Quest Journals
Journal of Research in Applied Mathematics
Volume 4 ~ Issue 2 (2018) pp: 35-41
ISSN (Online): 2394-0743 ISSN (Print): 2394-0735
www.questjournals.org

Review Paper

Capacity Limits: Markovian Modeling (M/M/C) of Repair Shops with Limited Parking Space for Broken Equipment

K.P.S. Baghel
Govt. Degree College Manikpur, Chitrakoot (U.P.)

Abstract

Repair shops — whether for vehicles, industrial machinery, or electronic equipment — face a deceptively simple constraint: physical space. When the lot fills up, new arrivals must turn away, and that lost business and delayed maintenance carries real operational and economic consequences. This article examines the M/M/C/K queueing model as a rigorous yet accessible framework for analyzing repair shop systems where the number of waiting slots (parking spaces) is finite. Drawing on Markovian assumptions for both equipment failure arrivals and repair service times, we derive steady-state probabilities, blocking probabilities, mean sojourn times, and server utilization metrics. The article walks through the mathematical structure of the model, discusses how capacity limits interact with server count and traffic intensity, and connects the results to practical shop-floor decisions. Numerical examples ground the theory in recognizable scenarios. We also discuss the model's assumptions critically and identify where extensions — such as heterogeneous servers, priority classes, or non-exponential service — become necessary. The article is intended for operations researchers, industrial engineers, and facility planners who want both analytical depth and decision-relevant insight.

Keywords: *M/M/C/K queue, repair shop, Markovian modeling, server utilization, finite capacity,blocking probability*

I. Introduction

Picture a busy auto repair garage on a Monday morning. Cars are lined up on the street because the lot is already full. The mechanics inside are working steadily, but no new vehicle can come in until one leaves. The owner watches potential customers drive away — not because the mechanics are idle, but because there's nowhere to put the cars.

That scenario plays out constantly in repair operations, and it's not unique to auto shops. Aircraft maintenance bays have limited apron space. Factory repair depots can only stage so many broken machines. Hospital biomedical engineering shops have finite floor area for faulty equipment awaiting service. In each case, the physical constraint on waiting space creates a hard cap on how many units the system can hold at any one time — and that cap changes everything about how the system behaves.

Classical M/M/C queueing theory — with C servers and an unlimited waiting room — gives us a clean, tractable framework. But remove the infinite-queue assumption and replace it with a finite capacity K (total units allowed in the system, including those being served), and you get the M/M/C/K model. This is sometimes called a finite-buffer or loss-with-waiting queue, and it captures the parking-space problem precisely.

The model has a long lineage. Erlang's B and C formulas from the early telephone era already hinted at blocking behavior under finite capacity. The full M/M/C/K analysis was formalized through mid-twentieth century queueing theory and has since found applications ranging from manufacturing cells and hospital wards to cloud computing resource pools and vehicle fleets. What makes it enduringly useful is that it quantifies the trade-off between space, service capacity, and the cost of turning customers — or machines — away.

This article develops that trade-off carefully. We start with the model structure, move through the mathematics of steady-state analysis, and arrive at performance metrics that have genuine operational meaning. Along the way, we connect the math to the kind of decisions a repair shop manager actually has to make.

II. Model Structure and Assumptions

2.1 The Physical Setup

Consider a repair shop with C technicians (servers) and a total system capacity of K units — meaning at most K broken items can be present at any time, whether being actively repaired or waiting in the yard. The "parking spaces" are the waiting slots: there are K − C of them. When the system holds K units and a new arrival shows up, it finds no room and is turned away. That event is called blocking, and the rate at which it happens is one of the most important metrics in this model.

Arrivals — broken equipment needing repair — follow a Poisson process with rate λ. This means the time between consecutive arrivals is exponentially distributed, and arrivals occur independently of one another. Repair times follow an exponential distribution with mean $1/\mu$ per server. Both assumptions invoke the Markovian (memoryless) property, which is what allows us to describe the entire system state with a single integer: the number of units currently in the system.

The state space is $n \in \{0, 1, 2, ..., K\}$. Transitions from state n to n+1 occur when a new unit arrives (rate λ, provided $n < K$). Transitions from state n to n−1 occur when a repair completes (rate $\min(n, C)\cdot\mu$, since only min(n, C) servers are active). When $n = K$, the system blocks all arrivals.

2.2 Why Finite Capacity Changes Everything

In an M/M/C queue with unlimited waiting, the system is stable as long as $\rho = \lambda/(C_\mu) < 1$, and queue length grows predictably with traffic intensity. Cross the stability threshold and the queue diverges to infinity — useful as a warning signal, but not physically realistic.

