ECE610 BROADBAND NETWORKS STATISTICAL SIMULATION OF QUEUES

ZHIRONG LI, UW#20156684

Abstract. This report is dedicated to demonstrating the simulation steps and corresponding simulation results in comparison with results obtained in class under various queueing models.

Contents

List of Figures

Date: Feb.28, 2006.

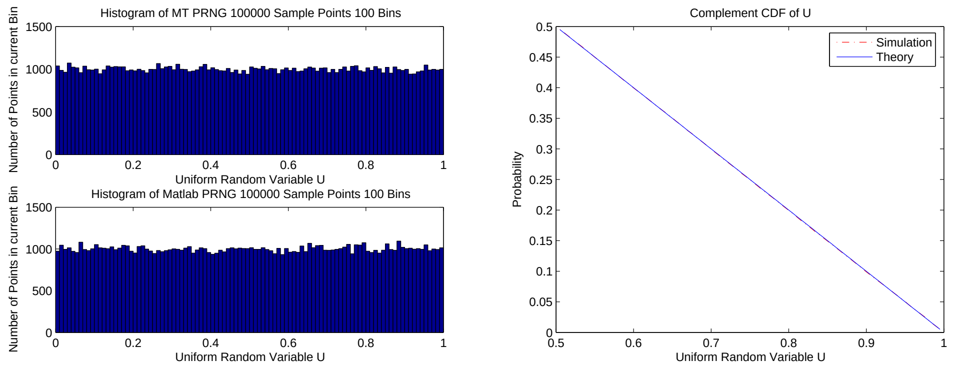

Figure 1.1. Histogram of PRNG

1. Introduction to Pseudo Random Number Generator (PRNG)

PRNG plays a very import role in statistical simulations. Arbitrary random variable with given distribution can be generated using PRNG. Many active research has been done to generate a good shaped random number and extensive randomness tests have been developed to test the performance of PRNG. Mersenne Twister[1] random number generator was proposed by two Japanese mathematicians in an ACM journal paper , which has been proved to be an efficient PRNG with period of $2^{19937} - 1$and with 623-dimensional equidistribution property. This MT-PRNG is often referred as "Many specialists in the field of random number generation consider MT as the current champion, ie., the best RNG for stochastic simulation"[2].

I modified the state-of-the-art C-version of MT PRNG to Matlab version MT PRNG in my simulations.[3]In order to test and compare the performance between MT PRNG and Matlab built-in PRNG function $rand()$, generate $100,000$samples using these two methods and plotted histograms as shown in Figure1. Both random number generator works well intuitively.I checked the course notes [4]and Kumar's book [6] and queueing theory [7] for theoretical results of the queueing model. Some simulation techniques are referred to Sheldon M Ross's book on simulation. [5]

2. Random Number Generation With Given Distributions

Arbitrarily distributed random variable can be generated using Uniform distributed random variable. Assume $CDF_X(X \leq a) = F$ and let $X = F^{-1}(U)$. By definition of the Cumulative Distribution Function and characteristics of uniform distribution , we get $CDF_X(X \leq a) = Pr\left(F^{-1}(U) \leq a\right) = Pr\left(U \leq F(a)\right) = F(a)$ [4], which implies that $X = F^{-1}(U)$ follows distribution with desired CDF. Hence $X = -\frac{1}{\lambda}ln\left(1-U\right)$ follows exponential distribution with mean of $\frac{1}{\lambda}$. Poisson random variable can be generated by searching the **look-up** table of CDF of Poisson distributed random variable: We generate a uniform random variable U and search the integer whose corresponding CDF is the right-nearest to U. It is a kind of inverse manipulation of CDF from $\mathbb{R}[0,1] \to \mathbb{N}$.

The CCDF exponential random variable and Poisson random variable are plotted shown in Figure 2.1

It is almost a perfect match to the CCDF in theory for both exponential and Poisson distributed random variables. Notice that for exponential distribution, the CCDF values deviate a little from theoretic values when X is large. It should be noted that the probability density almost reaches zero at this point and the deviation is partially due to precision limitation of matlab.

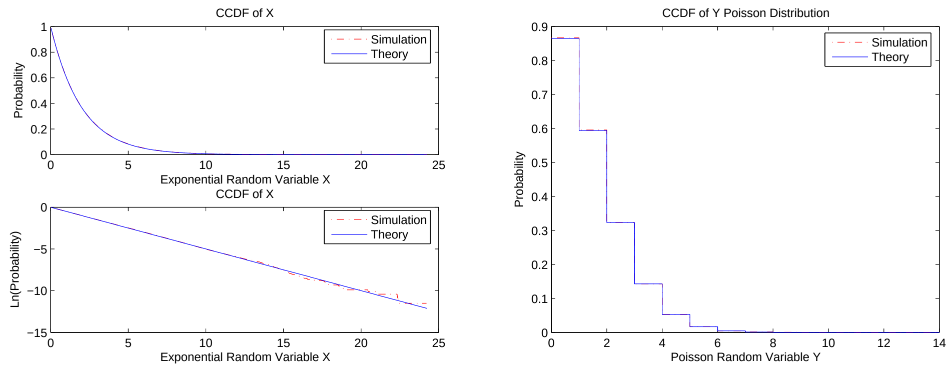

Figure 2.1. CCDF of X and Y

3. Statistical Simulation of Queues

I should claim that the server works in a **First Come First Serve**(FCFS) and in a **work-conserving** mode in the simulations.

I simulate the $M|M|1$ queueing system by considering the departures in each inter-arrival time. I constructed an array that stores the arrival index of the arrivals which are still in the system at each arrival time. In another word, the array is the queue seen by a newly arrived packet, not including itself. I generate the inter-arrival time and service time according to the requirement of the service system and we should notice that each packet's departure time is fully determined at its arrival time. It does not necessarily mean that the residual service time and waiting time are deterministic. From the simulation's point of view, the simulated process is always a realization (sample path) of the random process. We could associate each arrived packet with a random service time following the desired distribution. In this way, the newly arrived packet's departure time would be the packet's arrival time plus this random service time if the queue seen by this packet is empty and would be the immediately previous packet's departure time plus the service time otherwise. We assume the server works in a FCFS manner and in work-conserving mode. Once we have the recursive relation between departure times, the queueing system is easily simulated by updating the queue in each inter-arrival time. We want to make 100% sure that the queue seen by an newly arrived packet is the queue seen by this packet at its arrival time in reality. Assume t_{Dn},$\sigma_n$and t_{An}denote the departure time , service time and arrival time of this n-th arrival respectively. We have

$$t_{Dn} = \begin{cases} t_{An} + \sigma_n & if\ queue\ empty \\ t_{Dn-1} + \sigma_n & othewise \end{cases}$$

The first customer's departure time would be the summation of its arrival time and service time. The forthcoming packets may suffer delay in the queue. In simulation, I constructed an array which stores the arrival time, service time and departure time accordingly. Once this information is known and should be known when the simulation is done, the queue length at any time t is simply derived by examining if t falls within the life time of each packet and summation over these packets satisfying above condition.

