Introduction to Queuing Theory: Applications to Networks

Guevara Noubir

Textbook: D. Bertsekas, R. Gallagher, "Data Networks", Prentice-Hall.

Queuing Theory

- Study of the performance of systems composed of
  - Waiting lines
  - Processing units
- Allows to estimate
  - Time spent waiting
  - Expected number of waiting requests
  - Probability of encountering some states
- Useful for the design systems such as networks
  - Delay, blocking probability, links bandwidth, number of processors, buffers size

Examples

- Sliding window ARQ mechanism performance
  - Expected delay of a packet
- Medium access control protocol
  - E.g., IEEE 802.11
- Traffic/Packets multiplexing
  - Average delay when multiple links are grouped
  - Average queue size
- Cellular networks
  - Blocking probability
  - Dropping probability
- Webserver

Outline

- Delay Models
- Little's Theorem
- The $M/M/1$ queuing system
- The $M/G/1$ queuing system
- Other queuing systems

Delay Models

- *Delay* (or *latency*) of data packet is an important measure of the performance of a network

Delay	= *PropagationDelay* + *TransmissionDelay* + *QueuingDelay*
PropagationDelay	= *Distance/SpeedOfLight (independent of message size)*
TransmissionDelay	= *MessageSize/Bandwidth* (Bandwidth = data-rate here)
QueuingDelay	= delay due to time spent waiting in queues (**most important delay**)

- The queuing delay depends on several parameters:
  - Arrival process
  - Service discipline
  - Processing delay
  - Others: bandwidth of the link, buffer size

Queuing Theory Framework

- Queuing system:
  - Servers (one or several): e.g., router, computer processor, webserver with back-end processes
  - Customers: e.g., users, packets, web requests
  - Queues: customers wait in queues before getting services

`

Little Theorem (1961)

- Measurement quantities of interest:
  - T: average delay incurred by a customer
  - N: average number of customers in the system
- Little's Theorem:

  $N = \lambda T$ where λ is the rate of the arrival process
- Little's Theorem provides a general and fundamental relation between N, T, and λ. It is independent of the nature of the arrival process or of the service time distribution.

Proof of Little's Theorem

- Notation:
  - $\alpha(t)$: number of users that arrived before time t
  - $\beta(t)$: number of users that departed before time t
  - T_i time spent by user i within the system
  - $N(t)$ number of users in the system at time t
- Arrival rate: $\lambda(t) = \frac{\alpha(t)}{t}$
  - λ is the limit
- Average time within the system: $T(t) = \frac{1}{\alpha(t)} \sum_{i=0}^{\alpha(t)} T_i$
- Average number of users at time t:

$$N_t = \frac{1}{t} \int_0^t N(\tau) d\tau$$

Proof (Cont'd)

- The usage of the system can be bounded:

$$\sum_{i=0}^{\beta(t)} T_i \leq \int_0^t N(\tau)d\tau \leq \sum_{i=0}^{\alpha(t)} T_i$$

$$\frac{\beta(t)}{t}\frac{\sum_{i=0}^{\beta(t)} T_i}{\beta(t)} \leq \frac{1}{t}\int_0^t N(\tau)d\tau \leq \frac{\alpha(t)}{t}\frac{\sum_{i=0}^{\alpha(t)} T_i}{\alpha(t)}$$

- Taking the limit when t -> +∞

$$\lambda T \leq N \leq \lambda T$$

Application: Flow Control

- Sliding window flow control
  - e.g., Go-Back-N or Selective Repeat with window size: *W*

- The number of packets in the system is always less than *W*:
  $\lambda T = N \leq W$

- Conclusion:
  - for a given window size, if *T* increases, then the arrival rate has to be decreased
  - for a given arrival rate, if *T* increases, then the window size has to be increased
  - for a given *T*, if the arrival rate increases, then the window size has to be increased

The *M*/*M*/1 queuing system

- Notation:
  - Arrival Process/Departure Process/Number of servers
- Little's Theorem is a general tool that allows us to calculate the steady-state average delay of a queuing system
- Examples:
  - *M*: memoryless, *G*: general, *D*: deterministic, 1: number of servers in the system
- *M*/*M*/1:
  - Arrival rate is Poisson distributed
  - Service time is exponentially distributed
  - These two processes are independent

Poisson Process

- A Poisson process with arrival rate λ:
  - The probability distribution function (pdf):

$$\Pr(n \text{ arrivals in interval } [\text{t}, \text{t}+\tau]) = \frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!}$$

  - The arrival distribution of two disjoint intervals is independent

- Properties:
  - expected number of arrivals in a length-τ interval is: $\lambda\tau$.

