

The Waiting Room Problem: Managing Crowded Machine Repair Lines

K.P.S. Baghel

Government Degree College, Targawan Jaithra, Etah (UP)

Abstract

Every repair shop has a waiting room, even when it's just a row of dead machines on a factory floor. This article treats that queue as a mathematical object and asks a practical question: how many repairers do you need, how should you organize them, and what happens when machines wait too long? We start with the birth-death models in the Markovian tradition of Erlang (1909) and Kendall (1953). Then we build the finite-source machine interference model, where every breakdown removes a machine from the pool that creates future breakdowns. In a worked example with five machines and a failure-to-repair ratio of 0.1, a second repairer cuts the mean wait for repair from about 0.47 repair times to 0.02 and lifts availability from 87.2% to 90.7%. We then show why waiting grows hyperbolically with utilization, using the Pollaczek-Khinchine and Kingman formulas, and why pooling repairers beats splitting them. A pooled two-repairer system waits 1.78 time units, against 4.00 for two separate lines at the same load. Next we add reneging, where impatient jobs leave the line, and show that such a system stays stable even when arrivals exceed capacity, at the price of losing about 37% of demand in our example. We finish with priority rules from Cobham (1954).

Keywords: machine repair queue, machine interference, M/M/R model, reneging, repairer pooling, maintenance scheduling

I. Introduction

1.1 The problem

Walk into any plant on a bad Monday and you'll see it. Three machines are down. One technician is bent over the first, and the other two sit idle, collecting dust and costing money. Nobody planned for that line. It just formed.

The line matters because waiting time is lost output. A machine that waits is a machine that isn't producing, and the wait often exceeds the repair itself. Managers tend to look at average repair time, which hides the real culprit. The queue does the damage.

1.2 Why mathematics helps

Intuition fails badly with queues. People expect waiting to grow in proportion to load, but it doesn't. A shop running at 80% utilization feels fine. At 95% it's a disaster. Queueing theory explains why, and it does so with a small set of formulas that anyone can check by hand.

Our starting point is the notation of Kendall (1953), where M/M/R means Markovian failures, Markovian repairs, and R repairers. The machine interference problem goes back to Palm (1947), and Taylor and Jackson (1954) applied birth-death reasoning to spare machines. More recently, Baghel (2013, 2014, 2017) has studied crew training, reneging, and scheduled maintenance in the same framework.

II. Mathematical Foundations

2.1 Birth-death processes

Let n be the number of machines in the repair system, counting those under repair and those waiting. Suppose the system moves from n to $n+1$ at rate λ_n and from n to $n-1$ at rate μ_n . Exponential times are memoryless (Feller, 1968), which is why the state n alone describes the system.

At steady state, flow across each cut between $n-1$ and n balances:

$$\lambda_{n-1}p_{n-1} = \mu_n p_n$$

Solving recursively gives:

$$p_n = p_0 \cdot \prod_{k=1..n} (\lambda_{k-1}/\mu_k), \text{ with } p_0 = [1 + \sum_{n \geq 1} \prod_{k=1..n} (\lambda_{k-1}/\mu_k)]^{(-1)}$$

Everything in this article comes from that one product formula (Gross et al., 2008; Kleinrock, 1975).

2.2 Little's law

For any stable system, the mean number L in the system equals the effective arrival rate times the mean time W in the system (Little, 1961):

$$L = \lambda_{eff} \cdot W, \text{ and likewise } Lq = \lambda_{eff} \cdot Wq$$

We'll use it constantly. It needs no distributional assumptions, so it holds even where the exponential model is shaky.

2.3 The infinite-source benchmark

If failures come from a huge fleet, $\lambda_n = \lambda$ for all n . With R repairers, $\mu_n = \min(n, R)\mu$. Define $a = \lambda/\mu$ and $\rho = a/R$, and require $\rho < 1$. The probability that an arriving breakdown must wait is Erlang's C formula (Erlang, 1909):

$$C(R, a) = [a^R / (R! (1 - \rho))] / [\sum_{k=0}^{R-1} a^k / k! + a^R / (R! (1 - \rho))]$$

The mean wait in queue follows directly:

$$Wq = C(R, a) / (R\mu - \lambda)$$

For $R = 1$ this collapses to the formula we'll lean on most:

$$Wq = \rho / (\mu(1 - \rho))$$

III. The Finite-Source Machine Repair Model

3.1 Setting up the chain

Real plants own a finite number of machines. N machines run, and each fails independently at rate λ while running. Repairers fix at rate μ each. When n machines are down, $N-n$ are still running, so:

$$\lambda_n = (N - n)\lambda, \mu_n = \min(n, R)\mu$$

This is the machine interference model of Palm (1947). Failures slow down as the queue grows, because fewer machines remain to fail. The system regulates itself, and that's why it's always stable, even at high load.