We have 3 types of averages: mean queue length seen by arrival, mean queue length seen by departure and time average of the queue lengths. The first two average is event average and by PASTA, if the arrival process is Poisson, the event average should be equal to the time average of the queue length. Arrival average can be derived by examining the packets arrived before the packet under consideration and check if their departure times are greater than current packet's arrival time. Symmetrically, departure average can be derived by examining the packets arrived after the packet of interest and check if their arrival times are smaller than the current packet of interest's departure time.

In M|G|1 system, Polleczek-Khinchine Formula can be represented in Laplace-Stieljes form. If arrival is Poisson, PASTA(Poisson Arrival see Time Average) tells the distribution of queue length seen by an arrival is the stationary distribution of queue thus the average number of customers in the system(not including arrival itself) is equal to the average number of customs in the system. Since the process is ergodic, that event average is also equal to the time average of customs in the system. ($E[L] = \sum_{n=1}^{\infty} n\pi_n = \lim_{t\to\infty} \frac{1}{t}\int_{t=0}^{\infty} Q(t)dt.$)

3.1. **M|M|1 Queueing Model.** In theory, $M|M|1$ queue with arrival rate $\lambda$and service rateμrespectively has the steady state queue length distribution with $\pi_n = (1-\rho)\rho^n$ where$\rho = \frac{\lambda}{\mu}$. The average queue length is $E[L] = \sum_{n=1}^{\inf} n\pi_n = \frac{\rho}{1-\rho}$or by Polleczek-Khinchine Formula we get $E[L] = \rho + \frac{\lambda^2 E[\sigma^2]}{2(1-\rho)} = \rho + \frac{\rho^2(C^2+1)}{2(1-\rho)} = \frac{\rho}{1-\rho}$. By Little's Formula, we have $E[L] = \lambda E[S]$where S is the average sojourn time. If we have $\lambda = 5, \mu = 6$, the mean length of the queue is $\frac{\rho}{1-\rho} = 5$and the mean sojourn time $E[S] = 1$second. The sojourn time S distribution in M|M|1 system can be derived as follows:

$S = \sum_{k=1}^{L_a+1} \sigma_k$ where $\sigma_k$is iid. exponentially distributed with parameter μ. The probability for

$$\Pr(S > t) = \Pr(\sum_{k=1}^{L_a+1} \sigma_k > t) = \Pr(\Pr(\sum_{k=1}^{n+1} \sigma_k > t)) \Pr(L_a = n) = \exp(-\mu(1-\rho)t)$$

Thus the mean sojourn time is $\frac{1}{\mu(1-\rho)}$. The distribution of waiting time is $\Pr(W > t | W > 0) = \rho \exp(-\mu(1-\rho)t)$.

From my simulation shown in Figure5.1, both the time and event average are very close to the theoretical value which is 5 customers in the queue. The average sojourn time which is 1.0089 also verifies the Little's Formula!

3.2. $M|E_K|1$**Queueing Model.** In order to give a fair comparison with $M|M|1$model, we must have the mean service time equaled to $E[\sigma] = \frac{1}{\mu} = \frac{1}{6}$.Recall$E_k$distribution corresponds to the distribution of k-independent and identically distributed exponentially distributed random variables. Hence $\sigma = \sum_{i=1}^{k} \sigma_i$where each $\sigma_i$with mean$\frac{1}{k\mu}$ and variance$\frac{1}{(k\mu)^2}$ and the mean and variance of σwould be $\frac{1}{\mu}$and$\frac{1}{k\mu^2}$respectively. According to the Pollaczek-Khinchine formula, we have the $M|G|1$system with mean number of customers/packets equal to $E[L] = \rho + \frac{\rho^2(C^2+1)}{2(1-\rho)} = \rho + \frac{\rho^2}{2(1-\rho)}\frac{k+1}{k}$. If we have $k = 4$ in this simulation, then the expected number of customers in the queue is $\frac{5}{6} + \frac{(5/6)^2}{1/3}\frac{5}{4} = 3.4375$ and the mean sojourn time is$E[S] = \frac{E[L]}{\lambda} = 0.6875$ second. The simulation results are shown in Figure5.2. We should notice that the average number of customers in system $M|E_4|1$ is smaller than that in system $M|M|1$. This is because though these two systems have the same$\rho = \frac{\lambda}{\mu}$, the coefficients of variation C^2 of $E_k$is smaller than that of Exponential distribution and eventually $E_k$converges to deterministic distribution.

If we set $k = 40$, the simulation results are shown in the following Figure5.3.

$E[L] = \frac{5}{6} + \frac{(5/6)^2}{1/3}\frac{41}{40} = 2.96875$ and $E[S] = \frac{E[L]}{\lambda} = 0.59375$

In a general M|G|1 system, the Laplace-Stieljes transform of queue length at departure time is $P_L^d(z) = \frac{(1-\rho)P_A(z)(1-z)}{P_A(z)-z}$ where $P_A(z)$can be derived wrt. Laplace-Stieljes transform of service time.

