PRIORITY QUEUEING

Applies to $M/G/1$ systems in which all arriving customers are divided in $K$ classes. Class $i$ is served before class $i+1$, i.e. class $i$ has higher priority. We analyze two methods for assigning priorities:

NONPREEMPTIVE PRIORITY

In this case, a customer with a lower priority who is in service when a customer with a higher priority arrives, finishes service first and then the higher priority customer is served.

We can derive an expression for $W_k$ - average waiting time for a class $k$ customer. Let

$N^k_Q = \text{Average # of waiting customers of class } k$

$\rho^k = \frac{\lambda^k}{\mu^k}$

$R = \text{average residual service time}$

First, consider the class with the highest priority.

Class 1:

$$W_1 = R + N^1_Q \cdot \frac{1}{\mu_1} \quad (1)$$

Applying Little’s theorem $N^1_Q = \lambda_1 W_1$

$$\Rightarrow W_1 = R + \rho_1 W_1 \quad (2)$$

$$\Rightarrow W_1 = \frac{R}{1 - \rho_1} \quad (3)$$

Class 2: Here, we have to include the service times for users from a higher class (Class 1) that arrived while the customer from Class 2 was waiting. There will be $\lambda_1 W_2$ such users, each with service time $\frac{1}{\mu_2}$

$$W_2 = R + N^1_Q \cdot \frac{1}{\mu_1} + N^2_Q \cdot \frac{1}{\mu_2} + \lambda_1 W_2 \cdot \frac{1}{\mu_1} \quad (4)$$

Again, applying Little’s Theorem we get

$$\Rightarrow W_2 = R + \rho_1 W_1 + \rho_2 W_2 + \rho_1 W_2 \quad (5)$$

$$\Rightarrow W_2 = \frac{R + \rho_1 W_1}{1 - \rho_1 - \rho_2} \quad (6)$$

Substituting equation 3, we obtain

$$W_2 = \frac{R}{(1 - \rho_1)(1 - \rho_1 - \rho_2)} \quad (7)$$

Class 3:

$$W_3 = R + N^1_Q \cdot \frac{1}{\mu_1} + N^2_Q \cdot \frac{1}{\mu_2} + N^3_Q \cdot \frac{1}{\mu_3} + \lambda_1 W_3 \cdot \frac{1}{\mu_1} + \lambda_2 W_3 \cdot \frac{1}{\mu_2} \quad (8)$$
where
\( \lambda_1 W_3 = \# \) of class 1 arrivals while waiting
\( \lambda_2 W_3 = \# \) of class 2 arrivals while waiting

Applying the same procedure as in the previous two cases, we get

\[
W_3 = \frac{R}{(1 - \rho_1 - \rho_2)(1 - \rho_1 - \rho_2 - \rho_3)}
\]

(9)

In general, the waiting time for class \( k \)

\[
W_k = \frac{R}{(1 - \rho_1 - \cdots - \rho_{k-1})(1 - \rho_1 - \cdots - \rho_k)}
\]

(10)

We now have to determine \( R \).
The residual service time, as in the case of regular \( M/G/1 \), is shown in the following graph:

Figure 1: Residual service time.

Where each service time could belong to a customer of any class. For the regular \( M/G/1 \) system of arrival rate \( \lambda \), the mean residual time is

\[
R = \frac{1}{2} \lambda \overline{X}^2
\]

(11)

Residual service time in a nonpreemptive priority queueing system is the same as in \( M/G/1 \) system with \( \lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_K \). Only the order of service will change which amounts to rearranging the order of the triangles in the graph, but the effective area still remains the same. This is true only for \( t \to \infty \) since all users from all \( k \) classes will eventually be served.

\[
R = \frac{1}{2} \lambda \overline{X}^2
\]

(12)

where
\( \lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_K \)

\[
\overline{X}^2 = \frac{\sum_{M(t)} \overline{X}_j^2}{M(t)} \kappa \text{ sum of all service times regardless of the class}
\]

\( M(t) \) = # of service completions in \([0,t]\) over all \( k \) classes

\[
R = \frac{1}{2} \lambda E[X^2] = \frac{1}{2} \lambda \sum_{j=1}^{K} P(A_j) E[X_j^2 | A_j] = \frac{1}{2} \lambda \sum_{j=1}^{K} \frac{\lambda_j}{\lambda} \overline{X}_j^2 = \frac{1}{2} \sum_{j=1}^{K} \lambda_j \overline{X}_j^2
\]