3.2 Steady-state solution

Let $\rho = \lambda/\mu$. Applying the product formula gives:

$$p_n = p_0 \cdot C(N, n) \rho^n, \text{ for } 0 \leq n \leq R$$

$$p_n = p_0 \cdot C(N, n) \cdot n! \cdot \rho^n / (R! \cdot R^{(n-R)}), \text{ for } R < n \leq N$$

For one repairer this simplifies neatly:

$$p_n = p_0 \cdot N! / (N - n)! \cdot \rho^n$$

The performance measures are:

$$L = \sum n p_n, \text{ availability } A = (N - L)/N, \text{ throughput } \lambda_{eff} = \lambda(N - L), \text{ and } W = L/\lambda_{eff}$$

3.3 A worked example

Take $N = 5$ machines, $\lambda = 0.1$ per unit time, $\mu = 1$, so $\rho = 0.1$. With one repairer, the unnormalized terms for $n = 0$ to 5 are 1, 0.5, 0.2, 0.06, 0.012, and 0.0012. They sum to 1.7732, so $p_0 = 0.5640$. The numerator of L is $0.5 + 0.4 + 0.18 + 0.048 + 0.006 = 1.134$, giving $L = 0.6395$.

Then $A = (5 - 0.6395)/5 = 0.8721$. Throughput is $0.1 \times 4.3605 = 0.4360$, which matches the repairer's busy rate $\mu(1 - p_0) = 0.4360$, a useful consistency check. Little's law gives $W = 0.6395/0.4360 = 1.467$, so $Wq = W - 1/\mu = 0.467$.

Machines wait nearly half a repair time before anyone touches them.

Now add a second repairer. The terms become 1, 0.5, 0.1, 0.015, 0.0015, and 0.000075, which sum to 1.616575. So $p_0 = 0.6186$ and $L = 0.7514/1.6166 = 0.4648$. Availability rises to 0.9070, throughput to 0.4535, and W falls to 1.025, so $Wq = 0.025$. Table 1 compares the two cases.

Table 1: Machine interference model, $N = 5$, $\lambda = 0.1$, $\mu = 1$ (computed from the formulas in Section 3.2; time in mean repair times)

Measure	R = 1	R = 2
Probability repair shop idle (p_0)	0.564	0.619
Mean number down (L)	0.640	0.465
Availability (A)	0.872	0.907
Throughput (λ_{eff})	0.436	0.454
Mean wait before repair (Wq)	0.467	0.025

A 3.5-point gain in availability sounds small. It isn't, because the wait for repair falls by about 95%. The second repairer is mostly idle, with utilization of roughly 0.45 per head versus 0.44 for the lone worker, and yet it removes almost all of the queueing. Idle capacity is what you pay for short lines. As shown in Figure 1 below, this nonlinearity comes from a general law about utilization, which we turn to next.

IV. Variability, Utilization, and Pooling

4.1 Why waiting explodes

For the single-server queue with Poisson arrivals and exponential service:

$$Wq = \rho / (\mu(1 - \rho))$$

Measured in mean service times, Wq equals 1 at $\rho = 0.5$, 4 at 0.8, 9 at 0.9, 19 at 0.95, and 99 at 0.99. Each extra point of utilization buys more delay than the last. Running a repair crew at 95% looks efficient on a spreadsheet, but machines will wait 19 repair times on average.