$$P_A(z) = \sum_{n=0}^{\infty} \int_{t=0}^{\infty} \frac{(\lambda t)^n}{n!} e^{-\lambda t} f_\sigma(t) dt Z^n = \int_{t=0}^{\infty} e^{-(\lambda - \lambda z)t} f_\sigma(t) dt = G(\lambda - \lambda z)$$

where $G(z)$is the Laplace-Stieljes transform of service time σ. Hence

$$P_L^d(z)=\frac{(1-\rho)G(\lambda-\lambda z)(1-z)}{G(\lambda-\lambda z)-z}$$

For Erlang-K distribution with mean $\frac{1}{\mu}$, the corresponding Laplace-Stieljes transform of Erlang-K distribution is $G(s)=\left(\frac{k\mu}{k\mu+s}\right)^k$, substitute $G(\lambda-\lambda z)$into $P_L^d(z)=\frac{(1-\rho)\left(\frac{k\mu}{k\mu+\lambda-\lambda z}\right)^k(1-z)}{\left(\frac{k\mu}{k\mu+\lambda-\lambda z}\right)^k}$. Inverse Laplace transform of this term we can derive the queue length distribution for $M|E_k|1$system. The distribution of the number of customers left behind upon departure of customers is equal to $\{d_n\}$thus

$$d_n=\int_{t=0}^{\infty}\frac{(\lambda t)^n}{n!}e^{-\lambda t}f_s(t)dt$$

hence $P_{L_d}(z)=S(\lambda-\lambda z)$and we get the Laplace-Stieljes transform of sojourn time is

$$S(s)=\frac{(1-\rho)G(s)s}{\lambda G(s)+s-\lambda}$$

which is another form of Polleczek-Khinchine formula. In our case, the Laplace-Stieljes transform of sojourn time is denoted as

$$S(s)=\frac{(1-\rho)\left(\frac{k\mu}{k\mu+s}\right)^k s}{\lambda\left(\frac{k\mu}{k\mu+s}\right)^k+s-\lambda}$$

3.3. $M|D|1$**Queueing Model.** We set $\sigma=\frac{1}{\mu}=\frac{1}{6}$and we have the following results:

In theory, the average number of customers and mean sojourn time is $E[L]=\rho+\frac{\rho^2(C^2+1)}{2(1-\rho)}=\frac{2\rho-\rho^2}{2(1-\rho)}=2.9167$and $E[S]=\frac{2.9167}{5}=0.5833$. Actually the Erlang-$k$ distribution when $k\rightarrow\inf$ will converge to deterministic distribution with the same first moment. Proof is shown as follows:

Since$\sigma=\sum_{i=1}^{k}\sigma_i$, the MGF of MGF_σ is $MGF_\sigma(s)=\left[\frac{\mu k}{s+\mu k}\right]^k\rightarrow e_{k\rightarrow\infty}^{-\frac{s}{\mu}}$ which implies a deterministic distribution with mean value of $\frac{1}{\mu}$. Hence the $M|D|1$ Queueing System is shown as bellows (Figure5.4) to compare with the $M|E_{40}|1$system.

$$P_L^d(z)=\frac{(1-\rho)e^{-\frac{\lambda-\lambda z}{\mu}}(1-z)}{e^{-\frac{\lambda-\lambda z}{\mu}}-z}$$

It is obviously seen that $M|D|1$ and $M|E_{40}|1$ have similar distributions, which justifies the above conclusion. Besides, the $M|D|1$system has the smallest waiting time among $M|E_k|1$. From the mean-value approach, the mean waiting time $E[W]=\frac{\rho E[R]}{1-\rho}=\frac{5/6*E[R]}{1/6}=5/12$. By Little's law, the $E[L]=\lambda[E[W]+\frac{1}{\mu}]=5[5/12+1/6]=35/12=2.9167$ which is the same to the results derived from Polleczek-Khinchine formula.

3.4. $GI|M|1$**Queueing Model**[7]. Let GI process to be random process with $E_k$distribution. We investigate queueing model when $k=1,4,25$ respectively and $\lambda=\frac{1}{5},\mu=\frac{1}{3.5}$.

G|M|1 system queue length distribution seen by arrival/departure is not equal to the stationary distribution of the queue length since arrival is not Poisson any more. The queue length distribution seen by an arrival actually is quite similar to M|M|1 but with different parameters. According to the embedded Markovian chain upon arrival, the queue length distribution is $P_L^a(z)=\sum_{n=0}^{\infty}\sigma^n(1-\rho)z^n$where σcan be derived from this equation $\sigma=A(\mu-\mu\sigma)$and $A(S)$is the moment generating function of inter-arrival time.

The distribution of sojourn time for G|M|1 system is$S(s) = E[e^{-}] = \sum_{n=0}^{\infty} \sigma^n (1-\sigma) \left(\frac{\mu}{\mu+s}\right)^{n+1} = \frac{\mu(1-\sigma)}{\mu(1-\sigma)+s}$ which implies the sojourn time distribution is exponential with parameter $\mu(1-\sigma)$.

3.4.1. $E_1|M|1$ *Model shown in Figure* 5.5. This model is M|M|1 model in essence. The mean number of customers in the system is $E[L] = \frac{\rho}{1-\rho} = 2.33$ in theory, the average number of customers seen by arrival is 2.2488 and time average of customers is 2.2862 which is very close.

3.4.2. $E_4|M|1$ *Model shown in Figure* 5.6. The average number of customers seen by arrival is 1.2192 and time average of customers is 1.5422 which has a larger variation. The probability the queue is not empty seen by an arrival σ is derived from $\sigma = \left[\frac{\lambda k}{\mu-\mu\sigma+\lambda k}\right]^k$, solve this equation in interval $(0,1)$and let $k=4$we get $\sigma = 0.5529$, thus the mean number of customers seen by an arrival is $E[L^a] = \frac{\sigma}{1-\sigma} = 1.2366$ which is very close to our simulation results. $a_0 = 1-\sigma = 0.4471$ and our simulation gives0.45, the same! The expected sojourn time of an arbitrary packet is $\frac{1}{\mu(1-\sigma)} = 7.828$ secs and our simulation gives 7.7411secs. By Little's formula, $E[L] = \lambda E[S]$which implies the average number of customers in the system is $E[L] = \frac{1}{5}7.828 = 1.5656$ and our simulation gives 1.5422. The simulation results are very close to theoretical values which shows the validity of our simulation.

3.4.3. $E_{25}|M|1$ *Model shown in Figure* 5.7. The average number of customers seen by arrival is 0.88696 and time average of customers is 1.3111 which has a significant variation. If the arrival process is not Poisson, the timing average $E[L]$is not equal to the event average$E_A[L]$or$E_D[L]$ which implies PASTA does not hold any more. The theoretical values are derived from $\sigma = \left[\frac{\lambda k}{\mu-\mu\sigma+\lambda k}\right]^k$, we get $\sigma = 0.4828$ and we get $\sigma_{sim} = 1 - a_0 = 0.48$, almost the same results again!

3.5. $D|M|1$ **Queueing Model.** Let the arrival process be a deterministic process with inter-arrival time $\frac{1}{\lambda}$. Note that the $D|M|1$system (Shown in Figure 5.8) has the smallest waiting time among all $E_k|M|1$systems. In theory $\sigma = e^{-\frac{\mu-\mu\sigma}{\lambda}}$which immediately gives $\sigma = 0.4670$and our simulation gives $\sigma_{sim} = 1-a_0 = 1-0.53 = 0.47$.

