Simulation and Optimisation of Queueing Systems

Jimmy Lin[1] Robert Lambert[2]

[1]University of Glasgow

[2]Lancaster University

Motivation

Motivation

Figure 1: Tesco Queue

Why Queues?

Queueing systems are a fundamental part of our every day lives. For example, we see queues in:

Why Queues?

Queueing systems are a fundamental part of our every day lives. For example, we see queues in:

- Airports
- Retail
- Logistics
- Computing Systems

Why Queues?

Queueing systems are a fundamental part of our every day lives. For example, we see queues in:

- Airports
- Retail
- Logistics
- Computing Systems

Our goal is to be able to accurately model these queues, in hopes that we are able to optimise the system.

Why Simulation?

- The issue often with analytical mathematical modelling is that many assumptions are often made which are unrealistic for the real world.

Why Simulation?

- The issue often with analytical mathematical modelling is that many assumptions are often made which are unrealistic for the real world.
- As a result, we propose the usage of simulations as an approach to model queueing systems.

Queueing Theory Introduction

Distribution and Kendall's Notation

Within queueing systems, there are two key features that we are required to model.

Distribution and Kendall's Notation

Within queueing systems, there are two key features that we are required to model.

- Arrival Process: $X_t \sim \text{Poi}(\lambda)$
- Departure Process: $Y_t \sim \text{Exp}(\mu)$

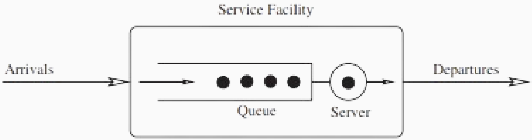
Distribution and Kendall's Notation

Within queueing systems, there are two key features that we are required to model.

- Arrival Process: $X_t \sim \text{Poi}(\lambda)$
- Departure Process: $Y_t \sim \text{Exp}(\mu)$

A key feature of these distributions are that they are memoryless processes.

Hence, a system with one server would be denoted using Kendall's notation as an M/M/1 queue.



Markov Chain

The state of the queues can also be modelled similar to that of a continuous time Markov Chain.

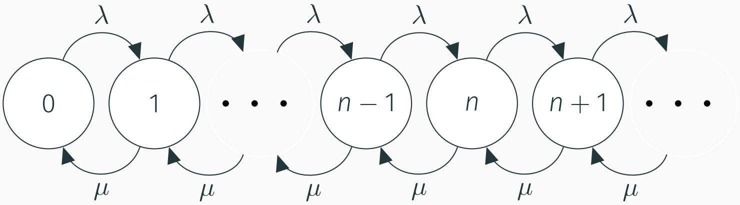


Figure 2: Markov Chain Respresentation of Queue States

M/M/1 Queue

Simulation of M/M/1 Queues

- The simulation of queues uses the interarrival time distribution to simulate discrete time points where events occur.

Simulation of M/M/1 Queues

- The simulation of queues uses the interarrival time distribution to simulate discrete time points where events occur.
- This interarrival time is distributed by an exponential distribution.

Simulation of M/M/1 Queues

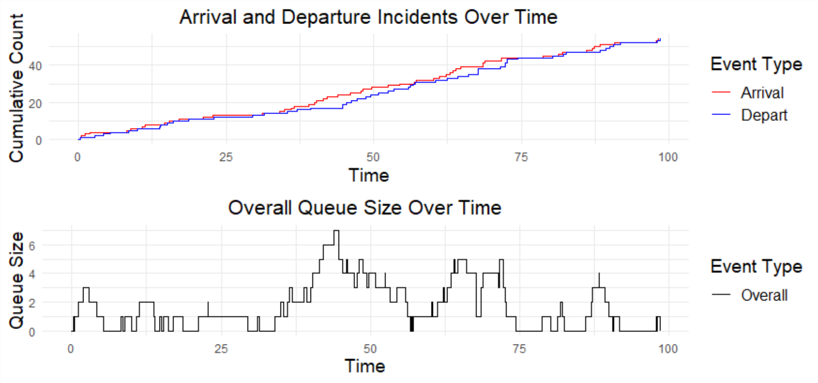


Figure 3: Queueing Simulation with $\lambda = 0.6$ and $\mu = 0.8$ up to $t = 100$

Steady State Solution

How do we know our simulation is working?

