Week 9: Continuous-time Markov chains
Cellular network design
Solutions

A Departure process. Since $T_{di} \sim exp(\mu)$ is the time at which customer i hangs up, the random variable T_k describing the time until the next departure can be written as

$$T_k = \min(T_{d1}, T_{d2}, ..., T_{dk}).$$

Indeed, the first departure occurs as soon as the first costumer hangs up. To show T_k is exponentially distributed with parameter $k\mu$, let us derive its cdf. To do so, note that $T_k > s$, for some time s if and only if $T_{di} > s$ for all i. Since the costumers are independent, we obtain

$$\mathbb{P}\left[T_k > s\right] = \mathbb{P}\left[T_{d1} > s\right] \mathbb{P}\left[T_{d2} > s\right] \dots \mathbb{P}\left[Tdi > s\right].$$

Using the fact that the T_{di} are exponentials yields

$$\mathbb{P}\left[T_k > s\right] = (e^{-\mu s})^k = e^{-(k\mu)s}$$

Using the definition of the cdf of the exponential distribution, T_k is therefore exponentially distributed with parameter $k\mu$.

B Four simple questions on the departure process.

(1) Since the call duration of all costumers follows the same distribution (costumers are independent and identically distributed, i.e., i.i.d.), they are all equally likely to be the first to hang up. Hence,

$$\mathbb{P}\left[\text{Customer 1 is the first to hang up}\right] = \frac{1}{k}.$$

(2) Recall that exponential distributions have a memoryless property, i.e., for $X \sim \exp(\lambda)$, $\mathbb{P}\left[X > t + s \mid X > t\right] = \mathbb{P}\left[X > s\right]$. Since call durations are exponentially distributed, the amount of time a customer has already been on a call is irrelevant to know how much longer the call will last. Since costumers are i.i.d., both calls are equally likely to end first and

$$\mathbb{P}\left[\text{Customer } i \text{ hangs up before Customer } j\right] = 1/2.$$

(3) We can find the probability that $T_{di} > 3$ minutes for $1/\mu = 3$ minutes directly from the exponential CDF:

$$\mathbb{P}\left[T_{di} > 3\right] = e^{-\mu 3} == e^{-(1/3)3} \approx 0.37.$$

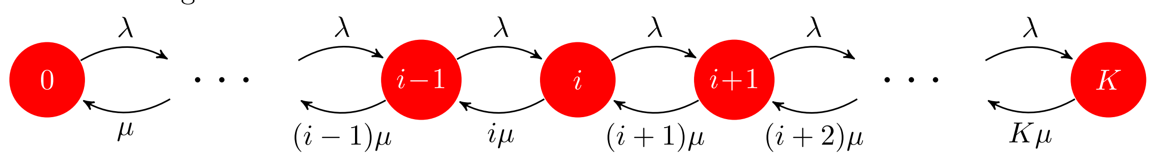
(4) Using the memoryless property as in (2), $\mathbb{P}\left[T_{dj} > 3\right] = \mathbb{P}\left[T_{di} > 3\right] \approx 0.37$.

C Continuous time Markov chain (CTMC) model. Due to the memoryless property of the exponential distribution, the number of calls at time t depends only on the number of calls in the previous instant (the process is memoryless). Moreover, the probability of going from any state i into any state j is the same regardless of when we are in that state (the process is time invariant). Hence, the number of calls established at time t can be modeled as a CTMC (refer to slide 35 of `continuous_time_markov_chains`).

The transition rates q_{ij} are given by

$$q_{ij} = \begin{cases} \lambda, & j = i+1 \\ i\mu, & j = i-1 \\ 0, & \text{otherwise} \end{cases}$$

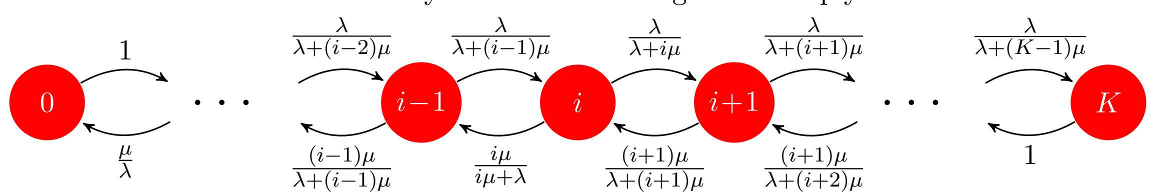
The transition diagram is:



D Alternative CTMC representation. The transition rate out of state k is $\nu_k = \lambda + k\mu$ for all states k. The probabilities P_{ij} are given by

$$P_{ij} = \begin{cases} 1, & i = 0 \text{ and } j = 1 \\ \frac{\lambda}{\nu_i}, & 0 < i < K \text{ and } j = i+1 \\ \frac{i\mu}{\nu_i}, & 0 < i < K \text{ and } j = i-1 \\ 1, & i = K \text{ and } j = K-1 \\ 0, & \text{otherwise} \end{cases}$$

E Embedded Markov chain (MC) and ergodicity of the CMTC. The embedded discrete MC associated with this CTMC has the same transition probabilities P_{ij}, but the transitions happen in uniform intervals rather than randomly. Its transition diagram is simply



The CMTC $X(t)$ is ergodic because its embedded DTMC is irreducible (there is only one class) and positive recurrent (since it is finite). The question of periodicity is irrelevant since periodic transitions occur with probability zero in continuous time.

F System simulation. The MATLAB function to simulate the call requesting process is given below.

```
function [ t, X ] = cell_model( lambda, avg_duration, K, t_max )
%CELL_MODEL Simulate calls made to a base station using CTMC

mu = 1/avg_duration;
```

```matlab

% Initialize CTMC
X = 0;
t = 0;
index = 1;

% Simulate CTMC
while t(index) < t_max
    % Draw time until next transition
    tau = exprnd( 1/(lambda + mu*X(index)) );
    t(index+1) = t(index) + tau;

    % State transition
    if X(index) == 0
        % Costumer requests a call
        X(index+1) = 1;
    elseif X(index) == K
        % Costumer hangs up
        X(index+1) = K-1;
    else
        % Draw if a costumer hangs up or requests a call
        u = rand;

        if u < X(index)*mu/(lambda+X(index)*mu)
            % Costumer hangs up
            X(index+1) = X(index) - 1;
        else
            % Costumer requests a call
            X(index+1) = X(index) + 1;
        end
    end

    index = index + 1;
end

end
```

To simulate the setup of the exercise, we can use the script:

```matlab
% Delete all variables and close figures
clear all
close all

lambda = 25;    % Call arrival rate (calls per minute)
mu = 56/60;     % Average durantion of calls (minutes)
K = 32;         % Number of channels
t_max = 30;     % Duration of simulation (minutes)

% Simulation
[ t, X ] = cell_model( lambda, 1/mu, K, t_max );

% Plot
figure();
stairs(t, X, 'Linewidth', 2)
xlabel('Time (minutes)')
ylabel('Number of calls in progress [X(t)]');
xlim([0 t_max]);
ylim([0 K]);
grid;


%%% Export figure %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
set(gcf,'Color','w');
export_fig -q101 -pdf HW9_F.pdf
```

A sample realization of this process is shown in Figure 1.

G Limit distribution. The simplest way to derive the limit distribution is to follow the hint from the exercise and express all P_i in terms of P_0. To do so, we can write the set of balance equations

$$\lambda P_i = (i+1)\mu P_{i+1}, \quad \text{for } i = 0, 1, \ldots, K-1,$$

from which we obtain

$$P_{i+1} = \frac{1}{(i+1)} \frac{\lambda}{\mu} P_i, \quad \text{for } i = 0, 1, \ldots, K-1.$$

This implies $P_1 = \lambda/\mu P_0$, $P_2 = \lambda/\mu P_1 = 1/2(\lambda/\mu)^2 P_0 \ldots$ Solving the recursion we obtain

$$P_i = \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i P_0, \quad \text{for } i = 0, 1, \ldots, K.$$

Using the fact that the P_i form a probability distribution, we obtain the additional equation we need to satisfy, namely

$$\sum_{i=0}^{K} P_i = 1$$

Thus,

$$P_0 = \left[\sum_{i=0}^{K} \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i\right]^{-1} \tag{1}$$

and

$$P_i = \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i \left[\sum_{i=0}^{K} \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i\right]^{-1}.$$

You may be tempted to try to find a closed form for (1). Resist that temptation: it does not exist! Notice that the summation in (1) looks similar to the CDF of a Poisson random variable. It turns out that its value depends on *upper incomplete gamma function*, which can only be computed numerically.

