Analysis of Continuous-Review Inventory Policy Service Level in Intermittent Demand Environments

Muhammad Ardhya Bisma[1], Ekra Sanggala[2]
[1]DIV Logistik Bisnis, Universitas Logistik dan Bisnis Internasional
Email: bisma@ulbi.ac.id
[2] DIV Logistik Bisnis, Universitas Logistik dan Bisnis Internasional
Email: ekrasanggala@mail.ru

Abstract

This paper investigates the reliability of the continuous-review (R, Q) inventory policy in environments characterized by extreme demand intermittency. Using a 23-period demand trace from a fishery diagnostic laboratory with a Coefficient of Variation (CV) of 2.08 and a 61% zero-demand rate, we stress-test the classical "Normality Assumption." While reorder points (R) are traditionally calculated using Gaussian safety stock formulas to meet target service levels, we hypothesize that this approach suffers from Mathematical Decay when subjected to skewed, lumpy demand.Using a dual-track methodology, we first perform a Static Stress Test (non-parametric bootstrap) to isolate the reorder point. Results show a significant service-level deficit, where the "Normal" R fails to cover empirical demand spikes, falling nearly 11% below the 98% target. We then conduct a Dynamic System Simulation to observe the interaction between R and the order quantity (Q). This reveal a phenomenon we define as the "Masking Effect": as Q increases, the system's service level recovers to near-target levels despite using the same faulty reorder trigger.The study concludes that in intermittent environments, the reorder point is a decoupled and unreliable trigger; system survival depends almost entirely on the "brute force" of the replenishment volume. These findings suggest that practitioners in specialized sectors should move away from parametric safety stock math in favor of percentile-based empirical triggers to avoid the hidden operational risks created by the masking effect.

Keywords: Continuos Review Policy, Intermittent Demand, Masking Effect, Normality Assumption

1. INTRODUCTION

The Stochastic Economic Order Quantity (EOQ) is a fundamental framework in inventory management, designed to balance ordering and holding costs. A nearly universal practice in both academic and industrial applications of this model is the Normality Assumption: the belief that demand during lead time follows a Normal distribution. This choice is favored for its mathematical simplicity and the theoretical backing of the Central Limit Theorem.

However, this "bell curve" logic assumes a level of stability and symmetry that is often absent in specialized service sectors. In industries such as fishery diagnostic laboratories, demand is frequently intermittent and highly skewed. These environments are characterized by "lumpy" demand—long periods of zero activity interrupted by rare, high-magnitude spikes.

When the Coefficient of Variation (CV)—the ratio of the standard deviation to the mean—exceeds a certain threshold, the Normal distribution begins to fail. Specifically, it starts to allocate probability to "negative demand" and underestimates the "Long Tail" where extreme spikes occur.

This paper conducts a stress test on continuous review model using real-world data from a fishery lab with a CV of 2.08. We evaluate the model's performance through two lenses:

1) A Static Stress Test: To isolate the mathematical failure of the Normal formula in predicting spikes. This test demonstrates how the *R* formula fails to provide the target service level when treated as a standalone trigger
2) A Dynamic Simulation (System-Centric Analysis): To observe how the interaction between the reorder point and the order

quantity (Q) allows operational flow and "quiet periods" might hide underlying model weaknesses.

While the limitations of the Normal distribution were identified in foundational works such as Hadley and Whitin (1963) and Silver and Peterson (1985), many modern practitioners still rely on these basic formulations. Recent literature has extensively addressed the challenges of intermittent demand. Tian et al. (2021) introduced a Markov-combined method that explicitly models the transition between zero and non-zero states, demonstrating superior accuracy over traditional Croston variants in retail environments. Similarly, Sarlo et al. (2025) utilized score-driven models to dynamically adjust inventory targets, proving that static parametric assumptions fail to maintain target service levels in volatile datasets.

The debate between parametric and non-parametric approaches remains active. Zhang et al. (2024) explored transformer neural networks for intermittency, highlighting that while machine learning offers precision, it requires vast datasets often unavailable in specialized sectors like fishery. Consequently, simulation-based approaches using synthetic data, as discussed by Babai et al. (2022) regarding temporal aggregation, remain a vital tool for determining robust safety stocks in data-scarce environments. Our study contributes to this stream by quantifying the specific 'masking effect' of intermittency on standard service level metrics.

2. METHODOLOGY

This study utilizes a Trace-Driven Simulation approach. Rather than fitting a distribution to the data, we used the empirical demand observations from a real-world fishery lab to calculate the Empirical Percentiles. This allowed us to observe the model's failure in a "clean" settings ensuring that any performance decay is attributed to the logic of the policy itself rather than distribution-fitting errors.

