Simulation & Analysis of M|G|1 Queuing Systems

Ahmed Almostafa Gashgash

Project Report
Computer Communication Networks
Instructor - Prof. Ness Shroff

20 November 2017

1 Introduction

In this project we simulate an M|M|1, M|E_k|1, and M|D|1 queuing systems and study the performance of each. We compare the mean queue length for all systems using the Pollaczek Khinchin formula.

2 Generating Random Variables

2.1 Uniform Random Variable

A uniform random variable distribution denoted by $U[a, b]$ entails that for n values within the interval $[a, b]$, each has a probability of $\frac{1}{n}$. The rand command in MATLAB generates random numbers between $[0, 1]$ according to a uniform random variable. In Figure 1 we plot $P(X > x)$ for $x \in (0.5, 1)$, where X is a uniformly distributed random variable.

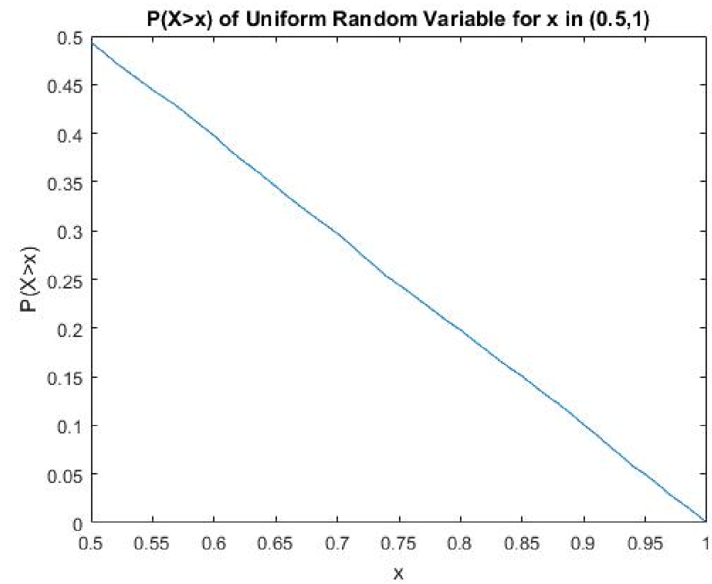

Figure 1: 1-CDF of Uniform Random Variable.

2.2 Exponential Random Variable

We can generate the exponential random variable X from its cumulative distribution function as follows:

$$F_X(x) = 1 - e^{-\lambda x} = P(U \geq e^{-\lambda x}) = P(\frac{-ln(U)}{\lambda} \leq x) = P(X \leq x)$$

where $U \in [0,1]$ is a uniformly distributed random variable and $\lambda > 0$ is the rate. We can see that $X = \frac{-ln(U)}{\lambda}$. The following MATLAB function generates exponential random variables:

Matlab Code

```
function e = exp_rv(n,avg)
% n : a vector of uniformly distributed random variables
% avg : the average of the desired exponential = 1/rate
% e : exponentially distributed random variables

e=-log(n)*avg;
end
```

In Figure 2 we plot $P(X > x)$, where X is a exponentially distributed random variable with mean value of 2.

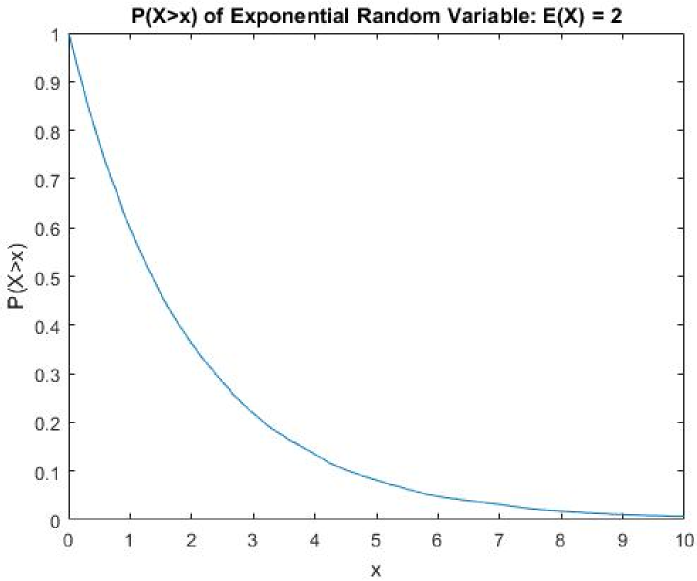

Figure 2: 1-CDF of Exponential Random Variable.

