Performance and Reliability Analysis for PBFT-Based Blockchain Systems With Repairable Voting Nodes

Yan-Xia Chang, Qing Wang, Quan-Lin Li, Yaqian Ma, and Chi Zhang

***Abstract*—In a practical blockchain system based on the Practical Byzantine Fault Tolerance (PBFT) protocol, the voting nodes can fail at any time due to non-Byzantine errors, such as autonomous shutdowns, device crashes, and communication link failures caused by mobility or obstacles. These errors may cause voting nodes to exit the PBFT-based blockchain system unpredictably, resulting in a variable number of voting nodes available at any given time. To maintain optimal performance and consistency while adapting a PBFT-based blockchain system to this dynamic change, this paper proposes an extension to the PBFT protocol by introducing a repair process for failed nodes. The new PBFT-based blockchain system with repairable voting nodes is then analyzed for performance and reliability analysis by using multi-dimensional Markov processes, queueing theory, and the first passage time method. Additionally, we validate the accuracy of our theoretical findings by conducting numerical examples and simulation experiments. These experiments demonstrate that the introduction of a repair process can improve the performance and reliability of the PBFT-based blockchain system. Furthermore, we illustrate how various system parameters impact the performance measures of the PBFT-based blockchain system with repairable voting nodes. We hope that the methodology and results presented in this paper will establish a common framework for deriving theoretical analysis of existing PBFT-based blockchain systems and inspire future research efforts in this field.**

***Index Terms*—Practical Byzantine fault tolerance (PBFT), blockchain, repairable voting nodes, Markov process, queueing theory, performance evaluation, reliability.**

I. Introduction

The consensus problem has gained significant attention in recent years due to its extensive applications to a new distributed system known as blockchain. Driven by the opportunities in the concept of cryptocurrencies and blockchain, the Practical Byzantine Fault Tolerance (PBFT) consensus protocol, which evolved from "the Byzantine General problem" proposed by Pease et al. [1], plays a crucial role in the evolution of the blockchain ledger consensus and has been applied to various blockchain systems. For example, Ethereum 2.0 with the Casper protocol [2], Hyperledger Fabric with the PBFT protocol [3], Facebook Libra with the LibraBFT protocol [4], and the cross-chain system Cosmos with the Tendermint protocol [5].

To enhance the practicality of the PBFT protocol, researchers have made significant progress in its development, thereby effectively improving the performance of PBFT-based blockchain systems. These solutions include, but are not limited to, dividing replicas into groups or categories [6], [7], [8], [9]; simplifying the PBFT process [10], [11], [12], [13], [14]; introducing a credit mechanism [6], [8], [11], [15], [16], [17], [18]; using multiple consensus algorithms in different scenarios [6], [19], [20], [21]; improving the selection of primary nodes [9], [11], [12], [13], [18], [19], [22]; and introducing geographical factors [23], [24]. At present, the PBFT protocol and its variants are still evolving and have been widely used in various fields, such as edge computing wireless networks [9], the Internet of Things (IoT) [24], [25], [26], Internet of Vehicles (IoV) [27], cloud computing [28], energy trading [29], and many other fields. However, the most fundamental processes, such as the leader election (combined with the DPoS protocol, node credit or reputation, and even smart contracts), block proposal, delivery, and commitment, are still maintained.

In contrast to public blockchains like Bitcoin [30] and Ethereum 1.0 [31], which rely on the Proof-of-Work (PoW) consensus protocol, PBFT-based blockchain systems typically involve only a subset of participating nodes running the PBFT protocol. These nodes can be tightly controlled by the membership, and once more than 2/3 of the voting nodes have approved a proposal proposed by the primary node, a round of voting consensus is reached. Thanks to the ability of PBFT-based blockchain systems to tolerate failures of less than 1/3 of voting nodes, these systems can continue to work even in the presence of malicious attacks, software errors, operator errors, and other potential issues. Meanwhile, these systems provide several advantages, including access control, low power consumption, high throughput, fast consensus speed, and scalability. For more details on the advantages, readers may refer to several survey papers, such as those by Correia et al. [32], Vukolić [33], Gramoli [34], Berger and Reiser [35], Gupta et al. [36],

Manuscript received 21 June 2023; revised 2 December 2023 and 21 March 2024; accepted 29 March 2024. Date of publication 23 April 2024; date of current version 21 August 2024. This work was supported by the National Natural Science Foundation of China under grant No. 71932002. The associate editor coordinating the review of this article and approving it for publication was Y. Liu. *(Corresponding author: Chi Zhang.)*

Yan-Xia Chang, Quan-Lin Li, Yaqian Ma, and Chi Zhang are with the School of Economics and Management, Beijing University of Technology, Beijing 100124, China (e-mail: czhang@bjut.edu.cn).

