A Performance Evaluation Method for a Class of Cross-Chain Systems

Ou Wu[1,2](✉), Binbin Huang[3], Shanshan Li[1], Yanze Wang[1], and Haoming Li[1]

[1] State Key Laboratory of Novel Software Technology, Software Institute, Nanjing University, Nanjing, China
lss@nju.edu.cn, ouwu123@126.com

[2] Department of Basic Courses, The PLA Army Engineering University, Nanjing, China

[3] School of Computer, Hangzhou Dianzi University, Hangzhou, China
huangbinbin@hdu.edu.cn

Abstract. In recent years, the heterogeneity among the blockchains has become the driving force behind the development of cross-chain technologies. Due to the limited processing speed of cross-chain system, excessive cross-chain transactions in the short term may cause network congestion and negatively impact. For this reason, it is essential to evaluate and optimize the performance of the cross-blockchain transaction process. However, existing research ignores the limitations of cross-chain systems. Much research is carried out to model, simulate and analyze the performance of traditional blockchain systems rather than cross-blockchain processes. To bridge this gap, our study proposes a queuing theoretical model based on system finite space, using the case of Cosmos, a typical cross-blockchain implemented by the relay mode. The solution of the steady-state equations are established by two-dimensional continuous time Markov process, and the performance measures such as average queue length, transaction rejection probability, and transaction response time are given. Finally, we simulated the analytical solutions of the relevant performance measures through experiments to verify the model's effectiveness. We believe this analytical approach can be generalized to other cross-blockchain systems.

Keywords: Blockchain · Cross-blockchain · Relays · Performance modeling · Queueing theory · Simulation

1 Introduction

Most of the blockchains in the current mainstream blockchain platforms are independent, vertical closed systems [1]. In commercial application scenarios with increasingly complex business forms, the interconnection between chains has become particularly important. As an important technical means for blockchain to achieve interoperability and improve scalability, cross-chain technology has been valued by more and more scholars.

Y. Cao and X. Shao (Eds.): MONAMI 2022, LNICST 474, pp. 265–281, 2023.
https://doi.org/10.1007/978-3-031-32443-7_19

As a bridge connecting multiple blockchains, cross-blockchain technology can overcome the problem of blockchain interoperability and save the blockchain from the island of decentralization. Many solutions and tools have been proposed to further solve the problem of blockchain interoperability [2,3]. Among them, the side chain/relay mode, Cosmos, Polkadot and Irisnet, are widely used by many representative solutions for asset portability, atomic exchange or any other more complex use cases [4,5]. In terms of the specific technology using the side chain/relay pattern, Cosmos is a highly scalable, robust, and easily upgradeable blockchain cross-chain network architecture. Each of its blockchains is supported by the BFT algorithm and can be used as a blockchain, providing reliable underlying technical support for the formation of the Internet.

Nowadays, performance has become a major factor hindering the expansion of blockchain systems. This is particularly true for high-performance systems such as real-time payment system [6]. Therefore, it is necessary to compare, evaluate and optimize the new solutions to show their efficiency and effectiveness. In addition, performance evaluation of the blockchain can also identify the system's performance bottlenecks. This can be used to further optimize the blockchain system in a specific ways. For these reasons, many researches have proposed performance evaluation solutions for blockchain systems based on mathematical modeling methods [7–12].

To our knowledge, there is no research on the use of mathematical modeling methods to evaluate the performance of cross-blockchain system. Due to the difference of consensus algorithm and communication protocol, the process of cross-blockchain technologies may be different. And the mathematical modeling method involves the complexity of theoretical derivation and the application of conditions [6,13]. Therefore, it is promising to use this way to study, analyze, and set up the fundamental understanding about the performance capability and bottleneck of specific cross-blockchain processes.

This paper takes the typical solution Cosmos as a case, and applies queuing theory to model its cross-blockchain process.

This paper has the following contributions.

(1) For the complex cross-chain process of Cosmos, consider several key factors, such as transaction arrival rate, block size and block generate time, transaction pool capacity, etc., we propose a model for transaction batch service based on queuing theory.
(2) Considering that the transaction service process is block generation and k-round block consensus. Through the established two-dimensional continuous time Markov process, we solve the state probability vector of stationary equations with faster convergence of the sub-rate matrix. Finally, the expressions of system queue length, transaction rejection probability, transaction execution time and other performance measures are obtained.
(3) We built MATLAB R2016a software platform to simulate the established model. The system capacity, transaction arrival rate, block size and other parameters are adjusted to simulate the impact on the system performance measures. Experiments show that the stability of our proposed model is good and efficient.

2 Related Work

The use of mathematical modeling methods to evaluate blockchain performance has been studied by many scholars. Kasahara et al. [14] gave the earliest study on the analysis of Bitcoin transaction confirmation time performance using queuing theory, where they provided some abstract ideas and queuing theory model to inspire the follow-up research. Based on the work of Kasahara, Li et al. [15] divided the service process into block generation and blockchain construction, established a $G/M/1$ queuing model and provided system expressions. Jiang et al. [16] provided a series model of transaction processing on the Hyperledger Fabric platform. The performance indexes such as system throughput, transaction rejection probability and transaction response delay are derived by using the state transition graph of queuing theory. Memon et al. [17] using queuing theory models $M/M/1$ and $M/M/c$ to simulate and model the memory and mining pool of the blockchain system, and provide transaction performance indicators through the memory and mining pool. Ricci et al. [18] proposed an optimized framework that combines machine learning and queuing theory model to identify confirmed transactions and characterizes confirmation times.