Poisson Process (Cont'd)

- Probabilities for small intervals:
  - Pr(0 arrival) = $e^{-\lambda\delta}/0! = 1 - \lambda\delta + o(\delta)$
  - Pr(1 arrival) = $\lambda\delta\, e^{-\lambda\delta}/1! = \lambda\delta + o(\delta)$
  - Pr(2 arrivals) = $(\lambda\delta)^2\, e^{-\lambda\delta}/2! = o(\delta)$

  If δ tends to 0, then we have Pr(0 arrivals) = $1 - \lambda\delta$, *and* Pr(1 arrival) = $\lambda\delta$.

- Inter-arrival times:
  - Let t_n be the arrival time of the n^{th} customer and $\tau_n = t_{n+1} - t_n$
  - Then: $\Pr(\tau_n > s) = e^{-\lambda s}$ (=> exponential distribution)

Other Properties

- Poisson processes are used to model the traffic of a large number of similar and independent users
- If n independently and identically distributed packet arrival processes (rate λ/n) occur at the head of a link then the aggregated process can be shown to be well approximated by a Poisson process of rate λ. n is considered to be a large value.

- The aggregation of k independent Poisson processes of rates $\lambda_1, \lambda_2, \ldots, \lambda_k$ yields a Poisson process of rate: $\lambda_1 + \lambda_2 + \ldots + \lambda_k$

Exponential Service Time

- Let s_n denote the service time for the n^{th} customer. The service time distribution is exponential with parameter μ if:

  $\Pr[s_n \leq s] = 1 - e^{-\mu s}$

- The expected service time for a job is: $1/\mu$

- The exponential service time is memoryless in the sense that:

  $\Pr(s_n > r + t \mid s_n > t) = \Pr(s_n > r)$

- Poisson processes are closely related to exponential distributions: inter-arrival times of a Poisson process with rate λ have an exponential distribution with parameter λ.

Analysis of the *M*/*M*/1 Queuing System

- The state of the system is captured by the number of customers in the system at time t
- We consider a discrete version of the process evolution:
  - Time: 0, δ, 2δ, 3δ, …$k\delta$, …
  - N_k: number of customers at time $k\delta$,
- Properties:
  - $\Pr[N_{k+1} = l \mid N_k = l] = \sum_{i \geq 0} \Pr[i$ arrivals and i departures in δ interval$]$
  - $\Pr[N_{k+1} = 0 \mid N_k = 0] \approx 1 - \lambda\delta + (\lambda\delta)(\mu\delta) \approx 1 - \lambda\delta$

$M/M/1$ Analysis (Cont'd)

- Let: $P_{i,j} = \Pr[N_{k+1} = j \mid N_k = i]$
- $P_{0,0} = 1 - \lambda\delta + o(\delta) \approx 1 - \lambda\delta$
- $P_{i,i} = 1 - \lambda\delta - \mu\delta + o(\delta) \approx 1 - \lambda\delta - \mu\delta$ (for $i \geq 1$)
- $P_{i,i+1} = \lambda\delta + o(\delta) \approx \lambda\delta$ (for $i \geq 0$)
- $P_{i,i-1} = \mu\delta + o(\delta) \approx \mu\delta$ (for $i \geq 1$)
- $P_{i,j} = o(\delta) \approx o(\delta)$ (for $j \neq i, i+1, i-1$)
- The state transitions represent a Markov chain

Stationary Distribution of a System

- After a long period of time the system reaches a *steady state*
- Let: $p_i = \lim_{k \to +\infty} \Pr[N_k = i]$

- From the Markov chain diagram we have:
  - $p_i = p_{i-1}(\lambda\delta) + p_i(1- \lambda\delta - \mu\delta) + p_{i+1}(\mu\delta)$
  - Hence: $(p_i – p_{i+1}) = \rho(p_{i-1} - p_i)$, where $\rho = \lambda/\mu$
  - Let $\Delta_i = p_i – p_{i+1}$, then $\Delta_i = \rho\Delta_{i-1}$ (for i>0)
  - We also have: $\Delta_0 = (1-\rho)\, p_0$

