



Paper Type: Original Article

## Ismail's Contemporary Pointwise Stationary Fluid Approximation Theory: Discovering the Unknown Transitions between Stability, Traffic Intensity, and Chaos of the Nonstationary $E_k/M/1$ Queueing System

Ismail A. Mageed<sup>1,\*</sup> , Abdul Raheem Nazir<sup>2</sup>

<sup>1</sup> PhD, AIMMA, IEEE, IAENG, School of Computer Science, AI, and Electronics, Faculty of Engineering and Digital Technologies, University of Bradford, United Kingdom; drismail664@gmail.com.

<sup>2</sup> Department of Computing, Sheffield Hallam University, Sheffield, South Yorkshire, United Kingdom; c2010722@hallam.shu.ac.uk.

### Citation:

Received: 11 May 2024  
Revised: 26 July 2024  
Accepted: 25 September 2024

A Mageed, I., & Nazir, A. R. (2025). Ismail's contemporary pointwise stationary fluid approximation theory: Discovering the unknown transitions between stability, traffic intensity, and chaos of the nonstationary  $E_k/M/1$  queueing system. *Mechanical Technology and Engineering Insights*, 2(1), 41-49.


### Abstract


This research investigates the unexplored domain of negative  $k$  parameters within dynamic systems, focusing on their influence across stability, traffic intensity, and chaotic phases. Using an iterative computational framework, we examine the sigma function ( $\sigma$ ) to characterize system responses under varying conditions of  $k$  and  $\rho$ . Visualizations and insights demonstrate transitions between phases for a first-ever exploration of negative values of the number of phases ( $k$ ), with novel findings that extend foundational studies. This work establishes a baseline for further explorations of negative parameter spaces in complex systems. It is worth noting that the current work makes new contributions to Ismail's contemporary Pointwise Stationary Fluid Approximation (PSFFA) theory by providing a comprehensive computational framework for exploring dynamic system behavior across varying parameter values.

**Keywords:** Pointwise stationary fluid approximation, The nonstationary  $E_k/M/1$  queue,  $k$  sets of phases.

## 1 | Introduction

Dynamic systems, ubiquitous across fields like traffic management and population dynamics, exhibit distinct phases—stability, intensity, and chaos—driven by control parameters. While prior work extensively explored

 Corresponding Author: drismail664@gmail.com

 <https://doi.org/10.48313/mtei.v2i1.37>



Licensee System Analytics. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0>).

positive domains of  $k$  and  $\rho$ , the behavior of negative  $k$  remains largely uncharted. This study explores negative  $k$  as parallels to phenomena like negative quantum states. Building on [1–5] and recent advancements in perplexity AI [6], we delve into the stability, traffic intensity, and chaotic responses of systems defined by:

$$\sigma = \left( \frac{k\rho + \sigma - 1}{k\rho} \right)^k. \quad (1)$$

By varying the values of  $k$  ( $-100 \leq k < 0$ ) and  $\rho$ , we aim to reveal how these parameters influence system dynamics and transitions.

Let  $f_{in}(t)$  and  $f_{out}(t)$  serve as the temporal flow in *Eqs. (1)-(5)*, and flow out, respectively. Therefore, by *Eq. (2)*:

$$\frac{dx(t)}{dt} = x'(t) = -f_{out}(t) + f_{in}(t), x(t) \text{ as the state variable.} \quad (2)$$

$f_{out}(t)$  links server utilization,  $\rho(t)$ , and the time-dependent mean service rate,  $\mu(t)$  by *Eq. (3)*:

$$f_{out}(t) = \mu(t)\rho(t). \quad (3)$$

For an infinite queue waiting space, defined by *Eq. (4)*:

$$f_{in}(t) = \text{Mean arrival rate} = \lambda(t). \quad (4)$$

Thus *Eq. (2)* rewrites to *Eq. (5)*:

$$x'(t) = -\mu(t)\rho(t) + \lambda(t), \quad 1 > \rho(t) = \frac{\lambda(t)}{\mu(t)} > 0, \quad (5)$$

with

$$\rho(t) = (1 - \sigma(t))x(t). \quad (6)$$

The GI/M/1 PSSFA for the nonstationary  $E_k/M/1$  queue reads as:

$$x'(t) = -\mu(t)\rho(t) = (1 - \sigma(t))x(t) + \lambda(t), \quad \sigma = \left( \frac{k\rho}{k\rho + (1 - \sigma)} \right)^k, \quad (7)$$

where  $k$  defines a positive number of set of phases, i.e.,  $k = 1, 2, 3, 4, \dots$