However, there is no research on performance modeling and analysis of cross-blockchain system. So we use queuing theory to model the process of a typical case system to solve this problem. It can support a better understanding of the performance characteristics of the cross-blockchain system.

3 Cosmos Architecture

So far, the cross-chain projects in the side chain/relay mode account for the largest proportion of the entire cross-chain projects, and the proportion is still increasing [19]. At the same time, there have been many mature solutions for the side chain/relay model, which have a great impact on cross-chain technology, the most representative of which is Cosmos[1] in 2016.

3.1 Cosmos Architecture and Consensus

Cosmos is a scalable, easy-to-use, and interoperable decentralized network of multiple independent parallel blockchains. There are three important components: The Hub is the relay chain, maintained by the government and used as a trust center for cross-chain messages. Zones are parachains participating in the Cosmos network. To support cross-chain interoperability between parachains, Cosmos proposes the Inter-Chain Communication Protocol (IBC) to perform cross-chain operations with Hub [20]. The Hub is used for cross-chain management in the parachain area. It connects the blockchain developed based on the Cosmos-SDK module to the Hub, and uses the Tendermint consensus algorithm to achieve cross-chain. As the first central chain, Cosmos hub enables network changes and updates through a simple administrative mechanism.

[1] https://github.com/cosmos.

In a single hub system, cross-chain transactions are queued in the mempool of the proposer node through different zones, waiting for the tendermint core to perform consensus verification. Tendermint's consensus mechanism is based on the Byzantine fault-tolerant algorithm. According to the rules, validators must reach a consensus on each block in rounds. Each round consists of three steps:

- Proposal stage. The proposer in each round is selected deterministically from an ordered list in proportion to the voting weight. The voting ratio in the whole process is calculated based on the Stake ratio, and each validator node has a different voting weight according to the number of tokens pledged by each validator node.
- Prevoting stage. Each validator broadcasts their own prevote. When a block in the round receives more than 2/3 of the prevote, it enters the next stage.
- Precommit phase. Each validator broadcasts their precommitted vote. When the vote exceeds 2/3, it enter the next stage.

The process is shown in Fig. 1. The block can only enter the commit phase when the adivators of 2/3 is consistent in the prevoting and precommit phases. Otherwise, it means that the block submission failed. In this case, the Tendermint protocol will choose the next validator to propose a new block at the same height and start voting again. At this time, the consensus time of this block is much longer than that of other blocks, but there is no rollback phenomenon.

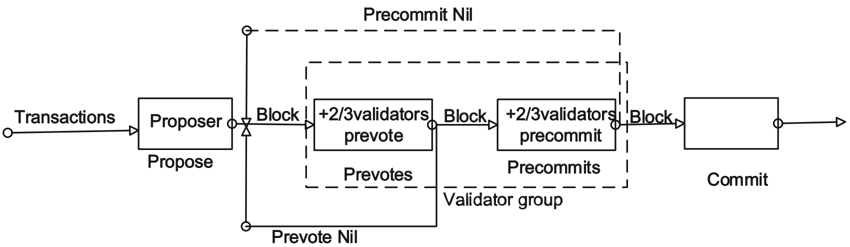

Fig. 1. Cosmos consensus process

4 Modeling Using Queuing Theory

Queuing theory is a mathematical method for solving different types of queuing system performance and service quality. In this paper, we model and test the core cross-blockchain processes of Cosmos, and use the batch service queuing theory to evaluate the performance of Cosmos hub system.

4.1 Cosmos Cross-Blockchain Queuing Representation

In the cross-chain activities of Cosmos, multiple Hub (relay)-centric blockchain alliances will form a huge network. Due to the complexity of the blockchain alliance, we first solve the cross-chain process problem of a single-relay system.

In the Cosmos hub, n validators are randomly selected to form a validator group to provide consensus verification for cross-chain transactions. The consensus process of the validator group is briefly as follows: Firstly, one of the validators be selected as the proposer node (generated by the validator in turn), then the proposer node starts to monitor and collect all transactions of the whole network, and store them in the memory pool to wait for consensus. Secondly, the proposer node will assemble a new block of cross-chain transactions, the proposal block, and broadcast it to other validators. Finally, when all validator nodes in the entire network receive the proposal block, they read all transactions in this block, vote after confirming that there is no problem, and broadcast the voting message to all validators again. However, when the prevote or precommit fails to pass the vote, the block is sent back for verification, and this process loops until the block is successfully verified. The process is shown in the Fig. 2.

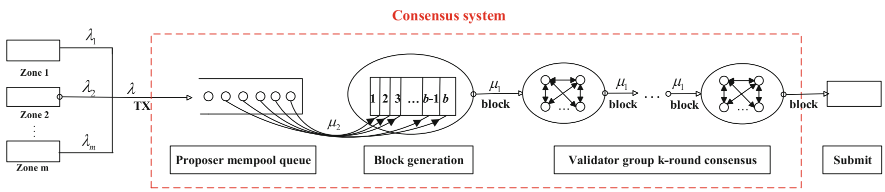

Fig. 2. Cross-chain process of Cosmos hub system.

We build a queuing system for Cosmos as follows.

Arrival Process: Assuming there are m zones in the system, the ith zone sends λ_i cross-chain transactions per second on average, which is equivalent to randomly receiving $\lambda = \sum_{i=1}^{m} \lambda_i$ cross-chain transactions per second in mempool. That is, in the entire Cosmos hub system, the number of cross-chain transactions that arrive is a Poisson process with parameter λ.

Service Rules: For cross-chain process of Cosmos, we set the service rule as first-come, first-served (FCFS).