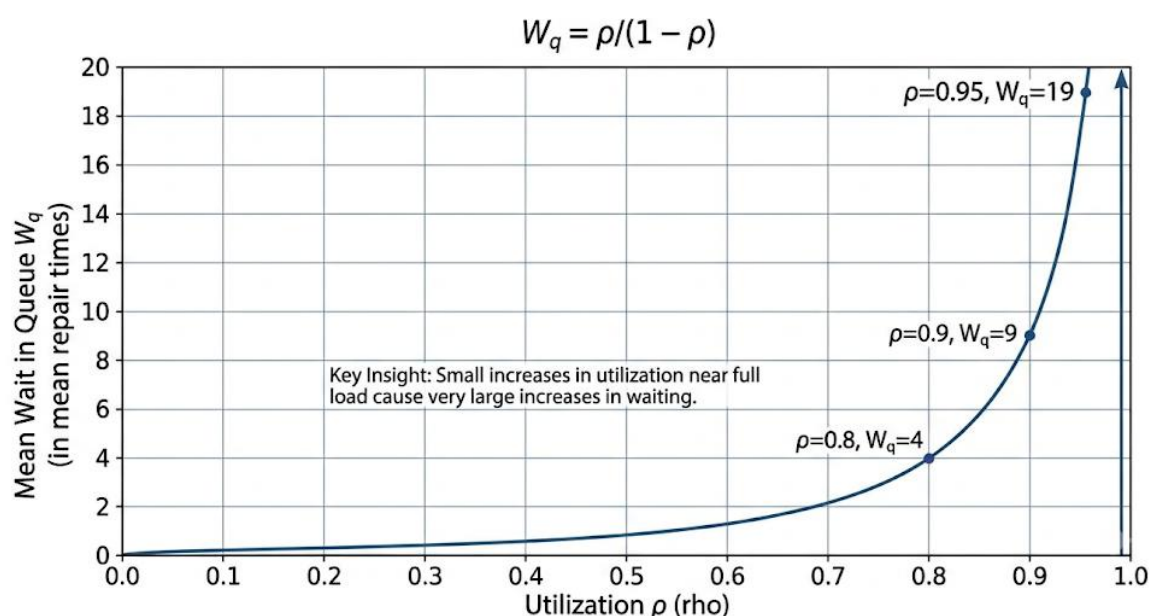


Figure 1: Mean Queue Wait versus Repairer Utilization in the M/M/1 Model

The horizontal axis shows utilization ρ from 0 to 0.99, and the vertical axis shows the mean wait in queue Wq measured in mean repair times, plotted from the formula $Wq = \rho / (1 - \rho)$. The curve is nearly flat below $\rho = 0.5$ and then rises almost vertically past 0.9, passing through the points (0.8, 4), (0.9, 9), and (0.95, 19). The key insight is that small increases in utilization near full load cause very large increases in waiting. Values computed from the M/M/1 result in Kleinrock (1975) and Gross et al. (2008).

4.2 The role of variability

Exponential service is a convenient assumption, not a law. For general service times, the Pollaczek-Khinchine formula for M/G/1 gives:

$$Wq = \rho E[S](1 + c_s^2) / (2(1 - \rho))$$

where c_s^2 is the squared coefficient of variation of repair time. Exponential repairs have $c_s^2 = 1$, and fixed-length repairs have $c_s^2 = 0$, which halves the wait. For general arrivals, Kingman's (1961) heavy-traffic approximation reads:

$$Wq \approx [\rho / (1 - \rho)] \cdot [(c_a^2 + c_s^2) / 2] \cdot E[S]$$

Whitt (1993) extended the idea to multiple servers. Hopp and Spearman (2008) put the lesson plainly: variability and utilization multiply each other. Standardizing repair procedures lowers c_s^2 , and that's often cheaper than hiring.

4.3 Pooling

Suppose two repairers each have their own line, and breakdowns arrive at 0.8 per unit time per line with $\mu = 1$. Each line is an M/M/1 queue with $\rho = 0.8$, so $Wq = 0.8 / 0.2 = 4.0$.

Now pool them into one line with total arrival rate 1.6. Here $a = 1.6$, $R = 2$, and $\rho = 0.8$. The Erlang C numerator is $a^2/(2(1 - \rho)) = 2.56/(2 \times 0.2) = 6.4$. The denominator is $1 + 1.6 + 6.4 = 9.0$, so $C = 0.711$. Then:

$$Wq = C/(R\mu - \lambda) = 0.711/0.4 = 1.778$$

Same people, same load, and the wait falls by 56%. Pooling works because a free repairer never sits idle next to a waiting machine. Baghel (2013) compares generalist and specialist repair crews in an M/M/R model, and this arithmetic is the reason generalists often win: they behave like a pooled system. Specialists can still win when repair times differ sharply by skill, so test it with your own numbers.

4.4 Scheduled maintenance as load reduction

Preventive work reduces λ , since fewer breakdowns arrive, but consumes repairer time. If a preventive visit takes time τ every T units per machine, the effective load is:

$$\rho_{total} = N\lambda(T)/(R\mu) + N\tau/(RT)$$

where $\lambda(T)$ is the failure rate under a T -unit maintenance cycle. Optimizing T balances the two terms. Baghel (2017) studies exactly this trade-off for scheduled maintenance cycles in an M/M/C framework. The point for managers is that maintenance isn't free. It borrows from the same crew that fixes breakdowns, so it can raise the queue if you schedule it badly.