Qing Wang is with the Monash Business School, Monash University, Caulfield East, VIC 3145, Australia.

Digital Object Identifier 10.1109/TNSM.2024.3384506

Stifter et al. [37], Alqahtani and Demirbas [38], Zheng and Feng [39], Gan et al. [40], and others.

Although PBFT-based blockchain systems offer the advantages mentioned above, they also have limitations. For example, most existing PBFTs lack system flexibility, that is, the voting nodes cannot freely join or exit the PBFT network. Therefore, a more general PBFT-based blockchain system cannot accommodate the behavior of failed nodes that leave the PBFT-based blockchain system or of repaired nodes that reconnect to the PBFT network. This limitation contradicts the principle that all nodes in the PBFT-based voting network have the right to vote on a proposal or suggestion at any time. Additionally, as the unreliability mentioned by Jaafar et al. in [9] and the single-point failure mentioned by Gai et al. in [14], the voting nodes may fail at any time. If more than one-third of nodes fail and these failed nodes cannot be repaired promptly, then the PBFT-based blockchain system will never be able to achieve consensus on a specific proposal. This limitation is also harmful to the liveness, availability, and security of the PBFT-based blockchain system. Thus, it is necessary to consider a new PBFT consensus protocol in which the repaired nodes can rejoin the network. Simultaneously, it is crucial to evaluate the performance of the blockchain system based on this extended PBFT consensus. By introducing a repair process for failed nodes, the static PBFT protocol evolves into a dynamic and adaptive protocol. This improvement enables us to effectively address the impact of node failure or exit on the blockchain system. Furthermore, the introduction of repaired nodes could enable the PBFT network to achieve voting consensus, even in situations where the PBFT-based blockchain system is at risk of becoming unavailable due to a higher number of failed nodes.

Motivated by the above descriptions, our focus is on analyzing the performance and reliability of the PBFT-based blockchain system with repairable voting nodes using multi-dimensional Markov processes, queuing theory, and the first passage time method. Note that Markov processes and queueing theory play a key role in research on blockchain systems. A growing body of literature has applied Markov processes and queueing theory to study blockchain systems. For example, Eyal and Sirer [41], Göbel et al. [42], Javier and Fralix [43], Li et al. [44], Song et al. [45], Ma et al. [46], Li et al. [47], [48], and Chang et al. [49]. It can be seen from these works that Markov processes and queueing theory can effectively describe various consensus processes in blockchain systems. They possess universality and superiority in evaluating the performance measures of blockchain systems.

This paper, in terms of method, is closely related to the works of Ma et al. [46], Chang et al. [49], Nischwitz et al. [50], and Hao et al. [51]. Hao et al. [51] presented a dynamic PBFT-based blockchain system with an uncertain total number of voting nodes. In this system, nodes can join or leave the PBFT network using the JOIN and EXIT protocols. Based on these two special protocols, Chang et al. [49] conducted a performance analysis using Markov processes and an approximate queueing system. Compared with the studies conducted by Hao et al. [51] and Chang et al. [49], this paper uses a three-dimensional Markov process to describe and analyze the actions of failed nodes leaving the PBFT network and repaired nodes rejoining the PBFT network while ensuring a constant total number of voting nodes. Also, Chang et al. [49] proposed a steady-state rate approximation method such that the block generation time and orphan block generation time were exponential and then developed an approximate queueing model to calculate the throughput of the PBFT-based blockchain system. In this paper, we obtain that the probability distribution of the block generation time and the orphan block generation time are phase-type, which is more practical than theirs. Ma et al. [46] adopted a two-dimensional Markov process to describe the voting process, which can be considered as a simple and specific case of our work. Nischwitz et al. [50] used a probabilistic model to evaluate BFT protocols in the presence of dynamic link and crash failures, which differs from the approach adopted in our paper.