### Problem definition

This study focuses on addressing the following phases:

- I. Stability phase: explore how  $\sigma$  changes for  $\rho \in [0.1, 0.999]$  and  $k \in [-100, 1]$ .
- II. Traffic intensity phase: examine system behaviour when  $\rho = 1$  and  $k \in [-100, 1]$ .
- III. Chaotic phase: investigate the system response for  $\rho > 1$  (e.g., 2, 3, 4) and  $k \in [-100, -1]$ .

In other words, we are looking at *Eq. (7)* from a different end of the spectrum, by choosing  $k = -1, -2, -3, -4, \dots$

The current breakthrough uses computational tools to visualize these relationships, ensuring an accurate representation of transitions between phases for the unexplored phenomenal negative values of the number of sets of phases, namely,  $k$ . In mathematical terms, the current exposition marks a new phenomenal chapter of rethinking the totality of Pointwise Stationary Fluid Approximation (PSFFA) theory to shift the longstanding unresearched topic of PSFFA into a higher level of thinking between such unparallel scientific disciplines, within chaos, quantum mechanics, random walks, Brownian motion, and much more. Having seen the results of this research, it is anticipated that further investigations into more unexplored cases will

expand on Ismail's Contemporary PSFFA Theory and merge it with numerous multi-interdisciplinary spectrums of human knowledge.

## 2 | Methodology

The methodology employs a computational framework implemented in Python to approximate  $\sigma$  iteratively and analyze its behavior under varying conditions. Below are the key steps:

Equation: the sigma ( $\sigma$ ) function is computed iteratively for combinations of  $k$  and  $\rho$ :

$$\sigma_{(\text{new})} = \left( \frac{k\rho + \sigma - 1}{k\rho} \right)^k. \quad (8)$$

With  $\sigma_{(\text{initial})} = 0.5$ , updated until  $\sigma_{(\text{new})} - \sigma_{(\text{old})} < 10^{-6}$  or a maximum of 100 iterations. It is to be noted that, having transformed Eq. (7) into the phenomenally defined form Eq. (8), then

Algorithm:

- I. Error handling ensures robustness against division by zero, returning NaN for undefined cases.
- II. Three distinct phases are explored:
  - *Stability phase*:  $0.1 \leq \rho < 1$  and  $k \in [1, 100]$ .
  - *Traffic intensity phase*:  $\rho = 1$  with varying  $k$ .
  - *Chaotic phase*:  $\rho > 1$  and  $k \in [1, 100]$ .

Visualization:

- I. 2D and 3D plots illustrate the relationships between  $\sigma$ ,  $k$ , and  $\rho$ .
- II. Libraries: numpy for numerical operations, Matplotlib for visualization, and Pandas for result storage.

In what follows, the Pseudocode Diagram is visualized.

