Module 8:Application of stochastic processes in areas like scheduling

Lecture 32:Scheduling Rules

The Lecture Contains:

Some scheduling rules

- Rules
- Examples

Previous Next

Some scheduling rules

Rule 8.1

It is easy to find the optimal permutation schedule for the stochastic counterpart of $1||\sum w_j C_j$, when the processing time of job j is j^{th} and that is from an arbitrary distribution F_j and the objective is $E\left(\sum w_j C_j\right)$. This leads to the concept of ***weighted shortest expected processing time*** (***WSEPT***), under which one sequences the jobs in decreasing order of the ratio $\frac{w_j}{E(X_j)}$ or $\lambda_j w_j$. It can be proved that WSEPT rule minimizes the expected sum of the weighted completion times in the class of nonpreemptive static list policies as well as in the class of nonpreemptive dynamic policies.

Rule 8.2

Let us consider the stochastic version of $1||\sum w_j\left(1-e^{-rC_j}\right)$, where r is the discounted factor, with arbitrary processing times. Thus it leads us to the concept of ***weighted discounted shortest processing time first*** (***WDESPT***), under which one can minimize the expected weighted sum of the discounted completion times in the class of nonpreemptive static list policies as well as in the class of nonpreemptive dynamic policies.

Example 8.1 for rule 8.2

Let us take a simple example where there are 3 jobs with priority 1,2 and 3. Furthermore the processing time distribution of the 3 jobs namely J_1, J_2 and J_3 is $U(0,2)$, $UD(0,2,1)$ and $UD(0,2,1)$ respectively. Given this information we can easily find that $E\left(X_{J_1}\right)=E\left(X_{J_2}\right)=E\left(X_{J_3}\right)=1$, thus $r=\frac{1}{10}$. Furthermore the priority indices of the jobs are calculated as follows:

1. Job # 1, i.e.,: J_1: $E\left(e^{-rX_{J_1}}\right)=\int_0^{\infty}e^{-rt}f_1(t)dt=\int_0^{2}\frac{1}{2}e^{-0.1t}dt=0.9063$
2. Job # 2, i.e.,: J_2: $E\left(e^{-rX_{J_2}}\right)=\sum_{t=0}^{\infty}e^{-rt}P\left(X_{J_2}=t\right)=\frac{1}{3}+\frac{1}{3}e^{-0.1}+\frac{1}{3}e^{-0.2}=0.9078$
3. Job # 3, i.e.,: J_3: $E\left(e^{-rX_{J_3}}\right)=\sum_{t=0}^{\infty}e^{-rt}P\left(X_{J_3}=t\right)=\frac{1}{3}+\frac{1}{3}e^{-0.1}+\frac{1}{3}e^{-0.2}=0.9078$

Thus the priority indices are: 9.678, 9.852 and 9.852, so once can either schedule J_2 or J_3 first and then J_1In order to make a judicious decision between J_2 and J_3, one needs to find the variance and schedule the one which has larger variance, and here is where the concept of stochastic dominance with respect to expected value as well variance comes into the picture..

Previous Next

Rule 8.3

The ***expected due date*** (***EDD***) rule minimizes the maximum lateness for arbitrary distributed processing times and deterministic due dates in the class of nonpreemptive static list policies, the class of nonpreemptive dynamic policies, and the class of preemptive dynamic policies.

Rule 8.4

The policy that maximizes the total discounted expected reward in the class of preemptive dynamic policies prescribed, at each point in time, is found out using the value of largest ***Gittin's index***, where Gittin's index is given by the formulae, $G_j\left(x_j\right) = \max_\pi \frac{E_\pi\left[\sum_{s=0}^{\tau-1}\beta^{s+1}W_j\{x_j(s)\}|x_j(0)=x_j\right]}{E_\pi\left[\sum_{s=0}^{\tau-1}\beta^{s+1}|x_j(0)=x_j\right]}$, where β is the discounted factor between 0 and 1.

