111803099
Chaitanya Bharat Shinge
Division 1: B3 Batch

Distributed Systems

Assignment 1

Prove the equations defining QoS parameters for a centralized system using mathematical modeling.

In this centralized system, there is a single server processing requests waiting in a single queue

Assumptions:

Infinite Queue Length

Notations:

λ = arrival rate of requests to the system

μ = requests served per unit time

1. Fraction of time having k requests in the system.

$$\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k$$

P(k): $\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k$

For k = 0,

$\rho_0 = (1 - \frac{\lambda}{\mu})$

For k = 1,

$\rho_1 = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu}) = \rho_0(\frac{\lambda}{\mu})$

For k = 2,

$\rho_2 = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^2 = \rho_0(\frac{\lambda}{\mu})^2$

For k = 3,

$\rho_3 = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^3 = \rho_0(\frac{\lambda}{\mu})^3$

So, for varying k, ρ_0 never changes and only $(\frac{\lambda}{\mu})^k$ changes. We can write it further as,

$\rho_k = \rho_0(\frac{\lambda}{\mu})^k = \rho_{k-1}(\frac{\lambda}{\mu})\ldots (eq^n 1)$

Base Case: for k = 0, P(0) is true.

$\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k$ => $\rho_0 = (1 - \frac{\lambda}{\mu})$

As $\frac{\lambda}{\mu}$ indicates utilization of service and indicates that the server is busy.

Although, we have assumed infinite queue length, we should keep $\lambda < \mu$, otherwise if $\lambda > \mu$ then the arrival rate will be more than the requests served. So we will be having more requests pending in the queue, and over time we will need to increase queue length, which should be practically inefficient. So, better to keep $\lambda < \mu$.

RHS = $(1 - \frac{\lambda}{\mu})$ => the fraction of time the server is idle.

LHS = ρ_0 => the fraction of time the server served 0 requests i.e. when it was idle.

Thus, LHS = RHS.

P(k) for k = 0 is true.

Next step, For k = n, P(n) is true => P(n+1) is true.

Inductive Hypothesis:

P(n) : $\rho_n = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^n = \rho_0(\frac{\lambda}{\mu})^n$ is true.

P(n+1): $\rho_{n+1} = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^{n+1}$

LHS = $\rho_{n+1} = \rho_n(\frac{\lambda}{\mu}) = \rho_0(\frac{\lambda}{\mu})^{n+1}$, $from\ eq^n 1$

RHS = $(1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^{n+1} = \rho_0(\frac{\lambda}{\mu})^{n+1}$, $from\ eq^n 1$

Thus, LHS = RHS.

P(n+1) is true.

Therefore, by mathematical induction,

P(k): $\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k$

Hence, proved.

2. Utilization of a server is fraction of time that it is busy

$$U = \sum_{k>0} \rho_k \; or \; \rho_k = (1 - U).U^k$$

Utilization indicates the fraction of time, server is busy in serving arrived requests

i.e. 1- probability that server served 0 requests.

$U = (1 - \rho_0) = 1 - (1 - \frac{\lambda}{\mu}) = \frac{\lambda}{\mu}$

It can be formulated in other words as, fraction of time the server serves at least one request i.e.

$U = \sum_{k>0} \rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu}) + (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^2 + (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^3 + \ldots$

$= (1 - \frac{\lambda}{\mu})[(\frac{\lambda}{\mu})^1 + (\frac{\lambda}{\mu})^2 + (\frac{\lambda}{\mu})^3 + \ldots]$ …(Infinite Geometric Progression)

$= (1 - \frac{\lambda}{\mu})[\frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}}] = \frac{\lambda}{\mu}$

Hence, proved.

Also, we can substitute utilization as $U = \frac{\lambda}{\mu}$ in $\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k$, so we get,

$\rho_k = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^k = (1 - U).U^k$. Hence, proved.

3. Average number of requests in the system

$$\overline{N} = \sum_{k\geq 0} k.\rho_k = \frac{U}{1-U}$$

ρ_k represents the fraction of time when k requests are in the system.

Product of the number of requests and the fraction of time (or probability of being in queue during that time frame) they were present in the queue will give expected value / average requests waiting in queue.

$$\overline{N} = \sum_{k\geq 0} k.\rho_k = \sum_{k\geq 0} k((1 - U).U^k) = (1 - U) \sum_{k\geq 0} kU^k$$

$\sum_{k\geq 0} kU^k$ is an arithmetico geometric sequence.

Let, $S = \sum_{k\geq 0} kU^k$

Multiplying both sides by U,

$$U.S = \sum_{k\geq 0} kU^{k+1}$$

Subtracting $U.S\ from\ S$,

$$S = \sum_{k\geq 0} kU^k \quad = 0 + 1.U^1 + 2.U^2 + 3.U^3 + \dots$$

$$- U.S = \sum_{k\geq 0} kU^{k+1} = \quad + 0.U^1 + 1.U^2 + 2.U^3 + 3.U^4 + \dots$$

$$S - U.S = 1.U^1 + 1.U^2 + 1.U^3 + 1.U^4 + \dots = \sum_{k=0}^{\infty} U^k = \frac{U}{1-U};\ (as\ U = \frac{\lambda}{\mu} < 1)$$

$$S.(1 - U) = \sum_{k=0}^{\infty} U^k = \frac{U}{1-U} \Rightarrow S = \frac{U}{(1-U)^2}$$

$$\overline{N} = (1 - U).\sum_{k\geq 0} kU^k = (1 - U).S = \frac{U}{1-U}$$

$$\overline{N} = \frac{U}{1-U}$$

Hence, proved.

4. Average Throughput

$$X = \lambda$$

System Throughput: The total amount of requests processed by the system over the defined period of time. (definition from Google)
Server can be in two states either busy or idle.
When the server is idle, it processes 0 requests, server utilization here is $1 - U$(i.e. the fraction of time frame when the server is idle).
When the server is busy, $\lambda = U.\mu$ requests are processed on an average.
Here, U is utilization of server (i.e. the fraction of time frame when server processes the requests) and μ is the request serving rate.
Overall requests processed by server in unit time frame
= requests processed by server when idle + requests processed by server when busy
= 0 + λ
= λ = Average Throughput
Hence, proved

5. Response Time

$$R = \frac{\overline{N}}{X} \; or \; \frac{R}{S} = \frac{1}{1-U}$$

System Response Time is the duration between the time when the request arrived and when the server started to process it.
More response time less is throughput and more is the average number of requests waiting in the system to be processed.
$R = \frac{\overline{N}}{X} = \frac{(\frac{\lambda}{\mu})}{(1-U).\lambda} = \frac{S}{(1-U)}$; $\overline{N} = \frac{U}{(1-U)}$, $X = \lambda, U = \frac{\lambda}{\mu}, S = \frac{1}{\mu}$;
λ is arrival rate of requests,
μ is request serving rate,
S is request serving time,
$\overline{N}$ is average number of requests waiting in the system to be processed,
U is utilization of server (fraction of time when server processed requests),
R is response time on average of a request arrived in the system.
$\frac{R}{S} = \frac{1}{1-U}$.
Hence, proved.