V. When Machines Quit the Line: Reneging

5.1 The model

Sometimes a waiting job leaves. A customer withdraws a repair order, or a unit gets scrapped or cannibalized for parts. Haight (1957) and Ancker and Gafarian (1963) formalized balking and reneging. Suppose each waiting job leaves at rate θ . With one repairer, the service-completion rate in state n is:

$$\mu_n = \mu + (n - 1)\theta, \text{ for } n \geq 1$$

So the stationary distribution is:

$$p_n = p_0 \cdot \lambda^n / \Pi_{k=1..n} (\mu + (k - 1)\theta)$$

Because μ_n grows without bound, the sum converges for any λ . The queue stays stable even when $\rho = \lambda/\mu$ exceeds 1. Baghel (2014) studies this class of reneging systems in an M/M/R setting with limited spares.

5.2 Numerical example

Let $\lambda = 1.2$, $\mu = 1$, $\theta = 0.5$, so $\rho = 1.2$. Without reneging, this queue would grow forever. Terms for $n = 0$ through 6 are 1, 1.2, 0.96, 0.576, 0.2765, 0.1106, and 0.0379. The tail from $n = 7$ up adds about 0.0152. The total is roughly 4.176, so $p_0 = 0.2395$.

Figure 2 shows the resulting distribution. The probability of an idle repairer is 0.24, and the probability of one machine in the system is 0.29.

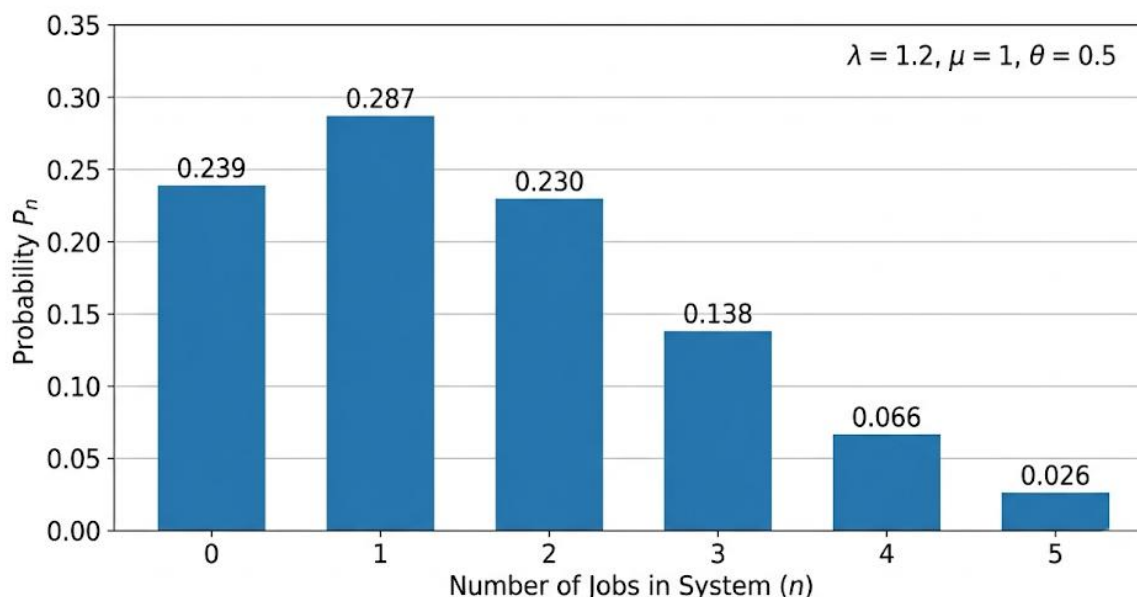


Figure 2: Steady-State Distribution of Jobs in a Single-Repairer Queue with Reneging

A bar chart shows the probability p_n of having n jobs in the system, for $n = 0$ to 5 , with $\lambda = 1.2$, $\mu = 1$, and $\theta = 0.5$. The bars read approximately 0.239, 0.287, 0.230, 0.138, 0.066, and 0.026, and they fall off quickly after $n = 2$. The key insight is that the line stays short even though demand exceeds repair capacity, because impatient jobs leave. Values computed from the birth-death formula in Section 5.1 (after Ancker & Gafarian, 1963).