In addition, numerous researchers have also conducted ongoing evaluations of the performance of PBFT-based blockchain systems from different perspectives and employing various analysis methods. For example, Hao et al. [52] proposed a method to evaluate the performance of PBFT consensus in Hyperledger and showed that PBFT consistently outperforms PoW in terms of latency and throughput under varying workloads. Sukhwani et al. [53] modeled the PBFT consensus process using Stochastic Reward Nets to compute the mean time required to complete consensus for networks with a maximum of 100 peers. Lorünser et al. [54] presented a performance model for PBFT that specifically considers the impact of unreliable channels and the use of different transport protocols over them. Pongnumkul et al. [55] developed a method for evaluating Hyperledger Fabric and Ethereum. They demonstrated that Hyperledger Fabric consistently outperforms Ethereum across all evaluation metrics, including execution time, latency, and throughput. Different from these works, in this paper, we use Markov processes and queueing theory to derive theoretical expressions for various performance measures. These measures include block generation time, orphan block generation time, average time before the first failure, reliability, and throughput. Also, we employ numerical examples and simulation experiments to demonstrate that our theoretical analysis matches simulation results and highlights the positive influence of a repair process on the performance and reliability of the PBFT-based blockchain system.

Regarding simulation experiments, there are also numerous studies on performance analysis using simulation models. For example, Jaafar et al. [9] and Diouf et al. [22] relied on simulation models to provide the performance analysis of the adaptive blockchain for edge computing and the Kubernetes multi-master robust platform, respectively. These two aforementioned works considered the non-Byzantine errors and the Byzantine failures. However, they mainly focus on detailing the main components and operations of the blockchain. Meshcheryakov et al. [25] simulated the primarily distributed ledger scenarios using PBFT and evaluated the performance of the blockchain system. Monrat et al. [56] provided a performance and scalability analysis of popular private blockchain platforms. Ahmad et al. [57] developed a blockchain test platform to execute and test the latency and throughput of various blockchain systems, including PBFT, PoW, Proof of Equity, Proof of Elapsed Time, and Clique. Zheng et al. [58] simulated the time response of a PBFT-based healthcare blockchain network using continuous-time Markov chain models. In comparison to these simulation models, we mainly use Monte Carlo simulation to validate our theoretical findings and evaluate the advantages of introducing the repair process.

From the above literature streams, relevant literature has examined the performance of PBFT-based blockchain systems from three perspectives: protocol, analytical models, and simulation models. The protocol perspective focuses on improving the PBFT protocol to evaluate and enhance the performance of PBFT-based blockchain systems. The latter two perspectives mainly provide the performance analysis of PBFT-based blockchain systems based on the blockchain's operating mechanism. However, there are few studies that analyze the performance and reliability of PBFT-based blockchain systems with repairable voting nodes using Markov processes and queueing theory. Additionally, there is a lack of research on the theoretical expressions for various performance measures of the PBFT-based blockchain systems. Motivated by these factors, we aim to address these gaps. Next, we summarize the main contributions of this paper as follows:

1. We extend the static PBFT consensus protocol to a dynamic and adaptive PBFT consensus protocol. This extension was developed by analyzing the problems faced by existing PBFT protocols and introducing a repair process for failed nodes caused by non-Byzantine errors, such as automatic shutdown, device crashes, or communication link failures caused by mobility or obstacles.
2. We use Markov processes, queueing theory, and the first passage time method to conduct performance and reliability analysis for the PBFT-based blockchain system with repairable voting nodes and derive theoretical expressions for several important measures, including block generation time, orphan block generation time, throughput, availability, reliability, and the average time before the first failure.
3. We validate our theoretical findings through numerical examples and simulation experiments. The experimental results demonstrate that introducing a repair process can improve the performance and reliability of the PBFT-based blockchain system. Additionally, we illustrate how various system parameters affect the performance measures of the PBFT-based blockchain system with repairable voting nodes.