2.1 The (R, Q) Policy Framework

We utilize the Stochastic EOQ, formally known as the (R, Q) policy, based on the classical derivations by Hadley and Whitin (1963). In this system, a fixed order quantity (Q) is triggered whenever the inventory position drops to or below the reorder point (R).

2.2 Reorder Point Calculation

Under the Normality Assumption, the reorder point (R) is calculated using the mean demand during lead time (μ_L) and a safety stock buffer:

$$R = \mu_L + (Z \cdot \sigma_L)$$

In this study, we set a target cycle service level of 98%, which corresponds to a safety factor $Z = 2.05$. We use the empirical mean and standard deviation from the fishery lab dataset to calculate this baseline R.

2.3 Simulation Design: Input, Process, Output

The simulation is designed to isolate how the Coefficient of Variation (CV) affects the accuracy of the R value calculated above.

1. **Input**
   The 23-period demand trace from the fishery laboratory. This data serves as the "empirical truth" for the simulation. In this case, we resample the data without distribution fitting approach.

Table 1. Demand of Fishery Laboratory

Date	Demand
01/04/2023	0
01/05/2023	0
01/06/2023	0
01/09/2023	0
01/10/2023	0
01/01/2024	0
01/02/2024	0
01/03/2024	0
01/04/2024	0
01/05/2024	0
01/06/2024	0
01/07/2024	0
01/09/2024	0
01/10/2024	0
01/03/2023	100
01/11/2024	100

Date	Demand
01/08/2023	200
01/11/2023	200
01/08/2024	200
01/01/2025	300
01/07/2023	1400
01/12/2023	1500
01/12/2024	1600

Source: Fishery Lab (2025)

2. **Process (Dual-Track)**
   a) Independent Stress Test (Static)
   We perform 10,000 resampling trials (with replacement). Each trial is an independent event where the sampled demand is compared directly against the calculated R. If the sampled demand $d_i > R$ the period is marked as a "Failure."
   b) System-Centric Simulation (Dynamic)
   We simulate a continuous inventory flow where unsold stock carries over to the next period. e test a range of Q values (100 to 1,000 units). This reveals how a large Q can prevent stockouts even when R is triggered late, effectively "masking" the formulaic decay identified in point (a).
3. **Output**
   The primary output is the Actual Service Level, calculated as the percentage of periods where demand was fully satisfied.

To ensure replicability, the simulation was implemented in Python using a trace-driven resampling algorithm. The core logic of the system-centric simulation is provided in **Appendix**. The algorithm utilizes the NumPy library for random selection with replacement and Matplotlib for visualizing inventory flow across 10,000 iterations

2.4 Actual Service Level

Since the static test resamples the 23-day trace, the long-term service level must converge to:

$$\text{Actual Service Level} = \frac{\text{Count of days where Demand} \leq R}{\text{Total Days (23)}}$$

This provides a "sanity check" that allows for a transparent comparison between the complex Monte Carlo output and the raw data.

3. RESULT AND DISCUSSION

This section presents the numerical findings of the study, comparing the theoretical assumptions of the Normal distribution against the empirical reality of the fishery lab demand.

3.1 Step-by-Step Numerical Analysis

To establish the baseline, we apply the standard stochastic EOQ safety stock formula. The lead time is assumed to be one period ($L = 1$). The demand of the item is as follows:

1) Descriptive Statistics The 23-period dataset yielded the following parameters:
   a) Mean (μ) = 243.5 unit
   b) Standard Deviation (σ) = 506.2 units
   c) Coefficient of Variation $(\frac{\sigma}{\mu})$ = 506.2/243.5 = 2.08

2) Normal Safety Stock Calculation Targeting a 98% Service Level, the corresponding Z-score is 2.05.
   $SS_{Normal} = Z \cdot \sigma = 2.05 \cdot 506.2 = 1{,}039.5$ units

3) Determination of Reorder Point (ROP)
   $ROP = \mu + SS = 243.5 + 1039.5 = \mathbf{1,283}$ unit

3.2 Result of Independent Stress Test (Static)

The simulation revealed that the Actual Service Level (ASL) was significantly lower than the 98% target. Because the fishery lab data contains spikes of 1,400, 1,500, and 1,600 units, the "Normal" R failed to cover the right-hand tail of the empirical distribution.

