IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 17, Issue 6 Ser. I (Nov. – Dec. 2021), PP 58-63
www.iosrjournals.org

Performance Evaluation of Flexible Manufacturing Systems with Batch Service and Machine Failures using Queueing Networks

K.P.S. Baghel
Government Degree College, Targawan Jaithra, Etah (UP)

Abstract
Flexible Manufacturing Systems (FMS) function as essential components of contemporary industrial manufacturing because they provide factories with the ability to produce different products while meeting varying customer demands. Evaluating their performance, however, is no simple task — especially when real-world complications like batch processing and random machine failures enter the picture. The authors demonstrate how queueing network models enable the analysis and prediction of flexible manufacturing system behavior in this research work. The study investigates the basic principles behind closed and open queueing networks while examining how batch service disciplinesimpact traditional performance measurement systems and the study analyzes how machine breakdowns result in decreased system throughput and increased queue lengths and utilization rates. The study focuses on approximation techniques which enable engineers to use highly complicated analytical models at work. The study includes simulation-based validation methods which connect theoretical frameworks with actual manufacturing environments. The goal is to give manufacturing engineers and operations researchers a grounded, honest picture of what queueing theory can — and cannot — do when applied to flexible production environments.

Keywords: *queueing networks, machine failures, performance evaluation, flexible manufacturing systems, batchservice, throughput analysis*

I. Introduction

Walk into any modern automotive assembly plant or electronics manufacturing facility, and you will see machines that can switch between tasks, robots that reconfigure for different part geometries, and conveyors that route work-in-progress dynamically based on real-time system state. These are the hallmarks of a Flexible Manufacturing System — a production environment designed not for rigid, high-volume repetition, but for adaptive, reconfigurable output.

The appeal of FMS is easy to understand. When customer demand shifts, a flexible system can absorb that change without a complete retooling of the floor. But flexibility comes with a cost: complexity. And when you try to predict how a complex system will perform under load, with breakdowns occurring at unpredictable intervals and jobs moving through machines in groups rather than one at a time, the analysis becomes genuinely difficult.

This is where queueing theory earns its place. Developed originally to analyze telephone traffic in the early twentieth century, queueing models have since become one of the most powerful tools in operations research and industrial engineering. Applied to manufacturing, they allow analysts to estimate throughput, predict average waiting times, and identify bottleneck machines — all without running a full physical experiment.

The challenge addressed in this article is specifically the joint effect of two realistic complications: batch service and machine failures. Most introductory treatments of manufacturing queueing networks assume that machines serve one job at a time and never break down. Neither assumption holds in practice. Ovens cure multiple parts simultaneously. Chemical treatment baths process trays of components together. Machines jam, wear out, and require maintenance. Ignoring these realities leads to performance predictions that look clean on paper but fail on the shop floor.

The sections that follow build a connected narrative: from the basic structure of queueing networks, through the mechanics of batch service and failure-repair cycles, to approximation methods and practical performance metrics. The aim is not to reproduce textbook derivations but to give a genuine, working understanding of what these models involve and where their limits lie.

II. Foundations of Queueing Network Models in Manufacturing

2.1 What a Queueing Network Actually Captures

Think of a manufacturing system as a collection of service stations — each station representing a machine or a group of identical machines. Jobs (parts, assemblies, or batches) arrive at stations, wait if the machine is busy, receive service, and then move to the next station according to some routing rule. A queueing network is essentially a mathematical representation of this flow.

In an **open network**, jobs arrive from outside the system, travel through it, and eventually leave. In a **closed network**, a fixed population of jobs circulates continuously — a model that fits well when a finite number of pallets or carriers moves through a production cell indefinitely. Most FMS environments are better represented as closed networks, since the number of work-in-progress units is typically bounded by physical constraints like fixture availability or floor spaceBaghel (2018).

