**CSC 777- Telecommunication Network Design**

**Professor: Rudra Dutta**

**12/1/2013**

**Individual Study Topic Tutorial:**

* **Linear Arrival Rate Approximations**
* **Balking Systems**

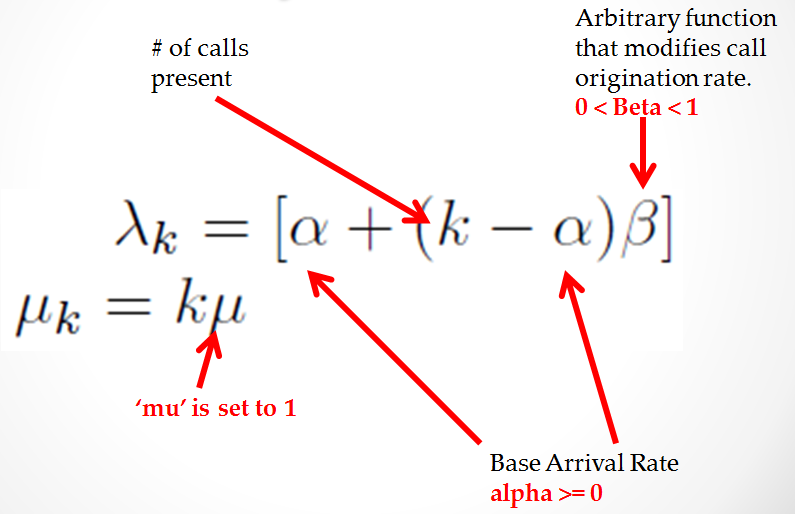
**By:**

**Nikhil Khatu**

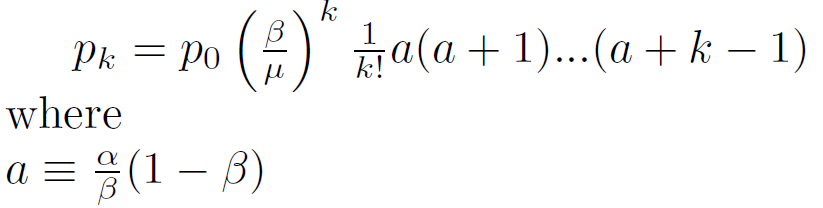
**Rushil Chugh**

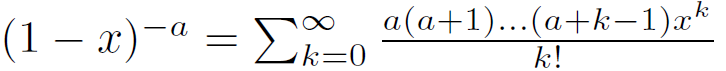
**Linear Arrival Rate Approximation**

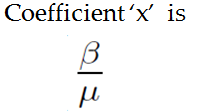
* **“What would be the physical description of a cause system with a variance smaller or larger than the Poisson?” - Wilkinson**
* **Physical Description: “If the variance is smaller, there must be forces at work which retard the call arrival rate as the number of calls recently offered exceeds a normal, or average, figure, and which increase the arrival rate when the number recently arrived falls below the normal level. Conversely, the variance will exceed the Poisson’s should the tendencies of the forces be reversed.” -Wilkinson**

