Discrete System

Nipun Thapa

Continuous Models

- Many systems comprise computers, communications networks, and other digital systems to monitor and control physical (electrical, mechanical, thermodynamic, etc.) processes.
- Models of these systems have some parts modeled as discrete event systems, other parts modeled with continuous (differential or differential-algebraic) equations, and the interaction of these parts is crucial to understanding the system's behavior.

Continuous Models

- The interaction of continuous and discrete event models is necessarily discrete. For example, a digital thermometer reports temperature in discrete increments, electrical switches are either open or closed, a threshold sensor is either tripped or it is not. Discrete interactions in a combined continuous-discrete event simulation are managed just as before: the models interact by producing output events and reacting to input events.

Discrete System)

Analog Computers:

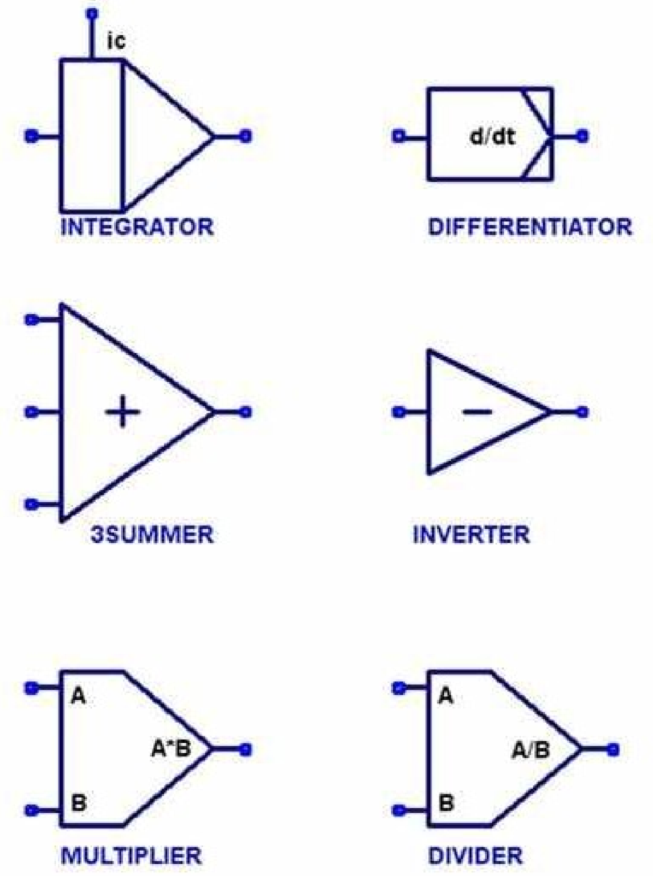
- Before general availability of digital computers, there existed devices whose behavior is equivalent to a mathematical operation. Putting together combination of such devices in a manner specified by a mathematical model of a system, allowed the system to be simulated.
- Specific devices have been created for particular system but with so general technique, it is customary to refer them as analog computers or when they are primarily used to solve differential equation models, as differential analyzers.
- The most widely used form of analog computer is the electronic analog computer, based on the use of high gain direct current (dc) amplifiers called operational amplifiers. Voltages in the computer are equated to mathematical variables and the operational amplifiers can add and integrate the voltages.

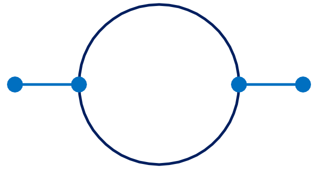
Discrete System)

Analog Computers..

- Electronic analog computers are limited in accuracy. It is difficult to carry the accuracy of measuring a voltage beyond a certain point. Secondly, many assumptions are made. Also, operational amplifiers have a limited dynamic range of output.
- But many users prefer to use analog computers. The analog representation of a system is often more natural as it directly reflects the structure of the system thus simplifying both the setting-up of a simulation and the interpretation of results. Also analog computer can be solving many equations in a truly simultaneous manner so is faster.

Analog Methods





SINGLE MULTIPLIER

Analog Methods

- The general method by which analog computers are applied can be demonstrated using second order differential equation.