Stationary Distribution of a System

$$p_i = p_0 - \sum_{j=0}^{i-1} \Delta_j$$

$$p_i = p_0 - (1-\rho)p_0 \frac{1-\rho^i}{1-\rho}$$

$$p_i = \rho^i p_0$$

- Since: $\sum_{i\geq 0} p_i = 1$, then $p_0 = 1-\rho$, and $p_i = \rho^i(1-\rho)$

Steady State Averages

- Steady state average number of customers:

$$\sum_{i=0}^{\infty} ip_i = \sum_{i=0}^{\infty} i\rho^i(1-\rho) = \frac{\rho}{1-\rho}$$

- Average delay *T* (using Little's Th.):

$$T = \frac{1}{\mu - \lambda}$$

- Average waiting time *W*:
  (delay-service time)

$$W = \frac{\lambda}{\mu(\mu - \lambda)}$$

Applications

- Scaling up the arrival rate and service rate
  - If we increase the arrival and service rates by the same factor then average number of customers in the system stays the same, while the average delay goes down
- Multiplexing several connections on one link
  - Benefit of statistical multiplexing

App1: Network Switch

- Consider a terminal concentrator:
  - 4 input lines, each line of 64 Mbps
  - 1 output line of 128 Mbps
  - Mean packet size is 12800 bits
  - Each of the four input lines delivers Poisson traffic with λ_i = 2,000 pkts/s
- Mean delay of a packet within the concentrator:
  - λ = 8,000 pkts/s, μ = 10,000 pkts/s, T = $1/(\mu - \lambda)$ = 500 us
- Average number of packets within the concentrator:
  - N = $\rho/(1-\rho)$ = 4

App1: Network Switch (Cont'd)

- Remarks:
  - The output line is capable of handling the generated traffic (128Mbps > 12800 * 8000), but a substantial input queue builds up.
  - The reason is the randomness of the arrivals
- Usefulness of modeling and analysis:
  - Delay estimation
  - Buffers dimensioning

App 2: Statistical Multiplexing vs. Dedicated Channels

- Let a system consist of:
  - Two computers connected using a 64Mbps line
  - 8 parallel sessions
  - Each session generates Poisson traffic with λ_i = 2000 pkts/s
  - Packets length is exponentially distributed with mean 2000 bits.
- Two possible strategies:
  - Give each session a dedicated portion of the channel (e.g. TDM or FDM)
  - Have all the packets compete for the shared channel

App 2: Statistical Multiplexing vs. Dedicated Channels (Cont'd)

- Dedicated channels (8*8Mbps):
  - λ = 2000 pkts/s, μ = 4000 pkts/s
  - T = $1/(\mu - \lambda)$ = 500 us
- Statistical multiplexing:
  - λ = 16000 pkts/s, μ = 32000 pkts/s
  - T = $1/(\mu - \lambda)$ = 62.5 us
- Explanation: because of the randomness of the arrival rate, some of the dedicated channel may be unused (because the corresponding session is idle) while packets are queued for other sessions

The $M/G/1$ System

- $M/G/1$ system:
  - Arrival rate is Poisson
  - Service time has a general distribution
- It is not possible to derive a closed-form stationary distribution (as in $M/M/1$) but we can derive other results
- Assume that:
  - Customers are served on a FCFS basis
  - X_i (service time of i^{th} arrival) identically distributed, mutually independent, and independent of the inter-arrival times

P-K Formula

- Average service time: $\overline{X} = E\{X\} = \dfrac{1}{\mu}$ $\overline{X^2} = E\{X^2\}$
- Second moment of service time:
- *Pollaczek-Khinchin* (P-K) formula: $W = \dfrac{\lambda \overline{X^2}}{2(1-\rho)}$

- Then: $T = \overline{X} + \dfrac{\lambda \overline{X^2}}{2(1-\rho)}$
- Using Little's Theorem: $N_Q = \dfrac{\lambda^2 \overline{X^2}}{2(1-\rho)}; N = \rho + \dfrac{\lambda^2 \overline{X^2}}{2(1-\rho)}$