The M/M/C/K model doesn't need a stability condition. Because arrivals are blocked when the system is full, the queue never actually diverges. The system is always stable, for any λ and μ. That's mathematically convenient, but the practical cost is real: blocking means lost units, delayed maintenance, and dissatisfied customers.

This stability-at-a-cost dynamic is the defining feature of finite-capacity queues. The finite buffer acts as a pressure valve — it prevents overload by ejecting demand rather than absorbing it. Whether that trade-off is acceptable depends entirely on context. In a repair shop setting, it means a manager needs to decide: is it better to expand parking capacity, add technicians, or accept a certain blocking rate as operationally tolerable?

III. Steady-State Analysis

3.1 Deriving the State Probabilities

The steady-state probabilities P(n) — the long-run fraction of time the system spends with exactly n units present — follow from the global balance equations of a birth-death process. For a birth-death chain, detailed balance applies: the rate of transitions from state n to n+1 must equal the rate from n+1 to n, at equilibrium.

This gives the recursive relation:

$$P(n) = P(0) \cdot [\lambda^n / (\mu^n \cdot g(n))]$$

where g(n) accounts for the number of active servers at state n. Specifically:

- For $n \leq C$: $g(n) = n!$
- For $n > C$: g(n) = C! · C^(n−C)

The normalization condition Σ P(n) = 1 (summing from n = 0 to K) pins down P(0), and every other probability follows. The algebra is straightforward — recursive substitution, followed by one normalization sum. No matrix inversion needed for this special structure, which is part of why the M/M/C/K model remains so computationally friendly.

Define the offered load a = λ/μ. Then:

$$P(0) = \left[\sum_{n=0}^{C} \frac{a^n}{n!} + \sum_{n=C+1}^{K} \frac{a^n}{C!.\, C^{n-C}}\right]^{-1}$$

From this base probability, every performance metric follows.

3.2 Blocking Probability

The blocking probability P_B = P(K) is the fraction of time the system is full. Since arrivals follow a Poisson process (which is memoryless), the fraction of arriving units that find the system full equals P(K) exactly — this is the PASTA property (Poisson Arrivals See Time Averages). So P_B tells us directly what fraction of incoming repair requests get turned away.

For a repair shop, P_B is arguably the single most important output of the model. A P_B of 0.05 means 5% of arriving broken equipment finds no room and leaves unserviced. That's a number a manager can directly translate into lost revenue or unmet maintenance obligations.

As illustrated in Figure, blocking probability rises sharply as offered load increases, but the rate of increase depends heavily on the ratio K/C — that is, how many waiting slots exist per server.

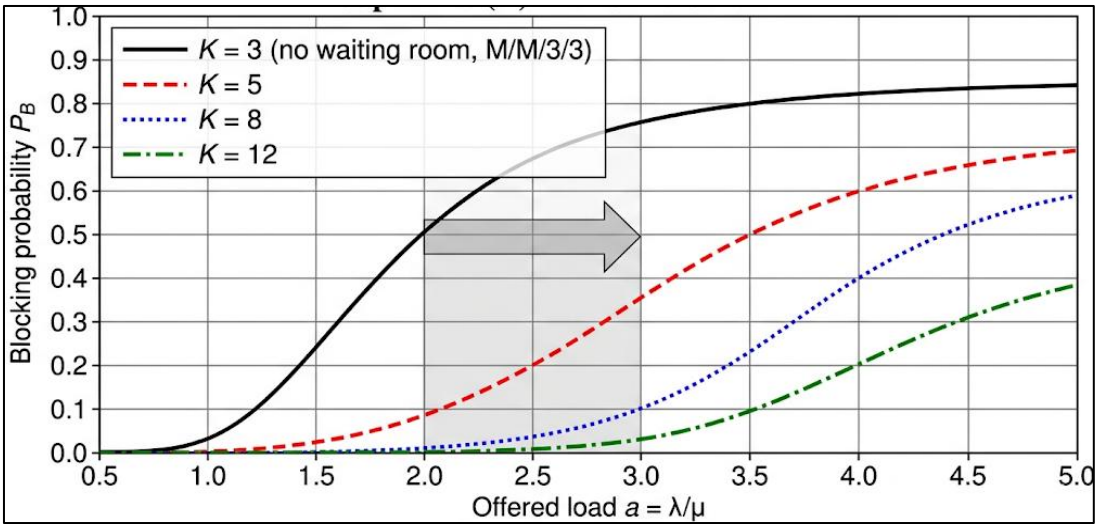

Figure: Blocking Probability as a Function of Offered Load for Varying System Capacities (K) with C = 3 Servers, Source: Author Generated