(13)

where
\( X_j \) = mean service time of class \( j \)
\( A_j \) = arrival of a class \( j \) customer

\[
W_k = \frac{\sum_{j=1}^{K} \lambda_j \overline{X}_j^2}{2(1 - \rho_1 - \cdots - \rho_{k-1})(1 - \rho_1 - \cdots - \rho_k)}
\]

(14)
The average delay is

\[ T_k = W_k + \frac{1}{\mu_k} \tag{15} \]

**PREEMPTIVE RESUME PRIORITY**

In this case a higher priority new arrival immediately starts service. After it is serviced, the low priority customer resumes service. Hence, for each class \( k \) \((k > 1)\) we can find \( T_k \) as if it was the lowest priority class in the system.

\[
T_k = \frac{1}{\mu_k} + \frac{R_k}{1 - (\rho_1 + \cdots + \rho_k)} + \frac{\lambda_1 T_k}{\mu_1} + \cdots + \frac{\lambda_{k-1} T_k}{\mu_{k-1}} \tag{16}
\]

where

- \( A \) = service time
- \( B \) = avg time spent to serve waiting customers of classes \( 1, \ldots, k \).

This is equal to the average waiting time in a \( M/G/1 \) system with \( \lambda = \lambda_1 + \cdots + \lambda_k \). This is because the customer in class \( k \) has to wait for higher priority customers and customers in his own class and hence sees \( \rho = \rho_1 + \cdots + \rho_k \)

- \( C \) = time to serve new arrivals while the customer is in the system

\[
\Rightarrow T_k = \frac{1}{\mu_k} + \frac{R_k}{1 - (\rho_1 + \cdots + \rho_k)} + \left( \sum_{i=1}^{k-1} \rho_i \right) T_k \tag{17}
\]

\[
\Rightarrow T_k = \frac{\left( \frac{1}{\mu_k} (1 - \rho_1 - \cdots - \rho_k) + R_k \right)}{(1 - \rho_1 - \cdots - \rho_{k-1})(1 - \rho_1 - \cdots - \rho_k)} \quad (k > 1) \tag{18}
\]

where \( R_k = \frac{1}{2} \sum_{i=1}^{k} \lambda_i X_i^2 \) \( \tag{19} \)

Class 1 customers see a \( M/G/1 \) queue with \( \lambda_1 \mu_1 \).

**NETWORKS OF TRANSMISSION LINES**

In a data network, a packet travels through several nodes and goes through a queue at each node.

**Problem:** After the packet goes through a queue, the inter-arrival times at subsequent queues become dependent on the packet length.

How do we find average delay per packet in data networks? ⇒

**THE KLEINROCK INDEPENDENCE APPROXIMATION**

Assume: Packet lengths are exponential and there are Poisson arrivals at the entry points of the network.

In the scenario shown in Fig 2 where there are several packet streams merging on a transmission line, we can assume that the independence of interarrival times and packet lengths on that line is restored. The communication link can therefore be modeled as a \( M/M/1 \) queue, enabling us to calculate the average delay per packet on the link.
In case of virtual circuit networks, the arrival rate at link \((i,j)\) can be calculated as
\[
\lambda_{ij} = \sum_{\text{all sessions } s \text{ using link}(i,j)} x_s
\]  
where \(x_s\) = packet rate of session \(s\)

In a more general case, when there are several paths used by one session
\[
\lambda_{ij} = \sum_{\text{all sessions } s \text{ using link}(i,j)} f_{ij}(s)x_s
\]  
where \(f_{ij}(s)\) = the fraction of the packets of session \(s\) going through link \((i,j)\).

Using the \(M/M/1\) model, we can now calculate the average number of packets at link \((i,j)\)
\[
N_{ij} = \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}
\]  
where \(\frac{1}{\mu_{ij}}\) = the average packet transmission time on link \((i,j)\)

and the average delay per packet as
\[
T = \frac{1}{\gamma} \sum_{(i,j)} \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}
\]  
where \(\gamma = \sum_s x_s\) = the total arrival rate in the system.

Processing and propagation delays are neglected.