Exploiting Adaptive Blockchain-based Decentralized Network Computing by Shortest Transaction First

Abinava Yeshwanth Kalayambakam Janardhan (Contribution: 16.666%)

Roopasree Pachipulusu (Contribution: 16.666%)

Venkata Ramana Reddy Peram (Contribution: 16.666%)

Rifat Rafiuddin (Contribution: 16.666%)

Md. Nazrul Huda Shanto (Contribution: 16.666%)

Sai Krishna Vennamaneni (Contribution: 16.666%)

Submitted on: April 26, 2024

Abstract

Blockchain technology often struggles with scalability and performance issues as transaction volumes grow. Recent research has introduced adaptive chain models that dynamically adjust to transaction demands, enhancing the blockchain's ability to manage large volumes of transactions. This paper extends these models by incorporating a minimum priority queue prioritization strategy that processes the simplest transactions first. This method speeds up validation and enhances overall efficiency by effectively scheduling transactions based on complexity. Our initial results indicate that this approach significantly improves blockchain throughput and processing time, presenting a viable solution to scalability challenges in blockchain networks.

I. Introduction

Blockchain technology, recognized for its pivotal role in secure digital transactions, encounters significant challenges related to scalability and performance, particularly when managing high transaction volumes. Addressing these challenges, recent studies have proposed various adaptive chain models that dynamically adjust blockchain configurations to efficiently handle fluctuating transaction demands [1, 4]. These models, by adapting to the volume and complexity of network transactions in real-time, offer a promising solution to the inherent limitations of traditional blockchains.

Building on this foundation, our research incorporates an innovative "easiest transaction first" strategy within the adaptive chain framework. This strategy employs a minimum priority queue to sort transactions based on their predefined estimated processing times, labeling them as either easy (faster processing) or complex (slower processing). By organizing transactions in this manner, the approach prioritizes and schedules all easy transactions first, ensuring that complex transactions are processed only afterward. This method is hypothesized to enhance transaction throughput and reduce overall processing time, thus optimizing blockchain performance [1, 2].

Our preliminary findings suggest that integrating the "easiest transaction first" methodology can significantly improve the efficiency of blockchain systems. This introduction sets the stage for a detailed exploration of the modified adaptive chain model, which will be further elaborated in subsequent sections of the paper. The following analysis will present empirical evidence and simulation results to validate the effectiveness of this approach in diverse operational scenarios, contributing to the scalability and practical applicability of blockchain technology.

II. Related Work

The literature on blockchain performance and decentralization offers a variety of models and enhancements aimed at improving transaction processing and system scalability. Seol et al. [1] introduce an adaptive blockchain-based decentralized network that uses a Variable Bulk Arrival and Variable Bulk Service (VBAVBS) model to enhance performance by accommodating variable transaction sizes and service rates. This model contrasts with the traditional Variable Bulk Arrival and Static Bulk Service (VBASBS) by allowing state transitions from any state back to the initial state, facilitating quicker recovery and higher throughput.

Kentner et al. [3] focus on the real-time performance modeling of Non-Fungible Token (NFT) chains. Their approach uses a Markovian queueing model to predict system behavior under various transaction loads, emphasizing the real-time constraints that affect blockchain performance. This work highlights the unique challenges posed by NFT transactions, including the handling of on-chain and off-chain transactions to optimize system responsiveness and capacity.

In a similar vein, Seol and Park [2] discuss an asynchronous chain model addressing the discrepancies between arrival and service rates of transactions through a Variable Bulk Arrival and Asynchronous Bulk Service model. Their model aims to manage the blockchain's performance under conditions of asynchronous transaction handling, which is crucial for maintaining system efficiency despite fluctuating network conditions.

Earlier work by Seol et al. [4] also tackled these issues by proposing a queueing model for blockchain that accounts for variable transaction arrivals and static service rates, offering foundational insights into the challenges of scaling blockchain technologies while ensuring consistent performance.

Research Challenges

The study of adaptive blockchain networks, as detailed by Seol et al. [1], highlights a key challenge: managing the variable transaction sizes and service rates. This adaptive model introduces new variables that adjust dynamically to changes in transaction volume and network conditions. These adjustments help improve system performance but also bring complexities that need careful management.

One significant research challenge in these adaptive systems is optimizing how these variables are adjusted in real-time. For example, in scenarios where the transaction load increases suddenly, the system must quickly adapt without compromising the blockchain's security or efficiency. This requires developing algorithms that can predict and react to changes in network activity swiftly and accurately.