The structure of this paper is organized as follows. Section II describes a new PBFT-based blockchain system with repairable voting nodes. Section III analyzes the probability distributions of block generation and orphan block generation times by means of two phase-type distributions of finite sizes. Section IV introduces the queueing model $\mathrm{M}\oplus\mathrm{PH}^b/\mathrm{PH}^b/1$ to provide performance analysis for the PBFT-based blockchain system with repairable voting nodes. Section V sets up two new Markov processes to analyze the reliability of the PBFT-based blockchain systems with repairable voting nodes. Section VI uses numerical examples and simulation experiments to validate our theoretical results and indicate how the key parameters influence the performance measures of the PBFT-based blockchain system with repairable voting nodes. Section VII gives some concluding remarks. Finally, an Appendix provides the non-zero matrix elements or blocks in three important infinitesimal generators.

II. Model Description

In this section, we provide a detailed model description for the PBFT-based blockchain system with repairable voting nodes. Also, we give mathematical notation, random factors, and necessary parameters used in our subsequent study.

(1) The failure process of voting nodes: We assume that each voting node can fail in the PBFT-based blockchain system, and the lifetime of a voting node follows an exponential distribution with a mean of $1/\theta > 0$. Here, a larger θ indicates a less stable network. Once a voting node fails, it cannot handle any other work or task until it is repaired well to enter the working state again.

(2) The repair process of voting nodes: Once any voting node fails, it immediately enters a repair state. To analyze the impact of the time taken to distribute workloads or transaction packages to a repaired node or a repaired node to perform the JOIN protocol (e.g., the one proposed by Hao et al. [51]), we assume that the repair time of the failed voting node follows an exponential distribution with a mean of $1/\mu > 0$.

(3) The total number of voting nodes: To prevent nodes from intentionally leaving the network to manipulate the voting process and to ensure the liveness, safety, and correctness of the PBFT-based blockchain system, we assume that the total number of voting nodes in the PBFT network is a fixed value $N = 3n + 1$ for the convenience of analysis, where n represents the maximum number of failed nodes.

(4) The voting process: We assume that the voting time of each node follows an exponential distribution with a mean of $1/\gamma > 0$. Moreover, each voting node is limited to one voting opportunity during a voting period.

(5) Probability of approving or disapproving the transaction package: According to the law of large numbers and voting statistics, we assume each voting node has a probability of p to approve a transaction package and a probability of $q = 1 - p$ to disapprove a transaction package. In this case, a larger θ is associated with a smaller p, indicating that the network becomes more unstable and the probability of a node voting to approve decreases.

(6) The voting result judgment: We denote by $N(t)$, $M(t)$, and $K(t)$ the number of voting nodes that approve the transaction package, the number of voting nodes that disapprove the transaction package, and the number of failed nodes at time $t > 0$, respectively. Based on this assumption, we give the following conditions to judge the voting result of a round of voting process:

(a) If $N(t) = 2n + 1$, then the number of approval votes exceeds 2/3 of the total number of voting nodes, the transaction package can be determined as a block and pegged on the blockchain;

TABLE I
MODEL NOTATION FOR PBFT WITH REPAIRABLE VOTING NODES

Symbol	Description
	Random Variables
θ	Failure rate of a voting node
μ	Repair rate of a faulty node
γ	Voting rate of each voting node
β	Pegging rate of a block or rolling back rate of an orphan block
λ	Arrival rate of external transactions
$N(t)$	The number of voting nodes that approve the transaction package at time $t > 0$
$M(t)$	The number of voting nodes that disapprove the transaction package at time $t > 0$
$K(t)$	The number of faulty nodes at time $t > 0$
	Probabilities
p	Probability of a transaction package being approved by a voting node
q	Probability of a transaction package being disapproved by a voting node
	Performance Measures
$E[W_B]$	The average block generation time
$E[W_O]$	The average orphan block generation time
TH	Transaction throughput
A_1 or A_2	Inherent stationary availability
A_3	Operational stationary availability
$R_1(t)$ or $R_2(t)$	Inherent reliability
MTTFF_1 or MTTFF_2	The average time before the first failure

(b) If $M(t) + K(t) = n + 1$, then the number of approval votes is less than 2/3 of the total number of voting nodes, the transaction package can be determined as an orphan block and needs to be rolled back to the transaction pool for further processing.