H Ergodic limits. The ergodic limit converges almost surely to the limit probability when the embedded DTMC of the CTMC in question is irreducible and positive recurrent, in other words, when the CTMC is ergodic. As we have argued in Part E, this is the case for the current model. Therefore, we can write

$$\bar{p}_k = P_k = \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i \left[\sum_{i=0}^{K} \frac{1}{i!} \left(\frac{\lambda}{\mu}\right)^i\right]^{-1}, \quad \text{almost surely for } 0 \le k \le K. \tag{2}$$

The expression equating the ergodic limit and the limit probability in (2) does *not* imply that $\bar{p}_k = P_k$ for *all* realization of the process $X(t)$. Recall that this equivalence holds *almost surely*, i.e., with probability one. Therefore, there may be realizations for which $\bar{p}_k \neq P_k$, but these realizations occur with probability zero. To make this argument more concrete, suppose that you ran an infinite number of realizations of this process and found that in 10 of them $\bar{p}_k \neq P_k$. This implies that

$$\mathbb{P}\left[\bar{p}_k = P_k\right] = 1 - \mathbb{P}\left[\bar{p}_k \neq P_k\right] = 1 - \frac{10}{\infty} = 1. \tag{3}$$

Notice that you can replace 10 by 10^6 or even $10^{10^{10}}$ without making any difference: as long as it is a finite number, you haven't violated anything. Hence, there may be a large number of realizations for which $\bar{p}_k \neq P_k$, but there is an overwhelmingly larger number of realizations for which $\bar{p}_k = P_k$ (an infinite number of them, in fact).

This is a very subtle question. You should probably read this argument at least once more and ask your TAs if things are not clear.

I Approximating P_k using a simulation. In this case, it is simpler to estimate P_k from a single run of the simulation instead of multiple ones. In the former case, we simply need to choose $t_{\max}$ long enough to ensure that we have enough samples to estimated the limit probabilities using the ergodic averages. Recall that the *limit* of the ergodic averages yields the limit probabilities. If we were to use ensemble averages (i.e., multiple runs), not only would we have to take $t_{\max}$ large enough to attain the limiting (steady-state) regime of the CTMC, but we would need to run this experiment multiple times (each time obtaining a single sample from where the CTMC has landed).

When we choose to use $t_{\max} = 10^4$ minutes (roughly one week), since the ergodic averages variation is below the required precision: the maximum change between the probabilities estimated with $t_{\max} = 10^3$ and $t_{\max} = 10^4$ is less than 4×10^{-3}. The MATLAB script below was used, together with the code from part F, to produce the results in Figure 2.

```
% Delete all variables and close figures
clear all
% close all

lambda = 25;          % Call arrival rate (calls per minute)
mu = 56/60;           % Average durantion of calls (minutes)
K = 32;               % Number of channels
t_max = [1e3 1e4];    % Duration of simulation (minutes)

% Initialize Pk
Pk = zeros(K+1,length(t_max));

for i = 1:length(t_max)
    % Simulate process
    [ t, X ] = cell_model( lambda, 1/mu, K, t_max(i) );

    % Estimate limit probability
    state = X(1:end-1);
    time_in_state = diff(t);
    total_time = t(end);

    for k = 0:K
        Pk(k+1,i) = sum(time_in_state(state == k))/total_time;
    end
end

disp(abs(max(Pk(:,1) - Pk(:,2))));

% Plot
figure();
bar(0:K, Pk(:,2))
xlabel('State k')
ylabel('P_k');
grid;