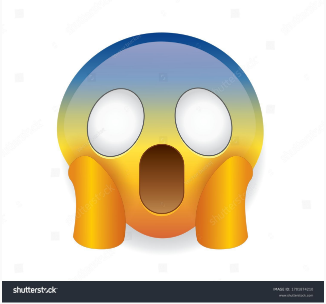
Steady State Solution

How do we know our simulation is working?

STEADY STATE SOLUTION!

How do we know our simulation is working?

STEADY STATE SOLUTION!



Steady State Solution

In an M/M/1 queue when $\lambda < \mu$, we have the closed form steady state solution:

$$p_n = (1 - \rho)\rho^n, \quad \rho = \frac{\lambda}{\mu}$$

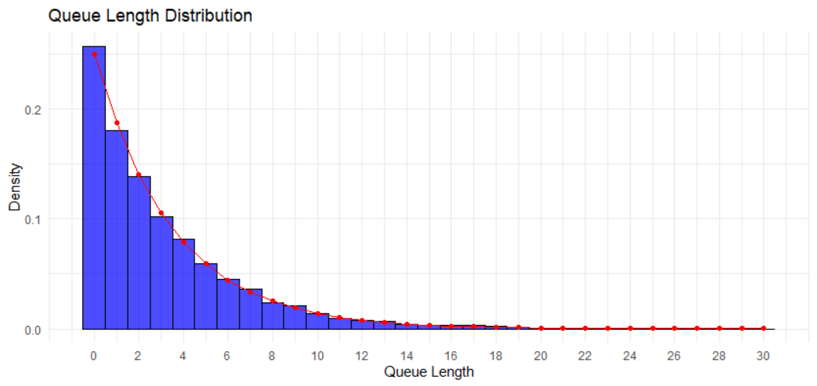


Figure 4: Density Histogram of 1000 Queue Simulations to Steady State Solution

Issues with M/M/1

M/M/1 is nice and basic but misses out key features which may occur in the real word, such as:

- Impatient Customers
- Scheduling Disciplines
- Group arrivals and departures
- and many more...

Issues with M/M/1

M/M/1 is nice and basic but misses out key features which may occur in the real word, such as:

- Impatient Customers
- Scheduling Disciplines
- Group arrivals and departures
- and many more...

Today, we will look at the issue of non-constant arrival rate.

M(t)/M/1 Queue

Inhomogeneous Poisson Process

Typically, it may be more realistic if λ is varying with time, i.e. $\lambda(t)$.
Today, we will explore $\lambda(t) = \sin(t) + 1$.

Inhomogeneous Poisson Process

Typically, it may be more realistic if λ is varying with time, i.e. $\lambda(t)$. Today, we will explore $\lambda(t) = \sin(t) + 1$.

What could be the issues with an inhomogeneous poisson process?

Inhomogeneous Poisson Process

Typically, it may be more realistic if λ is varying with time, i.e. $\lambda(t)$. Today, we will explore $\lambda(t) = \sin(t) + 1$.

What could be the issues with an inhomogeneous poisson process?

- Now the interarrival time will not necessarily be exponentially distributed. Could simulation be harder?
- With varying transitional properties, this makes the problem an inhomogeneous process, hence a steady state solution will likely not be available.

Thinning Algorithm

- We simulate a homogeneous queue with an arrival rate of $\bar{\lambda} = \sup_t \lambda(t)$.
- For each arrival, we want to accept the arrival with probability:

$$\mathbb{P}\{\text{Accept Arrival}\} = \frac{\lambda(t)}{\bar{\lambda}}$$

  where t represents the time in which the arrival at rate $\bar{\lambda}$ has occurred.

This is what is known as a thinning algorithm.