$$M\ddot{x} + D\dot{x} + kx = kF(t)$$

Solving the equation for the highest order derivate gives,

$$M\ddot{x} = kF(t) - D\dot{x} - kx$$

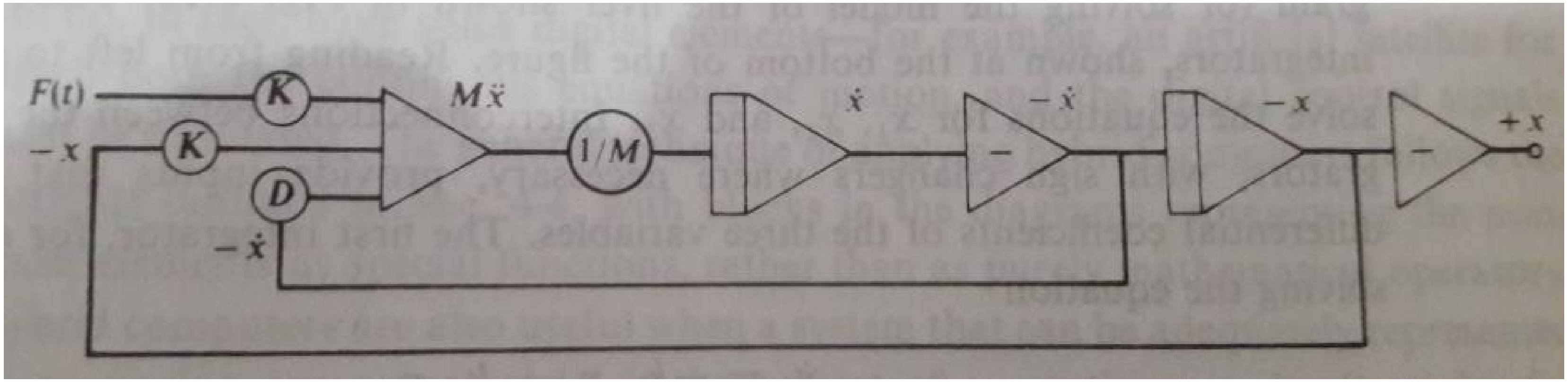


Fig: Automobile suspension problem

- Suppose a variable representing the input $F(t)$ is supplied, assume there exists variables representing $-x$ and $-\dot{x}$. These three variables can be scaled and added to produce $M\ddot{x}$. Integrating it with a scale factor $1/M$ produces $\dot{x}$. Changing sign produces $-\dot{x}$, further integrating produces $-x$, a further sign inverter is included to produce $+x$ as output.
- When a model has more than one independent variable, a separate block diagram is drawn for each independent variable and where necessary, interconnections are made between the diagrams.

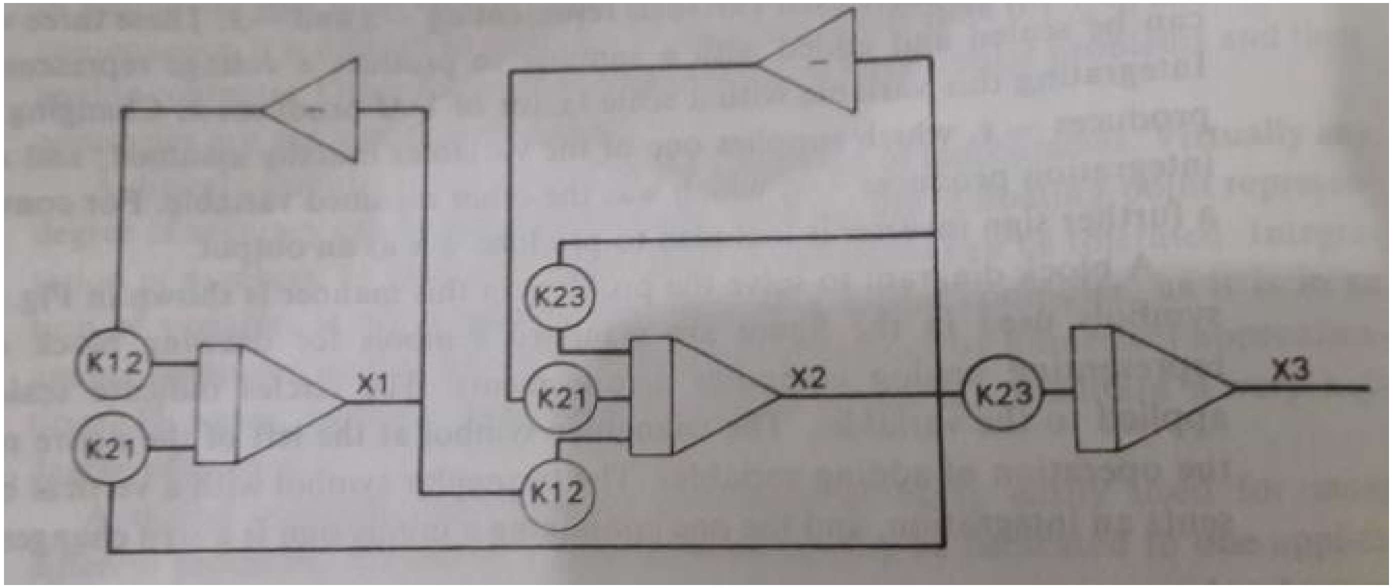
Discrete System)

