Other Scheduling Policies

Consider mix of scientific jobs (class 1) and compilations (class 2)

- fraction of class \(i\) jobs - \(\alpha_i, \ i = 1, 2\)
- service time of class \(i\) job - exponential with mean \(1/\mu_i\) \((i = 1, 2), 1/\mu_1 > 1/\mu_2\)
- density function for job service time, \(X\),

\[
f_X(t) = \alpha_1 \mu_1 e^{-\mu_1 t} + \alpha_2 \mu_2 e^{-\mu_2 t}, \quad t \geq 0
\]
Q: what is prob. that job is scientific given that $X > q$, $q > 0$?

$$P(\text{job is scientific}|X > q) = \frac{\alpha_1 e^{q(\mu_2 - \mu_1)}}{\alpha_1 e^{q(\mu_2 - \mu_1)} + \alpha_2},$$

$$> \frac{\alpha_1}{\alpha_1 + \alpha_2},$$

$$= \alpha_1$$

$\Rightarrow$ use of round robin policy with fixed quantum $q$ is good for bursty service times.
Processor Sharing

- idealized model of round robin with fixed quantum $q > 0$ as $q \to 0$

- consider server with rate $\mu$; if $n$ jobs in queue, each receives service with rate $\mu/n$

Q: how to model?
Processor Sharing

- idealized model of round robin with fixed quantum $q > 0$ as $q \to 0$

- consider server with rate $\mu$; if $n$ jobs in queue, each receives service with rate $\mu/n$

**Q:** how to model?

Consider same example of scientific jobs and compilations

- Poisson arrivals with rate $\lambda$
• can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class \(i\) jobs, \(i = 1, 2\).

• \(\pi_{i,j} = P(N_1 = i, N_2 = j), \ i, j \geq 0\)
can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class i jobs, \(i = 1, 2\).

\[
\pi_{i,j} = P(N_1 = i, N_2 = j), \quad i, j \geq 0
\]

balance equation is

\[
\pi_{i,j}(\lambda + 1\{i > 0\} \frac{i}{i+j} \mu_1 + 1\{j > 0\} \frac{j}{i+j} \mu_2)
\]
• can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class \(i\) jobs, \(i = 1, 2\).

• \(\pi_{i,j} = P(N_1 = i, N_2 = j), \ i,j \geq 0\)

Balance equation is

\[
\pi_{i,j}(\lambda + 1[i > 0] \frac{i}{i+j} \mu_1 + 1[j > 0] \frac{j}{i+j} \mu_2) = 1[i > 0] \pi_{i-1,j} \lambda \alpha_1
\]
• can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class \(i\) jobs, \(i = 1, 2\).

• \(\pi_{i,j} = P(N_1 = i, N_2 = j), \ i, j \geq 0\)

balance equation is

\[
\pi_{i,j}(\lambda + 1_{i > 0}\frac{i}{i+j}\mu_1 + 1_{j > 0}\frac{j}{i+j}\mu_2) \\
= 1_{i > 0}\pi_{i-1,j}\lambda\alpha_1 + 1_{j > 0}\pi_{i,j-1}\lambda\alpha_2
\]
- can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class \(i\) jobs, \(i = 1, 2\).

- \(\pi_{i,j} = P(N_1 = i, N_2 = j), \ i, j \geq 0\)

Balance equation is

\[
\pi_{i,j}(\lambda + 1\{i > 0\}\frac{i}{i+j}\mu_1 + 1\{j > 0\}\frac{j}{i+j}\mu_2)
= 1\{i > 0\}\pi_{i-1,j}\lambda\alpha_1 + 1\{j > 0\}\pi_{i,j-1}\lambda\alpha_2
+ 1\{i > 0\}\pi_{i+1,j}\frac{i + 1}{i + j + 1}\mu_1
\]
• can model as MC with state \((N_1, N_2)\) where \(N_i\) denotes number of class \(i\) jobs, \(i = 1, 2\).

