Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

HS 2015 Prof. R. Wattenhofer / G. Bachmeier, S. Brandt, K.-T. Foerster, J. Seidel

# Discrete Event Systems

## Solution to Exercise Sheet 10

## 1 PhD-Scheduling

**a)** (i) SmallLoad distributes the tasks as follows:

| | | | |
|---|---|---|---|
| PhD student 1: | 2 | 4 | 7 |
| PhD student 2: | 5 | 3 | |

Opt uses the following distribution (or another one with the same cost):

| | | | |
|---|---|---|---|
| PhD student 1: | 2 | 5 | 3 |
| PhD student 2: | 4 | 7 | |

SmallLoad thus distributes the tasks with cost $\textsc{Alg}(\sigma) = 13$ while Opt incurs a cost of $\textsc{Opt}(\sigma) = 11$. Hence,

$$\rho(\sigma) = \frac{\textsc{Alg}(\sigma)}{\textsc{Opt}(\sigma)} = \frac{13}{11} \ .$$

(ii) The following sequence results in a larger competitive ratio: $\sigma = 1, 1, 2$. We have $\textsc{Alg}(\sigma) = 3$ and $\textsc{Opt}(\sigma) = 2$ and thus

$$\rho(\sigma) = \frac{\textsc{Alg}(\sigma)}{\textsc{Opt}(\sigma)} = \frac{3}{2} \ .$$

(iii) See **b)**.

(iv) No, finding the optimal solution offline corresponds to solving the Partition-problem, which is NP-complete, thus presumably no efficient algorithm exists for the problem.

**b)** We first show a lower bound of $(2 - \frac{1}{m})$ on the competitive ratio of SmallLoad. To this end, we choose an input sequence that consists of $m(m-1)$ tasks of size 1 concluded with a task of size $m$, i.e. $\sigma = \underbrace{1, \ldots, 1}_{m(m-1)}, m$. After assigning the first $m(m-1)$ tasks, SmallLoad has assigned $m-1$ units to each of the $m$ PhD students. The last task of size $m$ incurs a load of $2m-1$ for the student to whom it is assigned.

The optimal algorithm assigns the first $m(m-1)$ taks to only $m-1$ students and the last (heavy) task to the remaining student. This results in a maximal load of $m$ and we get the following lower bound for the competitive ratio:

$$c \geq \frac{\textsc{Alg}(\sigma)}{\textsc{Opt}(\sigma)} = \frac{2m-1}{m} = 2 - \frac{1}{m}$$

Now we shall show a matching upper bound for the competitive ratio. Let $\sigma = (e_1, e_2, \ldots)$ be an arbitrary input sequence. Without loss of generality, we assume $s_1$ to be the student with the maximal load for $\sigma$. Furthermore, let $w$ be the effort of the last task $T$ assigned $s_1$ and $E$ the load of $s_1$ before assigning its last task. The load of all other students must be at least $E$ since $s_1$ was the student with minimal load when he was assigned task $T$ (otherwise another student would have received $T$). Hence, the sum of the loads of all students is at least $m \cdot E + w$ and hence

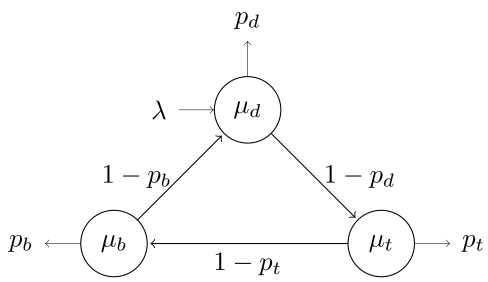
$$\textsc{Opt}(\sigma) \geq \frac{m \cdot E + w}{m} = E + \frac{w}{m} \ .$$