Service Process: The first stage of the service is the process of generating blocks for a transaction. We assume that the block generation time of a transaction follows the exponential distribution with μ_2. Then the proposer broadcasts the packaged blocks to other validators, and verifies the transactions in the block through prevote or precommit. Once the transaction fails to pass the verification, the block will be sent back to the proposer for re-verification. Here we assume that the verification time of each round of the block obeys the exponential distribution with μ_1, and assumes that the block has k rounds of verification, so the entire verification stage obeys the k-order Erlang distribution.

The Maximum System Capacity: The maximum capacity of the Cosmos hub system is N.

Independence: All parameters included in this paper are independent of each other.

4.2 A Continuous-Time Markov Process

We regard the validator group as a service desk, service time includes block generation time and k-round block verification time. When transactions arrive as Poisson process and the transaction pool is limited, we establish a continuous-time Markov process for this Cosmos hub system, and the stable probability vector is solved by constructing the a series sub-rate matrix.

Assume that $\xi(t)$ and $\eta(t)$ represent the number of transactions in the system queue and the number of transactions in the block at time slot t, respectively. Then $(\xi(t), \eta(t))$ is a state of the queuing system at time t. Here $i = 0, 1, \cdots, N$, $j = 0, 1, 2, \cdots, b$ and $i + j \leq N$. Then

$$\Omega = \{(\xi, \eta) : \xi = 0, 1, \cdots, N; \eta = 0, 1, 2, \cdots, b\}$$

where $(\xi(t), \eta(t))$ is considered to be a continuous-time Markov process on Ω. Figure 3 shows the transfer relationship among the states.

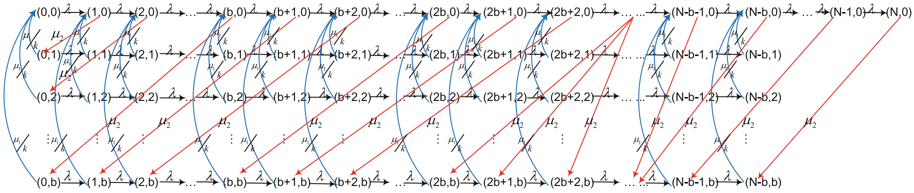

Fig. 3. State transition diagram.

The steady-state equations of cosmos hub system are obtained as follows.

- State $\{(0, 0)\}$

$$-\lambda P(0,0) + \frac{\mu_1}{k}[P(0,1) + P(0,2) + \cdots + P(0,b)] = 0 \tag{1}$$

- State $\{(0, \eta), \eta = 1, 2, \cdots, b\}$

$$-(\lambda + \frac{\mu_1}{k})P(0,\eta) + \mu_2 P(\xi, 0) = 0 \tag{2}$$

- State $\{(\xi, 0), \xi = 1, 2, \cdots, N - b\}$

$$\begin{aligned} -(\lambda + \mu_2)P(\xi,0) + \lambda P(\xi-1,0) + \frac{\mu_1}{k}[P(\xi,1) + P(\xi,2) \\ + ... + P(\xi,b)] = 0 \end{aligned} \tag{3}$$

- State $\{(\xi, b), \xi = 1, 2, \cdots, N - b - 1\}$

$$-(\lambda + \frac{\mu_1}{k})P(\xi,b) + \lambda P(\xi-1,b) + \mu_2 P(b+\xi,0) = 0 \tag{4}$$

– State $\{(N-b,b)\}$

$$-\frac{\mu_1}{k}P(N-b,b)+\lambda P(N-b-1,b)+\mu_2 P(N,0)=0 \tag{5}$$

– State $\{(N-b,\eta),\eta=1,2,\cdots,b-1\}$

$$-\frac{\mu_1}{k}P(N-b,j)+\lambda P(N-b-1,\eta)=0 \tag{6}$$

– State $\{(\xi,\eta),\xi=1,2,\cdots,N-b-1;\eta=1,2,\cdots,b-1\}$

$$-(\frac{\mu_1}{k}+\lambda)P(\xi,\eta)+\lambda P(\xi-1,\eta)=0 \tag{7}$$

– State $\{(\xi,0),\xi=N-b+1,N-b+2,\cdots,N-1\}$

$$-(\mu_2+\lambda)P(\xi,0)+\lambda P(\xi-1,0)=0 \tag{8}$$

– State $\{(N,0)\}$

$$-\mu_2 P(N,0)+\lambda P(N-1,0)=0 \tag{9}$$

If the system fails to generate blocks, then transactions continue to accumulate in the queue. Here, we're particularly analyzing this situation, namely the state $\{(\xi,0),\xi=N-b+1,N-b+2,\cdots,N-1\}$.

We can obtain the following conclusion from Eqs. (8) and (9):

$$\begin{cases} P(N-b+1,0)=\frac{\lambda}{\lambda+\mu_2}P(N-b,0) \\ P(N-b+2,0)=(\frac{\lambda}{\lambda+\mu_2})^2 P(N-b,0) \\ \vdots \\ P(N-1,0)=(\frac{\lambda}{\lambda+\mu_2})^{b-1}P(N-b,0) \\ P(N,0)=\frac{\lambda^b}{(\lambda+\mu_2)^{b-1}\mu_2}P(N-b,0) \end{cases} \tag{10}$$

