# Réseaux: Fiche d'exercices 4
7 octobre 2015

Scheduling in packet switches The most popular switch architecture is the crossbar switch architecture where at any given time, one input port can be connected to at most one output port, and vice versa. A simple way to represent a $N \times N$ crossbar switch is to use a $N \times N$ complete bipartite graph. A schedule is a set of connections from input ports to output ports such that no input is connected to two output ports and vice versa, i.e. a schedule has to be a matching in the bipartite graph representing the switch.

(a) Explain the phenomenon called Head-of-line (HOL) blocking illustrated by:
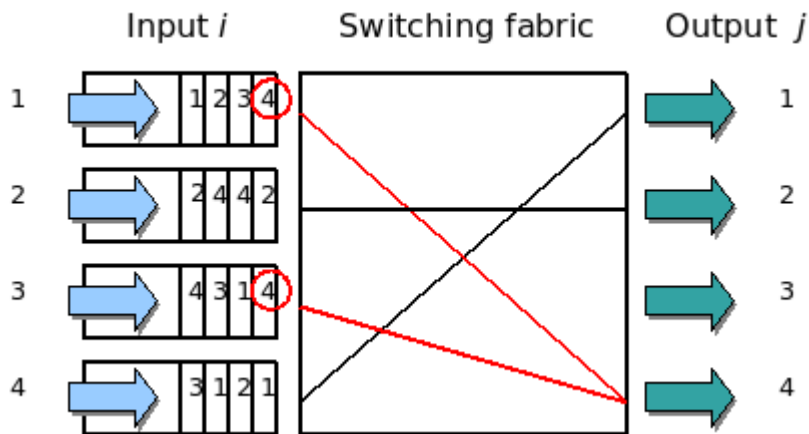
(b) In the Virtual Output Queues (VOQ) architecture, a separate queue is maintained for each output port at each input port. Explain why it solves the HOL blocking problem.

We let $VOQ(i,j)$ denote the VOQ for output port j and input port i and let $q_{ij}(t)$ denote the length of $VOQ(i,j)$ at time slot t. Further, $a_{ij}(t)$ denotes the number of packets arriving at input i at time slot t and destined to output port j. We assume that $a_{ij}(t)$ is a Bernoulli random variable with mean λ_{ij}. We define the arrival rate matrix λ to be the $N \times N$ matrix such that the (i,j)-th entry is λ_{ij}. We define

$$\mathcal{C} = \left\{ \lambda, \; \lambda_{ij} \geq 0, \; \sum_{i=1}^{N} \lambda_{ij} \leq 1 \forall j, \; \sum_{j=1}^{N} \lambda_{ij} \leq 1 \forall i \right\}.$$

(c) Show that if $\lambda \notin \mathcal{C}$ then no scheduling algorithm can support arrival rate matrix λ, i.e. $|\sum_{i,j} q_{ij}(t)|$ is transient.

MaxWeight scheduling: The switch finds a matching $M(t)$ such that

$$M(t) \in \arg\max_{M} \sum_{i,j} q_{ij}(t) M_{ij},$$

and transfers a packet from $VOQ(i, j)$ to output port j if $M_{ij}(t) = 1$ and $q_{ij}(t) + a_{ij}(t) > 0$. We will show that MaxWeight algorithm can support any arrival rate matrix λ such that $(1 + \epsilon)\lambda \in \mathcal{C}$.

A permutation matrix U is a $N \times N$ matrix such that $U_{ij} \in \{0, 1\}$ and $\sum_{i=1}^{N} U_{ij} = \sum_{j=1}^{N} U_{ij} = 1$.

(d) Show that a $N \times N$ matrix λ is doubly stochastic if and only if it is a convex combination of permutation matrices.

(e) Show that if a $N \times N$ matrix λ is substochastic then there exists a doubly stochastic matrix $\tilde{\lambda}$ such that $\lambda_{ij} \leq \tilde{\lambda}_{ij}$ for all i, j.

(f) Show that $q(t)$ is an irreducible Markov chain.

(g) Define the Lyapunov function $V(q(t)) = \sum_{i,j} q_{ij}(t)^2$. Show that

$$\mathbb{E}\left[V(q(t+1)) - V(q(t))|q(t) = q\right] \leq 2\sum_{i,j} q_{ij}\left(\lambda_{ij} - \mathbb{E}[M_{ij}(t)|q(t) = q]\right) + \sum_{i,j} \lambda_{ij} + N.$$

(h) Prove that

$$\mathbb{E}\left[V(q(t+1)) - V(q(t))|q(t) = q\right] \leq \sum_{i,j} \lambda_{ij} + N - 2\epsilon\sum_{i,j} \lambda_{ij}q_{ij},$$

and conclude.

A maximal matching is a matching such that if any other edge is added to the set, it is not a matching. The complexity of finding a maximal matching is $O(N \log N)$ which is much lower than the complexity of finding a maximum matching $O(N^3)$.

Maximal matching scheduling: The switch first removes all edges with $q_{ij}(t) = 0$ from the complete bipartite graph and then finds a maximal matching $M(t)$ in the remaining bipartite graph. One packet in $VOQ(i, j)$ is sent to output port j if $M_{ij}(t) = 1$.

We will show that Maximal matching scheduling can support an arrival rate matrix λ such that $2(1 + \epsilon)\lambda \in \mathcal{C}$ for some $\epsilon > 0$.

(i) Show that $\sum_h \lambda_{ih} + \sum_k \lambda_{kj} \leq \frac{1}{1+\epsilon}$ and that if $q_{ij}(t) \neq 0$, then

$$\sum_h M_{ih}(t) + \sum_k M_{kj}(t) \geq 1.$$

(j) Consider the Lyapunov function

$$V(q(t)) = \sum_{i,j} q_{ij}(t) \left(\sum_h q_{ih}(t) + \sum_k q_{kj}(t)\right),$$

and conclude.

Load-balanced switch: the load-balanced switching architecture consists of two stages. At the first stage, one FIFO queue is maintained at each input port and at the second stage, VOQs are maintained at input ports. At time t, set $h = t \mod N$. At the first stage switch, if the HOL packet at input queue i is destined to output port k, the packet is transfered to $VOQ(j, k)$ at the second-stage switch with $j = (i + h) \mod N$. For each input port i at the second-stage switch, a packet from $VOQ(i, j)$ is transfered to output j (with the same value as above).

We will show that the load-balanced switch architecture can support any λ such that $(1+\epsilon)\lambda \in \mathcal{C}$ for some $\epsilon > 0$. For simplicity, we modifiy the arrival process as follows: a packet arrives to queue i with probability λ_i and p_{ij} is the probability that the arrived packet is destinated to output port j. We define $\lambda_{ij} = \lambda_i p_{ij}$.

(k) Show that input queue at the first-stage switch are always empty. What is the mean arrival during N consecutive time slots to $VOQ(i, j)$ at the second-stage switch? Conclude.