2.3 Poisson Random Variable

Similar to the steps followed when We generating an exponential random variable, we can generate a poisson random variable x according to the following equation:

$$U = e^{-\lambda} \sum_{k=0}^{\lfloor x \rfloor} \frac{\lambda^k}{k!}$$

where $U \in [0,1]$ is a uniformly distributed random variable and $\lambda > 0$ is the rate. The following MATLAB function generates an exponential random variable x :

Matlab Code

```

function p= pois_rv(n,avg)
% n : a vector of uniformly distributed random variables
% avg : the average of the desired poisson = rate
% p : poisson distributed random variables

p = zeros(length(n),1);
for i =1:length(n)
```

```
        k=0;
        m=n(i);
        u=(avg.^k)/factorial(k);
        while m > exp(-avg)*u
            m=m*n(i);
            k=k+1;
            u=u+(avg.^k)/factorial(k);
        end
        p(i) = k;
end
end
```

In Figure 3 we plot $P(X > x)$, where X is a poisson distributed random variable with mean value of 2.

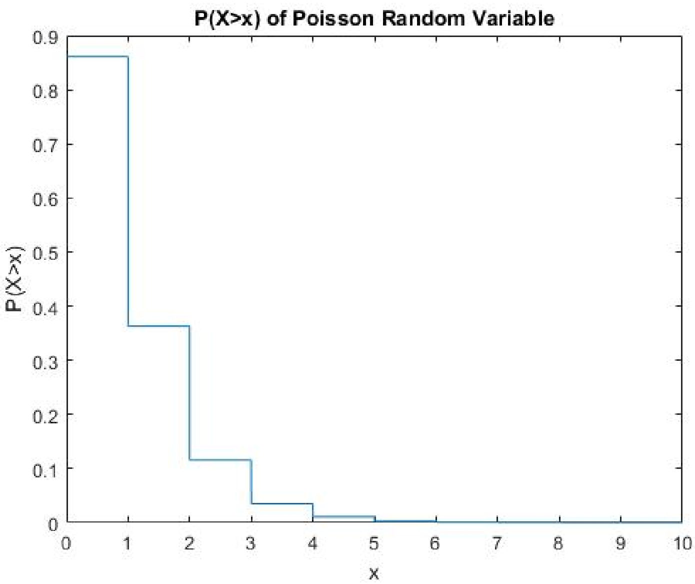

Figure 3: 1-CDF of Poisson Random Variable.

3 Simulation of an M|M|1 Queue

An M|M|1 has an infinite capacity queue and one server. The inter arrival time of packets to the queue follows a Poisson process with rate λ, and the service time for each packet follows an exponential distribution with rate μ. The following MATLAB code simulates an M|M|1 queue.

Matlab Code

```
%% M|M|1 Queue
c = 120000;                    %customers
step_size = 0.001;             %minutes
st =80*50/step_size;           %= hour simulation time
lambda = 5;                    % arrivals per minute
u = 6;                         %departure per minute
rho = lambda/u;

% Generate arrival times, service and departure times according to M|M|1
% queue
a_time = zeros(c,1);       %real time of arrival for each packet
s_time = zeros(c,1);       %service time for each packet
t_time = zeros(c,1);       %total time for each packet
```

```
%Interarrival time is generated by poisson
inter_a_time = exp_rv(rand(c,1),1/lambda);
a_time(1) = inter_a_time(1);
for i = 2:c
    a_time(i) = inter_a_time(i) + a_time(i-1);
end

%Service time is generated by exponential
s_time = exp_rv(rand(c,1),1/u);

%leaving time for each packet in system
t_time(1)=a_time(1)+s_time(1);
for i=2:c
    L = a_time(i)+s_time(i);
    K = t_time(i-1)+s_time(i);
    if L>K
        t_time(i) = L;
    else
        t_time(i) = K;
    end
end

%simulate arrival and departure
a_log = zeros(st,1);   % 1 if arrival happens, 0 o.w , for debugging
d_log = zeros(st,1);   % 1 if departure happens, 0 o.w , for debugging
n_log = zeros(st,1);   % #packets in system at each simulation step
cps = 0;               % current packets in system
tpa = 0;               % total packets arrived
tpl = 0;               % total packets that left
for i =1:st
    n_log(i)=cps;

    % Customer arrival
    if tpa < c
        if step_size*i >= a_time(tpa+1)
            a_log(i) = 1;
            cps = cps+1;
            tpa = tpa+1;
        end
    end

    %Customer departure
    if tpl < c
    if step_size*i >= t_time(tpl+1) && cps>0
        d_log(i) = 1;
        cps = cps-1;
        tpl = tpl+1;
    end
    end
end

v=250;
pn =zeros(v,1);  %probability of number of packets
for i =1:v
pn(i) = size(find(n_log==i))/size(n_log);
end
```

```
% average number
En=0;   %average number of customers
for i =1:v
    En = En + pn(i)*i ;
end
```

