Analysis and Optimisation of Queueing Systems

Jimmy Lin [1] Robert Lambert [2]

[1]University of Glasgow [2]Lancaster University

Motivation

Queues are a part of our everyday lives, from the long lunch queues, to peak hour traffic jams. As a result, queues are also a major topic within operational research.

In classical queueing models (i.e. M/M/1), we assume a poisson distributed entry and an exponentially distributed service time using one server. This is due to the memoryless property of these distributions.

The issue with classical queueing systems is that real world systems are far too complex and typically requires numerical solutions as they are analytically intractable, thus we choose to simulate.

M/M/1 Queues

To simulate queues, we consider generating and tracking discrete points where events occur, using the **interarrival distribution**, which is exponentially distributed.

To test the simulation, compare the simulation to the theoretical result. For M/M/1, we can simply use the steady state solution.

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

In this case, we simulate enough to reach steady state, then create a histogram based final state of simulation.

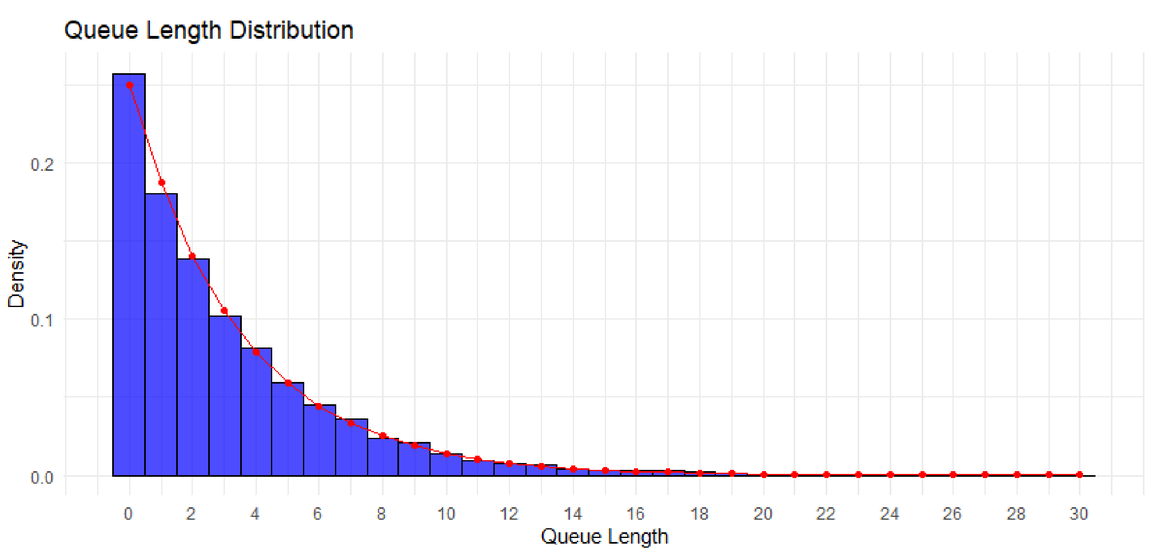

Figure 1. Histogram of Simulations Steady State Probabilities to Theory

However, due to assumption of constant arrival rate, M/M/1 models are typically far too unrealistic to model real world queues.

M(t)/M/1 Queues

Arrival rates often vary, complicating simulations which assumes an exponentially distributed interarrival times. To address this, we could use the **thinning algorithm**.

Suppose $\bar{\lambda} = \sup \lambda(t)$, we simulate a homogeneous poisson process with rate $\bar{\lambda}$. With each arrival occuring at t, we then **accept** the arrival with probability,

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

With changing arrival rate, we cannot use steady state solutions. In this scenario, to verify results of the simulation, use **numerical integration**.

Given that $p_n(t)$ denotes $\mathbb{P}\{n \text{ in system at time } t\}$:

1. First, set the beginning of the queue where $p_0(0) = 1$ and $p_n(0) = 0$.
2. For the next time point $t + d$, where d is a selected small time step,

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

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

M(t)/M/1 Results

Without a steady state solution, to compute the results, we can refer to average queue length at time t.