Thinning Algorithm

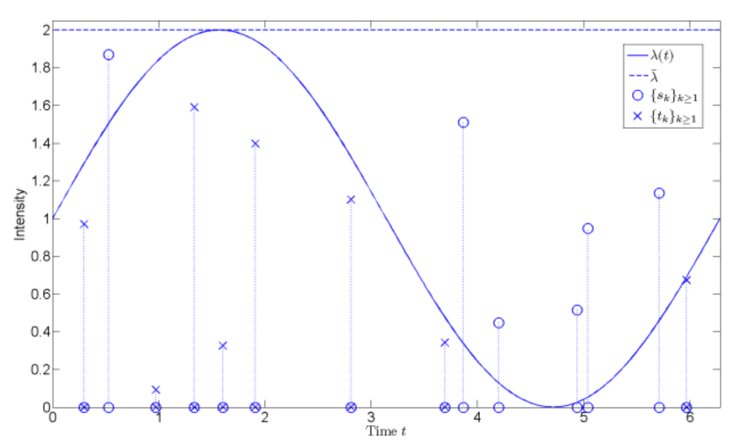


Figure 5: Thinning algorithm [Chen, 2016]

Inhomogeneous Poisson Process

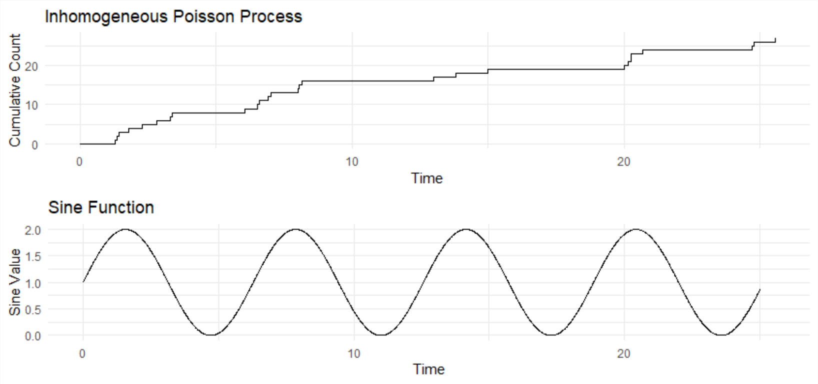


Figure 6: Inhomogeneous Poisson Process with $\lambda(t) = \sin(t) + 1$

Numerical Integration

Due to a lack of steady state solution, we use numerical integration for the comparison of our simulation.

Notation used:

$$p_n(t) = \mathbb{P}\{n \text{ in state at time } t\}$$

Numerical Integration

Choose a small time step d, and a large maximum queue capacity.

Numerical Integration

Choose a small time step d, and a large maximum queue capacity.

1. First, set the beginning of the queue where $p_0(0) = 1$ and $p_n(0) = 0$.

Numerical Integration

Choose a small time step d, and a large maximum queue capacity.

1. First, set the beginning of the queue where $p_0(0) = 1$ and $p_n(0) = 0$.
2. Apply classical queueing theory model for the next time step:

$$\begin{aligned} p_n(t+d) &= \lambda(t) d p_{n-1}(t) + [1 - \lambda(t) d - \mu d] p_n(t) + \mu d p_{n+1}(t) \\ p_0(t+d) &= [1 - \lambda(t) d] p_n(t) + \mu d p_{n+1}(t) \\ p_{n_{\max}}(t+d) &= \lambda(t) d p_{n_{\max}-1} + [1 - \mu d] p_{n_{\max}} \end{aligned} \tag{1}$$

Numerical Integration

Choose a small time step d, and a large maximum queue capacity.

1. First, set the beginning of the queue where $p_0(0) = 1$ and $p_n(0) = 0$.
2. Apply classical queueing theory model for the next time step:

$$\begin{aligned} p_n(t+d) &= \lambda(t) d p_{n-1}(t) + [1 - \lambda(t) d - \mu d] p_n(t) + \mu d p_{n+1}(t) \\ p_0(t+d) &= [1 - \lambda(t) d] p_n(t) + \mu d p_{n+1}(t) \\ p_{n_{\max}}(t+d) &= \lambda(t) d p_{n_{\max}-1} + [1 - \mu d] p_{n_{\max}} \end{aligned} \tag{1}$$

3. Increment time point: $t = t + d$
4. Repeat until you reach maximum time.

Comparison

To compare the results, we use average system length of both simulation and numerical integration result.