When Simulating this program, the average number of packets in the queue $E[n] = 4.939$ at any time, and the average delay $E[d] = 0.9878$ minutes, for arrival rate $\lambda = 5$ packets per minute and departure rate $\mu = 6$ packets per minute.

In Figure 4 we plot the probability distribution of the number of packets in the queue P_n and the number of packets n.

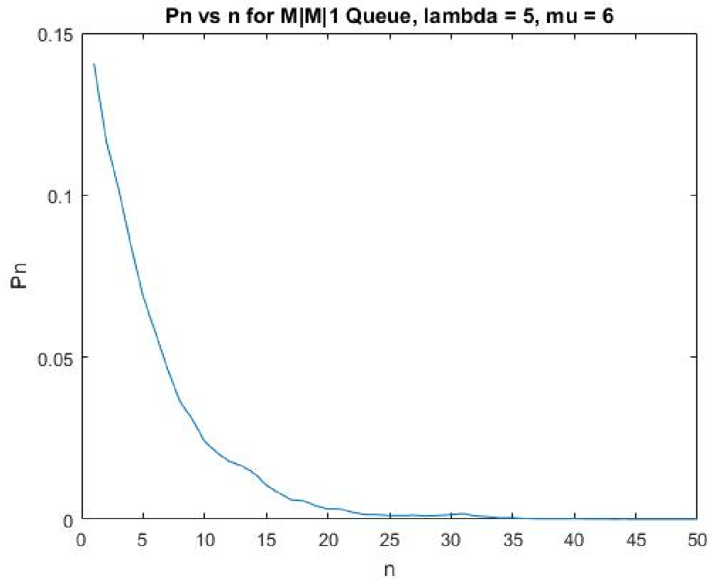

Figure 4: Pn vs n

4 Simulation of an M|E_k|1 Queue

The M|E_k|1 is different from the M|M|1 queue in that the service time for each packet follows an Erlang distribution, where

$$Erlang := Y_k = \sum_{i=1}^{k} X_i$$

$$X_i \sim exp(k\lambda)$$

The inter arrival time of packets to the queue also follows a Poisson process with rate λ. The MATLAB code to simulate an M|E_k|1 queue is the same as the previous code for the M|M|1 case except for the service time distribution. The following edit is made for k=4:

Matlab Code

```
%Service time is generated by an erlang random variable
k=4;
for i=1:k
    s_time =s_time + exp_rv(rand(c,1),1/(k*u));
end
```

When Simulating this program, the average number of packets in the queue $E[n] = 2.97$ at any time, and the average delay $E[d] = 0.59$ minutes, for arrival rate $\lambda = 5$ packets per minute, departure rate $\mu = 6$ packets per minute, and k=4.

In Figure 5 we plot the probability distribution of the number of packets in the queue P_n and the number of packets n.