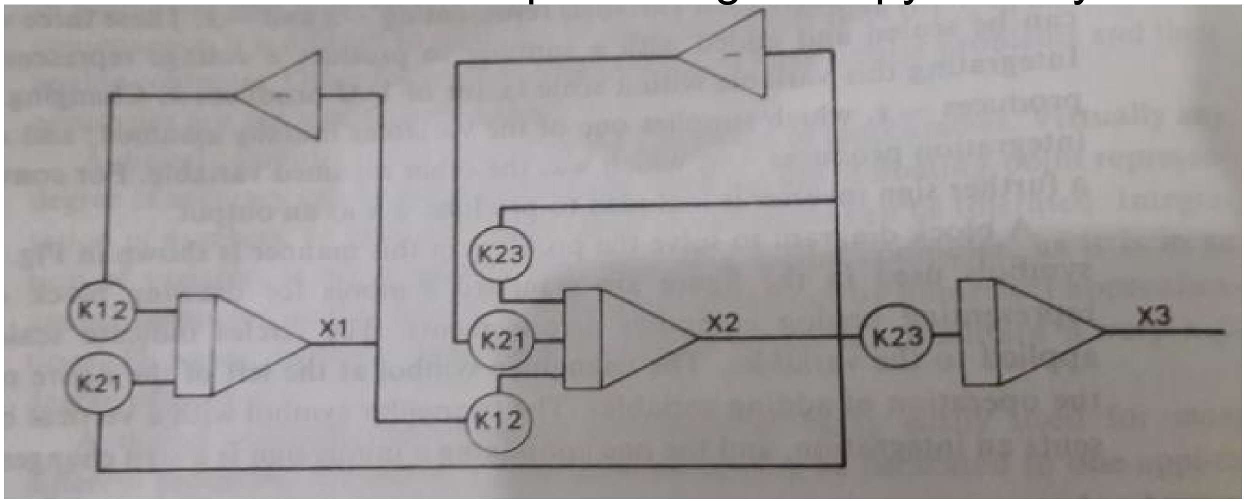
Q.> Design analog computer of

$$\dot{x1} = -k_{12}x_1 + k_{21}x_2$$

$$\dot{x2} = k_{12}x_1 - (k_{21} - k_{23})x_2$$

$$\dot{x3} = k_{23}x_2$$





There are three integrators. Reading from left to right, they solve the equations for $x1$, $x2$ & $x3$. Interconnections between the three integrators with sign changers where necessary provides inputs that define the differential coefficients of the three variables.

First integrator, for example is solving the equation,

$$\dot{x1} = -k_{12}x_1 + k_{21}x_2$$

The second integrator is solving the equation

$$\dot{x2} = k_{12}x_1 - (k_{21} + k_{23})x_2$$

In this case, the variable $x2$ is being used twice as an input to the integrator, so that the two coefficients $k21$ and $k23$ can be changed independently. The last integrator solves the equation

$$\dot{x3} = k_{23}x_2$$

Q.> Design analog computer of

$$\ddot{x}_1 - k\dot{x}_1 + k_1 \dddot{x}_1 = a x_3$$

$$\dddot{x}_2 + k_2 x_1 - k_1 x_3 - k_3 x_2 = 0$$

$$\dddot{x}_3 - k_2 x_2 + \ddot{x}_1 = f(x)$$

Q.> Design analog computer

of $\dot{y}_1 - \dot{x}_1 + k_1 \dddot{z}_1 = 0$

$$\ddot{x}_1 = x_1 - \dot{x}_1 z_1$$

$$\dddot{z}_1 = k_2 x_1 + \dddot{x}_1$$

Q.> Design analog computer

of $\dot{x} - k\dot{x} + k\dddot{x} = x$

$$\dot{y} = x - k z - k_3 x$$

$$\dddot{z} = k y + \ddot{z}$$

Hybrid Simulation

- For most studies the model is clearly either of a continuous or discrete nature and that is the determining factor in deciding whether to use an analog or digital computer for system simulation. However, there are times when an analog and digital computers are combined to provide simulation.
- In this circumstances hybrid simulation is used. Hybrid simulation is provided by combining analog and digital computers. The form taken by hybrid simulation depends upon the applications.
- Here one computer may be simulating the system being studied while other is providing a simulation of the environment in which the system is to operate. It is also possible that the system being simulated is an interconnection of continuous and discrete subsystems, which can be modeled by an analog and digital computer being linked together.

Discrete System)

Hybrid Simulation

- The introduction of hybrid simulation required certain technological developments for its exploitation. High speed converters are needed to transform signals from one form of representation to the other.
- Practically, the availability of mini computers has made hybrid simulation easier, by lowering costs and allowing computers to be dedicated to an application. The term "hybrid simulation" is generally reserved for the case in which functionality distinct analog and digital computers are linked together for the purpose of simulation.

Discrete System)

Digital- Analog Simulators

- ***To avoid the disadvantages of analog computers, many digital computer programming languages have been written to produce digital analog simulators.***
- They allow a continuous model to be programmed on a digital computer in essentially the same way as it is solved on an analog computer.
- ***The language contains macro instructions that carryout the actions of adders, integrators and sign changers.***
- More powerful techniques of applying digital computers to the simulation of continuous systems have been developed.
- ***As a result, digital analog simulators are not now in expensive use.***

Feedback System

The system takes feedback from the output i.e. input is coupled with output. Example can be; heat monitoring and control system.

- Issues – amplification and correction of feedback
- Negative feedback – control variable is proportional with output
- Positive feedback – control variable and output are inversely proportional

Other examples;

- Aircraft system
- Error Correction mechanism

Discrete System)

Feedback System

- A significant factor in the performance of many systems is that coupling occurs
between the input and output of the system.
- The term feedback is used to describe the phenomenon.
- A home heating system controlled by a thermostat is a simple example of a feedback system.
- The system has a furnace whose purpose is to heat a room, and the output of the system can be measured as a room temperature. Depending upon whether the temperature is below or above the thermostat setting, the furnace will be turned on or off, so that information is being feedback from the output to input. In this case there are only two states either the furnace is on or off.

Feedback System

- Next example of feedback system in which there is continuous control is the aircraft system.
- Here the input is a desired aircraft heading and the output is the actual heading. The **gyroscope** of the autopilot is able to detect the difference between the two headings.
- A feedback is established by using the difference to operate the control surface. Since change of heading will then affect the signal being used to control the heading.
- The difference between the desired signal θ_t and actual heading θ_0 is called the error signal, since it is a measure of the extent to which the system from the desired condition. It is denoted by є.