• \(\pi_{i,j} = P(N_1 = i, N_2 = j), \ i, j \geq 0\)

balance equation is

\[
\pi_{i,j}(\lambda + 1\{i > 0\})\frac{i}{i + j}\mu_1 + 1\{j > 0\}\frac{j}{i + j}\mu_2
\]

\[
= 1\{i > 0\}\pi_{i-1,j}\lambda\alpha_1 + 1\{j > 0\}\pi_{i,j-1}\lambda\alpha_2
\]

\[
\pi_{i+1,j} \frac{i + 1}{i + j + 1}\mu_1 + \pi_{i,j+1} \frac{j + 1}{i + j + 1}\mu_2
\]
Processor Sharing

Solution is

$$\pi_{i,j} = \pi_{0,0} \left( \frac{\lambda \alpha_1}{\mu_1} \right)^i \left( \frac{\lambda \alpha_2}{\mu_2} \right)^j \frac{(i + j)!}{i!j!}$$

Let $N = N_1 + N_2$. 
Processor Sharing

Solution is

\[ \pi_{i,j} = \pi_{0,0} \left( \frac{\lambda \alpha_1}{\mu_1} \right)^i \left( \frac{\lambda \alpha_2}{\mu_2} \right)^j \frac{(i + j)!}{i!j!} \]

Let \( N = N_1 + N_2 \),

\[ P(N = n) = \sum_{i=0}^{n} \pi_{i,n-i} \]

\[ = \pi_{0,0} \sum_{i=0}^{n} \binom{n}{i} \left( \frac{\lambda \alpha_1}{\mu_1} \right)^i \left( \frac{\lambda \alpha_2}{\mu_2} \right)^{n-i} \]
\[
= \pi_{0,0} \left( \frac{\lambda \alpha_1}{\mu_1} + \frac{\lambda \alpha_2}{\mu_2} \right)^n,
\]

\[
= \pi_{0,0} \lambda^n \left( \frac{\alpha_1}{\mu_1} + \frac{\alpha_2}{\mu_2} \right)^n,
\]

\[
= \pi_{0,0} (\lambda E[X])^n
\]

This is identical to \(M/M/1\) queue! Therefore,

\[
E[N] = \frac{\rho}{(1 - \rho)}
\]

\[
E[T] = \frac{E[X]}{(1 - \rho)}
\]
Processor Sharing \( M/G/1 \) Queue

- In general,

\[
P(N = n) = (1 - \rho)\rho^n, \quad n = 0, 1, 2, \ldots
\]

under processor sharing (PS) for arbitrary service times.

- if \( C_X > 1 \), PS is better

- if \( C_X < 1 \), FCFS is better

- considerable evidence that PS is good model for round robin with fixed quantum size \( q \) when \( q \) is small
$E[N]$ vs. $C_X$, PS and FCFS, $\rho = 0.9$
Priority $\text{M/G/1}$ Queue

- $n$ priority classes, $i = 1, \ldots, n$

- $i < j \Rightarrow i$ gets priority over $j$

- class $i$ customer enters queue ahead of all class $j$ customers, $j > i$

- at departure, highest priority customer in queue begins service

- **nonpreemptive** - once a customer enters service, it stays in service until completion
• **preemptive** - class $i$ customer arriving in system finding a class $j$ customer in service, $j > i$, replaces him
  - if class $j$ customer resumes service at place preempted, it is preemptive **resume**
  - if class $j$ customer restarts service, it is preemptive **restart**
**M/G/1 Priority Queue**

- \( X_i \) is service time of class \( i \) customer
- \( \mathbb{E}[X_i] \) and \( \mathbb{E}[X_i^2] \) are assumed known, \( i = 1, 2, \ldots, n \)
- Poisson arrivals with rate \( \lambda_i \) for class \( i \)

\[
\lambda = \sum_{i=1}^{n} \lambda_i
\]

\[
\mathbb{E}[X] = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \mathbb{E}[X_i], \quad \mathbb{E}[X^2] = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \mathbb{E}[X_i^2]
\]
\( \rho_i = \lambda_i E[X_i] \) is prob. class \( i \) customer is in service, \( i = 1, \ldots, n \); \( \rho = \sum_{i=1}^{n} \rho_i \)

