process with the intensity parameter $\lambda = 7$ hits per minute. On the average, how much time is needed to get $10,000$ hits? What is the probability that this will happen within 24 hours?

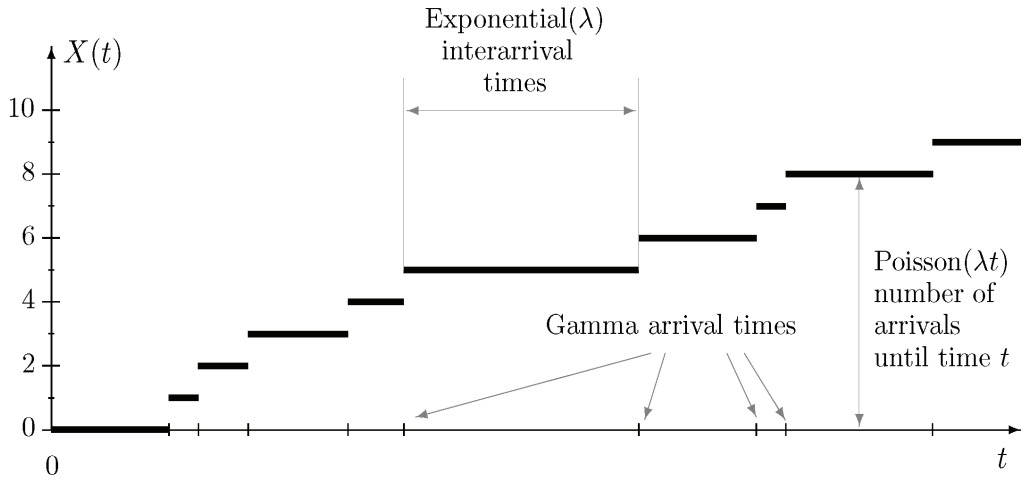

Fig. 2: Poisson process sample path

Solution. The time of the $10,000$-th hit T_k has $Gamma(k, 1/\lambda)$ distribution with parameters $k = 10,000$ and $\lambda = 7$ min^{-1}. Then, the expected time of the k-th hit is

$$\mu \;=\; E(T_k) \;=\; k \cdot \frac{1}{\lambda} \;=\; 1,428.6 \text{ minutes} \;=\; 23.8 \text{ hours}.$$

Also, the standard deviation is

$$\sigma \;=\; \text{Std}(T_k) \;=\; \sqrt{V(T_k)} \;=\; \sqrt{k} \cdot \frac{1}{\lambda} \;=\; 14.3 \text{ minutes}.$$

The probability that the $10,000$-th hit will happen within 24 hours is

$$P(T_k < 24 \cdot 60) \;=\; P(T_k < 1440) \;=\; \text{gamcdf}(1440, 10000, 1/7) \;=\; 0.7885.$$

Alternatively, with the Gamma-Poisson formula, (3.5),

$$\begin{aligned} P(T_k < 1440) \;&=\; P(T_k \le 1440) \;=\; P(X \ge k) \\ &=\; 1 - P(X < k) \;=\; 1 - P(X \le k-1) \\ &=\; 1 - \text{poisscdf}(9999, 7 \cdot 1440) \;=\; 0.7885, \end{aligned}$$

where X has a Poisson $\mathcal{P}(7 \cdot 1440)$ distribution.

■

Rare events

It can be shown that for a Poisson counting process, the following hold

$$\begin{aligned} a)\ P\Big(X(t+\Delta)-X(t)=1\Big) &= \lambda\Delta + o(\Delta), \\ b)\ P\Big(X(t+\Delta)-X(t)>1\Big) &= o(\Delta),\ \text{as}\ \Delta\to 0. \end{aligned} \tag{3.6}$$

The term $o(\Delta)$ denotes a *negligible* term, a quantity converging to 0 faster than Δ, i.e. $\dfrac{o(\Delta)}{\Delta}\to 0$, as $\Delta\to 0$.

For a Binomial counting process, the probability in part a) is the probability of 1 arrival during 1 frame and it equals $p=\lambda\Delta$, whereas the probability of *more* than 1 arrival is 0 (part b)).

For a Poisson process, these probabilities may be different, but only by "a little". The differences $X(t+\Delta)-X(t)$ are called *increments*. For a Poisson process, an increment is the number of arrivals during the time interval $(t, t+\Delta]$.