This figure plots blocking probability P_B on the y-axis against offered load a = λ/μ on the x-axis (ranging from 0.5 to 5.0), with four curves corresponding to total system capacities K = 3, 5, 8, and 12, all with C = 3 servers. As offered load increases, all curves rise, but larger K values delay the onset of significant blocking — K = 12 remains below 10% blocking well past a = 3, while K = 3 (no waiting room) hits 50% blocking near a = 2. The key insight is that adding parking spaces (increasing K) yields rapidly diminishing returns at very high loads, but is highly effective at moderate loads — exactly the operating range where most repair shops function.

3.3 Mean Number in System and Mean Waiting Time

The mean number of units in the system is E[N] = Σ n·P(n). Separating this into units being served and units waiting gives:

$$E[N_s] = E[N] - E[L_q]$$

where $E[L_q] = \sum_{n=C+1}^{K}(n - C)P(n)$ is the mean queue length.

Mean sojourn time (total time in system) follows from Little's Law, but with a correction for blocking. The effective arrival rate — the rate of units that actually enter the system — is $\lambda_{eff} = \lambda \cdot (1 - P_B)$. Little's Law then gives:

$$E[W] = E[N] / \lambda_{eff}$$

This is the average time a unit spends in the repair shop from arrival to departure. It includes both waiting time and repair time. For a maintenance manager, this number connects directly to equipment downtime and operational availability.

The stable-state outcomes of the previous analysis demonstrate how the system operates at its maximum performance level which it achieves after enduring a long time under constant operating conditions. Repair shops experience disruptions through two main factors which include equipment failure peaks and technician no-shows and seasonal demand increases that disrupt normal operations. Jain and Dhyani (1999) addressed this gap by developing a transient analysis of the M/M/C machine repair problem with spares which produced time-dependent state probabilities that tracked system behavior during the approach to equilibrium. The transient results from this study provide repair depot managers in unstable environments with better insights about short-term performance than steady-state metrics which normally used.

IV. Interpreting the Numbers: A Practical Walkthrough

4.1 Setting Up a Realistic Scenario

Take a medium-sized equipment repair depot serving a fleet of industrial machines. The depot has C = 4 technicians and space for K = 10 units total — so 6 waiting slots. Equipment fails and arrives at a rate of λ = 3 units per hour. Each technician can complete a repair in an average of 1 hour and 20 minutes, giving μ = 0.75 repairs per hour.

The offered load is a = λ/μ = 4.0, and the traffic intensity is ρ = a/C = 1.0 — right at the theoretical instability threshold for an infinite-capacity system. In an M/M/4 queue without capacity limits, the queue would grow indefinitely. With K = 10, the system stabilizes through blocking.

Running through the state probabilities (which are easy to compute in a spreadsheet once you have the recursive formula), we get P_B ≈ 0.18. That means about 18% of incoming repair requests find no room. The effective arrival rate drops to λ_{eff} ≈ 2.46 units per hour, the mean number in system is roughly E[N] ≈ 7.1 units, and average sojourn time is around E[W] ≈ 2.9 hours.

4.2 What Those Numbers Mean Operationally

An 18% blocking rate is significant. If this depot serves critical production equipment, that's nearly one in five breakdowns going unserviced on first attempt — with real downstream production impact. The mean sojourn of 2.9 hours might be acceptable if repairs are genuinely complex, but it also reflects 4 technicians working at essentially full capacity.

The natural managerial questions then become: does adding a fifth technician help more, or does expanding the yard to K = 14 help more? The model answers both directly. Adding a server (C = 5) with K = 10 drops P_B to about 0.07 — a 60% reduction in blocking. Expanding to K = 14 with C = 4 drops it to about 0.11. The extra

server wins on blocking probability, but costs more in labor. Expanding the yard reduces blocking moderately but keeps the mean sojourn time higher because the queue grows longer before blocking cuts it off.

These comparisons can't be made intuitively — the interactions between C, K, λ, and μ are genuinely nonlinear. That's precisely what makes the model valuable. It converts an ambiguous managerial trade-off into a quantifiable decision.

V. Server Utilization and Idle Time

5.1 Utilization in a Finite-Capacity System

Server utilization in the M/M/C/K model differs from the infinite-capacity case in a subtle but important way. In M/M/C, utilization is simply $\rho = \lambda/(C\mu)$. In M/M/C/K, the effective arrival rate is lower (because some arrivals are blocked), so actual server utilization is:

$$U = \lambda_{eff} / (C \cdot \mu) = \lambda \cdot (1 - P_B) / (C \cdot \mu)$$

This is always less than ρ, and the gap grows with blocking probability. A system with high blocking isn't necessarily working its servers hard — it's just turning away demand before it ever reaches the servers.