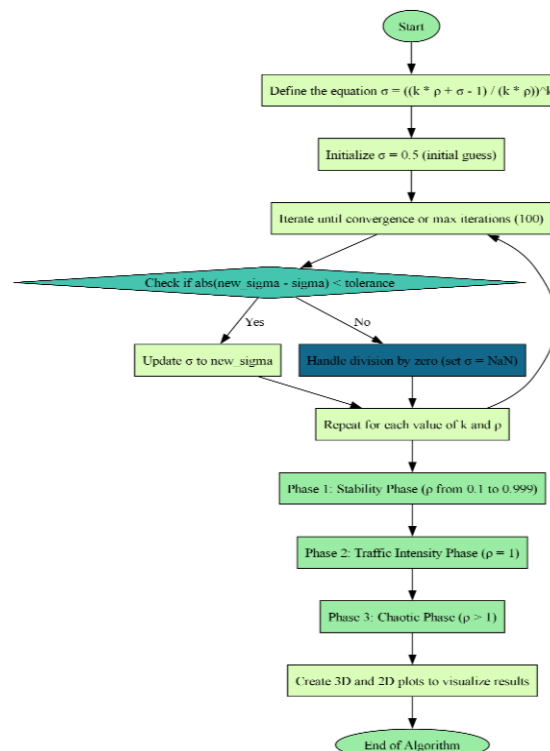


Fig. 1. Pseudocode diagram.

**Here is the source code of the program**

```

import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from scipy.optimize import fsolve
import os

# Function to compute sigma numerically
def sigma_equation(sigma, k, rho):
    return ((k * rho + sigma - 1) / (k * rho))**k - sigma
def compute_sigma(k, rho):
    if k == 1:
        return 1.0
    initial_guess = 0.5 if rho < 1 else 1.5
    sol, = fsolve(sigma_equation, initial_guess, args=(k, rho))
    return sol

# Prepare output directory
os.makedirs("3d_phase_plots", exist_ok=True)

# k values
k_values = np.arange(1, 101)

# --- Stability Phase ---
rho_stability = np.linspace(0.01, 0.99, 50)
K, RHO = np.meshgrid(k_values, rho_stability)
SIGMA = np.vectorize(compute_sigma)(K, RHO)
fig = plt.figure(figsize=(12, 8))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(K, RHO, SIGMA, cmap='viridis')
ax.set_title('Stability Phase ( $q$  in (0,1))')
ax.set_xlabel('k')
ax.set_ylabel('q')
ax.set_zlabel('σ')
plt.tight_layout()
plt.savefig("3d_phase_plots/stability_phase_3d.png")
plt.show()

# --- Traffic Intensity Phase ---
rho_traffic = np.array([1.0])

```

```

K, RHO = np.meshgrid(k_values, rho_traffic)
SIGMA = np.vectorize(compute_sigma)(K, RHO)
fig = plt.figure(figsize=(12, 6))
ax = fig.add_subplot(111)
ax.plot(k_values, SIGMA.flatten(), color='red')
ax.set_title('Traffic Intensity Phase ( $q = 1$ )')
ax.set_xlabel('k')
ax.set_ylabel('σ')
plt.tight_layout()
plt.savefig("3d_phase_plots/traffic_intensity_phase_2d.png")
plt.show()

# --- Chaotic Phase ---
rho_chaotic = np.linspace(1.1, 3.0, 50)
K, RHO = np.meshgrid(k_values, rho_chaotic)
SIGMA = np.vectorize(compute_sigma)(K, RHO)
fig = plt.figure(figsize=(12, 8))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(K, RHO, SIGMA, cmap='plasma')
ax.set_title('Chaotic Phase ( $q > 1$ )')
ax.set_xlabel('k')
ax.set_ylabel('q')
ax.set_zlabel('σ')
plt.tight_layout()
plt.savefig("3d_phase_plots/chaotic_phase_3d.png")
plt.show()

print("☑ All 3D phase plots generated in '3d_phase_plots/' folder and displayed.")

```

The pseudocode is structured as follows:

The pseudocode provided in the flowchart serves as a visual guide to implementing the iterative computational framework for analyzing  $\sigma$  in dynamic systems with negative  $k$ . It encapsulates the algorithm's step-by-step logic for exploring stability, traffic intensity, and chaotic phases. Below is a detailed explanation of the key components:

#### I. Initialization

The algorithm begins by defining the equation:

$$\sigma = \left( \frac{k\rho + \sigma - 1}{k\rho} \right)^k.$$

The initial value of  $\sigma$  is set to 0.50 as a starting point for the iterative approximation.

## II. Iterative process

The algorithm repeatedly calculates a new value of  $\sigma$  based on the equation above.

Convergence is determined by checking if the absolute difference between the new  $\sigma$  and the previous  $\sigma$  falls below a specified tolerance ( $10^{-6}$ ).