As these adaptive models incorporate more variables to handle fluctuations in transaction sizes and network loads, there is a need to ensure that these changes do not lead to unexpected issues or vulnerabilities in the blockchain. This involves rigorous testing and validation to ensure the reliability and stability of the blockchain under various conditions.

III. Method

Balance Equations:

The method section details the mathematical framework used to analyze the transaction processing dynamics within an adaptive blockchain system. We introduce several key balance equations that describe the flow and processing of transactions both externally and within the chain.

$$P_0 \frac{n(n+1)}{2}(\lambda_e + \lambda_c) = \left(\frac{\mu_e}{n} P_{1,e} + \frac{\mu_e}{n-1} P_{2,e} + \ldots + \mu_e P_{n,e} \right) + \left(\frac{\mu_c}{n} P_{1,c} + \frac{\mu_c}{n-1} P_{2,c} + \ldots + \mu_c P_{n,c} \right) \quad (1)$$

Equation (1) ensures that the overall input rate of transactions, considering both easy (λ_e) and complex (λ_c) transactions, matches the processing rates at different stages within the blockchain. This balance is crucial for maintaining system stability and efficiency under varying load conditions.

$$(\lambda_e + 2\lambda_e + 3\lambda_e + \ldots + (n-i)\lambda_e) + \frac{\mu_e}{n-(i-1)} P_{i,e} = \lambda_e P_{i-1,e} + 2\lambda_e P_{i-2,e} + 3\lambda_e P_{i-3,e} + \ldots + (i-1)\lambda_e P_{i,e} \quad (2)$$

$$(\lambda_c + 2\lambda_c + 3\lambda_c + \ldots + (n-i)\lambda_c) + \frac{\mu_c}{n-(i-1)} P_{i,c} = \lambda_c P_{i-1,c} + 2\lambda_c P_{i-2,c} + 3\lambda_c P_{i-3,c} + \ldots + (i-1)\lambda_c P_{i,c} \quad (3)$$

Equations (2) and (3) detail the flow balance for easy and complex transactions at each stage, from entry to exit. These equations model the dynamic nature of transaction processing, where the number of transactions being processed changes at each stage, influenced by both arrival rates and service rates.

$$P_{i,e} = q_{i,e}^{-1} P_0 \left(\sum_{j=1}^{i} j \left(\sum_{k=1}^{i-1} k \left(\prod_{\ell=1}^{k-1} q_{\ell,e}^{-1} \right) \right) + i \right) \quad \text{where } q_{i,e}^{-1} = \left(\frac{(n-i)(n-i+1)}{2} + \frac{\mu_e}{\lambda_e(n-i+1)} \right)^{-1}$$

$$P_{i,c} = q_{i,c}^{-1} P_0 \left(\sum_{j=1}^{i} j \left(\sum_{k=1}^{i-1} k \left(\prod_{\ell=1}^{k-1} q_{\ell,c}^{-1} \right) \right) + i \right) \quad \text{where } q_{i,c}^{-1} = \left(\frac{(n-i)(n-i+1)}{2} + \frac{\mu_c}{\lambda_c(n-i+1)} \right)^{-1}$$

The probabilities $P_{i,e}$ and $P_{i,c}$ represent the likelihood of transactions being in specific stages for easy and complex transactions, respectively. These expressions reflect the complexity and dynamic adaptations necessary in the processing chains, accounting for both transaction types and their handling within the system.

