Introduction to Queuing Theory and Its Use in Manufacturing

Rob Leachman
IEOR 130
Nov. 15, 2016
Purpose

• In most service and production systems, the time required to provide the service or to complete the product is important.
  – We may want to design and operate the system to achieve certain service standards.
• Generally, the time required includes “hands-on” time (actually processing) plus time waiting.
• Queuing theory is about the estimation of waiting times.
Terminology and Framework

- **Customers** arrive randomly for service and await availability of a **server**
  - When the server(s) has (have) finished servicing previous customers, the new customer can begin service
- **Time between arrival of customer and start of service** is called the **queue time**
- **Customer departs the system after completion of the service time**
- **Total time in system** = **queue time + service time**
Analytical Approximation

- The mathematics of queuing theory is much easier if we assume the customer inter-arrival time has an exponential distribution, and if we assume the service time also has an exponential distribution. The exponential distribution has the memoryless property:
  - Suppose the average inter-arrival time is $t_a$. Given it has been $t$ since the last customer arrival, what is the expected time until the next customer arrival? Answer: Still $t_a$!
  - Suppose the average service time is $t_s$. Given it has been $t$ time units since service started, what is the expected time until service ends? Answer: Still $t_s$!
The M/M/1 Queue

- Queuing notation: $A/B/n$ means inter-arrival times have distribution $A$, service times have distribution $B$, $n$ means there are $n$ servers
- M means Markovian (memoryless), 1 means one server
- In a Markovian queuing system, the only information we need to characterize the state of the system is the number of customers $n$ in the system
The M/M/1 Queue (cont.)

• We write \( \lambda = 1/t_a \) as the arrival rate and \( \mu = 1/t_s \) as the service rate.
• The utilization of the server is \( u = t_s / t_a = \lambda / \mu \).
• Note that we must have \( u < 1 \) for the queue to be stable.
The M/M/1 Queue (cont.)

- Markovian state-space: Node $n$ represents the state with $n$ customers in the system
- The arcs show the rate at which the system transitions to an adjacent state

![Diagram of M/M/1 Queue](image)

$\lambda$  $\mu$  $\lambda$  $\mu$  $\lambda$  $\mu$
The M/M/1 Queue (cont.)

- Let $p_n =$ probability system has $n$ customers in it.
- Because there is only one server, the system can only change by one unit at a time.
- The system moves from state $n$ to state $n+1$ at rate $\lambda$.
- The system moves from state $n+1$ to state $n$ at rate $\mu$.
- If the system is in a steady state, we must have
  $$\lambda p_n = \mu p_{n+1} \text{ or } p_{n+1} = \left(\frac{\lambda}{\mu}\right) p_n = u \ p_n$$
The M/M/1 Queue (cont.)

• Now  \[ 1 = \sum_{n=0}^{\infty} p_n = \sum_{n=0}^{\infty} p_0 u^n = p_0 \frac{1}{1-u} \]

• So  \[ p_0 = 1-u \]

• The expected total time a customer stays in the system is

\[
\sum_{n=0}^{\infty} t_s (1+n) p_n = t_s \sum_{n=0}^{\infty} p_n + \sum_{n=0}^{\infty} t_s np_0 u^n = t_s + \sum_{n=0}^{\infty} t_s n(1-u)u^n
\]

\[
= t_s + \sum_{n=0}^{\infty} t_s n(1-u)u^n = t_s + t_s (1-u)u \sum_{n=0}^{\infty} nu^{n-1}
\]

\[
= t_s + t_s (1-u)u \sum_{n=0}^{\infty} \frac{d}{du} u^n = t_s + t_s (1-u)u \frac{d}{du} \sum_{n=0}^{\infty} u^n
\]

\[
= t_s + t_s (1-u)u \frac{d}{du} \left( \frac{1}{1-u} \right) = t_s + t_s \frac{u}{(1-u)} = \frac{t_s}{(1-u)}
\]
The M/M/1 Queue (cont.)

- And so the expected queue time is

\[ QT = \frac{t_s}{1-u} - t_s = \frac{u}{1-u} t_s. \]
Numerical Example

• Suppose $t_s = 12$ minutes, $\lambda = 4$ per hour
• Then $u = \frac{\lambda}{\mu} = \lambda \cdot t_s = 4 \times \frac{12}{60} = 80\%$
• Probability server is idle $= 1 - u = 20\%$
• Expected queue time $= \frac{u}{1-u} t_s = \left(\frac{0.8}{0.2}\right) \times 12 = 48$ minutes
• Expected time in system $= 48 + 12 = 60$ minutes
Queuing in Manufacturing

- Customers = production lots. Total time a lot is at a production step (wait + process) is called the *cycle time* of the step.
- Servers = machines. Machines require maintenance. They are only available for processing work part of the time.
- Suppose the *availability* is $A$ and the process time is $PT$. The effective long-run service rate is $\mu = A \times (1/PT)$. $u$ becomes $u = \lambda/\mu = \lambda*(PT/A)$.
- Note that we decrease the service rate and we increase utilization to account for machine down time
Queuing in Manufacturing (cont.)