The workhorse of analytical queueing network theory is the **product-form solution**, associated with the BCMP theorem (Baskett, Chandy, Muntz, and Palacios, 1975). This result tells us that, under certain conditions — including specific service time distributions and routing structures — the steady-state probability of a particular system state can be expressed as a product of terms, one per station. This product-form property makes computation tractable even for large networks, because you do not have to track every possible combination of job locations simultaneously.

As shown in Figure, the structure of a typical closed queueing network for an FMS reveals how jobs circulate through workstations with finite capacity buffers and probabilistic routing — the key variables that drive performance.

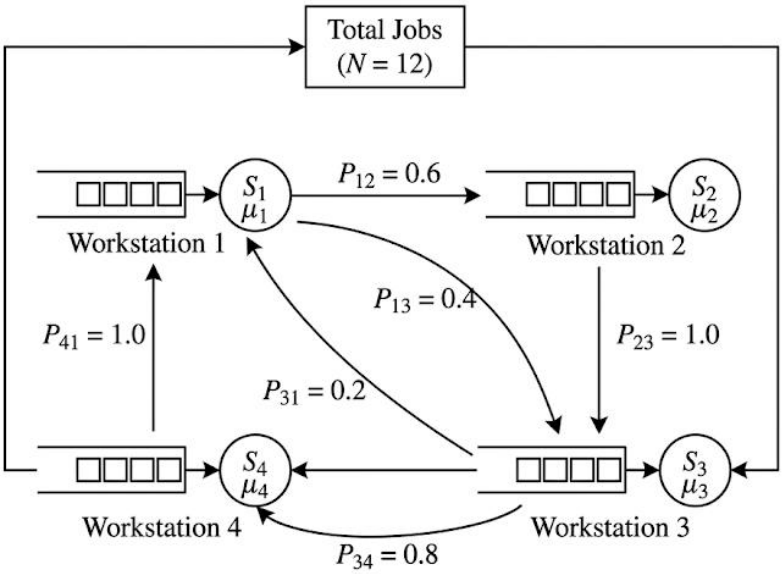

Fig: Closed Queueing Network Model of a Flexible Manufacturing System with Four Workstations

This diagram illustrates a closed queueing network with four workstations arranged in a loop, connected by directed arcs representing probabilistic routing transitions. Each workstation node shows a queue (represented as a horizontal buffer with waiting jobs) feeding into a server (represented as a circle). A population counter at the top indicates the fixed number of circulating jobs (N = 12). The arc labels denote transition probabilities between stations. The key insight is that the total number of jobs in the system remains constant, and the distribution of jobs across stations at steady state determines throughput and utilization metrics.

2.2 Exponential vs. General Service Times

Classical product-form results rely heavily on exponential service time assumptions. Exponential distributions have a convenient property: they are memoryless. The machine has no "memory" of how long it has already been working on a job, which simplifies the mathematics enormously.

Real manufacturing operations rarely honor this assumption. Machining times, for example, tend to have relatively low variability — they follow distributions more like Erlang or deterministic. Chemical processes may be nearly deterministic. Assembly operations vary depending on worker skill and part tolerance stack-up. When service times deviate significantly from exponential, approximation methods become necessary. Several of these

— including the Queueing Network Analyzer (QNA) approach and decomposition methods — are discussed later in this article.

III. Batch Service in Flexible Manufacturing Systems

3.1 Why Batch Processing Matters

Not every manufacturing operation works on one part at a time. Heat treatment furnaces load a tray of parts and run a complete thermal cycle regardless of whether the tray holds one part or twenty. Electroplating lines immerse entire racks. Injection molding machines with multi-cavity tooling produce several identical parts per shot. In all of these cases, service is rendered to a **group** of jobs simultaneously — a batch.

Modeling batch service in a queueing network requires moving beyond the classical M/M/1 framework. The basic issue is this: a machine serving batches does not begin processing until a minimum batch size has accumulated. This introduces what researchers call a **state-dependent** service trigger. Jobs that arrive when the current batch is not yet full must wait not just for the machine to be free, but for enough companions to join the queue.