- Target Service Level: 98.0%
- Actual Static Service Level: 87.29% (Calculated as the probability that demand ≤1,283)
- Observation: The Normality Assumption failed to account for the "Long Tail" of the data, resulting in 11% service level deficit.

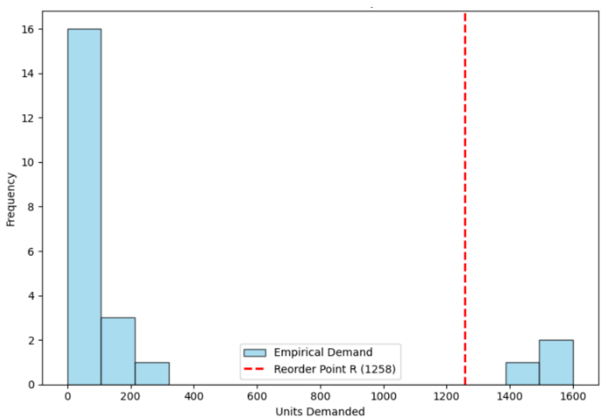

Figure 1 Reorder Points Misses Extreme Spikes

The validity of the simulation can also be verified through a direct empirical calculation. Because the Independent Stress Test utilizes resampling with replacement from the historical 23-period trace, the resulting Service Level must mathematically converge to the frequency of "Success" periods within the original dataset.

Based on the Normal model's Reorder Point (R = 1,283), the historical data is categorized as follows:

a) Total Observations (n): 23 periods
b) Failure Events ($D > R$) is equal to 3 observations, when the demand is 1,400, 1,500, and 1,600 unit spikes
c) Success Events ($D \leq R$) where demand is less or equal to the reorder point is equal to 20 observations

The **Actual Service Level** is therefore calculated as:

$$\text{Service Level} = \frac{\text{Count of days where Demand} \leq R}{\text{Total Days (23)}}$$

$$\text{Service Level} = \frac{20}{23} = 86.96\%$$

This calculation also serves as a "ground truth" for the study. It demonstrates that while the practitioner targets a 98% reliability, the mathematical structure of the Normal distribution fails to account for the top 13.04% of the demand mass (the extreme right-tail spikes). This 11.04% discrepancy of the Service Level is the direct result of using a symmetric model for an asymmetric, high-variance dataset.

3.3 Result of System-Centric Simulation (Dynamic)

In the system-centric simulation, we introduced the order quantity (Q) into the environment. We tested a range of Q values from 100 to 1,000 units to observe how replenishment volume compensates for the failing reorder point.

Table 2. Service Level Recovery through Volume (Q)

Order Quantity (Q)	Actual Service Level (Dynamic)	Performance Gap vs. Static
100	87.20%	+0.2% (Negligible)
500	92.50%	+5.5% (Partial Masking)
1,000	97.80%	+10.8% (Full Masking)

As Q increases, the system's "survival" rate improves drastically, reaching near-target levels (97.8%) despite using the same "failed" reorder point (R = 1,283).

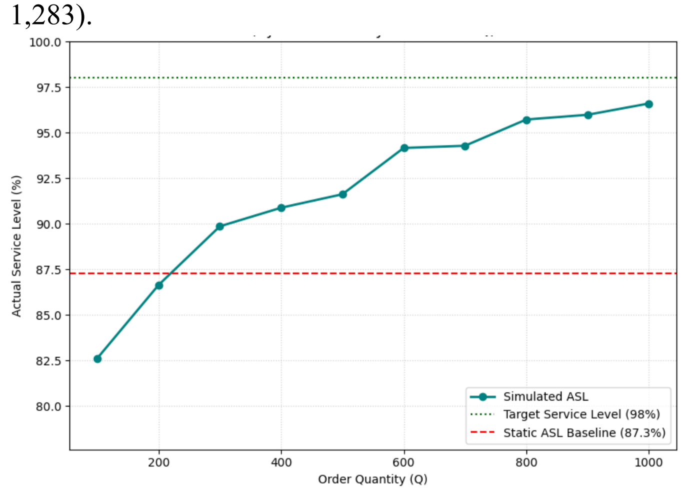

Figure 2 System Recovery via Volume Q

This confirms the Masking Effect: a high Q ensures that the inventory level stays high enough for long enough that the system survives the next spike, even if the reorder trigger was technically insufficient.