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

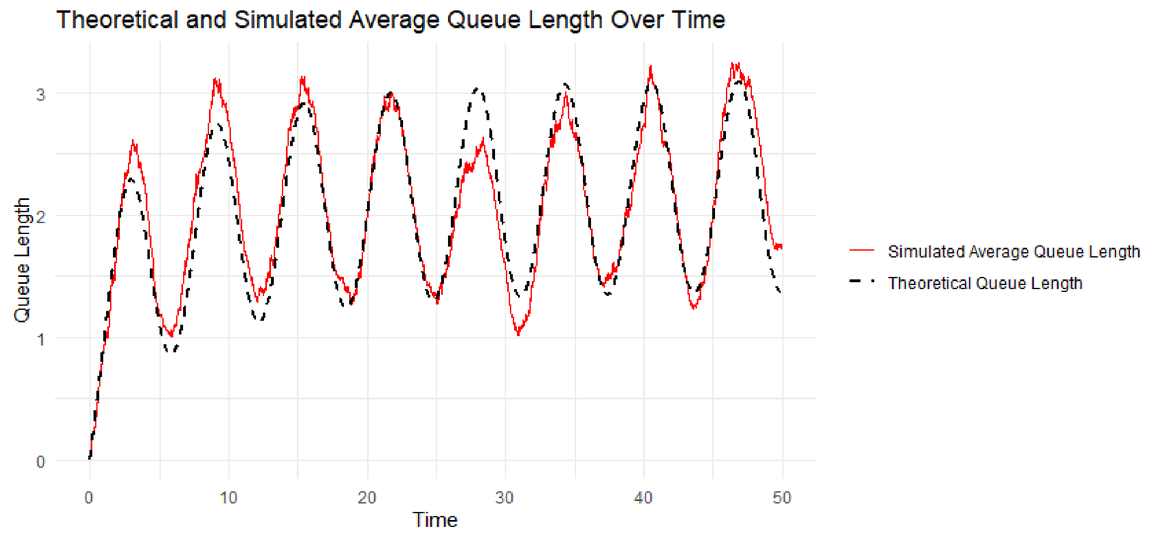

Figure 2. M(t)/M/1 Average System Length

Decision Making

Simulations can be incorporated within decision-making algorithms, where decision-makers optimises set objective function $F(\lambda)$ that captures the desired system dynamics.

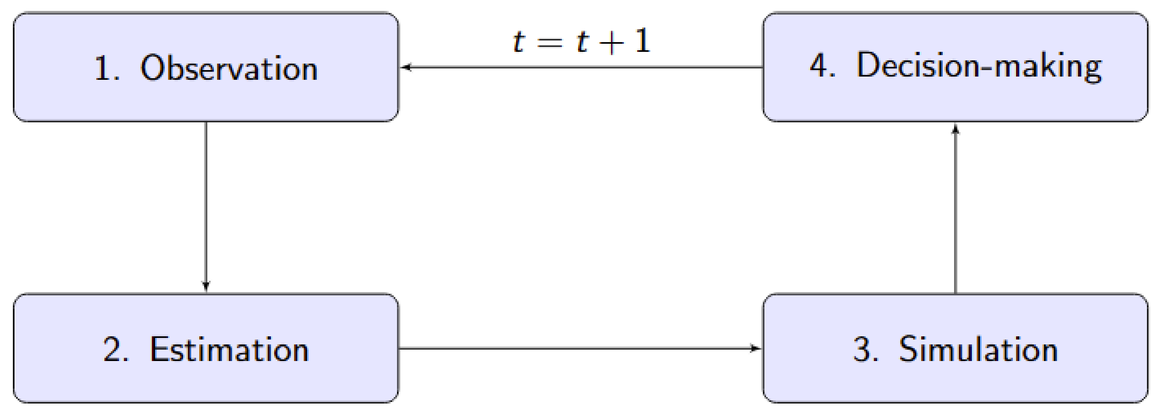

Figure 3. Decision Making Algorithm Flow Chart

For example, at an M/M/1 target system, we may choose to optimise λ based on observations for $\hat{\mu}$ to minimise waiting time and maximise number of arrivals.

1. Observe a period of time $t_{\max}$ with true μ and λ_t such that one would select a suitable λ_0.
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 $\lambda_{t+1} = \lambda_t^*$.

Future Work

- Improve computational time for simulations.
- Develop simulations for more advanced queues with less assumptions
- Explore further simulation optimisation methodologies.

References

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

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

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