Feedback System

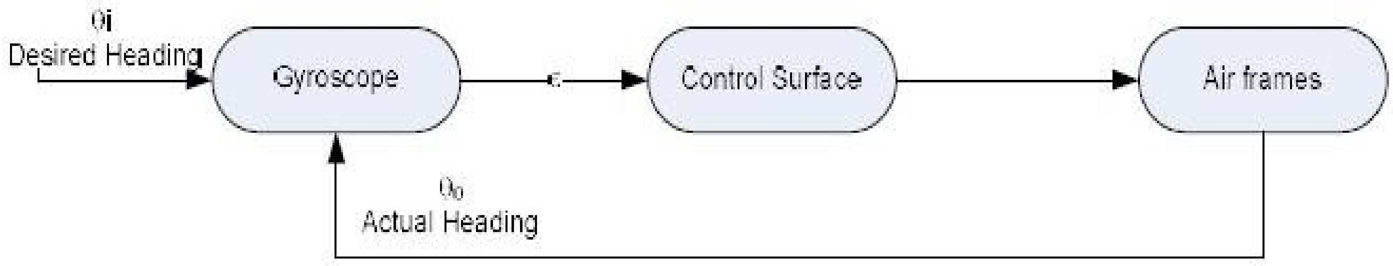


Fig: An aircraft under autopilot Control (Continuous System)

$\epsilon = \theta t - \theta_0$ --------- (1)

Torque = $K\epsilon - D\dot{\theta}_0$ --------------------(2)

We also know that, In terms of angular acceleration

Torque= $I\ddot{\theta}_0$ ____________________(3)

From equation (1), (2), (3)

$I\ddot{\theta}_0 + D\dot{\theta}_0 + K\theta_0 = K\theta_t$

If we divide both sides of equation by I

$\zeta w \theta_0$ =D/I

w^2 =K/I

$\ddot{\theta}_0 + 2\zeta w \theta_0 + w^2\theta_0 = w^2\theta_t$, this is a second order differential equation.

Simulation time and simulation clock

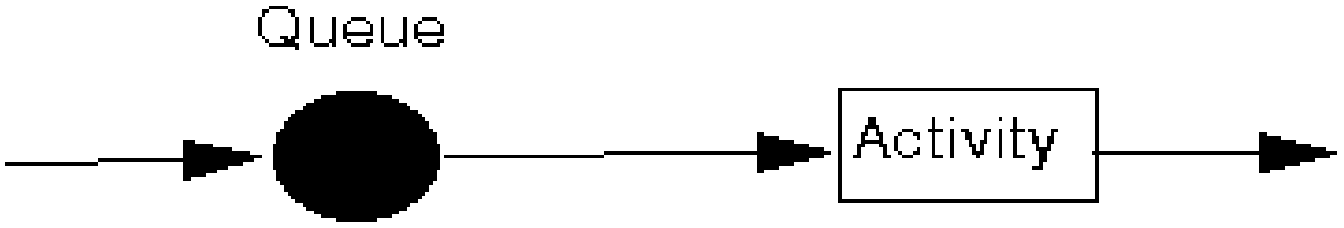
- When you simulate a model related to time (for example, a transition with a time trigger), Model Analyst will obtain simulation time from a simulation clock.
- The simulation time is the amount of time spent on simulating a model.
- Model Analyst also uses the simulation time in a timestamp of a signal instance in the Simulation Log, in a time series chart, and on messages of a generated Sequence diagram.

Simulation time and simulation clock

There are three types of simulation clocks in Model Analyst:

- **Built-in clock**. This is the default simulation clock.
- **Internal simulation clock** . This clock is designed to precisely control the simulation time. Its implementation is based on UML run-to-completion semantics and internal completion events.
- **Model-based clock**. You can select the model-based clock by making the property as the time value tag definition of a Simulation Config.

Arrival Processes



Arrival process:

- how customers arrive e.g. singly or in groups (batch or bulk arrivals)
- how the arrivals are distributed in time (e.g. what is the probability distribution of time between successive arrivals (the ***interarrival time distribution***))
- whether there is a finite population of customers or (effectively) an infinite number

Service mechanism:

- a description of the resources needed for service to begin
- how long the service will take (the ***service time distribution***)
- the number of servers available
- whether the servers are in series (each server has a separate queue) or in parallel (one queue for all servers)
- whether preemption is allowed (a server can stop processing a customer to deal with another "emergency" customer)

Queue characteristics:

- how, from the set of customers waiting for service, do we choose the one to be served next (e.g. FIFO (first- in first-out) also known as FCFS (first-come first served); LIFO (last-in first-out); randomly) (this is often called the *queue discipline*)
- do we have:
  - **balking** (customers deciding not to join the queue if it is too long)
  - **reneging** (customers leave the queue if they have waited too long for service)
  - **jockeying** (customers switch between queues if they think they will get served faster by so doing)
  - a queue of finite capacity or (effectively) of infinite capacity

