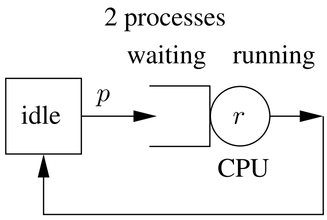
Problem 8.1



A slotted system consists of 2 processes and a CPU (see the figure). Processes can be idle, waiting in the queue or running in the CPU. Initially ($n = 0$) the 2 processes are idle. At each slot an idle process can awake with probability $p = 1/3$. Note that more than one idle process can awake simultaneously. In this case they are randomly sorted and introduced in the queue. If a process is in the queue and the CPU is empty (no process running in the CPU) the process in the head of the queue starts running immediately.

In the CPU there can be only 1 process running. After every slot a process running in the CPU can finish and go to idle state with probability $r = 1/2$. At the same time instant (i.e. if a process running in the CPU finish) an awaking process, or waiting in the queue can occupy the CPU. Define the stochastic process:

$$X(n) = \{\text{number of processes waiting or running in the CPU at time } n,\ n \geq 0\}$$

8.1.A Draw the transition diagram and derive the one step probability matrix.
8.1.B Compute the stationary distribution.
8.1.C Compute the throughput (mean number of processes dispatched by the CPU per slot).
8.1.D Let T be the random variable equal to the number of slots since a process awakes until the process is dispatched. Compute $E[T]$.

Problem 8.2
Assume the problem 7.3.

8.2.A Use the flux balancing method to compute the stationary distribution.
8.2.B Let T be the random variable equal to the number of steps between 2 consecutive losses. Compute E$[T]$ using Wald's equation.
8.2.C Compute the loss rate (mean number of packets lost per step).
8.2.D Compute E$[T]$ using the loss rate.

Problem 8.3 Google PageRank algorithm.
Idea: *a page has high rank if the sum of ranks of the pages pointing to it is high*. Defining

$$l_{ij} = \begin{cases} 1, & \text{if page } i \text{ points to page } j \\ 0, & \text{if page } i \text{ does not point to page } j \end{cases} \text{ and } p_{ij} = \frac{l_{ij}}{\sum_j l_{ij}}$$

the previous recursive idea can be described by a DTMC:

$$\pi_j = \sum_i \pi_i \, p_{ij}.$$

Computing the stationary distribution of the DTMC, π_i is the rank of page i. That is, the higher is π_i, the higher is the rank of page i.

8.3.A Assume the pages a, b, c, d with the links of the figure. Use the PageRank algorithm, solving the stationary distribution using the flux balancing method.