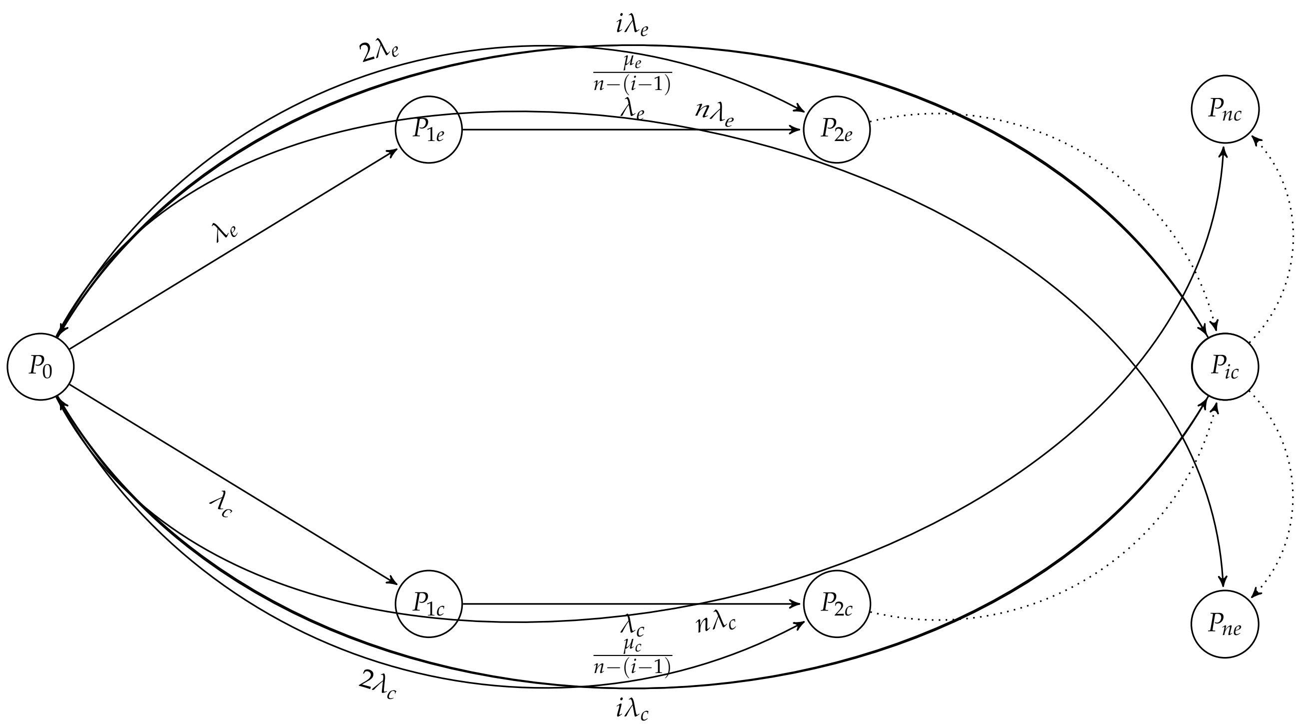
Adding all probabilities and making them equal to '1'.

$$P_0 \left[1 + A_e + A_c + \left(\frac{\lambda_e}{\mu_e} + \frac{\lambda_c}{\mu_c} \right) \frac{n(n+1)}{2} \right] = 1$$

$$P_0 = \frac{1}{1 + A_e + A_c + \left(\frac{\lambda_e}{\mu_e} + \frac{\lambda_c}{\mu_c} \right) \frac{n(n+1)}{2}} \quad \text{where } A_e = \sum_{i=1}^{n-1} \left(q_{i,e}^{-1} \left[\sum_{j=1}^{i} j \left[\sum_{k=1}^{i-1} \left(\prod_{\ell=1}^{k-1} q_{\ell,e}^{-1} \right) \right] + i \right] \right)$$

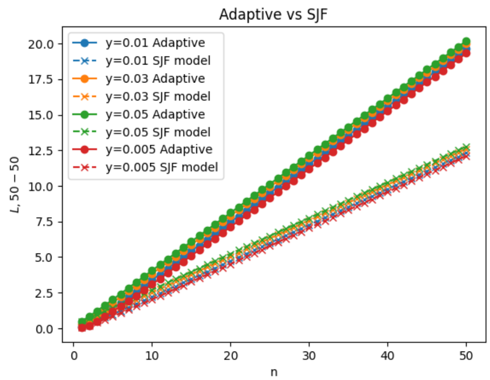
$$\text{where } A_c = \sum_{i=1}^{n-1} \left(q_{i,c}^{-1} \left[\sum_{j=1}^{i} j \left[\sum_{k=1}^{i-1} \left(\prod_{\ell=1}^{k-1} q_{\ell,c}^{-1} \right) \right] + i \right] \right)$$

This normalization ensures that all states in the system are accounted for, demonstrating the closure of the system and the comprehensive modeling of all possible transaction states within the blockchain.