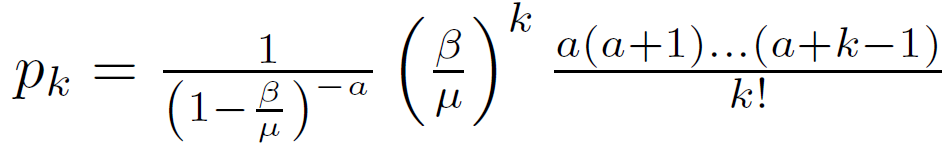


****

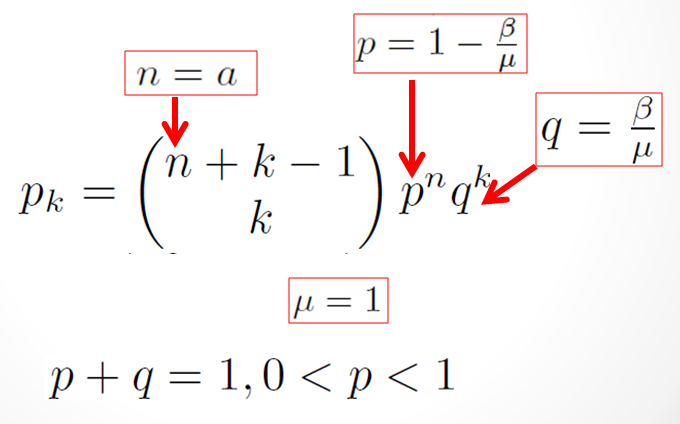
****

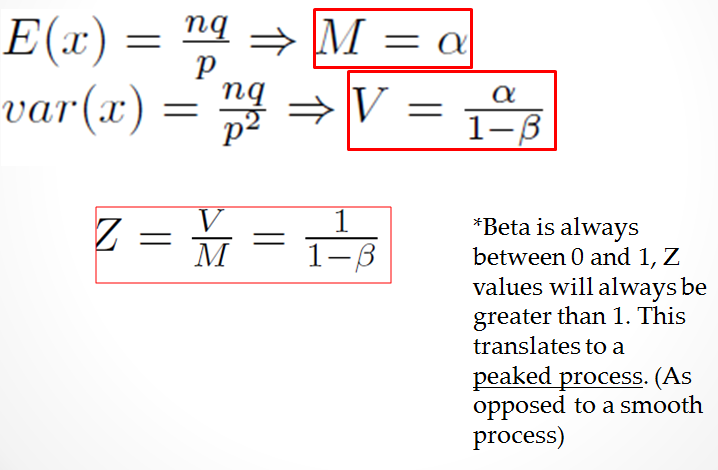
****



****



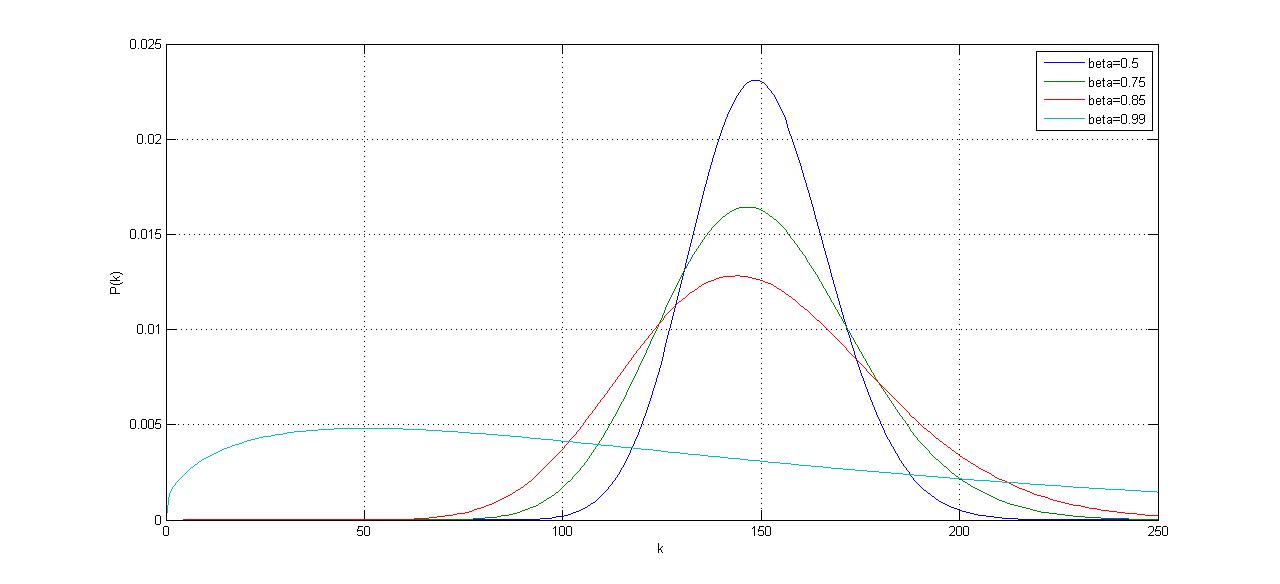


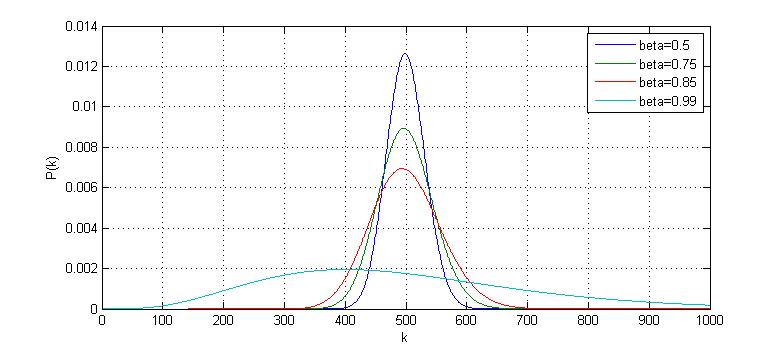


**Calculating the Pascal Distribution**

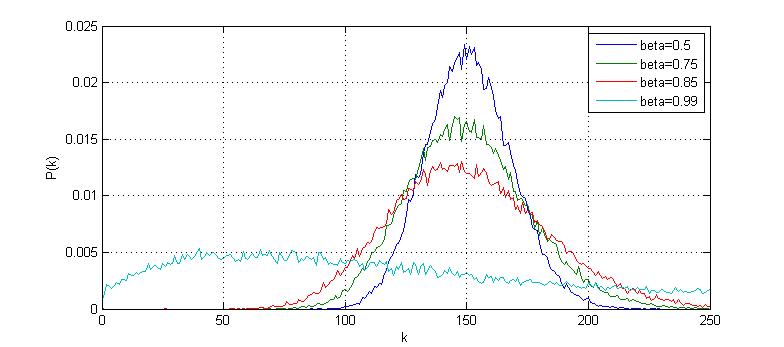
https://github.ncsu.edu/ngkhatu/CSC777.git

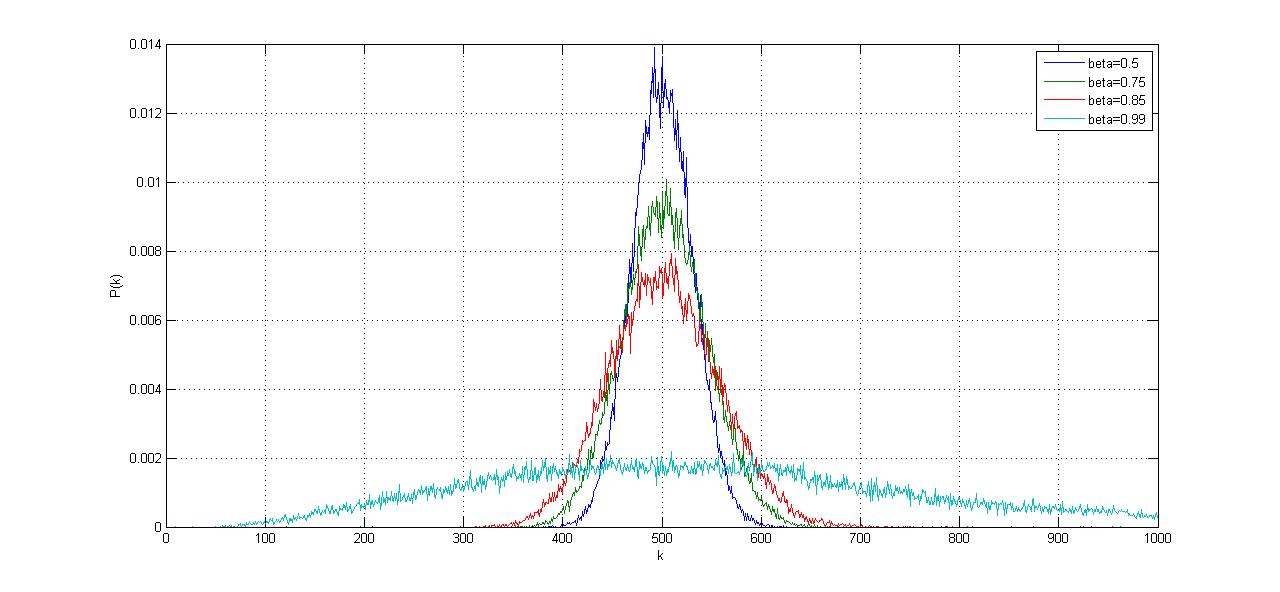
https://github.ncsu.edu/ngkhatu/CSC777/archive/master.zip

****

****

**Simulating with Linear Arrival Rates**

****

****

1. **Introduction**

A significant aspect in modeling call centers is customer impatience. Motivated by analyzing the call center operations, an M/M/s queuing system with impatient customers was considered. The customers arrive according to a Poisson process with rate λ, and request iid (independent and identically distributed) service times with an exponential distribution.