Comparison

To compare the results, we use average system length of both simulation and numerical integration result.

We use this formula to calculate average system length of queue.

$$L(t) = \sum_{n=0}^{n_{\max}} n p_n(t)$$

Comparison

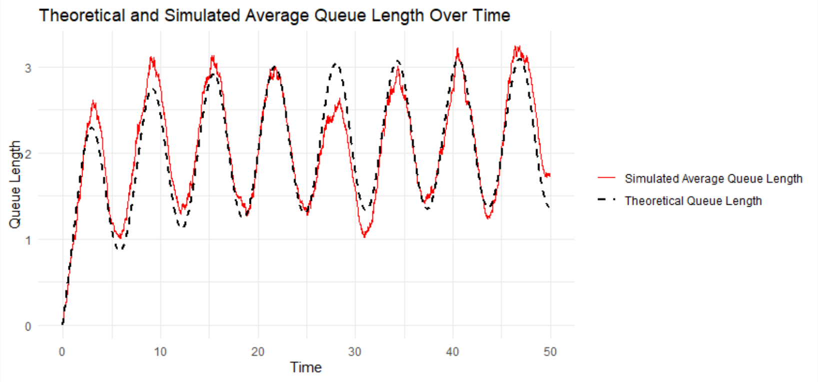


Figure 7: M(t)/M/1 Average System Length

Queue Extensions

Comparison

Figure 8: Taxi Queue in Ibiza

Double-ended Queues

One of the extensions we looked at was double-ended queueing systems.

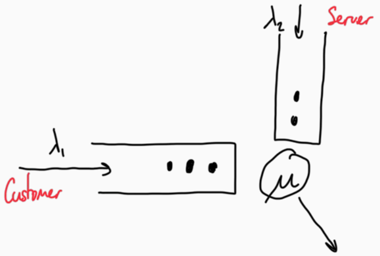


Figure 9: Double-Ended Queueing System

Simulation

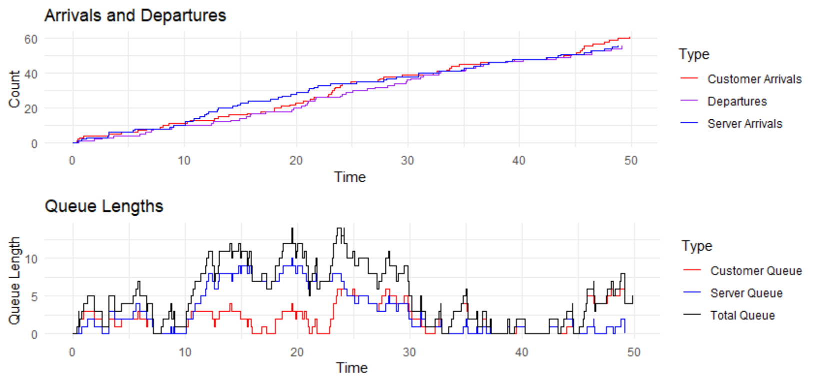


Figure 10: Simulation of a double-ended queue with $\lambda_1 = \lambda_2 = 1$ and $\mu = 2$.

Sanity Check

With $\mu \to \infty$, one side of the queue will always be empty.

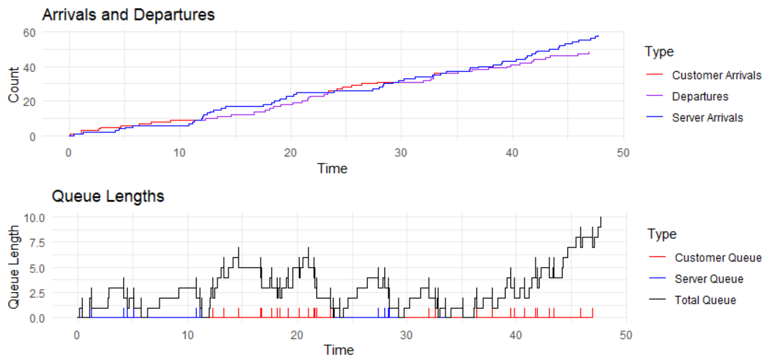


Figure 11: Simulation of a double-ended queue with $\lambda_1 = \lambda_2 = 1$ and $\mu = 10000$

Decision Making - Optimisation

Introduction to Simulation Optimisation

- Target variable is λ.
- We want to optimise to a certain objective denoted $F(\lambda)$.
- An example can be within an M/M/1 queue where we maximise entry with respect to a cost of waiting time.
- Typically in reality, we will also require observations of our service time to complete this $\hat{\mu}$.