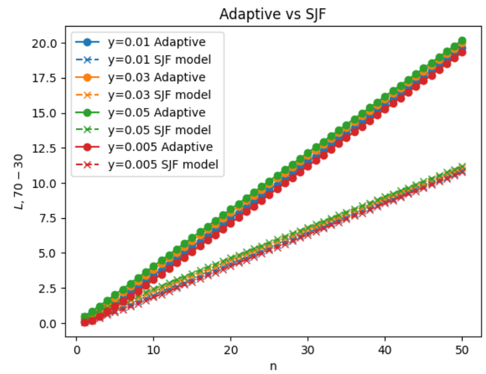


The state transition diagram represents the dynamics of transaction processing in a blockchain system, mathematically modeled through states and transitions for both easy (e) and complex (c) transactions. Initially, transactions enter the system at state P_0 and transition to processing states $P_{1e}, P_{2e}, \ldots, P_{ne}$ for easy transactions and $P_{1c}, P_{2c}, \ldots, P_{nc}$ for complex transactions. The system is designed to process all easy transactions first; transitions from P_0 to each state P_{ie} occur with rates $i\lambda_e$, and only after these are completed do transitions to P_{ic} for complex transactions begin, with rates $i\lambda_c$. This sequential processing ensures that the more straightforward transactions are prioritized, enhancing overall efficiency. Upon completion, transactions revert to P_0 from any state P_{ie} or P_{ic} at rates $\frac{\mu_e}{n-(i-1)}$ and $\frac{\mu_c}{n-(i-1)}$, reflecting the system's capacity to clear and reset for incoming transactions. This cyclic nature ensures the blockchain's efficiency and responsiveness in managing varying transaction volumes.

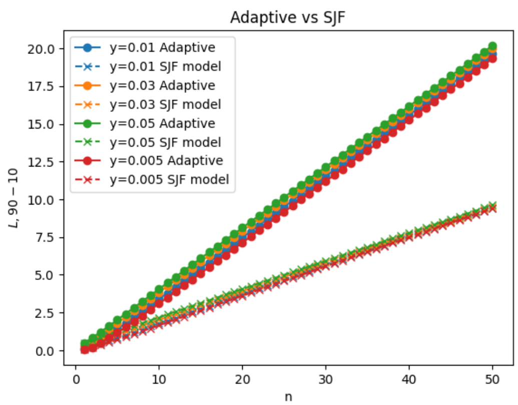
IV. Result and Analysis



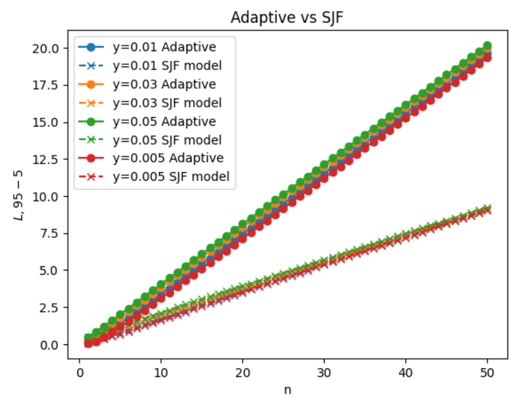
(a) L 50-50



(b) L 70-30



(c) L 90-10

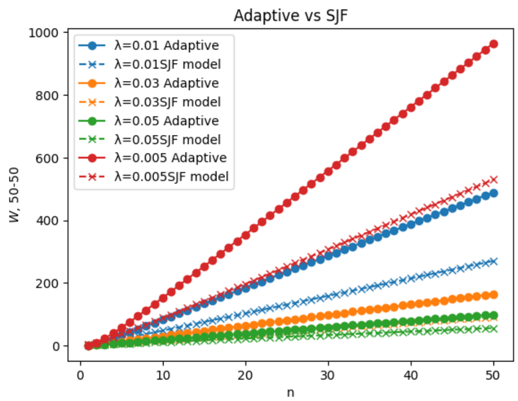


(d) L 95-5

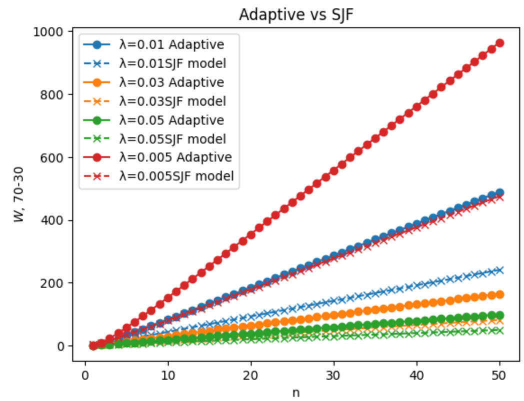
Figure 1: Comparative Analysis of Queue Lengths in Adaptive and SJF Models

Analysis on Queue Lengths:

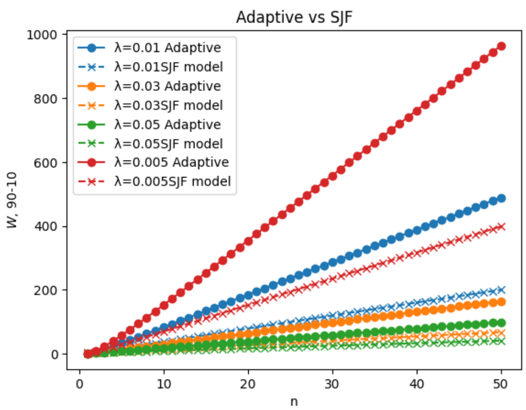
Analyzing the trends from the graphs, it is evident that both the SJF (Shortest Job First) and Adaptive models exhibit a linear increase in queue length L with the number of processes n. The SJF model consistently shows a lower queue length across all values of n, indicating its efficiency over the Adaptive model. As the parameter λ increases, representing either higher arrival rates or slower service rates, the queue lengths for both models increase, with the SJF model maintaining a relative advantage. This advantage diminishes slightly as λ decreases, suggesting the benefits of the SJF strategy are more pronounced under higher system loads.



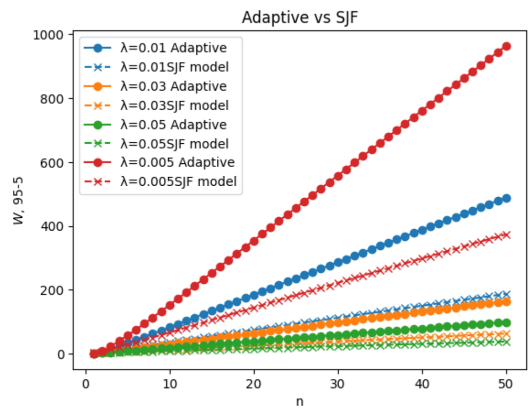
(a) W 50-50



(b) W 70-30



(c) W 90-10

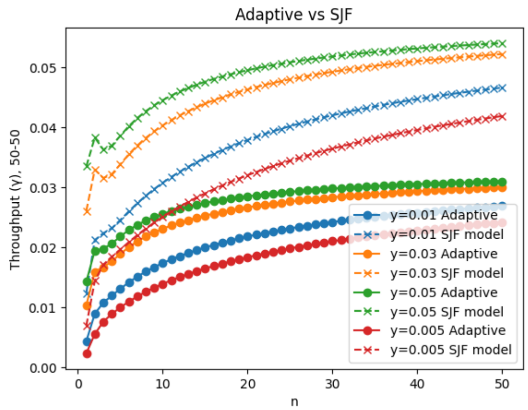


(d) W 95-5

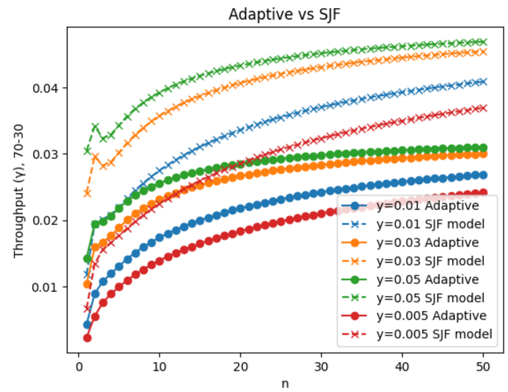
Figure 2: Comparative Analysis of Average Waiting Time in Adaptive and SJF Models

Analysis on Average Waiting Time:

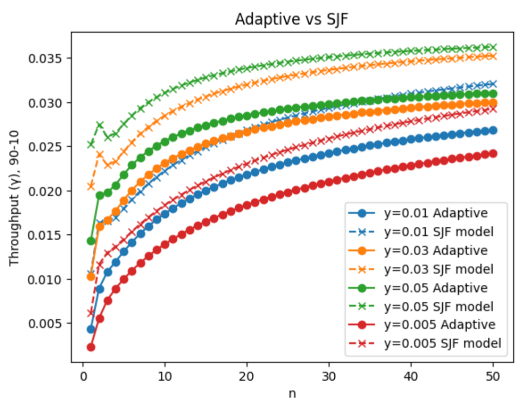
From the graphical data, it's clear that the waiting time, denoted as W, within the SJF (Shortest Job First) and Adaptive models increases with the number of processes, n. The SJF model displays a consistently lower waiting time across all process counts, signifying a more efficient handling of processes, particularly noticeable at higher values of λ. The trends suggest a near-linear relationship between W and n, with each increment in λ resulting in a steeper slope, thus a faster growth rate in waiting time. This indicates that higher arrival rates or processing complexities may result in longer wait times. However, the SJF model's performance suggests an ability to mitigate these delays better than the Adaptive model, especially noticeable as λ increases, where its lower waiting times imply a significant advantage in handling higher loads more efficiently.