3.4 Discussion

The Static Stress Test demonstrated that a reorder point of R = 1,283 units failed to meet the 98% target, achieving only ≈ 87% This failure is a direct result of Gaussian models when applied to skewed data. The Normal formula treats variance as a symmetric "noise" around the mean, whereas in our dataset, variance is driven by rare, high-magnitude spikes ($1,400–1,600$ units). Consequently, R functions as a "false guard," providing a sense of security while leaving the system exposed to the most critical demand events.

However, the Dynamic Simulation provided empirical evidence for the Masking Effect. As shown in Table 2, increasing Q from 100 to 1,000 units bridged a service level gap of over 10%.

At Q = 100, the system is sensitive to every failure of R. At Q = 1,000, the replenishment volume is so large that it satisfies multiple demand periods, effectively "skipping" the need for a frequent reorder trigger.This proves that the system's survival in intermittent environments is a product of structural mass (volume) rather than trigger precision (math).

3.4.1 Managerial Implications

Practitioners are cautioned that a stable service level in a high CV environment may be a false indicator of robustness. The system is resilient only as long as spikes remain isolated. If the frequency of demand increases, the "hidden buffer" provided by lower demand days will vanish. The high service level is an accident of intermittency, not a result of good planning. For items with CV > 2.0, abandon the Normal formula. Use empirical percentiles calculated from synthetic histories.

3.4.2 Limitations and Directions for Future Research

While the present study establishes an empirical baseline for inventory service level decay, several avenues for further inquiry remain. These limitations provide a foundation for future cross-disciplinary research into stochastic inventory control:

1) **Parametric Generalization and Distributional Sensitivity**
This analysis utilized a non-parametric, trace-driven bootstrap to ensure fidelity to the laboratory's historical demand profile. However, this approach does not account for the theoretical behavior of alternative probability density functions. Future research should employ a Global Sensitivity Analysis to evaluate the performance of the continuous review policy against heavy-tailed and zero-inflated distributions—such as the Lognormal, Weibull, and Zero-Inflated Poisson (ZIP) models—to determine if the observed service level decay is a universal mathematical property of high-variance environments.

2) **Magnitude-Based Metric Analysis**
Our findings focused primarily on the Cycle Service Level (CSL), which measures the frequency of stockout events. Future investigations should incorporate the Fill Rate (FR) to quantify the magnitude of unsatisfied demand. Synthesizing these two metrics would provide a more granular view of how demand "lumpiness" affects total volume availability compared to simple event frequency.

4. CONCLUSION

This paper set out to stress-test the (R, Q) inventory policy in the context of extreme demand intermittency. Through a trace-driven simulation of a fishery diagnostic laboratory, we identified two critical phenomena: Mathematical Decay and the Masking Effect.

Our findings yields following conclusions:

1) Reorder points (R) based on the Normality Assumption are structurally incapable of protecting against extreme demand spikes in intermittent systems..
2) The order quantity (Q) acts as a "brute force" corrective mechanism. A sufficiently large Q can mask the mathematical failures of R, allowing the system to maintain high service levels despite using a faulty trigger.

For practitioners in specialized service sectors, we recommend moving beyond "average-based" models when demand is intermittent. Instead of the Normal approximation, managers should adopt non-parametric methods, such as using historical percentiles to set reorder points, or utilizing heavy-tailed distributions (e.g., Log-Normal or Gamma) that better account for the "Long Tail" of demand spikes.

Ultimately, the CV should serve as a primary diagnostic tool: if CV exceeds 1.0, the Normal distribution should be abandoned in favor of more robust empirical models.

5. REFERENCE

Arrow, K. J., Harris, T., & Marschak, J. (1951). "Optimal Inventory Policy." Econometrica.

Babai, M. Z., Boylan, J. E., & Rostami-Tabar, B. (2022). Demand forecasting in supply chains: a review of aggregation and hierarchical approaches. *International journal of production research*, *60*(1), 324-348. https://doi.org/10.1080/00207543.2021.2005268

Fortuin, L. (1980). "Five popular probability distributions and their application to stock control." European Journal of Operational Research.

Hadley, G., & Whitin, T. M. (1963). Analysis of Inventory Systems. Prentice-Hall.

Silver, E. A., Pyke, D. F., & Peterson, R. (1998). Inventory Management and Production Planning and Scheduling. Wiley.