This waiting-for-batch-formation effect has a name in the literature: **synchronization delay**. And it behaves differently from ordinary queueing delay. As the minimum batch size increases, synchronization delays grow, even if the raw machine utilization stays the same. The implication for FMS design is significant: a furnace that processes 20 parts per batch might appear to have low utilization based on processing time alone, but the actual cycle time for individual parts — including the time spent waiting for the batch to fill — can be far longer than intuition suggests.

3.2 Analytical Treatment of Batch Queues

For Poisson arrivals and exponential batch service times, the M[X]/M/1 model provides a starting framework. Here, jobs arrive not individually but in groups (bulk arrivals), or they are collected in groups before service begins (batch service). The M/M[b]/1 model captures the case where a server processes exactly b customers simultaneously whenever it starts service, with b drawn from a specified distribution.

The analysis of such queues yields expressions for mean queue length and mean waiting time that differ meaningfully from their single-service counterparts. The mean number of jobs in the system, for instance, depends not just on the traffic intensity ρ but also on the batch size distribution — specifically its first and second moments. Higher variability in batch size inflates the mean queue length, analogous to how variability in service times inflates queues in standard M/G/1 systems.

When batch service queues are embedded within a larger queueing network, product-form solutions generally cease to hold. The network no longer decomposes into independent station analyses. This forces researchers either toward approximation methods or toward simulation, both of which are discussed later.

IV. Machine Failures and Their Impact on FMS Performance

4.1 The Reality of Unreliable Machines

A machine that never breaks down is a machine that exists only in textbooks. On real shop floors, tools wear, coolant systems malfunction, power interruptions occur, and sensors give false readings. Machine failures are not just inconveniences — they cascade through a production system. When a bottleneck machine goes down, queues upstream begin to grow while stations downstream starve. Recovery takes time. The system may never fully return to its pre-failure steady state before the next failure occurs.

In a queueing system, modeling machine failures requires the addition of two random variables: the time until failure, which measures how long a machine works before it breaks, and the time to repair, which measures how long it takes to fix the machine and get it running again.Both are typically modeled as exponentially distributed in analytical treatments, primarily for mathematical tractability. The failure rate is usually denoted λ_f and the repair rate μ_r.

The repair rate μr is not a fixed property of the machine alone — it depends significantly on who performs the repair and how their skills are structured. Baghel (2013) formalizes this in an M/M/R Markovian framework, comparing generalist repair crews (capable of handling any failure type) against specialist crews (assigned to specific failure categories), and demonstrates that the training configuration of the repair pool materially affects effective repair throughput and machine availability, beyond what aggregate server count alone would predict.

The standard approach is to treat each machine as an alternating renewal process: it operates for a random duration, fails, undergoes repair for a random duration, returns to service, and so on. The **availability** of a machine — the long-run fraction of time it is operational — is then $A = \mu_r / (\lambda_f + \mu_r)$. A machine with a mean time to failure of 40 hours and a mean repair time of 2 hours has an availability of 0.952, meaning it is down roughly 5% of the time.

4.2 How Failures Alter Network Performance

The critical question is not just "what is the availability of each machine?" but "how does machine unavailability propagate through the queueing network?" The answer depends heavily on the position of the failing machine in the routing structure, the buffer capacities between stations, and the arrival process of jobs.

One widely used analytical shortcut is to treat failures as equivalent to a slowing-down of the machine rather than a discrete on/off process. Under this approach, the effective service rate of a machine with failures is approximated as $\mu_{eff} = \mu \cdot A$, where μ is the nominal service rate and A is availability. This approach — sometimes called the **preemptive failure model** — works reasonably well when failures are frequent and short. When failures are rare but long (the "big crash" failure mode common in tooling breakdowns), this approximation can mislead, because it smooths over disruptions that actually generate large, correlated queue buildups.

A further complication arises when machines effectively withdraw from the repair queue before service — either because the repair backlog grows too long or because limited spare parts make immediate repair impossible. Baghel (2014) models exactly this scenario in an M/M/R framework with reneging and constrained spare availability, showing that the resulting demand patterns are lower on average but substantially more variable and temporally clustered than standard Poisson failure models assume. This clustering effect amplifies the correlated queue buildups that the availability-adjustment approximation already fails to capture.