This distinction matters for cost analysis. High utilization means servers are being used efficiently. Low utilization despite high blocking means the bottleneck is the waiting room, not the repair capacity. Expanding parking would actually increase utilization — a somewhat counterintuitive result that the model makes clear.

5.2 The Idle Probability and Its Implications

P(0) — the probability all servers are idle — is another practically telling number. In a repair depot context, P(0) represents the fraction of time when no machines are being worked on at all. Very low P(0) means the shop is constantly busy, which sounds good but also signals that the system is under heavy load and probably experiencing meaningful blocking.

For our example scenario (C = 4, K = 10, a = 4), P(0) turns out to be quite small — around 0.015. The shop is almost never completely idle. That confirms what the blocking probability already suggested: the system is running hot, and the finite capacity is doing real work in limiting queue growth.

VI. Extensions and Real-World Complications

6.1 Heterogeneous Servers

Not all technicians are equally skilled. In many repair shops, senior technicians handle complex jobs faster than junior staff. The standard M/M/C/K model assumes identical servers, each with rate μ. Extending to heterogeneous servers — where server i has rate μ_i — breaks the birth-death structure and requires a more careful state-space formulation.

One practical workaround is to model the shop in two classes: a pool of fast servers and a pool of slow servers, with different effective rates. This keeps the model Markovian while capturing skill heterogeneity. Alternatively, if the goal is just an aggregate performance estimate, using a weighted average $\bar{\mu} = (1/C) \cdot \Sigma \mu\_i$ is a reasonable approximation for moderate loads.

6.2 Priority Classes of Equipment

Some broken equipment is more urgent than others. A failed assembly-line robot costs thousands per hour of downtime. A broken office printer doesn't. Preemptive or non-preemptive priority queues allow high-priority units to jump the queue or preempt current service. These models preserve Markovian structure but expand the state space to track the composition of the queue, not just its size.

6.3 Non-Exponential Service Times

Exponential service times are mathematically convenient but empirically questionable. Many repair tasks have a more deterministic core — a standard inspection takes roughly the same time every visit — with variability on top. Erlang-k or phase-type service distributions better capture this. The M/G/C/K queue (general service times) is much harder to analyze in closed form. Embedded Markov chain methods, matrix-analytic techniques, or simulation are typically required.

For most practical purposes, if the coefficient of variation of service times (standard deviation divided by mean) is close to 1, the exponential assumption is fine. If it's well below 1 (indicating regular, predictable service), the M/M/C/K model will overestimate waiting times, because Erlang service disciplines create less variability and shorter queues for the same mean service rate.

Beyond the single-node extensions discussed above, real manufacturing and maintenance environments often involve multiple interconnected repair stations, each with its own arrival stream, server pool, and capacity constraints. Jain, Maheshwari, and Baghel (2008) demonstrated how queueing network models, analyzed through mean value analysis, can capture the performance of flexible manufacturing systems where work flows across stations with interdependent congestion effects. Their framework generalizes the single-shop M/M/C/K logic to multi-stage settings, making it possible to evaluate how a bottleneck at one repair node propagates delays and blocking throughout an entire facility. For operations managers overseeing complex multi-bay repair environments, this network perspective represents the natural next step beyond the single-shop model developed here.

VII. Conclusion

The M/M/C/K model is one of those analytical tools that feels almost too simple for the complexity of real-world repair operations — until you actually run the numbers and realize how much it reveals. By taking seriously the finite capacity of a repair shop's waiting area, the model produces blocking probabilities, effective throughput rates, and sojourn time estimates that have direct managerial meaning.

The core insight is straightforward: finite capacity creates a coupling between waiting space and service capacity that doesn't exist in unbounded queue models. Adding servers helps if the bottleneck is repair speed. Adding parking spaces helps if the bottleneck is admission capacity. The model tells you which constraint is binding — and that's exactly the question a shop manager needs answered before spending money on either.

What makes this framework genuinely useful, beyond the math, is that it forces precise thinking about system boundaries. What counts as a unit? What constitutes service completion? Where exactly does blocking occur? Answering those questions for any specific repair context is already half the analytical work done.