5.3 The price of stability

Balance on the whole system gives:

$$\lambda = \mu(1 - p_0) + \theta Lq$$

The repairer completes $1 \times (1 - 0.2395) = 0.7605$ jobs per unit time. So the reneging rate is $1.2 - 0.7605 = 0.4395$, giving $Lq = 0.4395/0.5 = 0.879$. About 36.6% of demand is lost.

A short queue here is a sign of failure, not success. The system looks tidy because a third of the work walks away. Judging a shop by its queue length alone is one of the most common mistakes in practice, and the abandonment fraction should always sit next to it.

VI. Priorities in the Repair Line

6.1 Cobham's formula

Not every machine matters equally. A bottleneck press should jump ahead of a spare lathe. For a nonpreemptive priority M/G/1 queue with classes 1 (highest) to K , Cobham (1954) showed:

$$W_k = W_0 / [(1 - \sigma_{k-1})(1 - \sigma_k)]$$

where $W_0 = \Sigma \lambda_i E[S_i^2] / 2$ and $\sigma_k = \Sigma_{i \leq k} \rho_i$.

6.2 Example

Take two classes with $\lambda_1 = 0.3$ and $\lambda_2 = 0.5$, both with exponential repairs of mean 1, so $E[S^2] = 2$. Then $W_0 = (0.3 \times 2 + 0.5 \times 2) / 2 = 0.8$, with $\sigma_1 = 0.3$ and $\sigma_2 = 0.8$. This gives:

$$W_1 = 0.8 / (1 \times 0.7) = 1.143$$

$$W_2 = 0.8 / (0.7 \times 0.2) = 5.714$$

Under first-come-first-served, everyone waits $0.8/0.2 = 4.0$. Check the conservation law: $(0.3 \times 1.143 + 0.5 \times 5.714) / 0.8 = 4.0$. Priorities don't reduce total waiting. They redistribute it.

So priority is a business decision wearing a mathematical costume. If a minute of downtime on class 1 costs ten times more than on class 2, the rule above is a bargain. If the costs are equal, you've only shifted pain from one group to another.

6.3 Networks of repair stations

Many shops have several stations, such as diagnosis, parts, and testing. Jackson (1957) proved that open networks of exponential queues behave as if each station were independent, with arrival rate set by the routing equations:

$$\lambda_j = \gamma_j + \Sigma_i \lambda_i P_{ij}$$

That makes bottleneck hunting simple. Compute $\rho_j = \lambda_j / (R_j \mu_j)$ at each station, and the largest value is where your waiting room is. Add capacity there, and nowhere else.

VII. Conclusion

The waiting room problem looks like a staffing question, but the formulas show it's really about three levers: load, variability, and structure.

Load comes first. Waiting grows like $\rho/(1 - \rho)$, so the last few points of utilization are extremely expensive. The five-machine example shows the flip side: a second repairer sat mostly idle yet removed 95% of the queueing delay, because idle time is the cost of a short line.

Structure comes next. Pooling the same two repairers into one line cut the mean wait from 4.0 to 1.78 at an identical 80% load. That matches the logic behind cross-trained crews in Baghel (2013). Variability is the third lever. By Pollaczek-Khinchine and Kingman, halving repair-time variance trims waiting directly, and standard work procedures cost far less than another technician.

Queue length alone can mislead you. In the reneging example, the system stayed stable at $\rho = 1.2$, yet 36.6% of demand disappeared. Always track the abandonment fraction beside the line length. Scheduled maintenance helps only if it lowers failure load by more than it adds to crew workload, which is the trade-off behind Baghel (2017). And priorities, as Cobham (1954) showed, move waiting around without making it vanish.

Here's the short version for a manager. Measure your arrival rate and repair-time variance. Compute utilization at each station. Pool crews wherever skills allow. Keep utilization below about 85% at bottleneck stations. Check abandonment. Then use priorities only where downtime costs genuinely differ.