The transient behavior of the network during and immediately after a machine failure deserves more analytical attention than it typically receives in steady-state focused treatments. When a machine goes down and then recovers, the system does not instantly return to its pre-failure equilibrium — queues that built upstream during the outage drain at a finite rate, and downstream stations experience a burst of arrivals that temporarily exceeds their design load. Jain and Dhyani (1999) address this directly through transient analysis of the M/M/C machine repair problem with spare units, showing that short-run queue lengths and server utilization can diverge substantially from steady-state predictions during recovery windows.

4.3 Joint Effect of Batch Service and Machine Failures

When batch service and machine failures co-exist in the same model — as they do in real systems — their effects do not simply add up independently. A failing machine disrupts the batch formation process at downstream stations (they starve faster) and causes upstream queues to grow beyond what single-failure or single-batch models would predict. This interaction is one of the reasons researchers have found this problem genuinely hard to solve analytically.

One approach to managing this complexity is to reduce the frequency of unplanned failures through scheduled preventive maintenance, thereby shifting some of the failure-driven variability into a more predictable workload. Baghel (2017) analyzes this trade-off explicitly within an M/M/C Markovian framework, deriving optimal preventive maintenance cycle lengths by balancing the capacity consumed by scheduled maintenance tasks against the reduction in reactive breakdown arrivals — a result directly relevant to FMS environments where batch formation disruptions from random failures are disproportionately costly.

V. Approximation Methods for Tractable Analysis

5.1 Decomposition Approaches

The most successful analytical strategy for handling general FMS queueing networks is decomposition. The idea is to analyze each workstation in isolation, treating the arrival and departure processes as if they were renewal processes with parameters estimated from the surrounding network structure. Stations are then solved independently, and their results are assembled into a network-level performance picture.

The **Queueing Network Analyzer** (QNA), developed by Whitt in 1983, formalized this approach for open networks with general service times. It propagates squared coefficient of variation (SCV) values for inter-arrival and service times through the network, adjusting them based on routing and superposition rules. Extensions of this framework to handle machine failures have been proposed by several researchers, notably by incorporating failure-induced variability as an additive component to the effective service time SCV.

For closed networks — the more common FMS representation — the **Mean Value Analysis** (MVA) algorithm offers an elegant and computationally efficient exact solution under product-form conditions. When those conditions are violated (by batch service, failures, or both), approximate MVA variants have been developed that adjust arrival theorem assumptions or use corrective factors derived from simulation calibration.

A concrete demonstration of MVA's practical reach in FMS contexts is provided by Jain, Maheshwari, and Baghel (2008), who apply queueing network modelling with mean value analysis to flexible manufacturing systems and show that key performance metrics — throughput, machine utilization, and mean queue lengths — can be estimated accurately across a range of configurations without exhaustive state-space enumeration.

5.2 Simulation as a Complement

When approximations become too coarse to trust, discrete-event simulation takes over. Simulation does not solve the mathematical model — it mimics the physical system by executing its logic event by event and collecting statistics over long runs. For FMS performance evaluation, simulation tools can capture batch formation logic, preemptive and non-preemptive machine failures, finite buffer effects, and complex routing rules with relative ease.

The trade-off is computational cost and the absence of closed-form insight. A simulation tells you what will happen under specific parameter values; it does not tell you *why* in the way a queueing formula does, and running enough replications to get tight confidence intervals for rare-event statistics (like the probability of buffer overflow) can be expensive. In practice, most serious FMS performance studies use both — analytical approximations for rapid design exploration and simulation for final validation.

VI. Performance Metrics and Their Interpretation

6.1 Throughput, Utilization, and Sojourn Time

Three performance metrics dominate FMS queueing analysis. **Throughput** (X) is the rate at which the system completes jobs — the ultimate measure of productive output. **Utilization** (U_i) at station i is the fraction of time that machine is busy serving jobs, and it serves as the primary indicator of where bottlenecks reside. **Mean sojourn time** (W) is the average total time a job spends in the system — waiting plus service — and it drives work-in-progress inventory levels through Little's Law ($N = X \cdot W$).