Verification of P-K Formula for Exponentially Distributed Service Time

- When service times are exponentially distributed as in the *M*/*M*/1 system:

$$\overline{X} = 1/\mu; \overline{X^2} = 2/\mu^2$$

$$W = \frac{\rho}{\mu(1-\rho)}; T = \frac{1}{\mu - \lambda}$$

- When the service time is identical for all customers: *M*/*D*/1:

$$\overline{X} = 1/\mu; \overline{X^2} = 1/\mu^2$$

$$W = \frac{\rho}{2\mu(1-\rho)}$$

M/*D*/1 provides lower bounds for W, T, N_Q, and N

Proof of the P-K Formula

- We use the concept of *mean residual service time*
- Notation:
  - W_i: waiting time of customer i
  - R_i: residual time to completion of the current customer at instant when i arrives (R_i=0, if no customer is being serviced)
  - Q_i: number of customers waiting in queue when i arrives
- Since customers are <u>serviced in order</u>, we have:

$$W_i = R_i + \sum_{j=i-Qi}^{i-1} X_j$$

$$W = R + N_Q \overline{X}$$

Proof of the P-K Formula (Cont'd)

- From Little's Theorem: $N_Q = W\lambda$, then: $W = R/(1-\rho)$

$$R = \frac{1}{t}\int_0^t R(\tau)d\tau$$

$$R = \frac{1}{t}\sum_{i=1}^{\beta(t)} \frac{1}{2} X_i^{\,2}$$

$$R = \frac{\lambda \overline{X^2}}{2}$$

- Thus the P-K formula

Why Poisson Assumption?

- Where did we use the Poisson Process arrivals assumption?
  - At the moment when a packet arrives the queue is typical
    - lim $P\{N(t) = n \mid \text{an arrival occurred just after } t\} = \lim \{N(t) = n\}$
    - Section 3.3.2
  - If arrival not Poisson:
    - Inter-arrival: uniformly distributed between 2 and 4 seconds
    - Customer service time is: 1 second
    - => An arriving customer finds the queue empty
    - => but an external customer sees a average queue length of 1/3

Unstable *M*/*G*/1 Systems

- For several probability distribution functions the second moment is finite (proportional to the square of the mean): e.g., exponential, constant, uniform. However, it is not general to all distributions.

- Let X be the random variable representing the service time for a customer s.t.:
  - $\Pr[X=1] = 2/3$; $\Pr[X=2^i]=1/4^i$ (for $i>0$)
  - The mean of X is finite, but the second moment is infinite
  - In this kind of systems we may have an accumulations of arrivals that exceeds the service capability

Applications of $M/G/1$: GBN ARQ

- Simplified analysis of Go-Back-n ARQ:
  - No-modulus, all acknowledgements are received
  - If the lowest number in the window is not ACKed by the end of the window the sender assumes that the error occurred and starts retransmitting
  - Errors are independent from one to another
  - All frames take a unit of time to be transmitted

- The service time distribution is:
  - $\Pr[X=1+ni] = p^i(1-p)$ $(i \geq 0)$

Applications of $M/G/1$: GBN ARQ

- If the packets are generated at the sender by a Poisson process, then we have an $M/G/1$ system:

$$\overline{X} = 1 + \frac{np}{1-p}$$

$$\overline{X^2} = 1 + \frac{2np}{1-p} + \frac{n^2(p+p^2)}{(1-p)^2}$$

$$W = \frac{\lambda \overline{X^2}}{2(1-\lambda \overline{X})}$$

$$T = \overline{X} + W$$

Formulas good to know:

$$\sum_{k=0}^{\infty} p^k = \frac{1}{1-p}, \sum_{k=0}^{\infty} kp^k = \frac{p}{(1-p)^2}, \sum_{k=0}^{\infty} k^2 p^k = \frac{p+p^2}{(1-p)^3}$$

$M/G/1$ with Priorities

- System with priority:
  - Customers are divided into classes: 1 … k
  - Customers in class i are given priority over customers of class j (for any $j>i$)
  - Non-preemptive
  - Customers are served in their order of arrival
- Notation:
  - Arrival process for class i: Poisson with rate λ_i
  - Service time of customers of class i: X_i
  - W_i average waiting time for a customer in class I
  - R average residual time
  - Q_i average number of customers of class i waiting in queue

$M/G/1$ with Priorities (Cont'd)