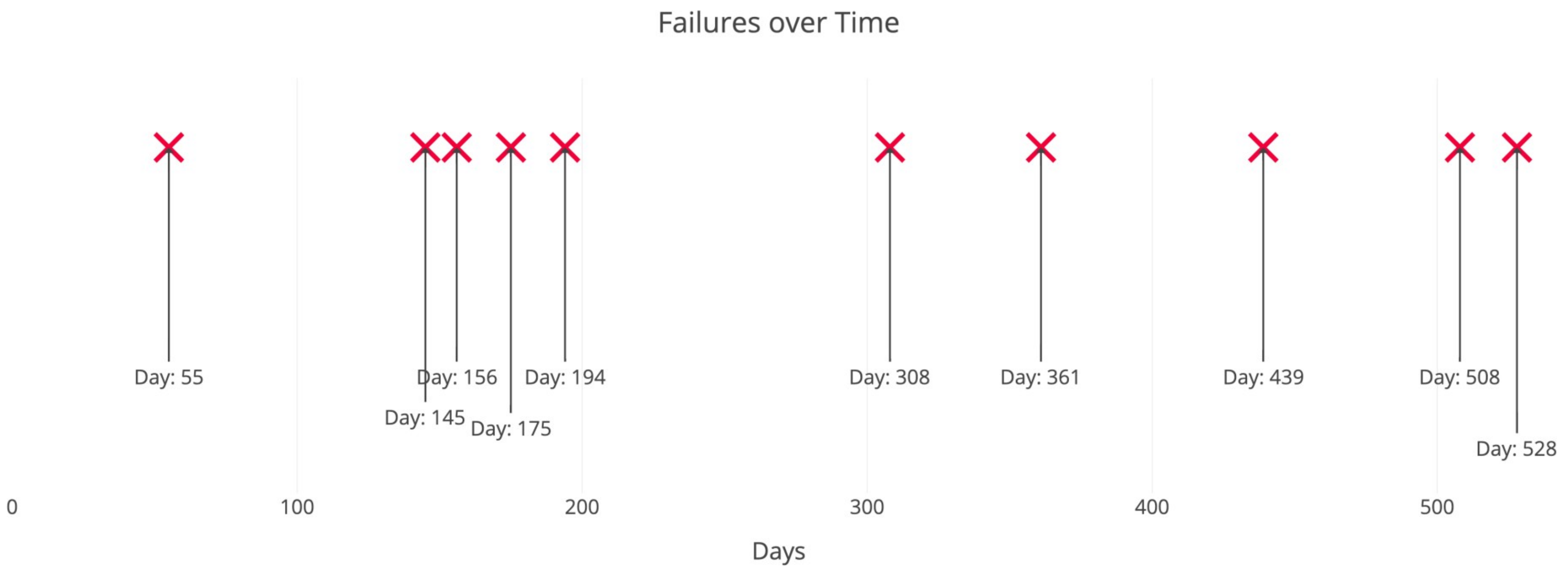
Poisson Process

- A Poisson Process is a model for a series of discrete event where the *average time* between events is known, but the exact timing of events is random.
- The arrival of an event is independent of the event before (waiting time between events is memoryless). For example, suppose we own a website which our content delivery network (CDN) tells us goes down on average once per 60 days, but one failure doesn't affect the probability of the next. All we know is the average time between failures.

Unit-02 (Simulation of continuos and Discrete System)

Poisson Process

- This is a Poisson process that looks like:



Poisson Process

- The important point is we know the *average time between events* but they are randomly spaced (stochastic).
- We might have back-to-back failures, butwe could also go years between failures due to the randomness of the process.

Poisson Process

A Poisson Process meets the following criteria (in reality many phenomena modeled as Poisson processes don't meet these exactly):

- Events are **independent** of each other. The occurrence of one event does not affect the probability another event will occur.
- The average rate (events per time period) is constant.
- Two events cannot occur at the same time.

Discrete System)

Poisson Process

- Common examples of Poisson processes are customers calling a help center, visitors to a website, radioactive decay in atoms, photons arriving at a space telescope, and movements in a stock price.
- Poisson processes are generally associated with time, but they do not have to be.
- In the stock case, we might know the average movements per day (events per time), but we could also have a Poisson process for the number of trees in an acre (events per area).

Discrete System)

Poisson Process

Poisson processes ,Here are some examples:

- At a drive-through pharmacy, the number of cars driving up to the drop off window in some interval of time.
- The number of hot dogs sold by Papaya King from 12pm to 4pm on Sundays.
- Failures of ultrasound machines in a hospital.
- The number of vehicles passing through some intersection
  from 8am to 11am on weekdays.
- Number of electrical pulses generated by a photo-detector that is exposed to a beam of photons, in 1 minute.

Unit-02 (Simulation of continuos and Discrete System)

Non-Stationary Poisson Process

- Assuming that a Poisson process has a fixed and constantrate λ over all time limits its applicability. (This is known as a time-stationary or time-homogenous Poisson process, or just simply a stationary Poisson process.)
- We've been looking at Poisson processes with a stationary arrival rate λ ○ In other words, λ doesn't change over time
- Today: what happens when the arrival rate is nonstationary, i.e. the arrival rate λ(τ) a function of time τ?
  - It turns out that a stationary Poisson process with arrival rate 1 can be transformed into a nonstationary Poisson process with any time-dependent arrival rate

Non-Stationary Poisson Process

- For example, during rush hours, the arrivals/departures of vehicles into/out of Manhattan is at a higher rate than at (say) 2:00AM.
- To accommodate this, we can allow the rate $\lambda = \lambda(t)$ to be a deterministic function of time t.
- For example,
  - consider time in hours and suppose $\lambda(t) = 100$ per hour except during the time interval (morning rush hour) (8, 9)
  - when $\lambda(t) = 200$,
  - that is $\lambda(t) = 200, t \in (8, 9), \lambda(t) = 100, t \in (8, 9)$