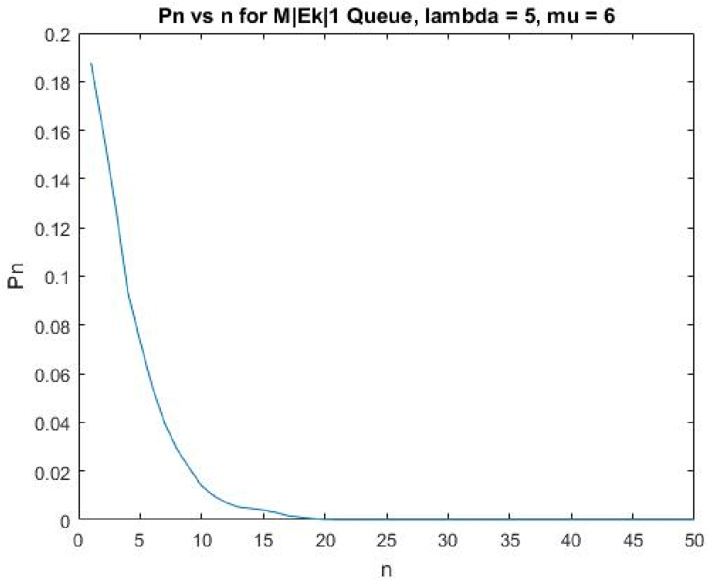

Figure 5: Pn vs n

When comparing the average number of packets E[n] in the queues, it's clear that this value is greater for the M|M|1 queue. To justify this we use the Pollaczek Khinchin formula for the mean queue length which states:

$$E[n] = \frac{\rho}{1-\rho}[1 - \frac{\rho}{2}(1 - \mu^2\sigma^2)]$$

where $\sigma^2 = \frac{1}{\mu^2}$ for the M|M|1 queue and $\sigma^2 = \frac{1}{k\mu^2}$ for the M|E_k|1 queue. Therefore the expected queue length for both queues becomes:

$$M|M|1 : E[n] = \frac{\rho}{1-\rho}$$

$$M|E_k|1 : E[n] = \frac{\rho}{1-\rho}[1 - \frac{\rho}{2}(1 - \frac{1}{k})]$$

Since $\rho \leq 1$ and $k > 1$, from the above equations, we can see that the expected queue length for the M|E_k|1 queue is less than that of the M|M|1 queue.

4.1 Comparison of an M|E_k|1 and an M|D|1 queue

In theory we learnt that an M|D|1 system provides a lower bound on congestion for a system with Poisson arrivals (lecture notes, Shroff). When calculating the expected queue length for an M|E_k|1 system it was noticed that as k increases the expected queue length decreases until it reaches a lower bound that is similar to that of an M|D|1 system.

To see this, we can compare the mean queue length of both systems again using the Pollaczek Khinchin formula. For a deterministic service time the variance $\sigma^2 = 0$. The mean queue length for both is :

$$M|E_k|1 : E[n] = \frac{\rho}{1-\rho}[1 - \frac{\rho}{2}(1 - \frac{1}{k})]$$

$$M|D|1 : E[n] = \frac{\rho}{1-\rho}[1 - \frac{\rho}{2}]$$

We notice that as $k \to \infty$, the mean queue length of both becomes equal.

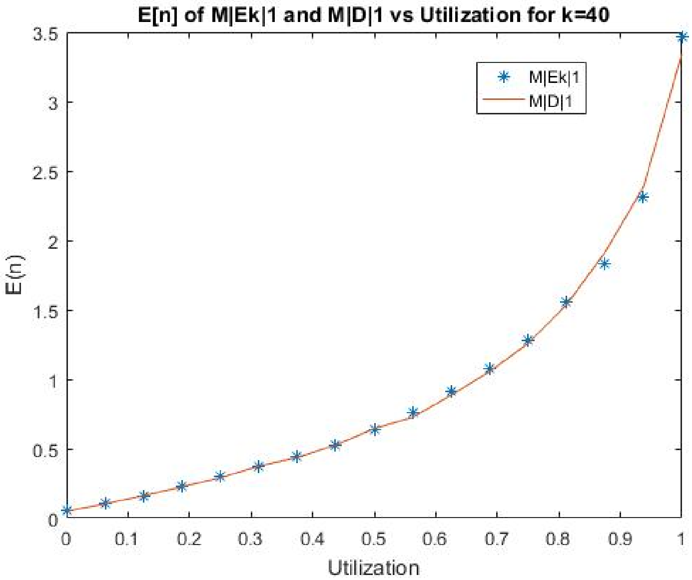

Figure 6: Expected number for M|E_k|1 and M|D|1 system with k=40

Above in Figure 6 we simulate the expected queue length for different values for the utilization ρ for an M|E_k|1 and an M|D|1 system, with $k = 40$ and show that the previous argument holds.