We plug Eq. (10) into Eqs. (4) and (5), then

$$\begin{cases} -(\frac{\mu_1}{k}+\lambda)P(\xi,b)+\lambda P(\xi-1,b)+\mu_2 P(b+\xi,0)=0, \\ \xi=1,2,\cdots,N-2b \\ -(\frac{\mu_1}{k}+\lambda)P(\xi,b)+\lambda P(\xi-1,b)+\mu_2(\frac{\lambda}{\lambda+\mu^2})^{\xi-(N-2b)}P(N-b,0)=0, \\ \xi=N-2b+1,N-2b+2,\cdots,N-b-1 \end{cases} \tag{11}$$

$$-\frac{\mu_1}{k}P(N-b,b)+\lambda P(N-b-1,b)+\frac{\lambda^b}{(\lambda+\mu_2)^{b-1}}P(N-b,0)=0 \tag{12}$$

Combine Eqs. (1)–(3), (6)–(9) and (11), (12), we have a $(N-b+1)$-order square matrix as minimum generator of this system:

$$Q = \begin{bmatrix} B_0 & A_0 & & & & & & & & \\ B_1 & A_1 & A_0 & & & & & & & \\ B_2 & & A_1 & A_0 & & & & & & \\ \vdots & & & A_1 & A_0 & & & & & \\ \vdots & & & & \ddots & \ddots & & & & \\ B_b & & & & & A_1 & A_0 & & & \\ & B_b & & & & & A_1 & A_0 & & \\ & & B_b & & & & & \ddots & \ddots & \\ & & & \ddots & & & & & A_1 & A_0 \\ & & & & B_b & C_1 & C_2 & \cdots & C_{b-1} & A_M \end{bmatrix}$$

where $A_0, A_1, B_0, B_i(i=1,2,\cdots,b), C_j(j=1,2\cdots,b-1), A_M$ are $(b+1)$-order square matrices, and

$$A_0 = \begin{bmatrix} \lambda & & & \\ & \lambda & & \\ & & \ddots & \\ & & & \lambda \end{bmatrix}, A_1 = \begin{bmatrix} -(\lambda+\mu_2) & & & \\ \frac{\mu_1}{k} & -(\lambda+\frac{\mu_1}{k}) & & \\ \vdots & & \ddots & \\ \frac{\mu_1}{k} & & & -(\lambda+\frac{\mu_1}{k}) \end{bmatrix},$$

$$B_0 = \begin{bmatrix} -\lambda & & & \\ \frac{\mu_1}{k} & -(\lambda+\frac{\mu_1}{k}) & & \\ \vdots & & \ddots & \\ \frac{\mu_1}{k} & & & -(\lambda+\frac{\mu_1}{k}) \end{bmatrix}, B_1 = \begin{bmatrix} 0 \ \mu_2 \ 0 \cdots 0 \\ \\ \\ \end{bmatrix}, B_2 = \begin{bmatrix} 0 \ 0 \ \mu_2 \cdots 0 \\ \\ \\ \end{bmatrix},$$

$\cdots,$

$$B_b = \begin{bmatrix} 0 \ 0 \ 0 \cdots \mu_2 \\ \\ \\ \end{bmatrix}, C_1 = \begin{bmatrix} 0 \cdots 0 \ 0 \ \mu_2(\frac{\lambda}{\lambda+\mu_2}) \\ \\ \\ \end{bmatrix}, C_2 = \begin{bmatrix} 0 \cdots 0 \ 0 \ \mu_2(\frac{\lambda}{\lambda+\mu_2})^2 \\ \\ \\ \end{bmatrix},$$

$\cdots,$

$$C_{b-1} = \begin{bmatrix} 0 \cdots 0 \ 0 \ \mu_2(\frac{\lambda}{\lambda+\mu_2})^{b-1} \\ \\ \\ \end{bmatrix}, A_M = \begin{bmatrix} -(\lambda+\mu_2) & 0 \cdots 0 & \frac{\lambda^b}{(\lambda+\mu_2)^{b-1}} \\ \frac{\mu_1}{k} & 0 \cdots 0 & 0 \\ \vdots & \vdots \quad \vdots & \vdots \\ \frac{\mu_1}{k} & 0 \cdots 0 & 0 \end{bmatrix}$$

Suppose $\pi = (\pi_0, \pi_1, \pi_2, \cdots, \pi_{N-b})$ is the state probability vector based on the minimum generator, where $\pi_i = (\pi_{i0}, \pi_{i1}, \cdots, \pi_{ib}), i = 0, 1, 2, \cdots, N-b$ is a $b+1$ dimensional row vector, then we have

$$\pi_0 B_0 + \pi_1 B_1 + \pi_2 B_2 + \cdots + \pi_b B_b = 0 \tag{13}$$

$$\pi_0 A_0 + \pi_1 A_1 + \pi_{b+1} B_b = 0 \tag{14}$$

$$\pi_{i-1}A_0 + \pi_i A_1 + \pi_{i+b}B_b = 0, i = 2, 3, \cdots, N-2b \tag{15}$$

$$\pi_{i-1}A_0 + \pi_i A_1 + \pi_{N-b}C_{i-(N-2b)} = 0, \quad i = N-2b+1, N-2b+2, \cdots, N-b-1 \tag{16}$$

$$\pi_{N-b-1}A_0 + \pi_{N-b}A_M = 0 \tag{17}$$

$$\pi e = 1 \tag{18}$$

The dimension of column vector e depends on the collocation matrix.

We use matrix analysis method to solve steady-state probability vector [21]. Here, the diagonal matrix A_0 is expressed as $A_0 = \lambda I$ (I is the $(b+1)$-order identity matrix). Let $R_{N-b} = I$, then

$$\pi_{N-b} = \pi_{N-b}R_{N-b} \tag{19}$$

From Eq. (17), we get

$$\pi_{N-b-1} = \pi_{N-b}(-\frac{1}{\lambda}A_M) = \pi_{N-b}R_{N-b-1} \tag{20}$$

Where $R_{N-b-1} = -\frac{1}{\lambda}A_M$ is a sub-rate matrix.