Discrete System)

In such a case, for a given rate function $\lambda(s)$, the expected number of arrivals by time t is thus given by

$$m(t) \stackrel{\text{def}}{=} E(N(t)) = \int_0^t \lambda(s)ds. \tag{1}$$

For a compound such process such as buses arriving: If independently each bus holds a random number of passengers (generically denoted by B) with some probability mass function $P(k) = P(B = k)$, $k \geq 0$, and mean $E(B)$. Letting $B_1, B_2, \ldots$ denote the iid sequential bus sizes, the number of passengers to arrive by time t, $X(t)$ is given by

$$X(t) = \sum_{n=1}^{N(t)} B_n, \tag{2}$$

where $N(t)$ is the counting process for the non-stationary Poisson process; $N(t)$ = the number of buses to arrive by time t. This is known as a *compound* or *batch* non-stationary Poisson arrival process. We have $E(X(t)) = E(N(t))E(B) = m(t)E(B)$.

We have already learned how to simulate a stationary Poisson process up to any desired time t, and next we will learn how to do so for a non-stationary Poisson process.

1.1 The non-stationary case: Thinning

In general the function $\lambda(t)$ is called the *intensity* of the Poisson process, and the following holds:

For each $t > 0$, the counting random variable $N(t)$ is Poisson distributed with mean

$$m(t) = \int_0^t \lambda(s)ds.$$

$$\begin{aligned} E(N(t)) &= m(t) \\ P(N(t) = k) &= e^{-m(t)}\frac{m(t)^k}{k!}, \ k \geq 0. \end{aligned}$$

More generally, the increment $N(t+h) - N(t)$ has a Poisson distribution with mean $m(t+h) - m(t) = \int_t^{t+h} \lambda(s)ds$.

2 An example

- Suppose we are conducting a time study of a helicopter maintenance facility
- Our data indicates that the facility is busier in the morning than in the afternoon:
  - In the morning (0900 – 1300): expected interarrival time of 0.5 hours
  - In the afternoon (1300 – 1700): expected interarrival time of 2 hours
- Let's say that $\tau = 0$ corresponds to 0900
- Therefore, the arrival rate $\lambda(\tau)$ as a function of τ (in hours) is:

$$\lambda(\tau) = \begin{cases} 2 & \text{if } 0 \le \tau < 4 \\ \frac{1}{2} & \text{if } 4 \le \tau < 8 \end{cases}$$

- Using this, we can compute the **integrated-rate function** $\Lambda(\tau)$, or the expected number of arrivals by time τ:

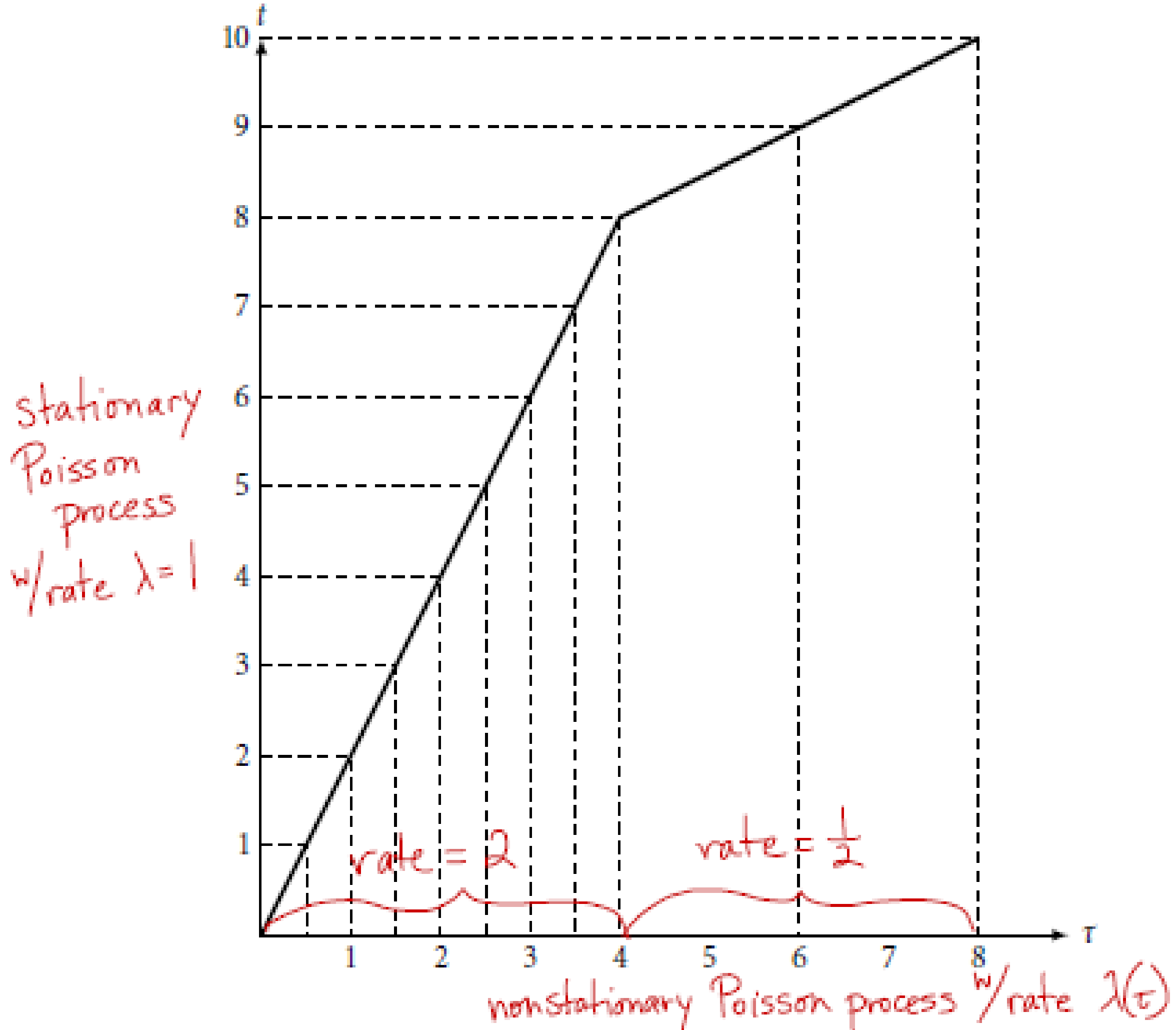
$$\Lambda(\tau) = \int_0^\tau \lambda(a)\, da$$