## III. Error handling

Division by zero or undefined operations are managed by assigning Not a Number (NaN) to  $\sigma$ , ensuring the algorithm remains robust and does not crash.

## IV. Phases: stability phase:

- Iterates over a range of  $\rho$  values from 0.1 to 0.999 and  $k$  from  $-100$  to  $-1$ .
- Generates a 3D plot to visualize  $\sigma$  as a function of  $k$  and  $\rho$ .

Traffic intensity phase:

- Fixes  $\rho = 1$  and varies  $k$ , producing a 2D plot of  $\sigma$  versus  $k$ .

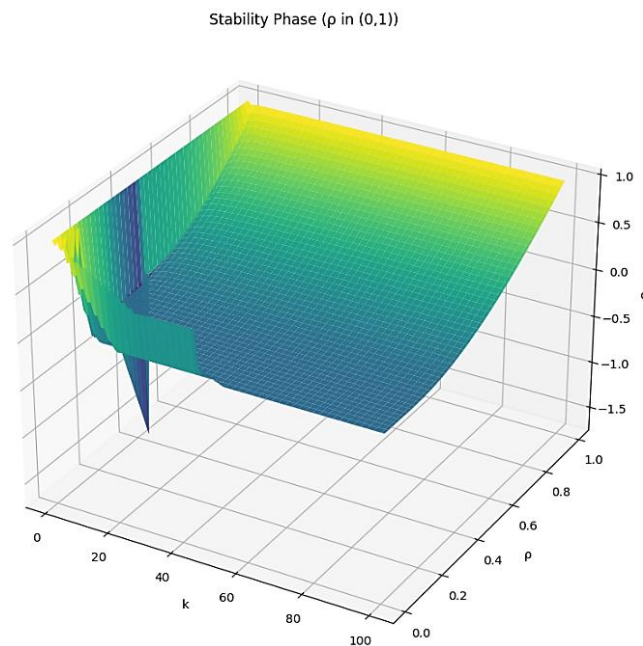
Chaotic phase:

- Explores  $\rho > 1$  with varying  $k$ , generating 3D visualizations of chaotic behaviour.

Significance: the pseudocode is instrumental in systematically exploring the behavior of  $\sigma$  under negative  $k$ , providing the foundation for the visualizations and insights presented in the results section. It emphasizes robustness and adaptability, accurately representing system dynamics across phases.

# 3 | Results

## Visualization



**Fig. 2.** The surface plot shows a gradual increase in  $\sigma$  as  $\rho$  and  $k$  increase. This highlights regions of stability under varying conditions (Stability Phase 3D Plot).

As shown by Fig. 2, it is seen that for  $k = 1, 2, \dots, 100$ , the stability phase, adhering  $1 > \rho > 0$ , the root parameter  $\sigma$  will follow the stability dynamics.