Using $\textsc{Opt}(\sigma) \geq w$, we get

$$\begin{aligned}
\textsc{Alg}(\sigma) &= w + E \\
&\leq w + \textsc{Opt}(\sigma) - \frac{w}{m} \\
&= \textsc{Opt}(\sigma) + \left(1 - \frac{1}{m}\right) w \\
&\leq \textsc{Opt}(\sigma) + \left(1 - \frac{1}{m}\right) \textsc{Opt}(\sigma) \\
&= \left(2 - \frac{1}{m}\right) \textsc{Opt}(\sigma)
\end{aligned}$$

# 2 Queuing Networks

**a)**




**b)** We have an open queuing network and hence we can apply Jackson's theorem (slides 97ff):

$$\begin{aligned}
\lambda_d &= \lambda + \lambda_b(1 - p_b) \\
\lambda_t &= \lambda_d(1 - p_d) \\
\lambda_b &= \lambda_t(1 - p_t)
\end{aligned}$$

Solving this equation system gives:

$$\lambda_d = \frac{\lambda}{1 - (1 - p_d)(1 - p_t)(1 - p_b)}$$

$$\lambda_t = \frac{(1 - p_d)\lambda}{1 - (1 - p_d)(1 - p_t)(1 - p_b)}$$

$$\lambda_b = \frac{(1 - p_d)(1 - p_t)\lambda}{1 - (1 - p_d)(1 - p_t)(1 - p_b)}$$

**c)** The waiting time is given by $W_t = \rho_t/(\mu_t - \lambda_t)$, where $\rho_t = \lambda_t/\mu_t$.

**d)** We apply the given values to the equations for $\lambda_d$, $\lambda_t$ and $\lambda_b$ and obtain:

$$\lambda_d = 10, \qquad \lambda_t = 25/3, \qquad \lambda_b = 20/3.$$

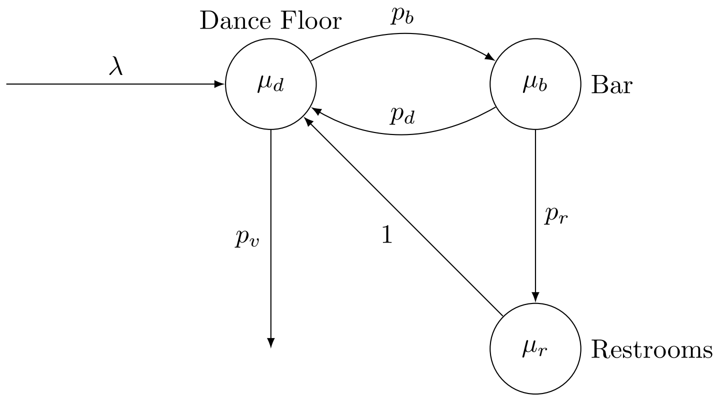
Therefore, by the formula of slide 73 and linearity of expectation, the expected number of customers in the system is given by

$$N = \frac{\lambda_d}{\mu_d - \lambda_d} + \frac{\lambda_t}{\mu_t - \lambda_t} + \frac{\lambda_b}{\mu_b - \lambda_b} = 8.$$

Applying Little's formula to the entire system gives $T = N/\lambda = 8/5$ hours.

# 3 A Night at the DISCO

**a)** As a queuing network, the DISCO can be modeled as follows.




**b)** We obtain the following system of linear equations:

$$\begin{aligned} \lambda_d &= \lambda + \lambda_b \cdot p_d + \lambda_r \\ \lambda_b &= \lambda_d \cdot p_b \\ \lambda_r &= \lambda_b \cdot p_r\,. \end{aligned}$$

Solving for $\lambda_d$ yields

$$\lambda_d = \frac{\lambda}{1 - p_b \cdot p_d - p_b \cdot p_r}\,.$$

**c)** We need to ensure that $\lambda_r/(m \cdot \mu_r) < 1$. Counting in hours, we have that $\lambda_r = 90$ and $\mu_r = 12$, which yields that $m$ must be at least 8 (since there is no such thing as half a toilet).

**d)** This is incorrect since with probability $p_v$, the visitor does not go to the bar, and even if he does, he does not go to the toilet with probability $p_d$.