4. Conclusions

The simulation results for various types of queueing models are shown as follows. The simulation results are very close to the theoretical values which demonstrates the correctness of our simulations. $E_k$distribution when k increases, it asymptotically approximates deterministic distribution which can be verified by the simulation results between $D|M|1$and $E_k|M|1$, $M|D|1$and$M|E_k|1$. When the arrival process is not Poisson distributed, the time average is not equal to event average any more. For $M|G|1$systems, we could use Pollaczek-Khinchine formula and Little's Formula to determine the mean customers in the queue and mean sojourn time and the simulation results justified the formulae well above.

5. Some Core Simulation Code

```
Stopping Time=10000;
lambda=5;
mu=6;
AccArrivals=0;
TimeElapsed=0;
QueueLenAtArrival=0;
UniRV=MTRNG;
```

```
InterArrival=-1/lambda*log(1-UniRV);
UniRV=MTRNG;
ServiceTime=-1/mu*log(1-UniRV);
while (TimeElapsed<StoppingTime)

    AccArrivals=AccArrivals+1;


    CustomerRec(AccArrivals,1)=TimeElapsed+InterArrival;%Arrival Time
    CustomerRec(AccArrivals,2)=ServiceTime;%service time

    if QueueLenAtArrival==0 %
        CustomerRec(AccArrivals,3)=CustomerRec(AccArrivals,1)+ServiceTime;
    else
        CustomerRec(AccArrivals,3)=CustomerRec(AccArrivals-1,3)+ServiceTime;
    end

    if QueueLenAtArrival==0 % Len Seen By an arrival
        QueueAtArrival(1)=AccArrivals;
        QueueLenAtArrival=1;
    else
        QueueAtArrival(QueueLenAtArrival+1)=AccArrivals;
        QueueLenAtArrival=QueueLenAtArrival+1;
    end

    %Calculate departures in this interarrival time
    TimeElapsed=TimeElapsed+InterArrival;

    UniRV=MTRNG;
    InterArrival=-1/lambda*log(1-UniRV);
    UniRV=MTRNG;
    ServiceTime=-1/mu*log(1-UniRV);

    MaxDeparturedIndex=max(find(CustomerRec(QueueAtArrival(1:QueueLenAtArrival),3)
<=(TimeElapsed+InterArrival)));
    if size(MaxDeparturedIndex,1)~=0 % Departure Exists
        QueueAtArrival(1:QueueLenAtArrival-MaxDeparturedIndex)=
QueueAtArrival(MaxDeparturedIndex+1:QueueLenAtArrival);
        QueueLenAtArrival=QueueLenAtArrival-MaxDeparturedIndex;
    end

end
%Calculate Qlength Distributions (in steady state) Event Average By
%Arrival/departure and Time Average
SteadyShift=ceil(AccArrivals/10); %Assume steady state starts from 10%
QA=zeros(1,AccArrivals-SteadyShift+1); %Event Average By arrival
```

```
for ArrivalIndx=SteadyShift:AccArrivals
    ArrivalTime=CustomerRec(ArrivalIndx,1);
    QA(ArrivalIndx-SteadyShift+1)=size(find(CustomerRec(1:ArrivalIndx-1,3)>ArrivalTime),1);
end
MaxQlen=max(QA);
PrAQn=zeros(1,MaxQlen+1);
for loopindx=1:AccArrivals-SteadyShift+1
   PrAQn(QA(loopindx)+1)=PrAQn(QA(loopindx)+1)+1;
end
PrAQn=PrAQn/(AccArrivals-SteadyShift+1);
EQA=sum(PrAQn.*(0:MaxQlen));
QD=zeros(1,AccArrivals-SteadyShift+1); %Event Average By Departure
for DepartureIndx=SteadyShift:AccArrivals
    DepartureTime=CustomerRec(DepartureIndx,3);
    QD(DepartureIndx-SteadyShift+1)=size(find(CustomerRec(DepartureIndx+1:AccArrivals,1)
<DepartureTime),1);
end
MaxQlen=max(QD);
PrDQn=zeros(1,MaxQlen+1);
for loopindx=1:AccArrivals-SteadyShift+1
   PrDQn(QD(loopindx)+1)=PrDQn(QD(loopindx)+1)+1;
end
PrDQn=PrDQn/(AccArrivals-SteadyShift+1);
EQD=sum(PrDQn.*(0:MaxQlen));
%Calculate Time Average
Q=zeros(3,2*AccArrivals);
Q(1,:)=[CustomerRec(:,1).',CustomerRec(:,3).'];
Q(1,:)=sort(Q(1,:));
Q(3,1)=1;
for loop=1:2*AccArrivals
    pos=find(Q(1,loop)==CustomerRec(:,1));
    if size(pos,1)~=0
       Q(2,loop)=1;
    else
       Q(2,loop)=-1;
    end
end
for loop=2:2*AccArrivals
    Q(3,loop)=Q(3,loop-1)+Q(2,loop);
end
Area=0;
for loop=1:2*AccArrivals-1
    Area=Area+Q(3,loop)*(Q(1,loop+1)-Q(1,loop));
end
EQ=Area/Q(1,2*AccArrivals);
```

References

[1] "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," Makoto Matsumoto et al., ACM Transactions on Modelling and Computer Simulations: Special Issue on Uniform Random Number Generation.1998
[2] HTTP://random.mat.sbg.ac.at/software/
[3] http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/MT2002/CODES/mt19937ar.c
[4] ECE610Course Notes by Prof. Mazumdar
[5] Sheldon M. Ross: Stochastic Processes, Wiley Series in Probability and Mathematical Statistics. ISBN 0-471-09942-2
[6] A. Kumar, D. Manjunath, and J. Kuri: Communication Networking: An analytical approach, Morgan-Kaufman (Elsevier), 2004, ISBN $0-12-428751-4$
[7] Ivo Adan and Jacques Resing: Queueing Theory, February 14,2001