• When we have one machine, we can estimate the avg. queue time as:
  \[ QT = \frac{u}{1-u} \frac{PT}{A} \]

• Queuing model: Time in system = queue time + service time =
  \[ QT + \frac{PT}{A} \]

• Real life: Time in system = wait time + process time = (wait time) + \( PT \)

• So (wait time) = \[ QT + \frac{PT}{A} - PT = QT + t_s - PT \]
Queuing in Manufacturing (cont.)

• The standard cycle time $SCT$ is the total time a lot is resident at the production step when there is no waiting. It is often somewhat larger than the process time $PT$ as it accounts for material handling time or other factors performed in parallel with the processing of other lots.

• Therefore, lot $CT = (\text{wait time}) + SCT$

$$= (QT + t_s - PT) + SCT$$

$$= QT + (1/A - 1)PT + SCT$$
Queuing in Manufacturing (cont.)

• Another way to think of this is:

\[
\text{(Cycle time)} = (\text{Time in queuing system}) + (\text{portion of cycle time not in queuing system})
\]

\[
= (QT + PT / A) + SCT – PT
\]

\[
= QT + (1/A – 1)\times PT + SCT
\]
Numerical example

- Availability of machine $A = 85\%$
- Arrival rate of lots $\lambda = 2$ per hour
- $PT = 0.25$ hours (i.e., 15 minutes), $SCT = 0.30$ hours (i.e., 18 minutes)
- $u = (2)(0.25)/0.85 = 0.588$
- $QT = [0.588/(1-0.588)]*(0.25/0.85) = 0.42$ hours = 25 minutes
- $CT = 25 + (1/0.85 -1)*15 + 18 = 45.6$ minutes
- Note that the avg. waiting time (30.6 mins) is much longer than the process time (15 mins)
More general queuing formula

- We may have $m$ machines instead of 1
- Service and arrival rates might not be exponential, machines may experience long down times (failures or major maintenance events)
- Generic formula for queue time per lot or batch (Kingman, Sakasegawa, Hopp and Spearman):

$$QT = \left( \frac{c_a^2 + c_e^2}{2} \right) \left( u \frac{\sqrt{2^{(m+1)-1}}}{m(1-u)} \right) \left( \frac{PT}{A} \right)$$
More general formula (cont.)

- $c_a^2$ is the normalized variance (the squared coefficient of variation, or “c.v.$^2$” for short) of the arrival rate, i.e., $c_a^2 = \sigma_a^2/\lambda^2$
- $c_e^2$ is the normalized variance of the (effective) service time, composed of the following:
  - $c_0^2$ is the normalized variance of the process time, i.e., $c_0^2 = \sigma_{PT}^2/PT^2$
  - $MTTR$ is the average length of a downtime event
  - $cr^2$ is the normalized variance of the length of an equipment-down event, i.e., $cr^2 = \sigma_r^2/MTTR^2$
More general formula (cont.)

- $A$ is the average availability of the machine type
- Then

$$c_e^2 = c_0^2 + (1 + cr^2)A(1 - A)\left(\frac{MTTR}{PT}\right)$$
Key point: Wait time =

\[
\text{Variability} \left\{ \frac{(u)^{\sqrt{2(m+1)}-1}}{m(1-u)} \right\} \text{Process time/Availability} + \{\text{Process time}\} \{1/\text{Availability} - 1\}
\]

- One can reduce cycle time if any of the above terms is reduced (i.e., reduce variability, reduce \(u\), increase \(m\), reduce \(PT\), or increase \(A\))
Queuing Analysis (cont.)

{Variability}

- \( ce = \) effective service time c.v. (reflects machine down time)
  - Let \( c_0 \) denote intrinsic process time c.v., \( cr \) denote repair time c.v., \( A \) denote availability, \( MTTR \) denote mean time to repair, \( PT \) denote avg. process time

\[
ce_k^2 = c_0^2 + (1 + cr_k^2) A_k (1 - A_k) \frac{MTTR_k}{PT_k}
\]

Utilization should be kept lower for machines with higher variability
{Utilization}

- System performance is very sensitive to high utilization levels.

- Balancing utilization reduces wait time.

- Increasing the number of qualified machines reduces wait time:

![Wait Time vs. Utilization Graph](image)