\( W^{(i)}, T^{(i)}, N_q^{(i)}, N^{(i)} \) are queueing time, sojourn time, no. in queue, and no. in system for class \( i \)
Analysis of Nonpreemptive Priority $M/G/1$ Queue

Consider a tagged class $i$ customer

$W^{(i)}$ - remaining time of customer in service (if any)

+ time of all customers of priorities $1, \ldots, i$ ahead in queue

+ times of all customers of priorities $1, \ldots, i-1$ that arrive after arrival of tagged customer but before it
initiates service

\[ E[W^{(i)}] = \rho \frac{E[X^2]}{2E[X]} + \sum_{l=1}^{i} E[N_q^{(l)}] E[X_l] \]

\[ + \sum_{l=1}^{i-1} E[W^{(i)}] \lambda_l E[X_l] \]

By Little’s law, \( E[N_q^{(i)}] = \lambda_i E[W^{(i)}] \), or

\[ E[W^{(i)}] = \frac{\lambda E[X^2]}{2} + \sum_{l=1}^{i} \rho_l E[W^{(l)}] + \sum_{l=1}^{i-1} \rho_l E[W^{(i)}] \]
Analysis (cont.)

Let $\sigma_i = \sum_{l=1}^{i} \rho_i$

Therefore,

$$E[W^{(i)}](1 - \sigma_i) = \frac{\lambda E[X^2]}{2} + \sum_{l=1}^{i-1} \rho_l E[W^{(l)}], \ i = 1, \ldots, n$$

These equations have the solution

$$E[W^{(i)}] = \frac{\lambda E[X^2]}{2(1 - \sigma_i)(1 - \sigma_{i-1})}, \ i = 1, \ldots, n$$

where $\sigma_0 = 0$. 
We have

\[ E[T^{(i)}] = \frac{\lambda E[X^2]}{2(1 - \sigma_i)(1 - \sigma_{i-1})} + E[X_i] \quad i = 1, \ldots, n \]

\[ E[N_q^{(i)}] \text{ and } E[N^{(i)}] \text{ can be obtained via Little’s law.} \]
Consider a tagged class \( i \) customer. Tagged customer *only* sees customers of classes \( 1, \ldots, i \), not \( i + 1, \ldots, n \).

\( T^{(i)} \) - time to clear out all class \( 1, \ldots, i \) customers in queue at arrival (*call it* \( R_i \))

+ times of all customers of priorities \( 1, \ldots, i - 1 \) that arrive after arrival of tagged customer but before it completes (incl. preemptions)
+ service time of tagged customer

\[ E[T^{(i)}] = E[R_i] + E[T^{(i)}] \sum_{l=1}^{i-1} \lambda_l E[X_l] + E[X_i] \]

or

\[ E[T^{(i)}] = \frac{E[R_i]}{1 - \sigma_{i-1}} + \frac{E[X_i]}{1 - \sigma_{i-1}} \quad i = 1, \ldots, n \]
Analysis of Preemptive Priority $M/G/1$ Queue

**Q:** what is $E[R_i]$

- if $i = 1$, $E[R_1]$ is the mean wait time in the $M/G/1$ queue consisting solely of class 1 customers

\[
E[R_1] = \frac{\lambda_1 E[X_i^2]}{2(1 - \rho_1)}
\]

- if $i > 1$, then $E[R_i]$ is the mean time in a $M/G/1$ queue consisting of classes $1, \ldots, i$ as if it were operating under
FCFS

\[ E[R_i] = \frac{\sum_{l=1}^{i} \lambda_l E[X^2_l]}{2(1 - \sigma_i)} \]

Therefore,

\[ E[T_i] = \frac{\sum_{l=1}^{i} \lambda_l E[X^2_l]}{2(1 - \sigma_i)(1 - \sigma_{i-1})} + \frac{E[X_i]}{1 - \sigma_{i-1}}, \quad i = 1, \ldots n \]

Note differences from nonpreemptive M/G/1 queue.