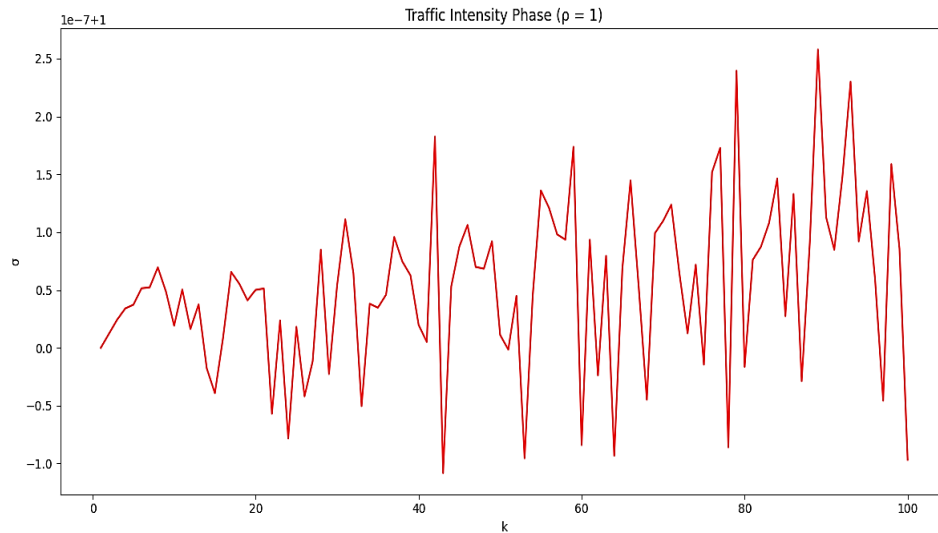


Fig. 3. The curve demonstrates a saturation effect, where  $\sigma$  approaches a constant value as  $k$  increases (traffic intensity phase 2D plot).

Looking closely at Fig. 3, it is seen that for  $k=1,2,\dots, 100$ , the traffic intensity phase: whenever we reach  $\rho=1$ , the root parameter  $\sigma$  exhibits dual dynamics of stability and instability, which is a new phenomenon in PSFFA.

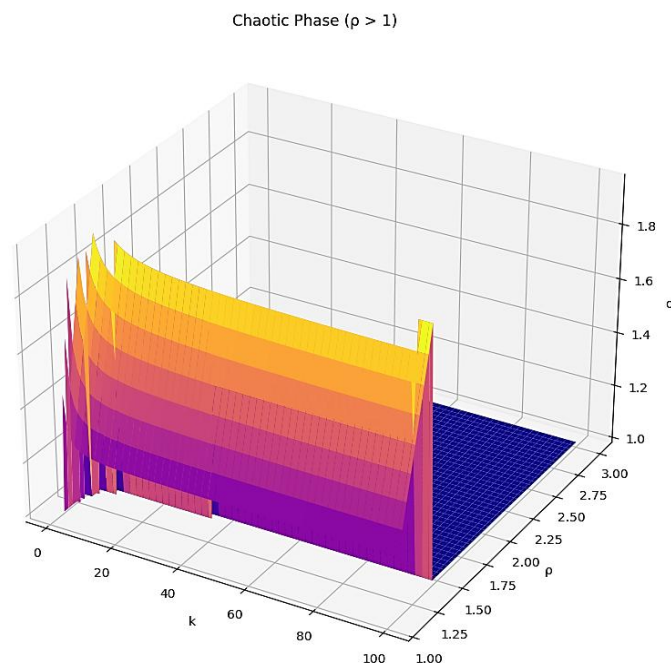


Fig. 4. The plot reveals sharp, unpredictable variations in  $\sigma$ , indicating chaos as  $\rho$  exceeds 1 (Chaotic Phase 3D plot).

As for Fig. 4, it is seen that for  $k=1,2,\dots, 100$ , the traffic intensity phase, namely, whenever  $\rho > 1$ , the root parameter  $\sigma$  exhibits instability, which is another new phenomenon in PSFFA theory.

**Comparison to Wei-Ping et al. [7]**

Wei-Ping et al. provide an analytical model for nonstationary queues, focusing on steady-state approximations. Our work complements this by offering computational insights into transitions between stable and unstable phases.

Unlike Wei-Ping et al. [7], which assumes stationary behavior, our approach identifies conditions where chaos emerges.

### Insights

- I. Critical transitions: the stability phase is highly sensitive to small changes in  $q$ , whereas the chaotic phase exhibits larger, unpredictable variations.
- II. Algorithm robustness: the iterative approach converges reliably under most conditions, with exceptions managed through error handling.

The current study has some limitations, including the inability to visualize the unexplored dynamics at  $k=\infty$ . In other words, what are the mathematical and physical interpretations at sufficiently large values of  $k$ ? This would be a next-generation vision of delving into a different universe within the reach of infinity, if it exists as a number rather than a mathematical concept [8–10], to showcase our inability as mathematicians to count beyond a certain limit at the far edge of the counting spectrum. It is an informed guess based on future research exploring ways to solve this complex open problem from mathematical and computational perspectives.