(7) The arrival of external transactions: We assume that the external transactions arrive at the transaction pool according to a Poisson process with an arrival rate $\lambda > 0$.

(8) The timing of pegging a block or rolling back an orphan block: Note that the processes of pegging a block and rolling back an orphan block are both performed in the PBFT-based blockchain network. These two processes reflect the latency and communication abilities among voting nodes in the network. To measure these factors, we regard the block-pegging time and the orphan block rolling-back time as identical, and we assume that both the block-pegging time and the orphan block rolling-back time follow an exponential distribution with a mean of $1/\beta$.

(9) Independence: We assume that all random variables defined above are independent of each other.

Table I summarizes the relevant random variables and probabilities used in our model. Additionally, we provide a summary of the symbols used for the performance measures discussed in this paper to help readers better read and understand the full text.

Remark 1: Based on Assumption (6), there are three cases in which a transaction package becomes an orphan block: **(1)** $M(t) = n + 1$, $K(t) = 0$; **(2)** $M(t) = 0$, $K(t) = n + 1$; **(3)** $M(t) + K(t) = n + 1$ for $M(t) \geq 1$ and $K(t) \geq 1$.

Remark 2: In the PBFT voting process, once a transaction package becomes a block (i.e., $N(t) = 2n + 1$), there is no need to further perform any subsequent process where $N(t) > 2n + 1$. Similarly, once a transaction package becomes an orphan block (i.e., $M(t) + K(t) = n + 1$), there is no need to further perform any subsequent process where $M(t)+K(t) > n + 1$.

III. DISTRIBUTIONS OF BLOCK GENERATION TIME AND ORPHAN BLOCK GENERATION TIME

In this section, we present the probability distributions of block generation time and orphan block generation time by means of the phase-type distributions of finite sizes.

Note that $N(t)$, $M(t)$, and $K(t)$ denote the number of voting nodes that approve the transaction package, the number of voting nodes that disapprove the transaction package, and the number of failed nodes at time $t \geq 0$, respectively. It is easy to see that $\{(N(t), M(t), K(t)) : t \geq 0\}$ is a three-dimensional continuous-time Markov process whose state space is given by

$$\Omega = \bigcup_{k=0}^{2n+1} \text{Level } k,$$

where for $0 \leq k \leq 2n$,

$$\begin{aligned}\text{Level } k = \{&(k,0,0),(k,0,1),\ldots,(k,0,n),(k,0,n+1);\\ &(k,1,0),(k,1,1),\ldots,(k,1,n-1),(k,1,n);\\ &(k,2,0),(k,2,1),\ldots,(k,2,n-1);\ldots;\\ &(k,n+1,0)\},\end{aligned}$$

and for $k = 2n + 1$,

$$\begin{aligned}\text{Level } k = \{&(k,0,0),(k,0,1),\ldots,(k,0,n-1),(k,0,n);\\ &(k,1,0),(k,1,1),\ldots,(k,1,n-2),(k,1,n-1);\\ &(k,2,0),(k,2,1),\ldots,(k,2,n-2);\ldots;\\ &(k,n,0)\}.\end{aligned}$$

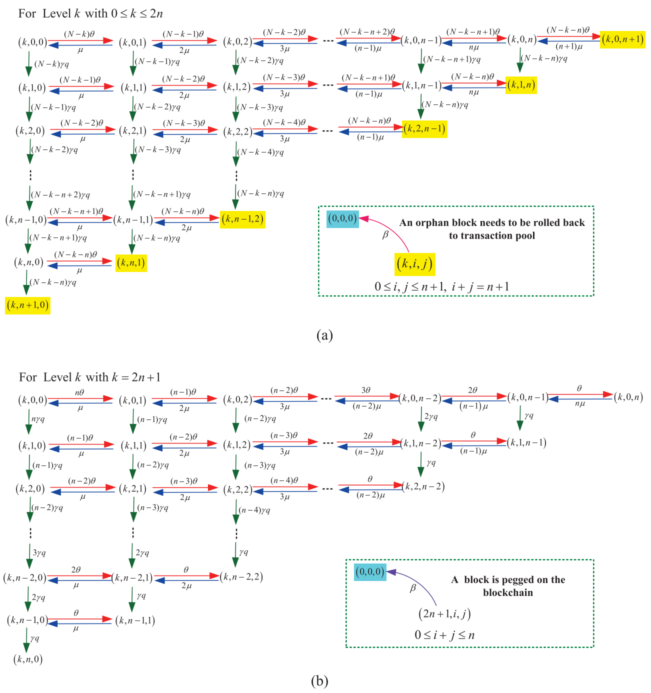
At the same time, the state transition relations of the Markov process $\{(N(t), M(t), K(t)) : t \geq 0\}$ are depicted in Figures 1 and 2.