There are s ≥ 1 servers in the system available to serve the customers. All servers are identical and unit-rate, i.e., each server is capable of processing one unit of service requirement per unit time.

Two common modes in which customers display their impatience are balking and reneging. A call-in customer who cannot be helped immediately by a human server might be told how long a wait he/she faces before an operator is available. Then the customer might hang up (i.e. balk) or decide to hold. This is an example of the balking behavior: a customer refuses to enter the queue if the wait is too long. On the other hand, a customer who is waiting for an operator might hang up (i.e. renege) before getting served if the wait in line becomes too long. This is the reneging behavior. Additionally, a combination of the two is possible.

Queuing models with balking incorporate the characteristics of the customers’ impatience or a specific admission control policy in force at a service system. In a typical queuing model with balking the service requirement of an arriving customer may not be (completely) accepted if the system is “too congested” at the time of its arrival. From the perspective of an arriving customer one natural measurement of system congestion is the queuing time he/she faces to get service started. A no-join decision based on this congestion measurement is the aforementioned wait-based balking. The model considered in this thesis uses a wait-based balking rule. Before stating the balking rule, we define the virtual queuing time (vqt) in the system. The vqt at time t in the system, denoted by W (t), is the queuing time (i.e., time spent in the system before commencing service) that would be experienced if a customer joins the system at time t. The process {W (t), t ≥ 0} is referred as vqt process. We call a queuing system with balking based on the vqt a wait-based balking queue. It works as follows. We assume that each customer knows his/her exact queuing time at the time of arrival. A customer arriving at time t joins the system if and only if W (t−) is no more than a pre-specified threshold (possibly random). The balking customers (i.e., customers who do not join) are lost forever. The entering customers wait in an infinite capacity FCFS (first-come, first-served) queue until a server is available and leave when the service completes.

Our justification of such a model proceeds as follows. Although the queueing time information in call centers is not precise, this model incorporates the right characteristics of the customer behavior. On the other hand, there are some systems, for instance, some communication systems, where the information about queueing time is always precisely available. What’s more, from the view of control problems, the balking rule can be regarded as a threshold type of customer acceptance/rejection 2 policy based on the system workload.

More complicated models involving balking and reneging behaviors can be found in the research with a focus on the comparison of performances under different operational characteristics presented in systems with customer impatience (cf. [1], [2],[18], [47]).

* 1. **Flow of a Typical Balking System**

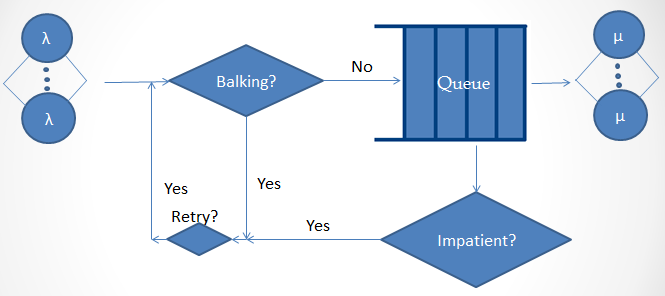
****

Figure 1: Flow of a typical balking system

The system assumes that calls arrive with an inter arrival rate of λ. The first flow decision is initially when the customer decides if he has to balk or not. If he decides to balk, the customer will enter the retrial queue and then retries with a retrial rate of α.

The second scenario is if the customer doesn’t balk and instead joins the queue, the customer might get impatient after some time and decides to renege. If the customer reneges, the customer will enter the retrial queue and then retries with a retrial rate of α.

The third scenario is when the customer doesn’t decide to balk or renege, but instead joins the queue and gets served with a service rate of μ.

An actual balking system can be one of the aforementioned scenarios or can be a combination of the three.

1. **Single Sever Queues**

Single server wait-based balking queues are studied under a variety of names in the literature: “finite workload capacity”, “dams”, “workload-dependent arrival rates”, or “queues with limited accessibility”. Notice for single server systems with a FCFS service discipline, the vqt coincides with the workload (work content) of the system, i.e., sum of the service times of all customs in queue and the remaining service time of the customer in service.

But the simulation and the presentation were done with respect to multi server queues, hence only an introduction was provided for the single server queues. The mathematical analysis and simulation results have been plotted for multi-server queues as shown below.

1. **Multi-Server Queues**

The reneging version of the model we consider has been studied by under the name “systems with limited waiting time”. They consider exponential service times to obtain a multidimensional Markov process for the number of busy servers and workload in each server. They derive a system of integro-differential equations for the limiting joint distribution and give explicit solution.