Relations (3.6) formally describe the concept of **rare events**. These events occur at random times and the probability of a new event occurring during a short interval of time is proportional to the length of that interval. Probability of *more* than 1 event during that time is much smaller, compared to the length of the interval. For such sequences of events, a Poisson process is a suitable stochastic model. Examples of rare events include telephone calls, message arrivals, virus attacks, errors in codes, traffic accidents, natural disasters, network blackouts, and so on. Relations (3.6) can also be considered as the definition of a Poisson process.

Example 3.3. Let us revisit the example from last time and model the arrivals of messages with a Poisson counting process, keeping the same arrival rate of 6 messages per minute.
a) Find the probability of no messages arriving during the next 1 minute.
b) Compute the probability of more than 35 messages arriving during the next 6 minutes.
c) Find the probability of more than 350 messages arriving during the next hour.
d) What is the average interarrival time and its standard deviation?
e) Compute the probability that the next message does not arrive during the next 20 seconds.

Solution.
a) We have $t=1$ minute and $\lambda=6$ / minute. The number of messages arriving during 1 minute, $X(1)$, has a Poisson distribution with parameter $\lambda t=6$. So the desired probability is

$$P\big(X(1)=0\big) \;=\; \mathrm{pdf}_{X(1)}(0) \;=\; poisspdf(0,6) \;=\; 0.0025.$$

b) Similarly, the number of messages arriving in $t = 6$ minutes, $X(6)$, has a Poisson distribution with parameter $\lambda t = 36$. Then the probability of more than 35 messages arriving during that time is

$$\begin{aligned} P\big(X(6) > 35\big) &= 1 - P\big(X(6) \le 35\big) \\ &= 1 - \mathrm{cdf}_{X(6)}(35) \\ &= 1 - \mathrm{poisscdf}(35, 36) \\ &= 0.5222. \end{aligned}$$

c) Again, in $t = 1$ hour $= 60$ minutes, the number of arriving messages, $X(60)$, has Poisson distribution with parameter $\lambda t = 360$. So, the probability of more than 350 messages arriving during the next hour is

$$\begin{aligned} P\big(X(60) > 350\big) &= 1 - P\big(X(60) \le 350\big) \\ &= 1 - \mathrm{cdf}_{X(60)}(350) \\ &= 1 - \mathrm{poisscdf}(350, 360) \\ &= 0.6894. \end{aligned}$$

Notice that again, as in the case of a Binomial process, "more than 35 messages in 6 minutes" is **not** the same as "more than 350 messages in 60 minutes".

d) The interarrival time, T, now has an $Exp(\lambda) = Exp(6)$ distribution, so

$$\begin{aligned} E(T) &= \frac{1}{\lambda} = \frac{1}{6} \text{ minutes} = 10 \text{ seconds}, \\ Std(T) &= \sqrt{V(T)} = \sqrt{\frac{1}{\lambda^2}} = \frac{1}{6} \text{ minutes} = 10 \text{ seconds}. \end{aligned}$$

Notice that the average interarrival time has not changed. This is to be expected, since jobs (messages) arrive at the same rate, λ, regardless of whether their arrivals are modeled by a Binomial or a Poisson process.

However, the standard deviation is slightly increased (it was 9.5 seconds before). That is because a Binomial process has a restriction on the number of arrivals during each frame, thus reducing variability.

e) Either we work with seconds (so $\lambda = \dfrac{1}{10}$ / second) and compute the probability $P(T > 20)$, where T has an $Exp(1/10)$ distribution), or in minutes ($\lambda = 6$ / minute, 20 seconds $= 1/3$ minutes)

and compute the probability $P(T > 1/3)$, where T has an $Exp(6)$ distribution. Either way, we have

$$\begin{aligned} P(T > 20) &= 1 - P(T \le 20) = 1 - \mathrm{cdf}_T(20) \\ &= 1 - \mathit{expcdf}(20, 10) = 0.1353, \\ P(T > 1/3) &= 1 - P(T \le 1/3) = 1 - \mathrm{cdf}_T(1/3) \\ &= 1 - \mathit{expcdf}(1/3, 1/6) = 0.1353. \end{aligned}$$

Again, this is the same as 0 arrivals in 20 seconds, where the number of arriving messages, $X(20)$, has a Poisson distribution with parameter $\lambda t = 2$ (or 0 arrivals in $1/3$ minutes, where the number of arriving messages, $X(1/3)$, has a Poisson distribution with parameter $\lambda t = 2$).

$$\begin{aligned} P\big(X(20) = 0\big) &= \mathrm{pdf}_{X(20)}(0) = \mathit{poisspdf}(0, 2) = 0.1353, \\ P\big(X(1/3) = 0\big) &= \mathrm{pdf}_{X(1/3)}(0) = \mathit{poisspdf}(0, 2) = 0.1353. \end{aligned}$$

■

Simulation of a Poisson counting process

Simulation of continuous-time processes has a clear problem. The time t runs continuously through the time interval, taking infinitely many values in this range. However, we cannot store an infinite number of random variables in the memory of our computer! For most practical purposes, it suffices to generate a discrete-time process with a rather short frame Δ (this process is called discretization).