In Fig. 1(a), a yellow state (k, i, j) with $i+j = n+1$ denotes that the transaction package is determined as an orphan block, and it needs to be rolled back to the transaction pool. In Fig. 1(b), each state in Level $(2n + 1)$ is determined as a block, and it can be pegged on the blockchain. Once a transaction package is determined as either a block or an orphan block, this round of the voting process is over. Then, the blockchain system enters the state (0, 0, 0) such that a new round of the voting process begins.

Using the Markov process $\{(N(t), M(t), K(t)) : t \geq 0\}$, it is easy to see that the transaction package becomes a block at each of states $(2n + 1, i, j)$ for $0 \leq i, j \leq n$; while the transaction package becomes an orphan block at states (k, i, j) with $0 \leq k \leq 2n$, $0 \leq i, j \leq n + 1$, $i + \ j = n + 1$. Also, the events that determine whether a transaction package becomes a block or an orphan block are mutually exclusive. Therefore, we can analyze the probability distribution of a transaction package being determined as a block or an orphan block, respectively.

A. Distribution of Block Generation Time

To analyze the distribution of block generation time, we set up a new Markov process with an absorption state Δ_1. To do


For Level k with $0 \le k \le 2n$
(k,0,0)
(k,0,1)
(k,0,2)
(k,0,n−1)
(k,0,n)
(k,0,n+1)
(k,1,n)
(k,2,n−1)
(k,n−1,2)
(k,n,1)
(k,n+1,0)
(0,0,0)
β
(k,i,j)
An orphan block needs to be rolled back to transaction pool
0 ≤ i, j ≤ n+1, i+j = n+1
For Level k with $k = 2n+1$
A block is pegged on the blockchain
(2n+1,i,j)
0 ≤ i+j ≤ n

(a)

(b)

Fig. 1. The state transition relations of the Markov process within one level.

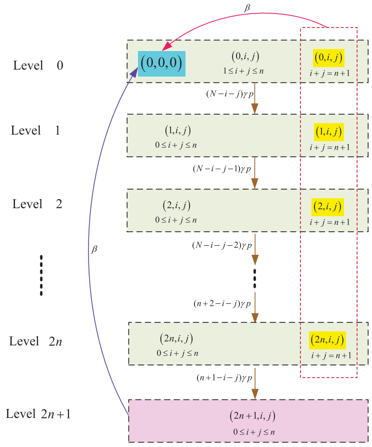


Fig. 2. The state transition relations of the Markov process among multiple levels.

this, all the states in the set $\{(2n+1, i, j) : 0 \le i, j \le n\}$ are regarded as an absorption state Δ_1. In this case, the Markov process $\{(N(t), M(t), K(t)) : t \ge 0\}$ operates on a new state space with the absorption state Δ_1 as follows:

$$\{\Delta_1\} \cup \{(k, i, j) : k = 0, 1, 2, \ldots, 2n, 0 \le i + j \le n\}.$$

At the same time, Figures 3 and 4 depict the state transition relations of the Markov process $\{(N(t), M(t), K(t)) : t \ge 0\}$ with the absorption state Δ_1. Additionally, we denote Ψ as the infinitesimal generator of the Markov process $\{(N(t), M(t), K(t)) : t \ge 0\}$ with the absorption state Δ_1, where,

$$\Psi = \begin{pmatrix} 0 & 0 \\ T^0 & T \end{pmatrix},$$

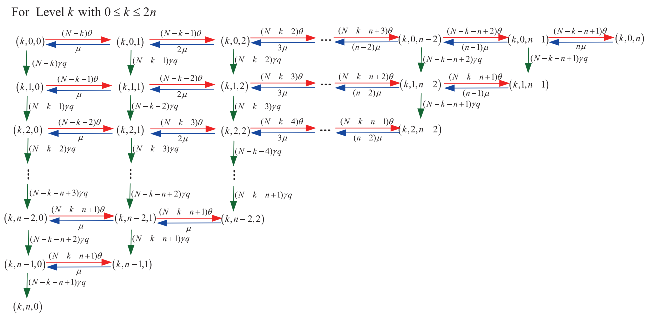


Fig. 3. The state transition relations of the Markov process Ψ: Part one.