They give formulas for the loss probability and average queueing time. They also give the limiting distribution of the vqt process. However, as Boots and Tijms [7] noted, the results in [17] are quite technical and not generally applicable. Instead, they give an alternative formula for the loss probability as a function of the tail probability of the stationary vqt process in a corresponding queue with no impatience.

They prove that their formula is exact in the M/M/s case and can be used as a heuristic for the M/G/s case. A severe restriction is that the formula is valid only when the traffic intensity is less than 1, which is not required for the reneging queue to be stable. The method we use in this thesis overcomes the preceding drawbacks and can be easily extended to the general case. Although we are unable to give the joint distribution for the workload and busy servers, we don’t lose much since many common performance measures can be derived directly from the limiting distribution of the vqt process.

Even in the absence of the balking behavior the M/G/s queueing system is notorious for its complexity which forbids analytical solutions. Analytical results are available for only a few special cases, while a handful of approximations for the limiting analysis have been proposed in the past decades (cf. Chapter 13, Heyman and Sobel [19]). In this thesis we focus on system approximations, i.e. approximations that take the results from an exact analysis of a simpler system as approximations of the true operating characteristics of the original system. Although the approximate methods vary by motivations and the techniques used, it turns out that all results can be viewed as the so-called “systems interpolation”, i.e., some mixture of the known analytical results for a few special cases, such as M/M/s, M/Ek /s, M/D/s, and M/G/∞. See Kimura [26] for details. We cannot find any system approximations of the M/G/s queueing system with impatient customers in the literature.

To develop a system approximation for the multi-server system with impatient customers, we borrow a simple idea used by Lee and Longton [33], Tak ́acs [44] (page 160), Newell [36] (page 86), Hokstad [21], Nozaki and Ross [37], Tijms et al. [45], and Miyazawa [34]. In brief, the idea is to treat the s-server system as an M/G/∞ system (or M/G/s − 1 loss system) when some servers are idle and an M/G/1 system when all servers are busy. Using such a system decomposition we construct a single server system whose operating characteristics approximate those of the M/G/s queueing system with wait-based balking. The approximation is exact when G = M , the 8balking threshold is zero or s = 1. The exact analysis of the approximate system follows the same line as Chapter 2, where we solve the s = 1 case. The approximation is evaluated by comparing performance measures against simulations. The connection between wait-based balking and reneging is discussed in Chapter 4, which reveals that the analysis of the vqt process in this thesis is indeed a treatment to a queue incorporating customer impatience, whether balking or reneging.

* 1. **Derivation for Multi-Server Queues (M-M-s)**

Let *P,, (r)* denote the transient-state probability that there are n in the system. The differential difference equations of the system are as follows:

P0` (t) = - λ P0 (t) + μ P1 (t)

Pn` (t) = - (λ+nμ) Pn (t) + λ Pn-1 (t) + (n+1) μ Pn+1 (t) 1<=n<=c-1

Pc` (t) = - (λp+cμ) Pc (t) + λ Pc-1 (t) + (cμ+α) μ Pc+1 (t) n=c

Pn` (t) = - (λp+cμ+ (n-c) α) Pn (t) + λp Pn-1 (t) + (cμ + (n+1-c) α) Pn+1 (t) n>c

Assuming steady state distribution exists:

0 = - λ P0 (t) + μ P1 (t)

0 = - (λ+nμ) Pn (t) + λ Pn-1 (t) + (n+1) μ Pn+1 (t) ----------- (1) 1<=n<=c-1

0 = - (λp+cμ) Pc (t) + λ Pc-1 (t) + (cμ+α) μ Pc+1 (t) -------- (2) n=c

0 = - (λp+cμ+ (n-c) α) Pn (t) + λp Pn-1 (t) + (cμ + (n+1-c) α) Pn+1 (t) n>c

From (1) and (2), we get:

Pn= [(1/n!)(λ/μ) n] P0 n<=c

Putting Pc-1 and Pc and using BD process derivation equations:

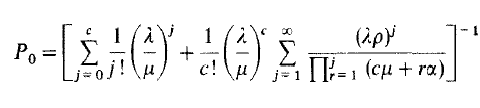
Pc+1= [(1/c!)(λ/μ)c (λp/cμ+α)]P0 n=c+1

Pc+2= [(1/c!)(λ/μ)c ((λp)2/(cμ+α)(cμ+2α)]P0 n=c+2

Generalizing, we get:

Pn= [(1/c!)(λ/μ) c [(λp) n-c] Po/] n>c

P0 can be obtained by applying the normalizing condition:



1. **Simulation Results**
   1. **Considerations for Simulation**

* Service rate was kept fixed at 1 customer per 1 ms.
* Number of buffers were fixed at 10
* Number of servers and arrival rate was varied to obtain respective response times.
* The customer balks with a probability of 0.2
  1. **Results of Simulation when Number of Servers =3**

|  |  |
| --- | --- |
| Response time vs Lambda | No. of servers = 3 |
| Response Time (ms) | Lambda (Customers per ms) |
| 0.275615 | 0.2 |
| 0.517 | 0.3 |
| 1.30433 | 0.4 |
| 1.51234 | 0.5 |
| 1.75072 | 0.6 |
| 2.16763 | 0.7 |
| 2.37624 | 0.8 |
| 2.45622 | 0.9 |
| 2.54721 | 1 |
| 2.64682 | 1.2 |
| 2.75313 | 1.4 |
| 2.83562 | 1.6 |
| 2.94453 | 1.8 |
| 3.008 | 2 |
|  |  |

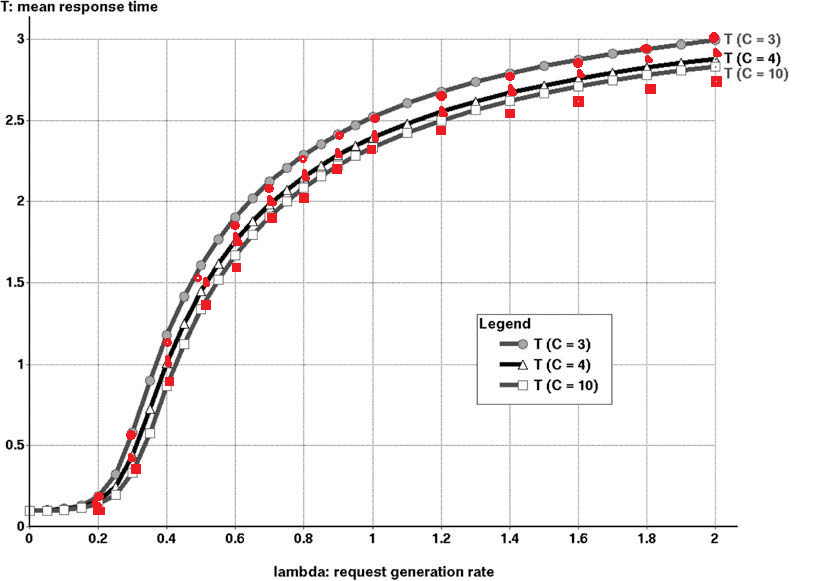
**4.3 Results of Simulation when Number of Servers =4**

|  |  |
| --- | --- |
| Response time vs Lambda | No. of servers = 4 |
| Response Time (ms) | Lambda (Customers per ms) |
| 0.24307 | 0.2 |
| 0.46532 | 0.3 |
| 1.03064 | 0.4 |
| 1.50169 | 0.5 |
| 1.55284 | 0.6 |
| 2.07193 | 0.7 |
| 2.18843 | 0.8 |
| 2.33869 | 0.9 |
| 2.40513 | 1 |
| 2.52783 | 1.2 |
| 2.64623 | 1.4 |
| 2.75441 | 1.6 |
| 2.84671 | 1.8 |
| 2.89864 | 2 |

**4.4 Results of Simulation when Number of Servers =10**

|  |  |
| --- | --- |
| Response time vs Lambda | No. of servers = 10 |
| Response Time (ms) | Lambda (Customers per ms) |
| 0.093043 | 0.2 |
| 0.365216 | 0.3 |
| 0.930796 | 0.4 |
| 1.351521 | 0.5 |
| 1.617698 | 0.6 |
| 1.864737 | 0.7 |
| 2.0790085 | 0.8 |
| 2.2217555 | 0.9 |
| 2.2848735 | 1 |
| 2.4014385 | 1.2 |
| 2.5139185 | 1.4 |
| 2.6166895 | 1.6 |
| 2.7043745 | 1.8 |
| 2.753708 | 2 |

**4.5 Expected vs Calculated Results for Simulation**



1. **Conclusion**