But, Poisson processes can be generated without discretization. Indeed, although they are continuous-time, the value of $X(t)$ can change only a finite number of times during each interval. The process changes every time a new "rare event" or arrival occurs, which happens a Poisson $\mathcal{P}(\lambda t)$ number of times during an interval of length t. Then, it suffices to generate these moments of arrival. As we know, the first arrival time has $Exp(\lambda)$ distribution, and each interarrival time is $Exp(\lambda)$ distributed, too. So, we generate them using the ITM $(-\frac{1}{\lambda}\ln U)$ and then, generate a segment of a Poisson process during a time interval $[0, M]$ by counting the number of such times in that interval.

Algorithm 3.4.

1. Given:
   T_{max} time period,
   λ arrival rate.
2. Initial arrival time:

$T = -1/\lambda \cdot \ln U$; growing array containing arrival times,

$last = T$; last (most recent) arrival time,

3. Count number of arrivals until time T_{max}:

    while $last \leq T_{max}$

      $last = last - 1/\lambda \cdot \ln U$; new arrival time

      $T = [T,\ last]$; array of arrival times extended

    end

Chapter 4. Queuing Systems

As mentioned before, stochastic processes have a wide variety of applications in many fields of research.
Among them, of special interest are *queuing systems*. Queuing systems are service facilities designed to serve randomly arriving jobs.
We are now in the position to analyze a broad range of queuing systems that play a crucial role in Computer Science and other fields.

1 Basic Notions; Main Components

Definition 1.1. *A **queuing system** is a facility consisting of one or several servers designed to perform certain tasks or process certain jobs, and a queue of jobs waiting to be processed.*
*A queuing system is called **stationary** if its distribution characteristics do not change over time.*

Jobs arrive at the queuing system at some arrival rate, wait for an available server, get processed by this server, and leave.

Example 1.2. Examples of queuing systems are:

- a computer executing tasks sent by its users;
- a printer processing jobs sent to it from different computers;
- cars at a toll booth, gas station, or auto service facility;
- an internet service provider whose customers connect to the internet, browse, and disconnect;
- people waiting in line at a cafeteria, or a bank;
- a medical office serving patients;
- airplanes waiting to take off or land, at an airport;
- a TV or radio channel viewed (listened to) by many people at various times;
- a customer service with one or several representatives on duty answering calls from their customers;
- people connecting to Facebook, Instagram, TikTok and so on.

We are now equipped to study a wide variety of queuing systems.

1.1 Main components of a queuing system

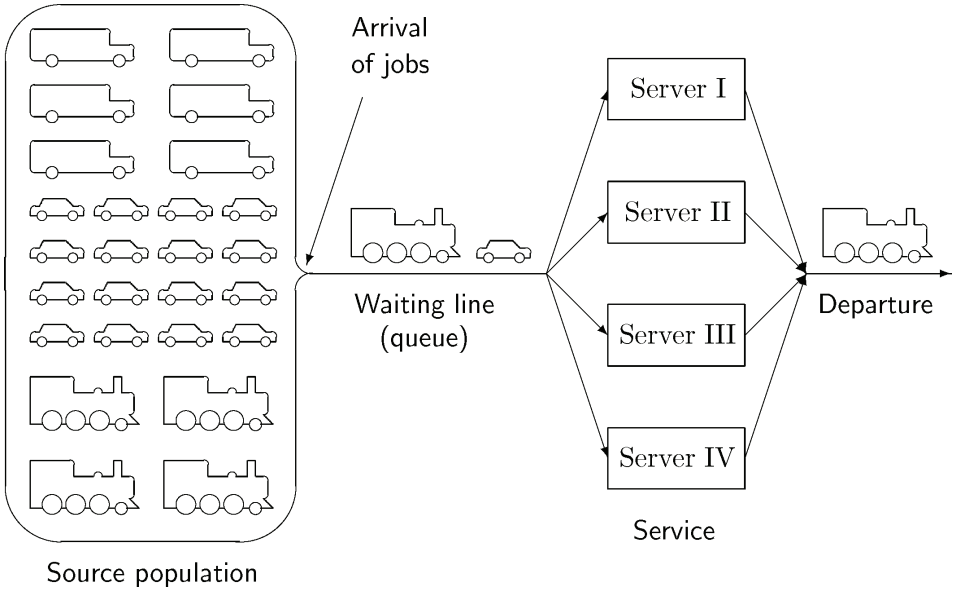

Fig. 3: Main components of a queuing system

The main stages of a queuing system are depicted in Figure 3.