If $\tau \in [0,4)$: $\Lambda(\tau) = \int_0^\tau 2\, da = 2\tau$

If $\tau \in [4,8)$: $\Lambda(\tau) = \int_0^4 2\, da + \int_4^\tau \frac{1}{2} da = 8 + \frac{1}{2}(\tau - 4) = \frac{1}{2}\tau + 6$

$$\Rightarrow \Lambda(\tau) = \begin{cases} 2\tau & \text{if } 0 \le \tau < 4 \\ \frac{1}{2}\tau + 6 & \text{if } 4 \le \tau < 8 \end{cases}$$

- A graph of the integrated-rate function $\Lambda(\tau)$:



- The inverse of the integrated-rate function $\Lambda(\tau)$:

Let $t = \Lambda(\tau)$

If $t \in [0,8)$: $t = 2\tau \Rightarrow \tau = \frac{1}{2}t$

If $t \in [8,10)$: $t = \frac{1}{2}\tau + 6 \Rightarrow \tau = 2(t-6)$

$$\Rightarrow \Lambda^{-1}(t) = \begin{cases} \frac{1}{2}t & \text{if } 0 \le t < 8 \\ 2(t-6) & \text{if } 8 \le t < 10 \end{cases}$$

- Key idea: τ and t represent <u>different time scales</u> connected by $t = \Lambda(\tau)$
  - t represents time scale for stationary Poisson process with arrival rate 1
  - τ represents time scale of nonstationary Poisson process
- Why does this work? Intuitively, can be seen from the graph above

3 Nonstationary Poisson processes, formally

- Let $\{Y_t; t \geq 0\}$ be a Poisson process (in particular, its output process) with arrival rate 1 and arrival times $\{T_n; n = 0, 1, 2, \ldots\}$
- Define a new arrival counting process with output process $\{Z_\tau; \tau \geq 0\}$ and arrival times $\{U_n; n = 0, 1, 2, \ldots\}$, where $U_n = \Lambda^{-1}(T_n)$
- The process $\{Z_\tau; \tau \geq 0\}$ is (the output process of) a **nonstationary Poisson process** with integrated-rate function $\Lambda(\tau)$
- A nonstationary Poisson process $\{Z_\tau; \tau \geq 0\}$ has the property:

$$\Pr\{Z_{\tau+\Delta\tau} - Z_\tau = m \mid Z_\tau = k\} = \Pr\{Z_{\tau+\Delta\tau} - Z_\tau = m\} = \frac{e^{-[\Lambda(\tau+\Delta\tau) - \Lambda(\tau)]}[\Lambda(\tau+\Delta\tau) - \Lambda(\tau)]^m}{m!}$$

← Poisson w/ parameter $\Lambda(\tau+\Delta\tau) - \Lambda(\tau)$

  - As a consequence, the expected number of arrivals in $(\tau, \tau + \Delta\tau]$ is:

$$E[Z_{\tau+\Delta\tau} - Z_\tau] = \Lambda(\tau+\Delta\tau) - \Lambda(\tau)$$

- In particular, a nonstationary Poisson process satisfies the independent-increments property
- The probability distribution of the number of arrivals in $(\tau, \tau + \Delta\tau]$ depends on both $\Delta\tau$ and τ

  $\Rightarrow$ The stationary-increments and memoryless properties no longer apply

Example 1. In the maintenance facility example above:

a. What is the probability that 2 helicopters arrive between 1200 and 1400, given that 5 arrived between 0900 and 1200?

b. What is the expected number of helicopters to arrive between 1200 and 1400?

a. $\Pr\{Z_5 - Z_3 = 2 \mid Z_3 = 5\} = \Pr\{Z_5 - Z_3 = 2\} = \dfrac{e^{-5/2}\left(\frac{5}{2}\right)^2}{2!} \approx 0.26$