E-mail address: `z32li@engmail.uwaterloo.ca`

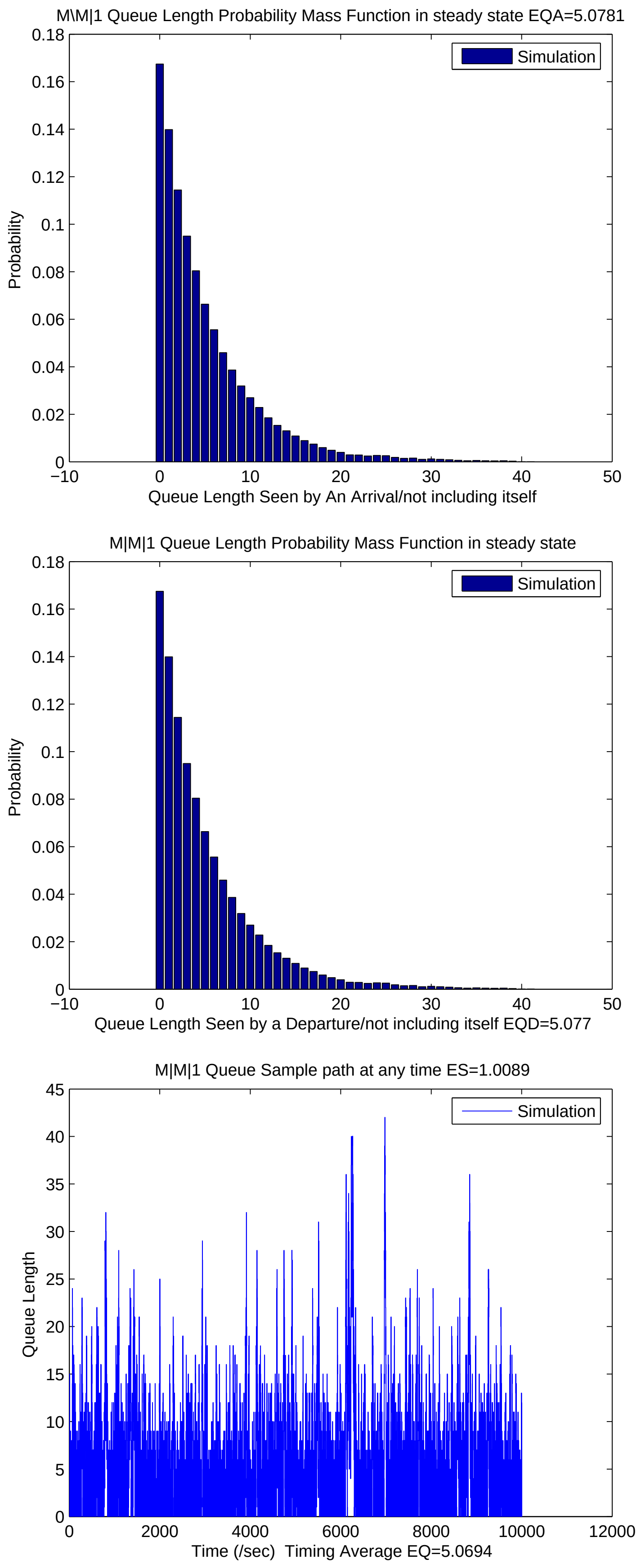

FIGURE 5.1. M|M|1 Queueing

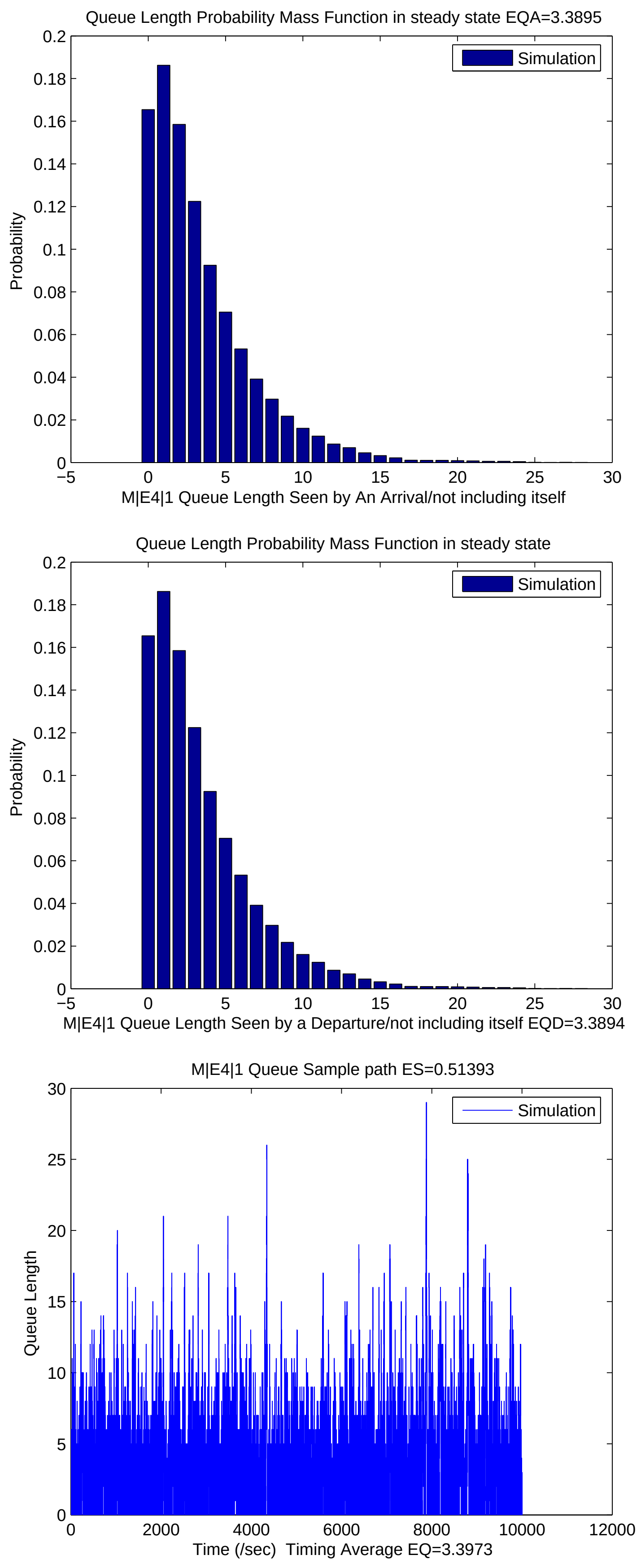