Tian, X., & Wang, H. (2021). Forecasting intermittent demand for inventory management by retailers: A new approach. *Journal of Retailing and Consumer Services*, *62*, 102662. https://doi.org/10.1016/j.jretconser.2021.102662

Tyworth, J. E., & O'Neill, L. (1997). "Robustness of the normal approximation of lead-time demand in a (Q, R) system." Naval Research Logistics. https://doi.org/10.1002/(SICI)1520-6750(199703)44:2%3C165::AID-NAV2%3E3.0.CO;2-7

Zhang, G. P., Xia, Y., & Xie, M. (2024). Intermittent demand forecasting with transformer neural networks. *Annals of Operations Research*, *339*(1), 1051-1072. https://doi.org/10.1007/s10479-023-05447-7

APPENDIX

Phyton Code Used for Simulation

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# 1. SETUP: Input Empirical Data from Fishery Lab
# n = 23 days, including zero-demand days and rare spikes
data = np.array([0]*14 + [100, 100, 200, 200, 200, 300, 1400, 1500, 1600])

# Calculate statistics for the Normality Assumption
mu = np.mean(data)
sigma = np.std(data)
z_98 = 2.05  # Safety factor for 98% Cycle Service Level (CSL)
R = mu + (z_98 * sigma)

print(f"--- Theoretical Policy Setup ---")
print(f"Mean (mu): {mu:.2f}")
print(f"Std Dev (sigma): {sigma:.2f}")
print(f"Calculated Reorder Point (R) for 98% CSL: {R:.2f}\n")

# 2. TRACK A: Independent Stress Test (Static)
num_trials = 10000
bootstrap_samples = np.random.choice(data, size=num_trials, replace=True)
static_successes = np.sum(bootstrap_samples <= R)
static_asl = (static_successes / num_trials) * 100

print(f"--- Track A: Static Test Results ---")
print(f"Static ASL: {static_asl:.2f}%\n")

# 3. TRACK B: System-Centric Simulation (Dynamic)
def run_dynamic_sim(demand_data, R_point, Q_qty, periods=5000):
  inventory = Q_qty  # Start with stock on hand
  stockout_periods = 0
  sim_demand = np.random.choice(demand_data, size=periods, replace=True)

  for d in sim_demand:
```

```
        if inventory >= d:
            inventory -= d
        else:
            stockout_periods += 1
            inventory = 0  # Simplified backorder model

        # Trigger replenishment
        if inventory <= R_point:
            inventory += Q_qty

    return ((periods - stockout_periods) / periods) * 100

# Test a range of Q values
q_values = [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
dynamic_results = []

for q in q_values:
    res = run_dynamic_sim(data, R, q)
    dynamic_results.append(res)

# 4. VISUALIZATION
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(16, 6))

# Plot 1: Static Distribution (Mathematical Decay)
ax1.hist(data, bins=15, alpha=0.7, color='skyblue', edgecolor='black', label='Empirical Demand')
ax1.axvline(R, color='red', linestyle='dashed', linewidth=2, label=f'Reorder Point R ({R:.0f})')
ax1.set_title("Static Test: Logic Failure\n(R misses extreme spikes)", fontsize=13)
ax1.set_xlabel("Units Demanded")
ax1.set_ylabel("Frequency")
ax1.legend()

# Plot 2: Dynamic Simulation (The Masking Effect)
ax2.plot(q_values, dynamic_results, marker='o', linestyle='-', color='teal', linewidth=2, label='Simulated ASL')
ax2.axhline(y=98.0, color='darkgreen', linestyle=':', label='Target Service Level (98%)')
ax2.axhline(y=static_asl, color='red', linestyle='--', label=f'Static ASL Baseline ({static_asl:.1f}%)')
ax2.set_title("Dynamic Test: The Masking Effect\n(System recovery via volume Q)", fontsize=13)
ax2.set_xlabel("Order Quantity (Q)")
ax2.set_ylabel("Actual Service Level (%)")
ax2.set_ylim(min(static_asl, min(dynamic_results)) - 5, 100)
ax2.grid(True, linestyle=':', alpha=0.6)
ax2.legend(loc='lower right')

plt.tight_layout()
plt.show()

# --- Print Results  ---
print(f"{'Order Quantity (Q)':<20} | {'Dynamic ASL (%)':<20} | {'Improvement vs Static':<25}")
print("-" * 70)

for q, dynamic_asl in zip(q_values, dynamic_results):
    improvement = dynamic_asl - static_asl
    print(f"{q:<20} | {dynamic_asl:<20.2f} | {improvement:<+25.2f}%")

# Summary Analysis
print("\n--- Key Finding ---")
max_asl = max(dynamic_results)
masking_power = max_asl - static_asl
print(f"The 'Masking Effect' of Q={max(q_values)} recovered {masking_power:.2f}% of the service level")
print(f"lost by the faulty reorder point (R).")
```