Arrival

Jobs arrive to a queuing system at random times. The **number of arrivals** that occurred by the time t is a counting process $A(t)$. In stationary queuing systems, arrivals occur at **arrival rate**

$$\lambda_A \;=\; \frac{E\big(A(t)\big)}{t},\ \forall\, t>0, \tag{1.1}$$

i.e. the expected number of arrivals per time unit.
Then, the expected time between arrivals, the **mean interarrival time** is

$$\mu_A \;=\; \frac{1}{\lambda_A}. \tag{1.2}$$

Remark 1.3. Usually, arrived jobs are processed in the order of their arrivals, on a "first come-first serve" basis. When a new job arrives, it may find the system in different states.
− If one server is available, it will take the new job.

– If several servers are available, the job may be randomly sent to one of them, or the server may be assigned according to some rules. For example, the fastest server or the least loaded server may be assigned to process the new job.
– If all servers are busy, the new job will join the queue, wait until a previously arrived job is completed (accumulating waiting time) and get routed to the next available server.

Other constraints may take place. For example, a queue may have a *buffer* that limits the number of waiting jobs (like a parking garage or a restaurant). Such a queuing system is a system with **limited capacity**. The total number of jobs in it at any time is bounded by some constant C (capacity). If the capacity is full, a new job cannot enter the system until another job departs.

Also, jobs may leave the queue prematurely, say, after an excessively long waiting time (think people waiting for a table at a busy restaurant).

Servers may also open and close during the day as people need breaks or servers need maintenance.

Complex queuing systems with many extra conditions may be difficult to study analytically; however, Monte Carlo methods can be employed, to find (estimate) characteristics that evaluate the performance of most queuing systems.

Service

Once a server becomes available, it starts processing the next assigned job. Service times are usually random, they depend on the amount of work required by each task and on the efficiency of the server (slow or rapid computer, some people may work faster than others, etc).

The **average service time** is denoted by μ_S and it may vary from one server to another. The **service rate** is defined as the average number of jobs processed by a continuously working server during one unit of time, i.e.

$$\lambda_S \;=\; \frac{1}{\mu_S}. \tag{1.3}$$

Departure

When the service is completed, the job leaves the system.

To summarize, the following parameters and random variables describe the performance of a queuing system.

$$\begin{aligned}
\lambda_A &= \text{arrival rate} \\
\lambda_S &= \text{service rate} \\
\mu_A &= 1/\lambda_A = \text{mean interarrival time} \\
\mu_S &= 1/\lambda_S = \text{mean service time} \\
r &= \lambda_A/\lambda_S = \mu_S/\mu_A = \textbf{utilization} \text{ (arrival-to-service ratio)}
\end{aligned} \tag{1.4}$$

Random variables:

$$\begin{aligned}
X_s(t) &= \text{number of jobs receiving service at time } t \\
X_w(t) &= \text{number of jobs waiting in a queue at time } t \\
X(t) &= X_s(t) + X_w(t) = \text{total number of jobs in the system at time } t \\
S_k &= \text{service time for the } k^{\text{th}} \text{ job} \\
W_k &= \text{waiting time for the } k^{\text{th}} \text{ job} \\
R_k &= S_k + W_k = \textbf{response time} \text{ for the } k^{\text{th}} \text{ job, i.e.} \\
&\quad \text{total time a job spends in the system, from arrival to departure}
\end{aligned} \tag{1.5}$$

A queuing system is **stationary** if the pdf's of S_k, W_k and R_k are independent of k, in which case, we omit the index k in notation.

Utilization r is an important parameter. It shows whether or not a system can function under the current (or higher) rate of arrivals, and whether the system is over- or underloaded.

The main goal in studying queuing systems will be finding the distribution of $X(t)$, the total number of jobs in the system. Then other characteristics can be assessed from that and a comprehensive performance evaluation of a queuing system can be made.