Steps to Optimisation

To begin the cycle, select a suitable λ_0. If unsure usually start rather high as it would provide better estimations for $\hat{\mu}$.

Steps to Optimisation

To begin the cycle, select a suitable λ_0. If unsure usually start rather high as it would provide better estimations for $\hat{\mu}$.

We also need to select our objective function $F(\lambda)$. This will depend on the needs of the decision maker.

1. Observe a period of time t_{max} with true μ and λ_t.

Steps to Optimisation

To begin the cycle, select a suitable λ_0. If unsure usually start rather high as it would provide better estimations for $\hat{\mu}$.

We also need to select our objective function $F(\lambda)$. This will depend on the needs of the decision maker.

1. Observe a period of time $t_{\max}$ with true μ and λ_t.
2. Use the MLE to estimate the departure parameter μ.

Steps to Optimisation

To begin the cycle, select a suitable λ_0. If unsure usually start rather high as it would provide better estimations for $\hat{\mu}$.

We also need to select our objective function $F(\lambda)$. This will depend on the needs of the decision maker.

1. Observe a period of time $t_{\max}$ with true μ and λ_t.
2. Use the MLE to estimate the departure parameter μ.
3. Simulate the queue over a selected set of λs.

Steps to Optimisation

To begin the cycle, select a suitable λ_0. If unsure usually start rather high as it would provide better estimations for $\hat{\mu}$.

We also need to select our objective function $F(\lambda)$. This will depend on the needs of the decision maker.

1. Observe a period of time t_{max} with true μ and λ_t.
2. Use the MLE to estimate the departure parameter μ.
3. Simulate the queue over a selected set of λs.
4. Choose $\lambda_t^* = \arg\max_\lambda F(\lambda)$. Set the next cycle $\lambda_{t+1} = \lambda_t^*$.

Flow Chart

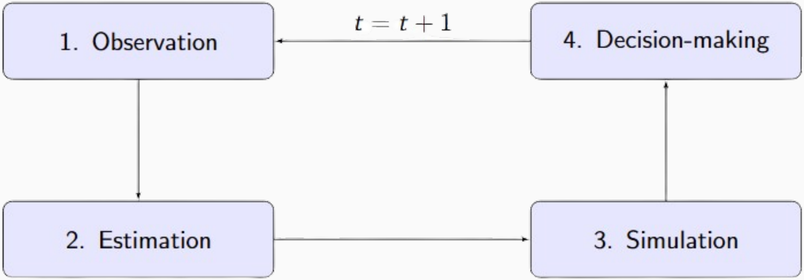


Figure 12: Simulation Optimisation Flow Chart

Assumptions

Decision-making assumptions made:

- At t_{max}, the queue no longer accepts entries but the simulation continues until all customers are served.
- Without this, computation may be more difficult as it would skew rate $\hat{\mu}$ upwards and average waiting time downwards.