FIGURE 5.2. $M|E_4|1$Queueing Model

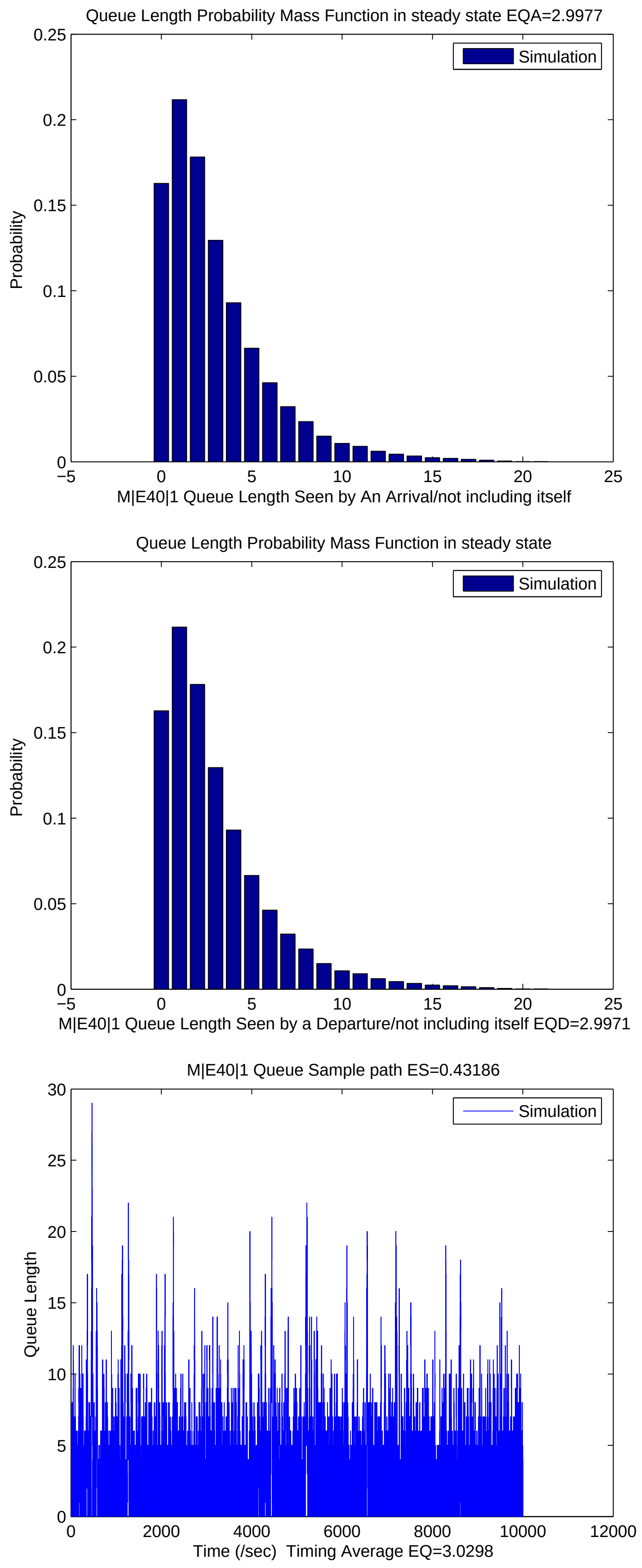

FIGURE 5.3. $M|E_{40}|1$Queueing Model

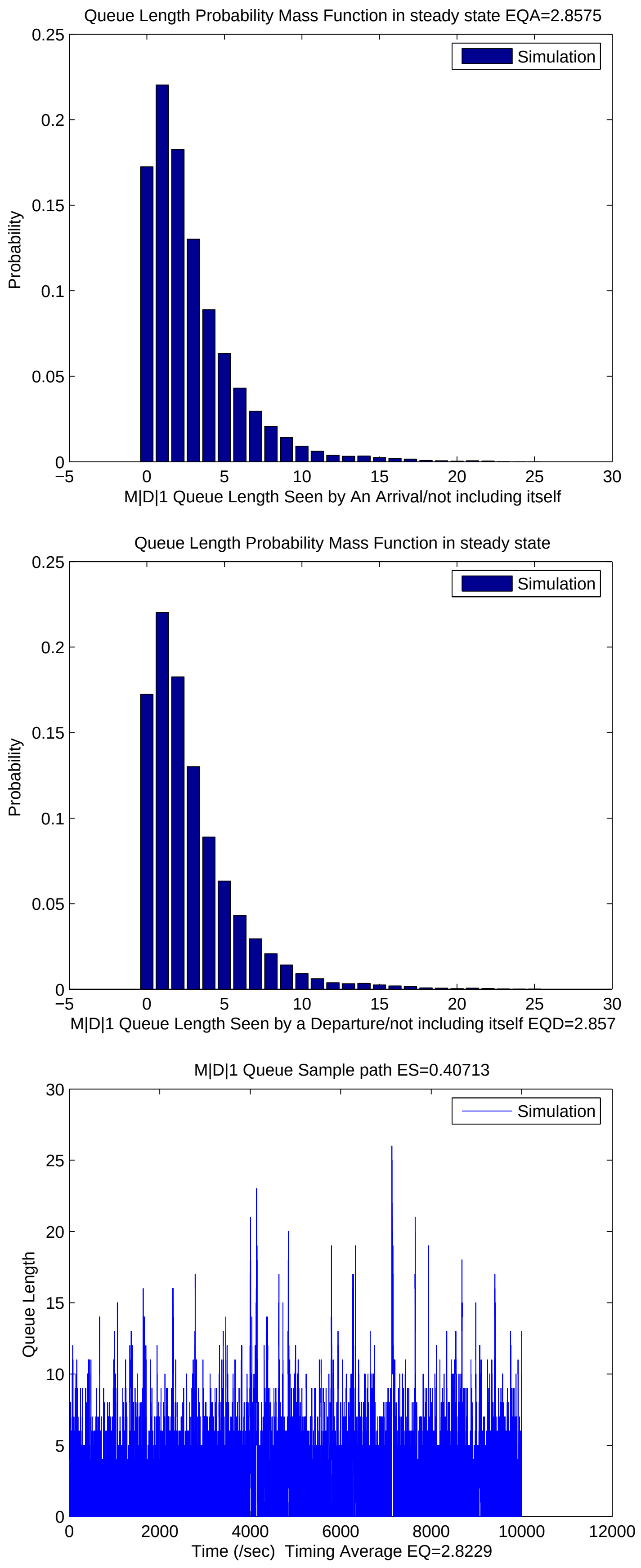