We put Eq. (20) into Eq. (16), then

$$\begin{aligned} \pi_{N-b-(i+1)} &= \pi_{N-b}[-\frac{1}{\lambda}(R_{N-b-i}A_1 + C_{b-i})] \\ &= \pi_{N-b}R_{N-b-(i+1)}, \\ i &= 1, 2, \cdots, b-1 \end{aligned} \tag{21}$$

where

$$R_{N-b-(i+1)} = -\frac{1}{\lambda}(R_{N-b-i}A_1 + C_{b-i}), i = 1, 2, \cdots, b-1.$$

We plug Eq. (21) into Eq. (15), then

$$\begin{aligned} \pi_{N-b-(i+1)} &= \pi_{N-b}[-\frac{1}{\lambda}(R_{N-b-i}A_1 + R_{N-i}B_b)] \\ &= \pi_{N-b}R_{N-b-(i+1)}, \\ i &= b, b+1, \cdots, N-b-1 \end{aligned} \tag{22}$$

where

$$R_{N-b-(i+1)} = -\frac{1}{\lambda}(R_{N-b-i}A_1 + R_{N-i}B_b), i = b, b+1, \cdots, N-b-1.$$

From Eq. (13), we get

$$\begin{aligned} \pi_0 &= -\pi_{N-b}(R_1B_1 + R_2B_2 + \cdots + R_bB_b)B_0^{-1} \\ &= \pi_{N-b}R_0, \end{aligned} \tag{23}$$

where $R_0 = -(R_1B_1 + R_2B_2 + \cdots + R_bB_b)B_0^{-1}$. Figure 4 shows the solving process of $R_i, i = 0, 1, 2, \cdots, N-b$.

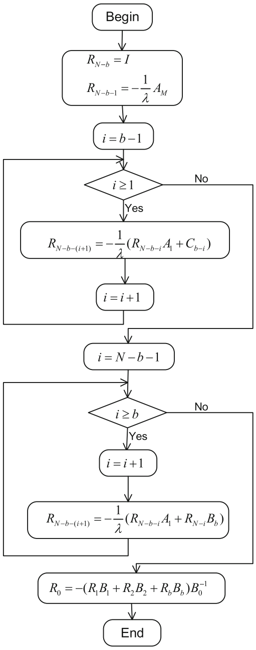

Fig. 4. R_i calculation flow chart.

Combine Eqs. (14) and (18), we get

$$\begin{cases} \pi_{N-b}(R_0 A_0 + R_1 A_1 + R_{b+1} B_b) = 0 \\ \pi_{N-b}(R_0 + R_1 + R_2 + \cdots + R_{N-b-1} + I)e = 1 \end{cases}. \tag{24}$$

π_{N-b} is substituted into the Eqs. (20)–(23) to solve the steady-state probability vector π.

Then we obtain the performance measures of consensus hub system, and analyze the influence of parameters on them. The conditions for a stable queue system are as follows:

$$\lim_{t\to+\infty} \xi(t) = \xi_q, \lim_{t\to+\infty} \eta(t) = \eta_b, \tag{25}$$

(a) Average number of transaction in queue

$$E(L_q) = \sum_{i=0}^{N-b} (i \sum_{j=0}^{b} \pi_{ij}) = \pi_{N-b}[R_1 + 2R_2 + \cdots + (N-b)R_{N-b}]e \tag{26}$$

(b) Average transaction execution time

$$E(T_{exe}) = \sum_{m=0}^{[\frac{N-b-l}{b}]} \sum_{l=0}^{b-1} \pi_{mb+l,0}(m+1)(\frac{k}{\mu_1} + \frac{1}{\mu_2}) + \sum_{m=0}^{[\frac{N-b-l}{b}]} \sum_{l=0}^{b-1} \sum_{j=1}^{b} \pi_{mb+l,j}[\frac{k}{\mu_1} + (m+1)(\frac{k}{\mu_1} + \frac{1}{\mu_2})] \tag{27}$$

The proof is analogous to the literature [15], where $[\frac{N-b-l}{b}]$ is a integer function.

(c) Transaction rejection probability

$$P_{rjc} = \sum_{j=0}^{b} \pi_{N-b,j} = \pi_{N-b}e \tag{28}$$

(d) Average transaction response time

$$E(T_{resp}) = \frac{E(L_q)}{\lambda(1-p_{rjc})} \tag{29}$$

(e) Throughput of system

$$E(TPS) = \lambda(1-p_{rjc}) \tag{30}$$

5 Model Simulation and Evaluation

This section, we simulated the performance measures of Cosmos hub system by conducting experiments and testing necessary data. We provided several graphs of important measures about the parameter λ, N, and μ_1, μ_2.

5.1 Test Framework

In this section, $MATLABR2016a$ software platform is installed to study parameter's impact on performance measures by setting parameter ranges, so as to verify the accuracy of this model.

5.2 Performance Evaluation of Cosmos Hub System

(A) Impact of arrival rate λ on system performance

We set the variation range of λ to be 0 to 5000 txs/s, $\mu_1 = 4, \mu_2 = 1$ txs/s, the Cosmos hub system capacity $N = 500$. On average, each transaction performs one round of consensus verification, that is $k = 1$. Figure 5 shows the change trend in performance measures such as rejection probability and transaction response time, etc.