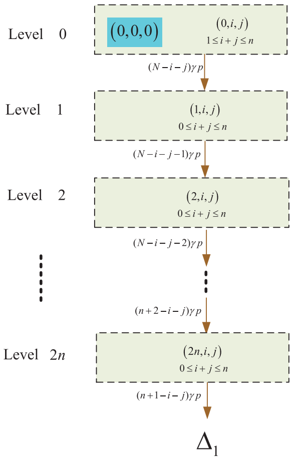


Fig. 4. The state transition relations of the Markov process Ψ: Part two.

$T^0 + T\mathbf{e} = 0$. For the non-zero matrix elements of the infinitesimal generator Ψ, readers may refer to Part A of the Appendix.

In the PBFT-based blockchain system with repairable voting nodes, we denote W_B as the block generation time, which refers to the interval between when the primary node proposes a new transaction package to vote on and the determination moment when a transaction package becomes a block. During this interval, the transaction package obtains enough approval votes. As a result, this block can be pegged to the blockchain.

Let (α_0, α) denote the initial probability distribution of the Markov process Ψ with absorption state Δ_1 at time $t = 0$ for $\alpha_0 = 0$. The vector $\alpha = (1, 0, \ldots, 0)$ shows that the Markov process T is at the state (0,0,0) at time 0. The following theorem provides an expression for the probability distribution of the block generation time W_B by means of the first passage times and the phase-type distributions of finite sizes.

Theorem 1: If the initial probability distribution of the Markov process Ψ with absorption state Δ_1 is (α_0, α) for $\alpha_0 = 0$, then the probability distribution of the block generation time W_B is of phase-type with an irreducible matrix representation (α, T) of finite sizes, and

$$F_{W_B}(t) = P\{W_B \le t\} = 1 - \alpha \exp\{Tt\}e, \quad t \ge 0.$$

Also, the average block generation time is given by

$$E[W_B] = -\alpha T^{-1} e,$$

where T^{-1} is the inverse of the matrix T of finite sizes.

Proof: For $k \in \{0,1,2,\ldots,2n\}, i \in \{0,1,2,\ldots,n\}, j \in \{0,1,2,\ldots,n\}, 0 \le i+j \le n$, we write

$$q_{k,i,j}(t) = P\{N(t)=k, J_1(t)=i, J_2(t)=j\},$$

which is the state probability that the Markov process Ψ with absorption state Δ_1 is at state (*k*, *i*, *j*) at time $t \ge 0$ before absorbed to state Δ_1,

$$q_k(t) = \big\{q_{k,0,0}(t),\ldots,q_{k,0,n}(t); q_{k,1,0}(t),\ldots,q_{k,1,n-1}(t); \ldots; q_{k,n-1,0}(t), q_{k,n-1,1}(t); q_{k,n,0}(t)\big\}$$

and

$$q(t) = \{q_0(t), q_1(t), q_2(t), \ldots, q_{2n-1}(t), q_{2n}(t)\}.$$

Using the Chapman-Kolmogorov forward differential equation, we can obtain

$$\frac{d}{dt} q(t) = q(t)T, \qquad (1)$$

with the initial condition

$$q(0) = \alpha. \qquad (2)$$

It follows from equations (1) and (2) together with $\alpha_0 = 0$ that

$$q(t) = \alpha \exp\{Tt\}. \qquad (3)$$

Thus we obtain

$$P\{W_B > t\} = q(t)e = \alpha \exp\{Tt\}e.$$

This gives

$$\begin{aligned} F_{W_B}(t) &= P\{W_B \le t\} = 1 - P\{W_B > t\} \\ &= 1 - \alpha \exp\{Tt\}e, \quad t \ge 0. \end{aligned}$$