FIGURE 5.4. M|D|1 Queueing Model

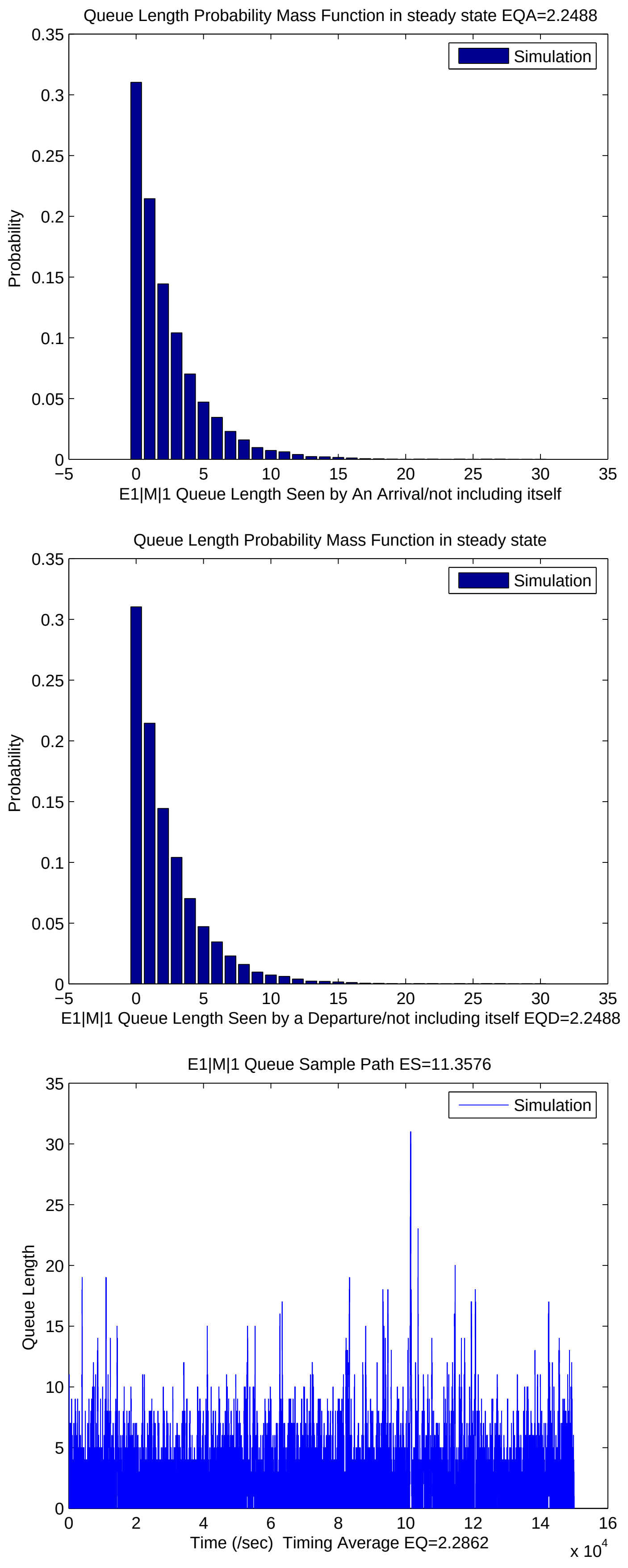

FIGURE 5.5. $E_1|M|1$Queueing Model

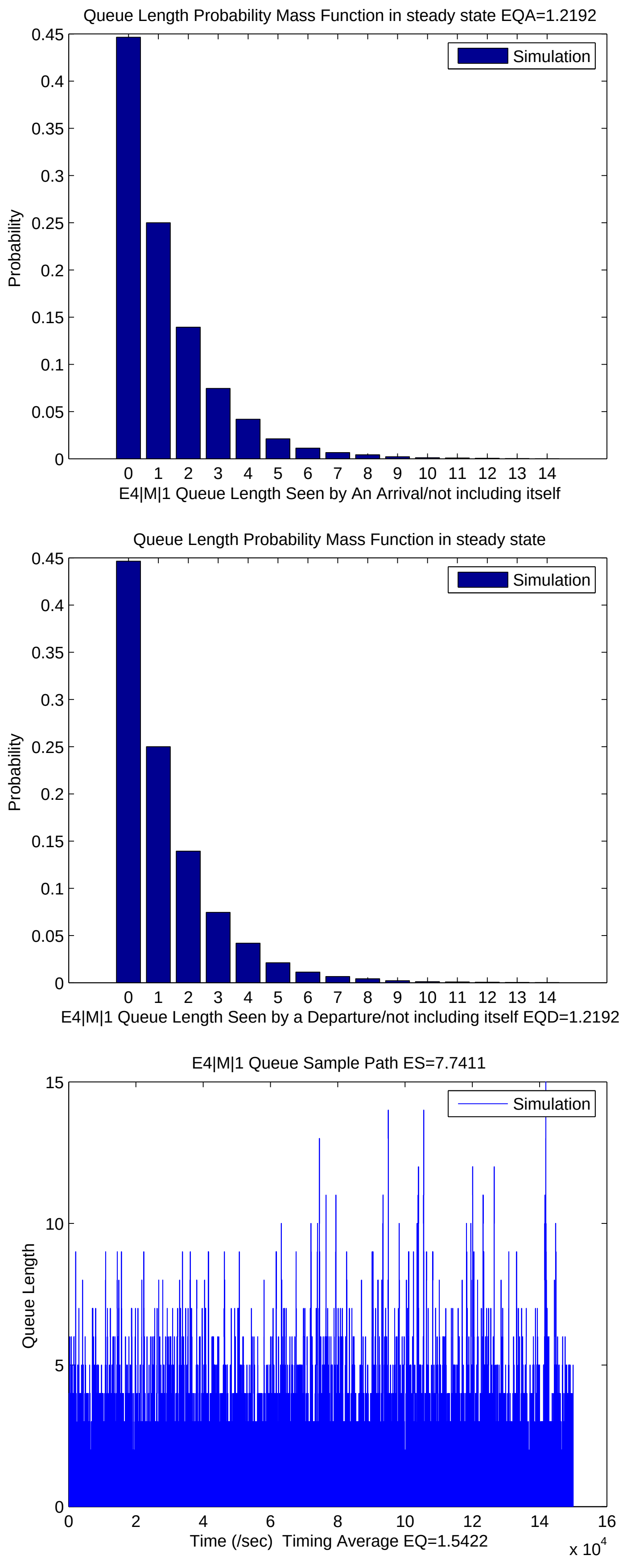

FIGURE 5.6. $E_4|M|1$Queueing Model