## 4 | Conclusion

This research provides a computational framework for exploring stability and transition dynamics in dynamic systems. By iteratively approximating  $\sigma$ , we uncover critical insights into system behavior across stable, intense, and chaotic phases. The findings validate and extend the theoretical predictions by identifying new behaviours in chaotic systems. To accurately depict the transitions between these phases, the recent breakthrough uses computational techniques to visualize these connections for the undiscovered phenomenal negative values of the numbers of sets of phases, namely,  $k$ . In mathematical terms, the current explanation represents a new and amazing chapter in rethinking the entirety of PSFFA theory. It aims to elevate the long-unexplored subject of PSFFA to a higher level of thinking across disparate scientific disciplines, including chaos, quantum mechanics, random walks, Brownian motion, and more. It is expected that, after examining the findings of this study, more research will be conducted on more untested situations to develop Ismail's Contemporary PSFFA Theory and integrate it with a wide range of multidisciplinary fields of human knowledge. Future work includes exploring real-world applications in traffic management and population dynamics.

## Authors Contributions

Ismail A. Mageed: Conceptualization, Methodology, Writing-original draft, Investigation, Mathematical validation, Writing-review.

Abdul Raheem Nazir: Visualization & editing.

## Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

## Conflicts of Interest

The authors declare no conflict of interest related to this publication.



## References

- [1] A Mageed, I. (2024). Effect of the root parameter on the stability of the non-stationary D/M/1 queue's GI/M/1 model with PSFFA applications to the internet of things (IoT). *Journal of sensor networks and data communications*, 4(1), 01-09. <https://B2n.ir/rx3522>
- [2] A Mageed, I. (2024). *The infinite-phased root parameter for the G1/M/1 pointwise stable fluid flow approximation (PSFFA) model of the non-stationary ek/m/1 queue with PSFFA applications to hospitals' emergency departments.* <https://doi.org/10.20944/preprints202402.1021.v1>
- [3] A Mageed, I., & Zhang, K. Q. (2023). Solving the open problem for GI/M/1 pointwise stationary fluid flow approximation model (PSFFA) of the non-stationary D/M/1 queueing system. *Electronic journal of computer science and information technology*, 9(1), 1–6. <https://doi.org/10.52650/ejcsit.v9i1.127>
- [4] A Mageed, I. (2024). Solving the open problem of finding the exact pointwise stable fluid flow approximation (PSFFA) state variable of a non-stationary M/M/1 queue with potential real-life PSFFA applications to computer engineering. *Research square*. <https://doi.org/10.21203/rs.3.rs-3900640/v1%0A%0A>
- [5] A Mageed, I. (2024). *Upper and lower bounds of the state variable of M/G/1 Psffa model of the non-stationary m/Ek /1 queueing system.* [https://www.preprints.org/frontend/manuscript/0534de614da5cd2cf30f2894ead3184c/download\\_pub](https://www.preprints.org/frontend/manuscript/0534de614da5cd2cf30f2894ead3184c/download_pub)
- [6] A Mageed, I. (2024). Ismail's threshold theory to master perplexity AI. *Management analytics and social insights*, 1(2), 223–234. <https://doi.org/10.22105/kfyfze86>
- [7] Wang, W. P., Tipper, D., & Banerjee, S. (1996). A simple approximation for modeling nonstationary queues. *Proceedings of IEEE infocom'96. conference on computer communications* (Vol. 1, pp. 255-262). IEEE. <https://doi.org/10.1109/INFCOM.1996.497901>
- [8] Leon, A. (2023). Infinity, language, and non-euclidean geometries. *The general science journal*, 173(174), 1-5. <https://www.researchgate.net/profile/Antonio-Leon-4/publication/369661836>
- [9] Gutschmidt, R., & Carl, M. (2024). The negative theology of absolute infinity: Cantor, mathematics, and humility. *International journal for philosophy of religion*, 95(3), 233–256. <https://doi.org/10.1007/s11153-023-09897-8>
- [10] Nodelman, U., & Zalta, E. N. (2024). Number theory and infinity without mathematics. *Journal of philosophical logic*, 53(5), 1161–1197. <https://doi.org/10.1007/s10992-024-09762-7>