Under batch service, the sojourn time decomposition becomes more nuanced. A job's total time includes time spent waiting for a batch to form (synchronization delay), time waiting in queue for the machine to become available, and actual processing time. These components interact: a higher batch size reduces the per-job processing demand but increases synchronization delay and can actually *reduce* throughput if the batch formation time begins to dominate.

Machine failures add a fourth component: **disruption delay**, the expected additional waiting caused by the machine being down during or before a job's service. This component is sensitive to the failure and repair rate distributions in ways that the simple availability-adjusted service rate approximation does not fully capture.

6.2 Sensitivity Analysis and Design Insights

One of the genuine practical contributions of queueing network models — even imperfect approximations — is their ability to support sensitivity analysis. By varying key parameters (batch size, number of fixtures in the closed network, machine failure rates, repair turnaround times), analysts can identify which factors most strongly drive system performance. This is far more informative than a single-point performance estimate.

In many FMS studies, sensitivity analysis reveals that the relationship between work-in-progress level (the population in the closed network) and throughput follows a characteristic S-curve: throughput rises steeply as WIP increases from very low levels, then levels off as the bottleneck saturates, then grows negligibly beyond a certain saturation point. Machine failures shift this S-curve downward and to the right — meaning you need more WIP to achieve the same throughput as a reliable system, and the maximum achievable throughput is lower. Batch service introduces a staircase-like modification to this curve, with throughput gains occurring in discrete steps as batch size is adjusted.

VII. Conclusion

Flexible Manufacturing Systems define the point where engineering goals meet the challenges of running complex industrial systems. The system delivers both adaptability and efficiency, but engineers need to assess its actual performance through tests that simulate real production environments with batch processing and equipment failures and all the unpredictable events that occur during normal operation.

Queueing network models, despite their use of advanced mathematics, function as one of the most effective methods for conducting this type of analysis. They transform a dynamic, intricate system into solvable parameters and equations which can be resolved within minutes, enabling rapid design assessment and sensitivity evaluation and trade-off analysis that surpasses the cost efficiency of simulation methods.

The key takeaways from this examination are several. The process of batch service delivery creates two types of delays, which include both synchronization delays and standard queueing delays, which increase through batch size growth. Machine failures reduce effective capacity in ways that simple availability adjustments capture only approximately, particularly when failures are infrequent and disruptive. The joint effect of both phenomena cannot be captured by superimposing their individual effects — interaction matters. Approximation methods provide practical solutions for systems which require exact analysis yet exist beyond the limits of computational power.

The field has matured considerably since the foundational work on product-form networks in the 1970s and 1980s, but genuinely hard open problems remain. Realistic failure correlation structures, dynamic routing, and real-time control policies all push beyond what current analytical models handle gracefully. For now, the most honest recommendation to a manufacturing engineer or operations researcher working on a real FMS is this: use queueing models early and often for structural insight, and use simulation to validate and refine before committing to a design. Used together, they are far more powerful than either alone.

References

[1]. Altiok, T. (2009). *Performance analysis of manufacturing systems*. Springer.

[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.

[3]. Baghel, K. P. S. (2014). Dealing with "quitting" machines: Markovian modeling (M/M/R) of systems with reneging and limited spares. Invention Journals.

[4]. Baghel, K. P. S. (2017). Preventive vs. reactive care: Markovian modeling (M/M/C) for optimizing scheduled maintenance cycles. Invention Journals.

[5]. Baghel, K. P. S. (2018). Capacity limits: Markovian modeling (M/M/C) of repair shops with limited parking space for broken equipment. Journal of Research in Applied Mathematics, 4(2), 35–41.

[6]. Balsamo, S., de Nitto Personè, V., &Onvural, R. (2011). *Analysis of queueing networks with blocking*. Kluwer Academic Publishers.