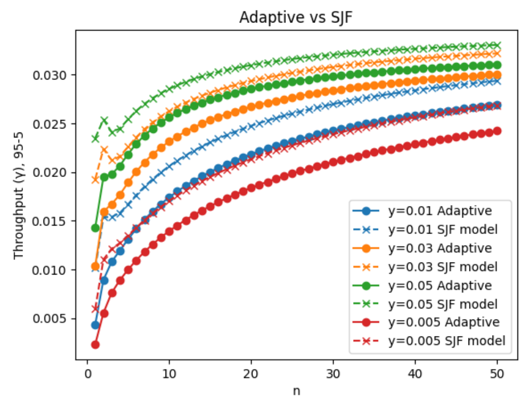
(a) λ 50-50



(b) λ 70-30



(c) λ 90-10



(d) λ 95-5

Figure 3: Throughput Comparison Between Adaptive and SJF Scheduling Models at Different System Loads

Analysis on Throughput:

From the graphical analysis, one might initially expect a lower throughput per block for the SJF (Shortest Job First) model, as transactions are processed faster, which could lead to fewer transactions per block. However, the data reveals a counterintuitive trend: the throughput, correctly denoted as γ, is actually higher for the SJF model in comparison to the Adaptive model. This unexpected result occurs in spite of the swifter transaction processing in the SJF model. Throughput γ consistently rises with the number of processes n, with a more significant increase observed at higher values of λ. As λ increases, the slope becomes steeper, indicating a more rapid growth in throughput, which suggests that the SJF model is adept at managing higher process arrival rates or processing complexities. The reasoning behind the SJF model's unexpected higher throughput will be further analyzed below.

V. Quantitative Analysis of Adaptive Chain and SJF Models

Our research on a new transaction model using a minimum priority queue to prioritize simpler transactions showed unexpected results: although we expected a decrease in transaction processing per block, we observed an increase across different mixes of simple and complex transactions. This suggests that by prioritizing simpler transactions, like a highway with lanes for faster vehicles, our model helps reduce congestion for more complex transactions, allowing them to be processed more quickly. As a result, even with fewer simple transactions per block, the overall processing speed for complex transactions improves, leading to a net increase in overall throughput. Moreover, as the proportion of simpler transactions increases, the benefits become less noticeable since there is less congestion to manage, which explains the decreasing performance gap between the standard and enhanced models. This finding underscores the model's effectiveness in improving block utilization and system efficiency by smartly managing transaction complexity.

VI. Conclusion

The analysis of the SJF (Shortest Job First) and Adaptive scheduling models' performance presents a trend: the SJF model outshines the Adaptive model. It keeps the queue length (L) and waiting time (W) low as the number of processes (n) increases. This is especially true as the system's workload, represented by λ, grows. Despite earlier thoughts that quicker transaction processing would reduce overall throughput, the SJF model actually shows higher throughput (γ), even under the stress of more complex processes, marked by λ. These results highlight the SJF model's strength in managing a busy system efficiently, positioning it as a strong and simpler option when compared to more intricate scheduling methods, especially in high-demand and complex situations.

References

[1] Seol, J., Kentner, K., Park, I.N., Joshi, S. and Park, N., 2023, October. An Adaptive Blockchain-Based Decentralized Network Computing and Performance Analysis. In 2023 Fifth International Conference on Blockchain Computing and Applications (BCCA) (pp. 335-342). IEEE.

[2] Seol, J. and Park, N., 2023, July. An Asynchronous Chain and A Variable Bulk Arrival and Asynchronous Bulk Service Model. In Proceedings of the 5th ACM International Symposium on Blockchain and Secure Critical Infrastructure (pp. 123-131).

[3] Kentner, K., Seol, J. and Park, N., 2023, October. A Real-Time Performance Model for NFT Chains. In 2023 Fifth International Conference on Blockchain Computing and Applications (BCCA) (pp. 221-228). IEEE.

[4] Seol, J., Kancharla, A., Ke, Z., Kim, H. and Park, N., 2020, October. A variable bulk arrival and static bulk service queueing model for blockchain. In Proceedings of the 2nd ACM International Symposium on Blockchain and Secure Critical Infrastructure (pp. 63-72).