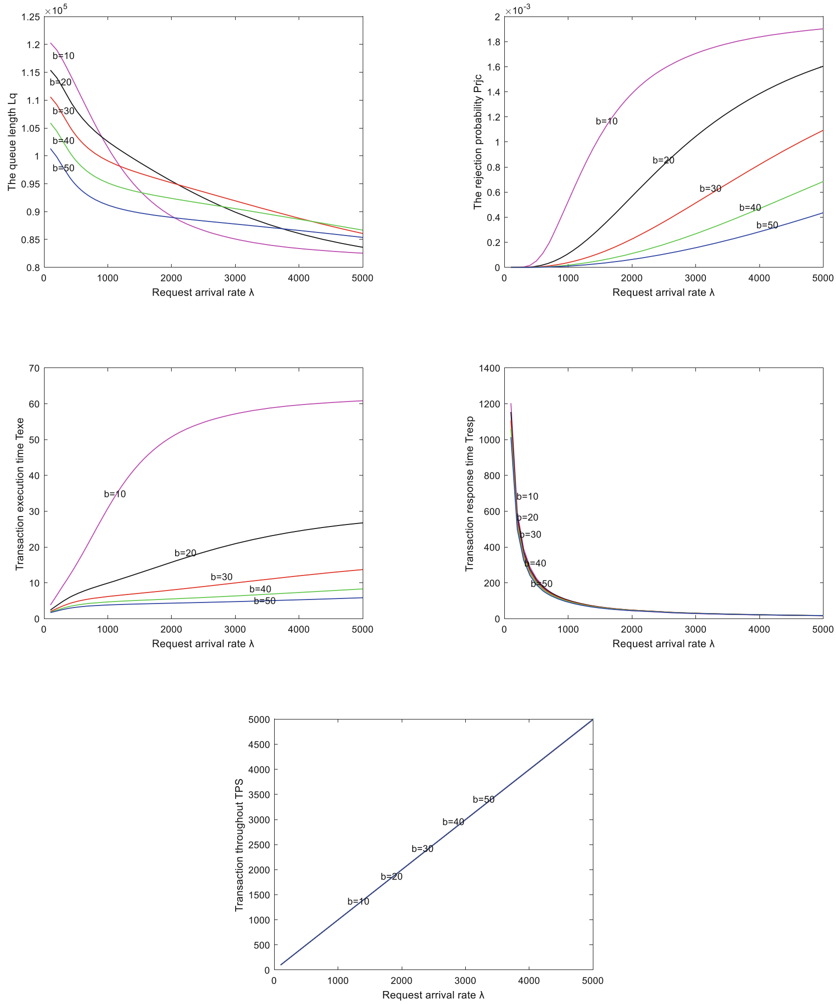

Fig. 5. Performance measures with λ

Figure 5 shows the trends of the five performance measures of the Cosmos hub system with respect to value of b and λ. When the value of b is determined, the larger the transaction arrival rate λ, the smaller the queue length Lq and the transaction response time T_{resp}, the larger is the rejection probability P_{rjc} and the transaction execution time T_{exe}, the transaction throughput TPS is proportional to the transaction arrival rate λ. When λ is determined, the queue length Lq decreases as the value of b increases, and the rejection probability P_{rjc} increases as the value of b increases. Execution time T_{exe} and system throughput TPS are independent of changes with block size b.

To sum up, when the value of λ is $\lambda \leq 1000$, we try to set a large block size. At this time, the queue length is small, and the consensus efficiency is also high. But when $\lambda \geq 1000$, the larger the block, the longer the queue length, so the system accumulation will increase, but the consensus efficiency is always high. Given several interrelated performance metrics, we cannot blindly pursue large block sizes when setting system parameters. An appropriate block size b ensures good system performance and small system builds.

(B) Influence of Cosmos hub system capacity (N)

We set the range of N to be 0 to 5000 transactions, $\mu_1 = 4, \mu_2 = 1$ txs/s. When the value of λ is $\lambda = 1000$ txs/s, and $k = 1$. Figure 6 shows the changes with the five performance measures.

Figure 6 shows the value of b is determined, and when the consensus system capacity is $N \geq 1000$, no matter the change of N, it has little effect on the rejection probability and throughput. When $N \leq 1000$, the system rejection probability P_{rjc} decreases with the increase of N, while the throughput increases with the increase of N. N has little effect on the transaction execution time T_{exe}, but its increase will lead to an increase in transaction response time. When N is determined, P_{rjc} decreases faster with the larger b, T_{exe} decreases with the increase of b, and the TPS growth rate also increases with the increase of b.

In summary, when the value of N increases, although the rejection probability gets closer to 0 and the transaction throughput (TPS) tends to stabilize (proportional to the transaction arrival rate λ), but the queue length and response time also increase. Therefore, an appropriately sized N will balance the performance of the entire system, rather than a larger N, the better the system performance.

(C) transaction consensus rate (μ_1) and consensus rounds (k)

The parameters μ_1, μ_2 represent a round transaction consensus rate and block generation rate. Their values are related to the number of ordering nodes and peer nodes. Consider the above test, let $\mu_1 = 4, \mu_2 = 1, N = 500, \lambda = 1000$, when the range of k varies from 1–20, the range of $\mu_{11} = \frac{\mu_1}{k}$ is 0.2–4. Figure 7 shows the trend of changes of five performance measures as the dependent variable μ_{11}.

From Fig. 7, we can see that when $\mu_{11} \leq 0.2$, the curve shakes violently, especially if the rejection probability is negative, obviously the system is wrong. When $\mu_{11} \geq 0.2$, the larger the μ_{11} is, the smaller the number of transaction consensus rounds k is, that is, the higher the transaction consensus efficiency is. At this time, when the block size b is fixed, the queue length, transaction response time and system throughput increase accordingly, while the rejection probability and transaction execution time decrease, indicating that the system performance is getting better and better. Compared to block size b, larger blocks also lead to better performance.