Poisson parameter: $\Lambda(5) - \Lambda(3) = \frac{17}{2} - 6 = \frac{5}{2}$

b. $E[Z_5 - Z_3] = \Lambda(5) - \Lambda(3) = \frac{5}{2}$

Example 2. Think back to the Darker Image case. Suppose the copy shop is open from 0900 ($\tau = 0$) to 1500 ($\tau = 360$), and the arrival-rate function is

$$\lambda(\tau) = \begin{cases} 1/6 & \text{if } 0 \le \tau < 180, \\ 1/5 & \text{if } 180 \le \tau < 360 \end{cases}$$

a. What is the expected number of customers by time τ?
b. What is the probability that 5 customers arrive between 1100 and 1300?
c. What is the expected number of customers that arrive between 1100 and 1300?
d. If 15 customers have arrived by 1100, what is the probability that more than 60 customers will have arrived throughout the course of the day?

a. $\Lambda(\tau) = \int_0^{\tau} \lambda(a)\,da$

If $\tau \in [0,180)$: $\Lambda(\tau) = \int_0^{\tau} \frac{1}{6}\,da = \frac{\tau}{6}$

If $\tau \in [180,360)$: $\Lambda(\tau) = \int_0^{180} \frac{1}{6}\,da + \int_{180}^{\tau} \frac{1}{5}\,da$

$= 30 + \frac{1}{5}(\tau - 180)$

$= \frac{1}{5}\tau - 6$

$$\Lambda(\tau) = \begin{cases} \frac{\tau}{6} & \text{if } 0 \le \tau < 180 \\ \frac{1}{5}\tau - 6 & \text{if } 180 \le \tau < 360 \end{cases}$$

b. $\Pr\{Y_{240} - Y_{120} = 5\} = \frac{e^{-22}(22)^5}{5!} \approx 0.000012$

Poisson w/ parameter

$\Lambda(240) - \Lambda(120) = 42 - 20 = 22$

c. $E[Y_{240} - Y_{120}] = \Lambda(240) - \Lambda(120)$

$= 22$

d. $\Pr\{Y_{360} > 60 \mid Y_{120} = 15\} = \Pr\{Y_{360} - Y_{120} > 45 \mid Y_{120} = 15\} = \Pr\{Y_{360} - Y_{120} > 45\}$

$= 1 - \Pr\{Y_{360} - Y_{120} \le 45\} = 1 - \sum_{j=0}^{45} \frac{e^{-46}(46)^j}{j!} \approx 0.52$

Poisson w/ param.

$\Lambda(360) - \Lambda(120) = 66 - 20 = 46$

Monte Carlo

- Monte Carlo simulations are used to model the probability of different outcomes in a process that cannot easily be predicted due to the intervention of random variables. It is a technique used to understand the impact of risk and uncertainty in prediction and forecasting models.
- Monte Carlo simulation can be used to tackle a range of problems in virtually every field such as finance, engineering, supply chain, and science.
- Monte Carlo simulation is also referred to as multiple probability simulation.

Monte Carlo: Estimating the value of Pi using Monte Carlo

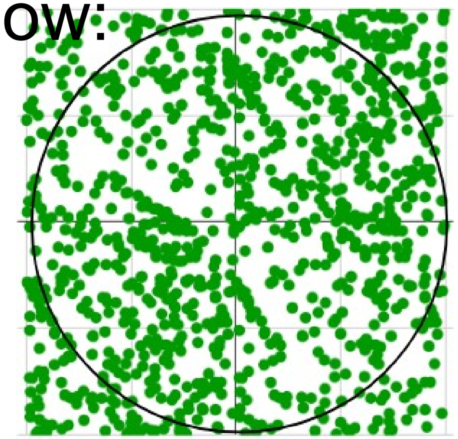
Monte Carlo estimation :

- Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. One of the basic examples of getting started with the Monte Carlo algorithm is the estimation of Pi.

Monte Carlo: Estimating the value of Pi using Monte Carlo

Estimation of Pi

The idea is to simulate random (x, y) points in a 2-D plane with domain as a square of side 1 unit. Imagine a circle inside the same domain with same diameter and inscribed into the square. We then calculate the ratio of number points that lied inside the circle and total number of generated points. Refer to the image below:



Monte Carlo: Estimating the value of Pi using Monte Carlo

Estimation of Pi

We know that area of the square is 1 unit sq while that of circle is $\pi * (\frac{1}{2})^2 = \frac{\pi}{4}$.

Now for a very large number of generated points,

$$\frac{\text{area of the circle}}{\text{area of the square}} = \frac{\text{no. of points generated inside the circle}}{\text{total no. of points generated or no. of points generated inside the square}}$$

that is,

$$\pi = 4 * \frac{\text{no. of points generated inside the circle}}{\text{no. of points generated inside the square}}$$

Monte Carlo: Estimating the value of Pi using Monte Carlo

Estimation of Pi

The beauty of this algorithm is that we don't need any graphics or simulation to display the generated points. We simply generate random (x, y) pairs and then check if $x^2+y^2<=1$.

If yes, we increment the number of points that appears inside the circle. In randomized and simulation algorithms like Monte Carlo, the more the number of iterations, the more accurate the result is.

Thus, the title is " **Estimating** the value of Pi" and not "Calculating the value of Pi". Below is the algorithm for the method:

Monte Carlo: Estimating the value of Pi using Monte Carlo

The Algorithm

1. Initialize circle_points, square_points and interval to 0.
2. Generate random point x.
3. Generate random point y.
4. Calculate d = x*x + y*y.
5. If d <= 1, increment circle_points.
6. Increment square_points.
7. Increment interval.
8. If increment < NO_OF_ITERATIONS, repeat from 2.
9. Calculate pi = 4*(circle_points/square_points).
10. Terminate.

Finished Unit 2