Challenge #2: load balancing

Definitions (1/2)

- Consider a cluster of N servers to which a flow of task is offered through a Load Balancer.
- Tasks arrive at the Load Balancer according to a renewal ON-OFF process.
  - Inter-arrival times are distributed according to the r.v. T:

$$T = \begin{cases} T_0 & \text{w.p. } q \\ T_0 + Y & \text{w.p. } 1-q \end{cases}$$

    It is $\mathrm{E}[T] = T_0 + (1-q)\mathrm{E}[Y]$
  - Y is a negative exponential random variable with mean $\overline{Y}$.
  - Assume T_0 as the unit of time ($T_0 = 1$), $q = 3/5$ and $\overline{Y} = 10$.

Definitions (2/2)

- Processing times of tasks are i.i.d. random variables distributed according to a Weibull random variable X. It is $\mathcal{P}(X > t) = \exp\left(-\left(\frac{t}{\beta}\right)^{\alpha}\right)$. Assume $\alpha = 1/2$.
- The mean of X is $\mathrm{E}[X] = \beta\Gamma(1 + 1/\alpha)$, where $\Gamma(\cdot)$ is the Euler Gamma function (remember that $\Gamma(k+1) = k!$ for integer k)..
- The mean serving rate for each server is $\mu = 1/\mathrm{E}[X]$.
- The utilization coefficient of the system is $\rho = \frac{\Lambda}{N\mu} = \frac{\mathrm{E}[X]}{N \cdot \mathrm{E}[T]}$.

Generation of random variables

A sample of the inter-arrival T can be generated as follows

- IF $R_1 <= q$
  - THEN $T = T_0$
  - ELSE $T = T_0 - \overline{Y} \log(R_2)$

If $\mathrm{E}[X] \gg 1$, a sample of processing time X can be generated as follows.

$$X = \max\left\{1, \min\left\{100 \cdot \mathrm{E}[X], f\left(\beta\left(-\log R_3\right)^{1/\alpha}\right)\right\}\right\}$$

where $f(x)$ is the integer part of x (`round()` function).

R_1, R_2, R_3 are random variables uniformly distributed in $[0, 1]$, obtained by means of a library function `rand()`.

Alternative generation of service times

In the order of tens of thousands samples of service times should be enough to achieve reliable statistical estimates of the performance metrics.

In case you have trouble obtaining reliable statistical measures with Weibull distributed service times, you can use the following fall-back alternative for generating service times (please, specifiy which kind of service times you are using in your final delivery):

$$X = \begin{cases} floor(b/2) & \text{with probability } 0.4 \\ ceil(b/2) & \text{with probability } 0.4 \\ 3b & \text{with probability } 0.2 \end{cases}$$

where the constant b is set so as to obtain a desired mean value of X (note that it is $\mathrm{E}[X] = b$).

Assignment

1. Write a script to simulate the load balancing system with the following policies:
   a. Pod for $d = 3$.
   b. JBT-d, for $d = 3$. Assume that the sampling time is $1000 \cdot T_0$
   c. JSQ
2. As a function of ρ, for $\rho \in [0.8, 0.99]$, and for $N = 20$
   1. Evaluate the mean system time $\mathrm{E}[D]$ of tasks through the system.
   2. Evaluate the mean number of messages-per-task-arrival sent in the system (overhead).
3. Assume the load balancer knows the required service time for each arriving task and invent a new scheduling algorithm to assign tasks to servers. Compare $\mathrm{E}[D]$ and the mean overhead of your algorithm with that of JSQ.

Delivery of the assignment

The delivery of the assignment consists of a **2 pages written report** (Font size: 12 pt). Remember to put your given and family names, and enrollment number as a header on top of each page.

1. **PAGE 1** - (i) Write the expression of the parameter β as a function of ρ, N, α, and $\mathrm{E}[T]$; (ii) Plot the mean delay; (iii) Plot the mean message overhead (mean number of messages per task). The curves in (ii) and (iii) are plotted as a function of ρ for $\rho \in [0.8, 0.99]$ and $N = 20$, for the three considered policies Pod-3, JBT-3 and JSQ
2. **PAGE 2** (i) Explain in a concise, clear and accurate way your invented algorithm. (ii) Present two plots (one for mean delay and one for mean overhead, as in (ii) and (iii) above) where you compare your invented policy with JSQ.