Class 1

$$W_1 = R + Q_1 \overline{X_1}$$

$$Q_1 = W_1 \lambda_1$$

$$\Rightarrow W_1 = \frac{R}{1 - \rho_1}$$

Class 2

$$W_2 = R + Q_1 \overline{X_1} + Q_2 \overline{X_2} + \lambda_1 W_2 \overline{X_1}$$

$$Q_i = W_i \lambda_i$$

$$\Rightarrow W_2 = \frac{R}{(1 - \rho_1)(1 - \rho_1 - \rho_2)}$$

Class i

$$W_i = \frac{R}{(1 - \rho_1 - \ldots - \rho_{k-1})(1 - \rho_1 - \ldots - \rho_k)}$$

$M/G/1$ with Priorities (Cont'd)

- As for the P-K formula: $R = \frac{1}{2}\sum_{i=1}^{k}\lambda_i \overline{X_i^2}$

- Thus the average waiting time for a customer in class *i:*

$$W_i = \begin{cases} \dfrac{\sum_{i=1}^{k}\lambda_i \overline{X_i^2}}{2(1-\rho_1)} & \text{if } i = 1 \\ \dfrac{\sum_{i=1}^{k}\lambda_i \overline{X_i^2}}{2(1-\rho_1-\ldots-\rho_{i-1})(1-\rho_1-\ldots-\rho_i)} & \text{if } i > 1 \end{cases}$$

$M/M/m$ Markov System

- m servers
- Steady state probabilities:

$$\rho = \frac{\lambda}{m\mu} < 1$$

$$p_n = \begin{cases} p_0 \dfrac{(m\rho)^n}{n!} & n \le m \\ p_0 \dfrac{m^m \rho^n}{m!} & n > m \end{cases}$$

$$p_0 = \left[\sum_{n=0}^{m-1} \frac{(m\rho)^n}{n!} + \frac{(m\rho)^m}{m!(1-\rho)} \right]^{-1}$$

Erlang C Formula: probability of having to wait for service

$$P\{Queuing\} = P_Q = \frac{p_0 (m\rho)^m}{m!(1-\rho)} \qquad W = \frac{\rho P_Q}{\lambda(1-\rho)}$$

$M/M/m/m$ Markov System

- m servers, no queuing
- Steady state probabilities:

$$p_n = p_0 (\frac{\lambda}{\mu})^n \frac{1}{n!}; p_0 = \left[\sum_{n=0}^{m} \left(\frac{\lambda}{\mu} \right)^n \right]^{-1}$$

Blocking probability: Erlang B formula

$$p_m = \frac{(\lambda/\mu)^m / m!}{\sum_{n=0}^{m} (\lambda/\mu)^n / n!}$$

Application: Throughput in a Time Sharing System

- Assumptions:
  - *N* terminals, one processor, one queue
  - Terminals are always occupied
  - System activity: users log-on, reflection (*R* on average), submit task to the processor, tasks are queued, tasks execution takes on average *P* units of time
- The delay for a user task is on average *T* s.t.:

$$R+P \le T \le R + NP$$

Using Little's Theorem: $\frac{N}{R+NP} \le \lambda \le \frac{N}{R+P}$

Time Sharing (Cont'd)

- The processor is also a queuing system where $N \leq 1$
- In the steady state mode: the arrival rate in the system is the same as for the processor
- Using Little's Theorem a second time: $\lambda P \leq 1$

- Combining these two bounds we get:

$$\lambda \leq \min\{\frac{1}{P}, \frac{N}{R+P}\} = \frac{1}{P}\min\{1, \frac{N}{1+R/P}\}$$

- The smalled term indicates the bottleneck

App 1: Blocking Probability

- Consider a queuing system with:
  - K servers
  - $N \geq K$ *in system* customers (in service + waiting)
  - Departing customers are immediately replaced by new customers
  - $\overline{X}$ is the average customer service time
- Average customer time in the system T ?
  - $T = N/\lambda$ and $K = \lambda \overline{X}$
  - Thus: $T = N\overline{X} / K$

App 1: Blocking Probability (Cont'd)

- Assume that customers are blocked (and lost) if the system is full:
  - β is the proportion of customers that are blocked
  - The system may go through moments where less than *K* servers are active
  - Then: $$\overline{K} = (1-\beta)\lambda\overline{X}$$

$$\beta = 1 - \frac{\overline{K}}{\lambda\overline{X}} \geq 1 - \frac{K}{\lambda\overline{X}}$$