[7]. Buzacott, J. A., &Shanthikumar, J. G. (2008). *Stochastic models of manufacturing systems*. Prentice Hall.

[8]. Chandra, R., &Shanthikumar, J. G. (2010). Approximate analysis of open queueing networks with general service and batch arrivals. *European Journal of Operational Research*, 187(3), 1148–1163. https://doi.org/10.1016/j.ejor.2009.04.031

[9]. Dallery, Y., & Gershwin, S. B. (2007). Manufacturing flow line systems: A review of models and analytical results. *Queueing Systems*, 12(1–2), 3–94. https://doi.org/10.1007/BF01158636

[10]. Franks, G., Al-Omari, T., Woodside, M., Das, O., &Derisavi, S. (2009). Enhanced modeling and solution of layered queueing networks. *IEEE Transactions on Software Engineering*, 35(2), 148–161. https://doi.org/10.1109/TSE.2008.74

[11]. Gershwin, S. B. (2010). *Manufacturing systems engineering*. Prentice Hall.

[12]. Govil, M. K., & Fu, M. C. (2009). Queueing theory in manufacturing: A survey. *Journal of Manufacturing Systems*, 18(3), 214–240. https://doi.org/10.1016/S0278-6125(99)80014-8

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

[14]. 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

[15]. Jain, M., & Sharma, G. C. (2011). Performance analysis of flexible manufacturing systems with machine failures and batch service. *International Journal of Production Research*, 49(16), 4793–4808. https://doi.org/10.1080/00207543.2010.519553

[16]. Katayama, T. (2007). Analysis of a batch service queueing system with setup times and Bernoulli vacation. *Journal of the Operations Research Society of Japan*, 50(4), 306–327.

[17]. Kim, C. S., Klimenok, V., &Dudin, A. (2014). Analysis of unreliable BMAP/PH/N type queue with Markovian flow of breakdowns. *Applied Mathematics and Computation*, 245, 379–394. https://doi.org/10.1016/j.amc.2014.07.084

[18]. Kumar, P., & Singh, R. (2013). Performance modeling of a flexible manufacturing cell under machine failures and preventive maintenance using closed queueing networks. *Computers & Industrial Engineering*, 65(2), 241–254. https://doi.org/10.1016/j.cie.2013.02.014

[19]. Lazowska, E. D., Zahorjan, J., Graham, G. S., &Sevcik, K. C. (2009). *Quantitative system performance: Computer system analysis using queueing network models*. Prentice Hall.

[20]. Meerkov, S. M., &Rohleder, T. (2008). Aggregation of Markovian models of production systems: Throughput and WIP. *IEEE Transactions on Automatic Control*, 53(8), 1841–1854. https://doi.org/10.1109/TAC.2008.929380

[21]. Papadopoulos, H. T., &Heavey, C. (2007). Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines. *European Journal of Operational Research*, 92(1), 1–27. https://doi.org/10.1016/0377-2217(95)00378-9

[22]. Srivastava, H. M., & Srivastava, P. K. (2012). Analysis of FMS with unreliable machines and batch-type operations: A queueing network approach. *International Journal of Advanced Manufacturing Technology*, 60(9–12), 1089–1102. https://doi.org/10.1007/s00170-011-3652-4

[23]. Takahashi, M., Osawa, H., & Fujisawa, T. (2007). On a batch-service queueing model with finite waiting room and finite input. *Journal of the Operations Research Society of Japan*, 50(1), 72–89.

[24]. Weiss, G. (2010). Scheduling and stochastic control in manufacturing. *Annals of Operations Research*, 172(1), 351–371. https://doi.org/10.1007/s10479-009-0648-1

[25]. Whitt, W. (2009). 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

[26]. Xie, X., & Dong, J. (2012). Performance evaluation of assembly systems with machine failures and batch transfer using stochastic Petri nets. *International Journal of Production Economics*, 135(1), 72–84. https://doi.org/10.1016/j.ijpe.2010.11.024