References

- [1]. Ancker, C. J., Jr., & Gafarian, A. V. (1963). Some queuing problems with balking and reneging. I. *Operations Research*, 11(1), 88–100. <https://doi.org/10.1287/opre.11.1.88>
- [2]. Baghel, K. P. S. (2013). Generalists vs. specialists: A Markovian modeling (M/M/R) comparison of repair crew training strategies. *Journal of Research in Applied Mathematics*, 1(1), 10–15. <https://doi.org/10.35629/0743-01011015>
- [3]. Baghel, K. P. S. (2014). Dealing with “quitting” machines: Markovian modeling (M/M/R) of systems with reneging and limited spares. *International Journal of Mathematics and Statistics Invention*, 2(1), 56–61. <https://doi.org/10.35629/4767-02015661>
- [4]. Baghel, K. P. S. (2017). Preventive vs. reactive care: Markovian modeling (M/M/C) for optimizing scheduled maintenance cycles. *International Journal of Mathematics and Statistics Invention*, 6(4), 26–31. <https://doi.org/10.35629/4767-06042631>
- [5]. Cobham, A. (1954). Priority assignment in waiting line problems. *Operations Research*, 2(1), 70–76. <https://doi.org/10.1287/opre.2.1.70>
- [6]. Erlang, A. K. (1909). The theory of probabilities and telephone conversations. *Nyt Tidsskrift for Matematik B*, 20, 33–39.
- [7]. Feller, W. (1968). *An introduction to probability theory and its applications* (Vol. 1, 3rd ed.). Wiley.
- [8]. Gross, D., Shortle, J. F., Thompson, J. M., & Harris, C. M. (2008). *Fundamentals of queueing theory* (4th ed.). Wiley.
- [9]. Haight, F. A. (1957). Queueing with balking. *Biometrika*, 44(3/4), 360–369. <https://doi.org/10.2307/2332869>
- [10]. Halfin, S., & Whitt, W. (1981). Heavy-traffic limits for queues with many exponential servers. *Operations Research*, 29(3), 567–588. <https://doi.org/10.1287/opre.29.3.567>
- [11]. Hopp, W. J., & Spearman, M. L. (2008). *Factory physics* (3rd ed.). McGraw-Hill.
- [12]. Jackson, J. R. (1957). Networks of waiting lines. *Operations Research*, 5(4), 518–521. <https://doi.org/10.1287/opre.5.4.518>
- [13]. Jain, M., Baghel, K. P. S., & Jadown, M. (2003). A repairable system with spares, state-dependent rates and additional repairman. *Journal of Rajasthan Academy of Physical Sciences*, 2(3), 181–190.
- [14]. Jain, M., Baghel, K. P. S., & Shekhar, C. (2003). M/G/1 system with standbys and common cause failure. *Journal of Rajasthan Academy of Physical Sciences*, 2(1), 45–56.
- [15]. Jain, M., & Maheshwari, S. (2008). Queueing network modelling of flexible manufacturing system using mean value analysis. *Applied Mathematical Modelling*, 32(5), 700–711. <https://doi.org/10.1016/j.apm.2007.02.007>
- [16]. Jain, M., & Preeti. (2009). N-policy for M/G/1 machine repair model with mixed standby components, degraded failure and Bernoulli feedback. *Journal of King Abdulaziz University-Engineering Sciences*, 21(2), 23–45.
- [17]. Jain, M., Sharma, G. C., & Baghel, K. P. S. (2002). Performance prediction of a flexible manufacturing system. *International Journal of Engineering (IJE), Transactions A: Basics*, 15(4), 355–364.
- [18]. Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. *The Annals of Mathematical Statistics*, 24(3), 338–354. <https://doi.org/10.1214/aoms/1177728975>
- [19]. Kingman, J. F. C. (1961). The single server queue in heavy traffic. *Mathematical Proceedings of the Cambridge Philosophical Society*, 57(4), 902–904. <https://doi.org/10.1017/S0305004100036094>
- [20]. Kleinrock, L. (1975). *Queueing systems: Vol. I. Theory*. Wiley.
- [21]. Little, J. D. C. (1961). A proof for the queueing formula: $L = \lambda W$. *Operations Research*, 9(3), 383–387. <https://doi.org/10.1287/opre.9.3.383>
- [22]. Palm, C. (1947). The distribution of repairmen in servicing automatic machines. *Industritidningen Norden*, 75, 75–80, 90–94, 119–125.
- [23]. Sharma, G. C., Shinde, V., & Jain, M. (2004). Performance modelling of machining system with mixed standby components, balking and reneging. *Journal of King Abdulaziz University-Engineering Sciences*, 16(1), 3–20.
- [24]. Taylor, J., & Jackson, R. R. P. (1954). An application of the birth and death process to the provision of spare machines. *Operational Research Quarterly*, 5(4), 96–104.
- [25]. Whitt, W. (1993). Approximations for the GI/G/m queue. *Production and Operations Management*, 2(2), 114–161. <https://doi.org/10.1111/j.1937-5956.1993.tb00094.x>