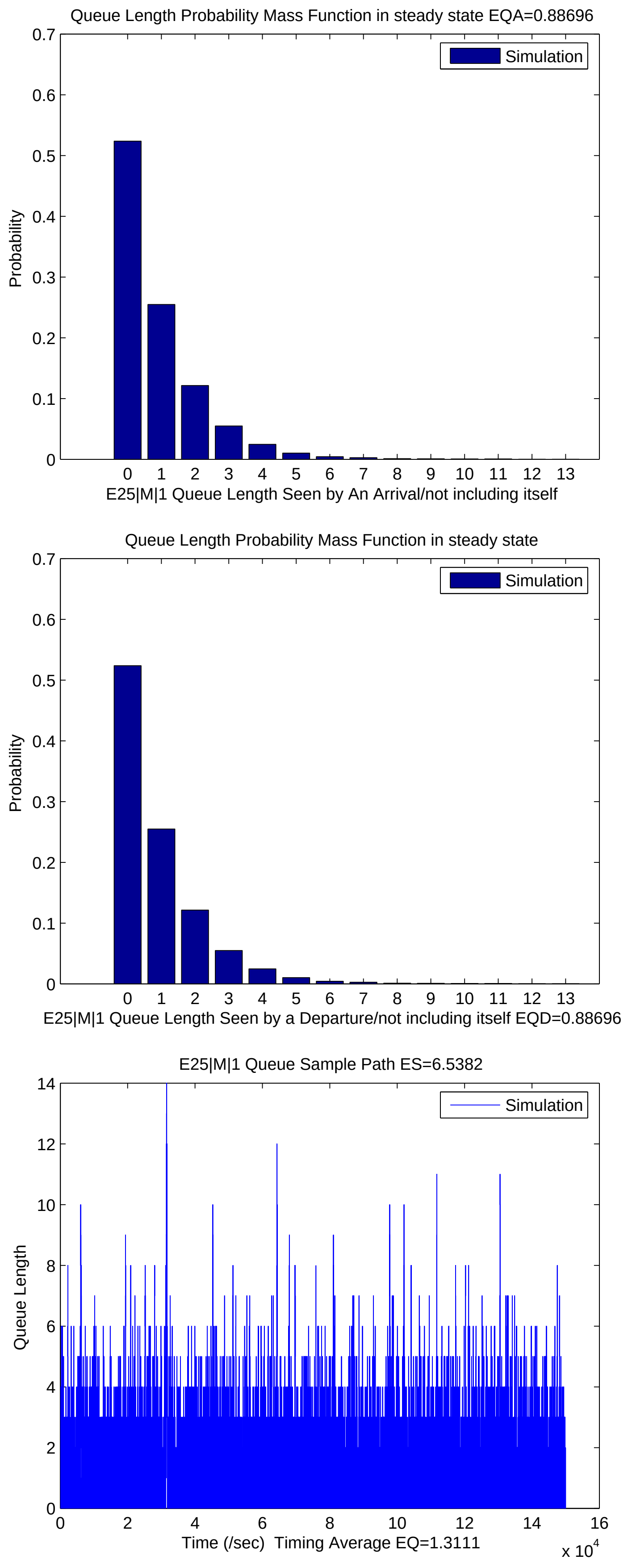

FIGURE 5.7. $E_{25}|M|1$Queueing Model

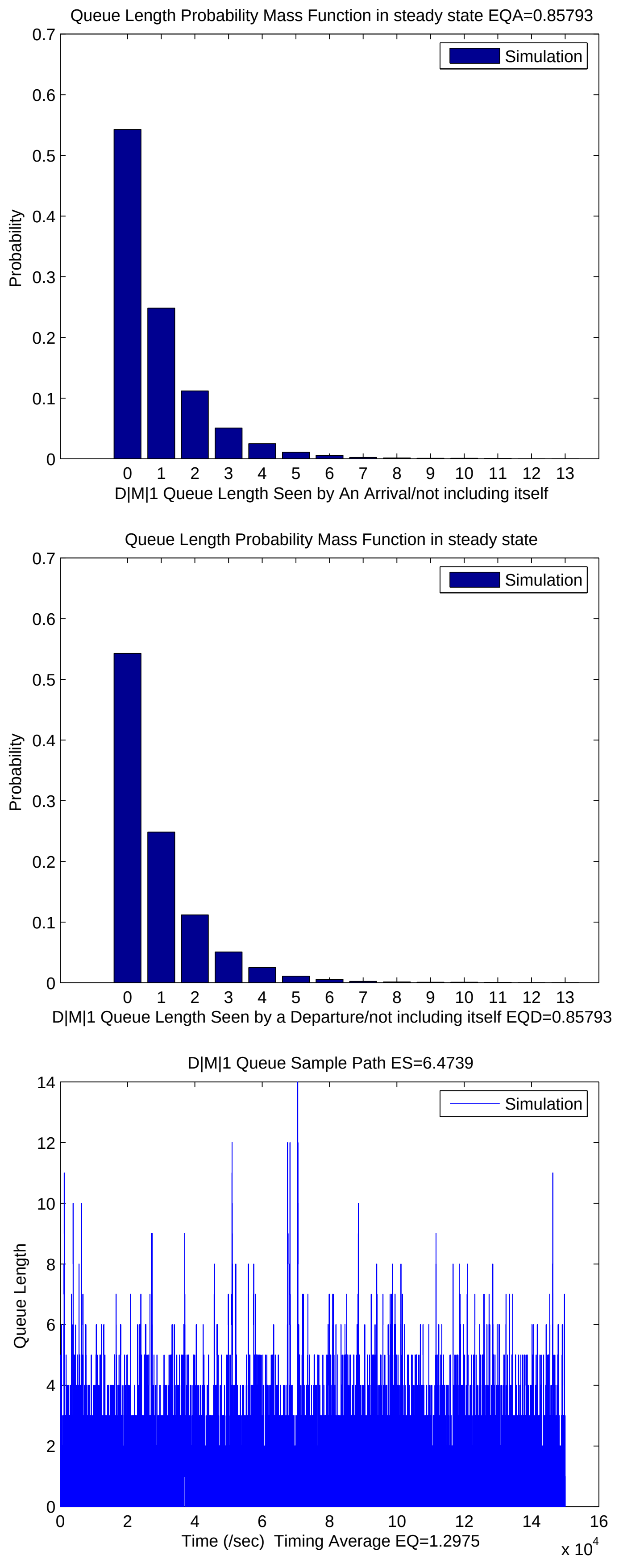

FIGURE 5.8. D|M|1 Queueing Model