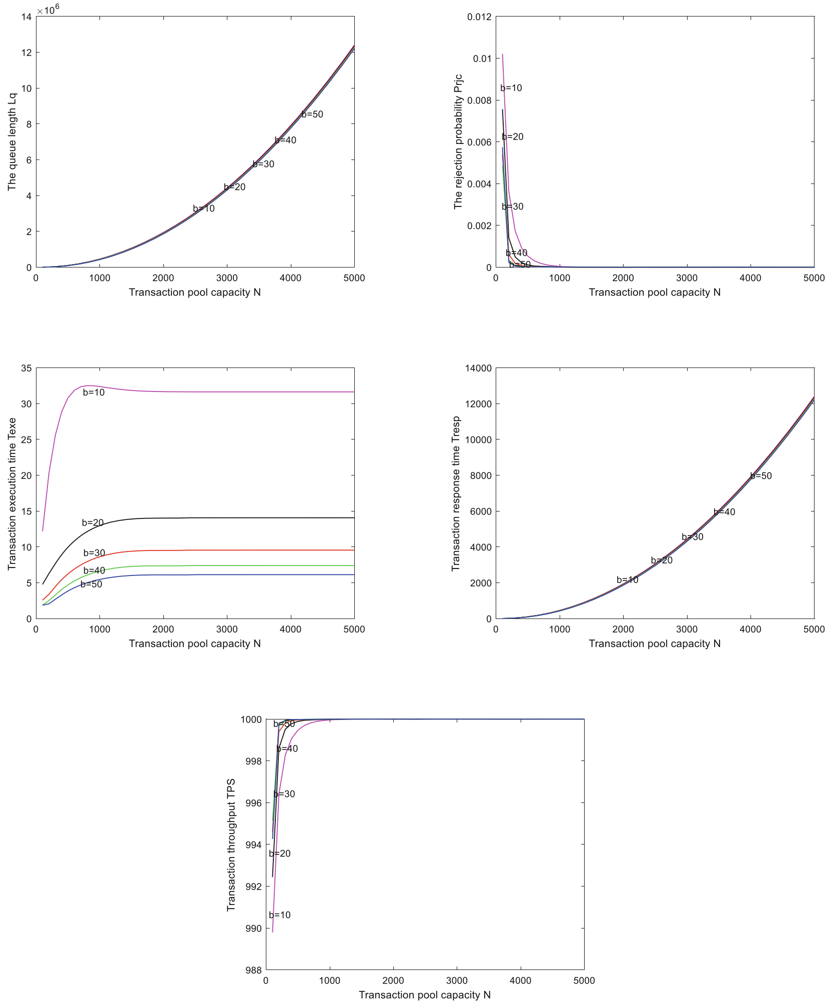

Fig. 6. Performance measures with N

To sum up, the variable μ_{11} is a reflection of the system settings. When the number of validator nodes is small, the system consensus efficiency is naturally high, but at the same time, there will be security problems. Therefore, when setting up the consensus system, choosing an appropriate number of nodes while taking into account both security and efficiency is also a problem that we need to consider.