Example Setup

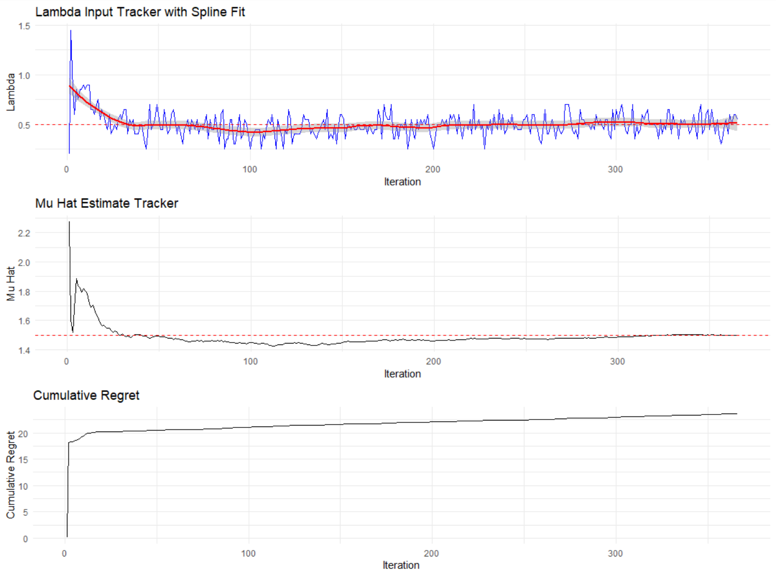
- Begin with $\lambda_0 = 0.2$ and true $\mu = 1.5$.
- Cycle through 24 time units per cycle for 365 cycles.
- 40 (week long) simulations for each λ when optimising.
- Objective function is set as:

$$F(\lambda) = \frac{\#\text{arrivals}}{t_{\text{max}}} - \frac{\sum_{i=1}^{n} w_i}{n}$$

- The steady state equivalent for this is

$$F(\lambda) = \lambda - \frac{\lambda}{\mu(\mu - \lambda)}$$

Example Optimisation



Pros and Cons

Pros

- Converges relatively quickly dependent on $\hat{\mu}$.
- Corrects a poorly chosen λ_0.
- With simulation, correctly accounts where steady state does not (if $\lambda > \mu$).
- Convexity is not required.

Pros and Cons

Pros

- Converges relatively quickly dependent on $\hat{\mu}$.
- Corrects a poorly chosen λ_0.
- With simulation, correctly accounts where steady state does not (if $\lambda > \mu$).
- Convexity is not required.

Cons

- Computationally slow with more complex systems.
- Does not settle on a solution and constantly fluctuates (hence the need for splines for understandability).

Conclusion

Further Work

- Improve computational time for simulations.
- Develop simulations for more advanced queues with less assumptions, i.e. multi-server queue
- Develop optimisations for more advanced queues, i.e. double-ended taxi queues.
- Explore further simulation optimisation methodologies.

References

References

Chen, Y. (2016).
Thinning algorithms for simulating point processes.
Presented in September, 2016.

Nelson, B. (2021).
Foundations and Methods of Stochastic Simulation.
Springer International Publishing, Cham, Switzerland.

Stewart, W. J. (2009).
Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling.
Princeton University Press, Princeton, NJ.

Any questions?

Appendix

Proof: Poisson Arrival Implies Exponential Interarrival

Let T_n and T_{n-1} denote the time difference between two arrivals, i.e. the interarrival time, and $p_n(t) = \frac{(\lambda t)^n}{n!} \exp(\lambda(t))$.

$$\begin{aligned} F(t) &= \mathbb{P}(T_n - T_{n-1} \leq t) \\ &= \mathbb{P}(T_1 \leq t) \quad \text{due to memoryless property} \\ &= \mathbb{P}\{\text{at least one event occured } (0, t]\} \\ &= 1 - \mathbb{P}\{\text{no event occured } (0, t]\} \\ &= 1 - p_0(t) \\ &= 1 - \exp(-\lambda t) \end{aligned} \tag{2}$$

Hence obtaining the PDF, we can differentiate $F(t)$

$$f(t) = \frac{dF(t)}{dt} = \lambda \exp(-\lambda t)$$

Which is the PDF for the exponential distribution.

Alternative Objective Functions

Objective functions can be chosen to suit needs of decision maker. Examples:

- Maintain unit of time by replacing arrival count with interarrival time.

$$F(\lambda) = -\frac{1}{\lambda} - \frac{\lambda}{\mu(\mu - \lambda)}$$

- Adjust severeness of reward or cost of existing objective function by α and β.

$$F(\lambda) = \alpha\lambda - \beta\left(\frac{\lambda}{\mu(\mu - \lambda)}\right)$$