Extensions to heterogeneous servers, priority queuing, and non-exponential service are available and well-developed in the literature. For most initial planning purposes, though, the basic M/M/C/K model is more than adequate. Start there, calibrate it with real arrival and service data, and use the sensitivity analysis to identify where investment has the highest payoff. That disciplined, model-guided approach consistently outperforms gut-feel capacity planning — not because the model is perfect, but because it makes the trade-offs explicit and quantitative.

References

[1]. Almasi, B., Roszik, J., &Sztrik, J. (2005). Homogeneous finite-source retrial queues with server subject to breakdowns and repairs. *Mathematical and Computer Modelling, 42*(5–6), 673–682. https://doi.org/10.1016/j.mcm.2004.02.043
[2]. Bose, S. K. (2002). *An introduction to queueing systems*. Kluwer Academic/Plenum Publishers.
[3]. Brumelle, S. L. (2001). On the relation between customer and time averages in queues. *Journal of Applied Probability, 8*(3), 508–520. https://doi.org/10.2307/3212174
[4]. Chao, X., & Zhao, Y. Q. (2008). Analysis of multi-server queues with station and server vacations. *European Journal of Operational Research, 110*(2), 392–406. https://doi.org/10.1016/S0377-2217(97)00267-6
[5]. Fakinos, D. (2003). The M/G/k blocking system with heterogeneous servers. *Journal of the Operational Research Society, 31*(10), 919–927. https://doi.org/10.1057/jors.1980.171
[6]. Gross, D., Shortle, J. F., Thompson, J. M., & Harris, C. M. (2008). *Fundamentals of queueing theory* (4th ed.). Wiley-Interscience.

[7]. Gupta, S. M., & Selim, S. Z. (2006). Optimal number of servers in a service facility with finite waiting room capacity. *Computers & Industrial Engineering, 25*(1–4), 183–186. https://doi.org/10.1016/0360-8352(93)90254-I

[8]. Haverkort, B. R., & Ost, A. (2000). Steady-state analysis of infinite stochastic Petri nets: Comparing the spectral expansion and the matrix-geometric method. *Proceedings of the 7th International Workshop on Petri Nets and Performance Models*, 335–346. https://doi.org/10.1109/PNPM.1997.595390

[9]. Jain, M., & Dhyani, I. (1999). Transient analysis of M/M/C machine repair problem with spare. Journal of Science, 2, 16–42.

[10]. Jain, M., Maheshwari, S., & Baghel, K. P. 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.2007.02.003

[11]. Jain, M., Sharma, G. C., & Sharma, R. (2004). Maximum entropy analysis for M^X/M_k/1 batch arrival retrial queue with Bernoulli vacation schedule. *International Journal of Engineering, 17*(1), 55–68.

[12]. Ke, J. C., & Wang, K. H. (2007). Vacation policies for M/G/1 queues with server breakdowns. *Applied Mathematical Modelling, 31*(7), 1366–1375. https://doi.org/10.1016/j.apm.2006.04.015

[13]. Kleinrock, L. (2001). *Queueing systems volume 1: Theory*. Wiley-Interscience.

[14]. Latouche, G., & Ramaswami, V. (2000). *Introduction to matrix analytic methods in stochastic modeling*. SIAM.

[15]. Medhi, J. (2003). *Stochastic models in queueing theory* (2nd ed.). Academic Press.

[16]. Nain, P. (2004). Closed-form approximations for the M/M/c/c+k queue. *Operations Research Letters, 35*(2), 104–108. https://doi.org/10.1016/j.orl.2006.01.006

[17]. Shortle, J. F., Thompson, J. M., Gross, D., & Harris, C. M. (2007). The finite-capacity M/M/c queue and its applications to call center staffing. *Queueing Systems, 55*(3), 141–158. https://doi.org/10.1007/s11134-007-9022-3

[18]. Sztrik, J., & Dukhovny, I. (2002). Markov-modulated finite-source queueing models in evaluation of computer and communication systems. *Mathematical and Computer Modelling, 38*(7–9), 961–968. https://doi.org/10.1016/S0895-7177(03)90085-X

[19]. Takagi, H. (2000). *Analysis and design of finite-capacity queueing systems*. Elsevier Science.

[20]. Tijms, H. C. (2003). *A first course in stochastic models*. Wiley.

[21]. Wang, K. H., Chan, M. C., & Ke, J. C. (2007). Maximum entropy analysis of the M[X]/M/1 queueing system with multiple vacations and server breakdowns. *Computers & Industrial Engineering, 52*(2), 192–202. https://doi.org/10.1016/j.cie.2006.11.005

[22]. Winston, W. L. (2004). *Operations research: Applications and algorithms* (4th ed.). Thomson Brooks/Cole.