```

```

%%% Export figure %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
set(gcf,'Color','w');
export_fig -q101 -pdf HW9_I.pdf
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

J Blocked call probability. Notice that customers are denied service whenever all channels are occupied when they request to try to make a call. Hence, we can write

$$\mathbb{P}\left[\text{blocked call}\right] = \mathbb{P}\left[X(t) = K\right],$$

which we know from Part G asymptotically becomes

$$\mathbb{P}\left[\text{blocked call}\right] = P_K = \frac{1}{K!}\left(\frac{\lambda}{\mu}\right)^K \left[\sum_{i=0}^{K} \frac{1}{i!}\left(\frac{\lambda}{\mu}\right)^i\right]^{-1}.$$

K Determining the need to add a new BS. Since we do not have the whole database, we cannot determine the two busiest days of the year. But let's assume that they are 12/24 (Christmas eve) and 11/23 (Thanksgiving day). Ignoring these days and the next two largest entries, we obtain our design target of 872 calls. Including the 5% predicted increase in demand, our target call rate becomes $\lambda^\star = 872(1.05)/30 \approx 31$ calls per minute. Using Part J, we then compute the probability of blocked call (numerically) with the current 32 channels (be careful with the units of μ!)

$$\mathbb{P}\left[\text{blocked call}\right] = \frac{1}{32!}\left(31 \times \frac{56}{60}\right)^{32} \left[\sum_{i=0}^{32} \frac{1}{i!}\left(31 \times \frac{56}{60}\right)^i\right]^{-1} \approx 0.0799 > 0.02.$$

Hence, we do need to install a new BS.

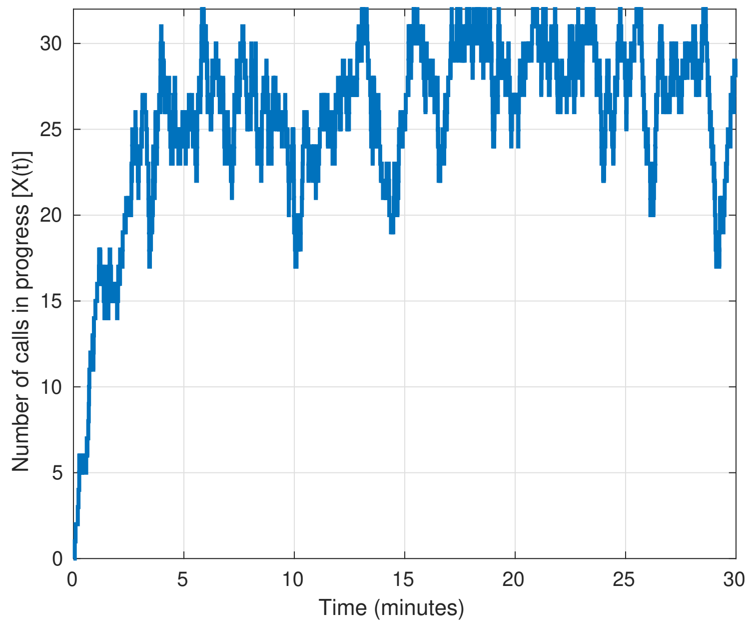


Figure 1: A realization of the calls process with $K = 32$ channels, $\lambda = 25$ calls per minute, and average call duration of $1/\mu = 56$ seconds (part F).

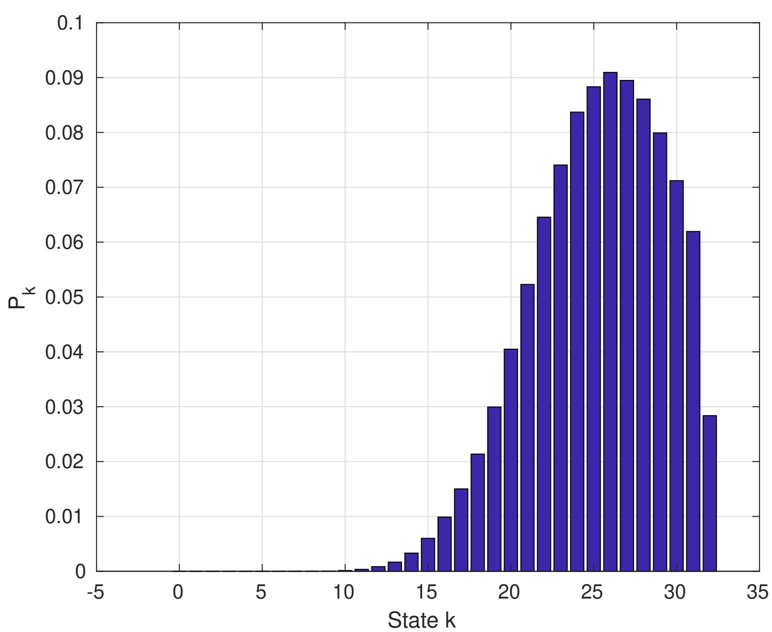


Figure 2: Estimated limit distribution of the CTMC from a single realization of length $t_{\max} = 10^4$ minutes (part I).