Example 8.2 for rule 8.4

Consider we have 3 jobs marked as J_1, J_2 and J_3. Let the corresponding weights of the 3 jobs be $w_1 = 60$, $w_2 = 30$ and $w_3 = 40$. For simplicity let us assume that the processing times of the 3 jobs can only be integer values and those values are 1, 2 and 3. Moreover the corresponding probabilities for the 3 jobs are as follows :

- Job J_1: $p_{11} = \frac{1}{6}$, $p_{12} = \frac{1}{2}$ and $p_{13} = \frac{1}{3}$
- Job J_2: $p_{21} = \frac{2}{3}$, $p_{22} = \frac{1}{6}$ and $p_{23} = \frac{1}{6}$
- Job J_3: $p_{31} = \frac{1}{2}$, $p_{32} = \frac{1}{4}$ and $p_{33} = \frac{1}{4}$

Let us assume the value of $\beta = 0.5$. Now the next step is to find the value of Gittin's index for each job and they are calculated as follows

Previous Next

Gittin's index for job J_1

When this job is put on the machine at time $t = 0$, then the discounted expected reward is given by $w_1 p_{11} \beta = 5$.
When this job is put on the machine at time $t = 1$, then the discounted expected reward is given by $w_1 p_{12} \beta^2 = 7.5$.
When this job is put on the machine at time $t = 2$, then the discounted expected reward is given by $w_1 p_{13} \beta^3 = 2.5$.

Thus $G_1\{x_1(0)\} = max\left\{\frac{5}{0.5}, \frac{5+7.5}{0.5+0.208}, \frac{5+7.5+2.5}{0.5+0.208+0.042}\right\} = 20$

Gittin's index for job J_2

When this job is put on the machine at time $t = 0$, then the discounted expected reward is given by $w_2 p_{21} \beta = 10$.
When this job is put on the machine at time $t = 1$, then the discounted expected reward is given by $w_2 p_{22} \beta^2 = 1.25$.
When this job is put on the machine at time $t = 2$, then the discounted expected reward is given by $w_2 p_{23} \beta^3 = 0.625$.

Thus $G_2\{x_2(0)\} = max\left\{\frac{10}{0.5}, \frac{10+1.25}{0.5+0.083}, \frac{10+1.25+0.625}{0.5+0.083+0.021}\right\} = 20$

Gittin's index for job J_3

When this job is put on the machine at time $t = 0$, then the discounted expected reward is given by $w_3 p_{31} \beta = 10$.
When this job is put on the machine at time $t = 1$, then the discounted expected reward is given by $w_3 p_{32} \beta^2 = 2.5$.
When this job is put on the machine at time $t = 2$, then the discounted expected reward is given by $w_3 p_{33} \beta^3 = 1.25$.

Thus $G_3\{x_3(0)\} = max\left\{\frac{10}{0.5}, \frac{10+2.5}{0.5+0.125}, \frac{10+2.5+1.25}{0.5+0.125+0.031}\right\} = 20.96$.

Thus considering the indices values of the jobs, out of J_1, J_2 and J_3, J_3 would be put on machine at time $t = 0$ and would be completed first. After the completion of job J_3, we can take up either job J_1 or job J_2 as the Gittin's index for both the jobs are the same.

Previous Next

Rule 8.5

The ***longest expected processing time first*** (***LEPT***) rule minimizes the expected makespan in the class of nonpreemptive static list policies as well as in the class of nonpreemptive dynamic policies.

Rule 8.6

Let us consider the stochastic version of $1|d_j = d|\sum w_j U_j$, where job j has an exponential distributed processing time with a rate of λ_j and a deterministic due date, d. Thus it leads to the concept of ***weighted discounted shortest processing time first*** (***WDESPT***), under which one can minimize the expected weighted number of tardy jobs in the classes of nonpreemptive static list policies, nonpreemptive dynamic policies as well as preemptive dynamic policies.

Now let us consider some simple rules with examples as applicable for problems which have release dates which are stochastic

Rule 8.7

Let us consider the stochastic version of $1|r_j, prmp|\sum w_j C_j$, where there is preemption for processing, and we also know that the highest preemptive rule is already defined. Then the optimal preemptive policy is the preemptive ***weighted discounted shortest processing time first*** (***WDESPT***) rule. Hence at any point of time, amongst jobs available, the one with highest $\lambda_j w_j$ must be processed.

Rule 8.8

Under the optimal nonpreemptive dynamic policy, the decision maker selects whenever the machine is free, from among the waiting jobs one with the highest value of $\lambda_j w_j$. This implies that the jobs are scheduled according to the ***weighted discounted shortest processing time first*** (***WDESPT***) rule.

Now let us consider some simple rules with examples as applicable for problems which have parallel machines

Rule 8.9

If there are two machines in parallel and the processing times are exponentially distributed, then the ***longest expected job processing time*** (***LEPT***) rule minimizes the expected makespan in the class of nonpreemptive static list policies.

Rule 8.10

The nonpreemptive ***shortest expected job processing time*** (***SEPT***) policy minimizes the total expected completion time in the class of preemptive dynamic policies.

Previous Next