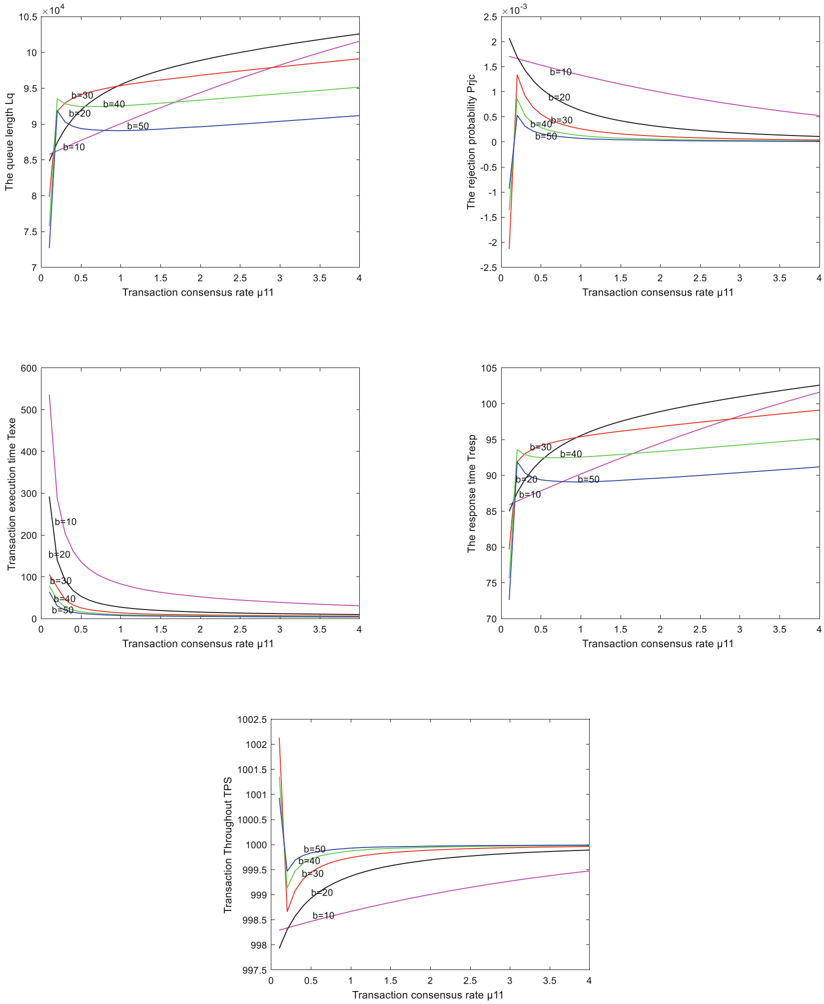

Fig. 7. Performance measures with μ_{11}

6 Conclusion

As a key technology in the development of blockchain field, the theory and performance of cross-chain technology is a main research topic. This paper first selects the representative cross-chain technology Cosmos to model the core cross-chain process. Secondly, we apply queuing theory to derive the performance evaluation measures of the Cosmos hub system. Finally, we simulated the model by testing the transaction consensus time and block generation rate of cross-chain transactions to verify the validity of the model. The model proposed and validated in this paper can still be used to evaluate the performance of other cross-chain technologies using similar patterns.

There are three future directions for this research: 1) Optimize the existing queuing theory model to improve the utilization of node group consensus in Cosmos Hub. 2) Improve the equipment configuration used to build the Cosmos cross-chain platform and identify the performance bottleneck of the relay system. 3) Compare other cross-chain systems with the relay model to expand the applicability of the model.

References

1. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system, vol. 4 (2008). https://bitcoin.org/bitcoin.pdf
2. Buterin, V.: Chain interoperability. R3 Research Paper (2016)
3. Deng, L., Chen, H., Zeng, J., Zhang, L.-J.: Research on cross-chain technology based on sidechain and hash-locking. In: Liu, S., Tekinerdogan, B., Aoyama, M., Zhang, L.-J. (eds.) EDGE 2018. LNCS, vol. 10973, pp. 144–151. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-94340-4_12
4. Kannengießer, N., Pfister, M., Greulich, M., Lins, S., Sunyaev, A.: Bridges between islands: Cross-chain technology for distributed ledger technology. In: Proceedings of the 53rd Hawaii International Conference on System Sciences (2020)
5. Borkowski, M., Frauenthaler, P., Sigwart, M., Hukkinen, T., Hladkỳ, O., Schulte, S.: Cross-blockchain technologies: review, state of the art, and outlook. White Paper (2019). http://dsg.tuwien.ac.at/projects/tast/pub/tast-white-paper-4.pdf
6. Fan, C., Ghaemi, S., Khazaei, H., Musilek, P.: Performance evaluation of blockchain systems: a systematic survey. IEEE Access **8**, 126 927–126 950 (2020)
7. Huang, D., Ma, X., Zhang, S.: Performance analysis of the raft consensus algorithm for private blockchains. IEEE Trans. Syst. Man Cybern. Syst. **50**(1), 172–181 (2019)
8. Geyer, F., et al.: Performance perspective on private distributed ledger technologies for industrial networks. In: 2019 International Conference on Networked Systems (NetSys), pp. 1–8. IEEE (2019)
9. Geissler, S., Prantl, T., Lange, S., Wamser, F., Hossfeld, T.: Discrete-time analysis of the blockchain distributed ledger technology. In: 2019 31st International Teletraffic Congress (ITC 31), pp. 130–137. IEEE (2019)
10. Sukhwani, H., Wang, N., Trivedi, K.S., Rindos, A.: Performance modeling of hyperledger fabric (permissioned blockchain network). In: 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA), pp. 1–8. IEEE (2018)
11. Yuan, P., Zheng, K., Xiong, X., Zhang, K., Lei, L.: Performance modeling and analysis of a hyperledger-based system using GSPN. Comput. Commun. **153**, 117–124 (2020)
12. Papadis, N., Borst, S., Walid, A., Grissa, M., Tassiulas, L.: Stochastic models and wide-area network measurements for blockchain design and analysis. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications, pp. 2546–2554. IEEE (2018)
13. Smetanin, S., Ometov, A., Komarov, M., Masek, P., Koucheryavy, Y.: Blockchain evaluation approaches: state-of-the-art and future perspective. Sensors **20**(12), 3358 (2020)
14. Kasahara, S., Kawahara, J.: Effect of bitcoin fee on transaction-confirmation process. J. Industr. Manage. Optim. **15**(1), 365 (2019)

15. Li, Q.-L., Ma, J.-Y., Chang, Y.-X.: Blockchain queue theory. In: Chen, X., Sen, A., Li, W.W., Thai, M.T. (eds.) CSoNet 2018. LNCS, vol. 11280, pp. 25–40. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04648-4_3
16. Jiang, L., Chang, X., Liu, Y., Mišić, J.V., Mišić, V.B.: Performance analysis of hyperledger fabric platform: a hierarchical model approach. Peer-to-Peer Netw. Appl. **13**(3), 1014–1025 (2020). https://doi.org/10.1007/s12083-019-00850-z
17. Memon, R.A., Li, J.P., Ahmed, J.: Simulation model for blockchain systems using queuing theory. Electronics **8**(2), 234 (2019)
18. Ricci, S., Ferreira, E., Menasche, D.S., Ziviani, A., Souza, J.E., Vieira, A.B.: Learning blockchain delays: a queueing theory approach. ACM SIGMETRICS Perform. Eval. Rev. **46**(3), 122–125 (2019)
19. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. Manubot, Technical report (2019)
20. Kwon, J., Buchman, E.: A network of distributed ledgers. Cosmos, dated, pp. 1–41 (2018)
21. Elhafsi, E.H., Molle, M.: On the solution to QBD processes with finite state space. Stochast. Anal. Appl. **25**(4), 763–779 (2007)