In what follows, we compute the average block generation time $E[W_B]$. Let *f*(*s*) be the Laplace-Stieltjes transform of the distribution function $F_{W_B}(t)$, then

$$f(s) = \int_0^\infty e^{-st} \mathrm{d}F_{W_B}(t) = 1 + \alpha(sI - T)^{-1}T^0, \text{for } s \ge 0,$$

where *I* denotes an identity matrix of finite size. Hence we obtain that

$$E[W_B] = -\frac{\mathrm{d}}{\mathrm{d}s} f(s)_{|s=0} = \alpha\Big[(sI-T)^{-2}\Big]_{|s=0} T^0 = -\alpha T^{-1}e$$

by using $T^0 + T\mathbf{e} = 0$ and $T^{-1}T\mathbf{e} = \mathbf{e}$. This completes the proof. ■

It is worth mentioning that we need to use the RG-factorizations of the Markov process *T* to compute the inverse matrix T^{-1} of finite size. To achieve this, we write

$$T^{-1} = \begin{pmatrix} J_{0,0} & J_{0,1} & J_{0,2} & \cdots & J_{0,2n} \\ & J_{1,1} & J_{1,2} & \cdots & J_{1,2n} \\ & & J_{2,2} & \cdots & J_{2,2n} \\ & & & \ddots & \vdots \\ & & & & J_{2n,2n} \end{pmatrix},$$

By using $T^{-1}T = I$, we can obtain that for $k = 0,1,\ldots,2n$,

$$J_{k,k} = T_{k,k}^{-1},$$

and for $k = 0,1,\ldots,2n-1$, $j = 1,2,\ldots,2n-k$,

$$J_{k,k+j} = (-1)^j \Big(T_{k,k}^{-1} T_{k,k+1} T_{k+1,k+1}^{-1} T_{k,k+2} \cdots T_{k+j-1,k+j} T_{k+j,k+j}^{-1}\Big).$$

Therefore, before computing T^{-1}, it is a key to compute the inverse matrices of diagonal block elements $T_{k,k}$ for $k = 0,1,\ldots,2n$. Note that $K_{i,i}^{(k)} \neq \mathbf{0}$, then the upper triangular matrix $T_{k,k}$ is invertible, and there exists a unique inverse matrix. Thus, we write

$$T_{k,k}^{-1} = \begin{pmatrix} X_{0,0}^{(k)} & X_{0,1}^{(k)} & X_{0,2}^{(k)} & \cdots & X_{0,n}^{(k)} \\ & X_{1,1}^{(k)} & X_{1,2}^{(k)} & \cdots & X_{1,n}^{(k)} \\ & & X_{2,2}^{(k)} & \cdots & X_{2,n}^{(k)} \\ & & & \ddots & \vdots \\ & & & & X_{n,n}^{(k)} \end{pmatrix}, \ 0 \le k \le 2n.$$

By using $T_{k,k}T_{k,k}^{-1} = I$, we can obtain that for $i = 0,1,\ldots,n$,

$$X_{i,i}^{(k)} = \Big(K_{i,i}^{(k)}\Big)^{-1},$$

and for $i = 0,1,\ldots,n-1$, $j = 1,2,\ldots,n-i$,

$$X_{i,i+j}^{(k)} = (-1)^j \left(\Big(K_{i,i}^{(k)}\Big)^{-1} K_{i,i+1}^{(k)} \Big(K_{i+1,i+1}^{(k)}\Big)^{-1} K_{i+1,i+2}^{(k)}, \ldots K_{i+j-1,i+j}^{(k)} \Big(K_{i+j,i+j}^{(k)}\Big)^{-1}\right).$$

Therefore, before computing $T_{k,k}^{-1}, 0 \le k \le 2n$, it is a key to deal with the inverse matrices of diagonal block elements $K_{i,i}^{(k)}$ with the order $n+1-i$ for $i = 0,1,\ldots,n$. Since $K_{i,i}^{(k)}$ is a birth and death process with finite states, we can use RG-factorizations to compute its inverse matrix. Readers may refer to [60, Ch. 1] for more details.

B. Distribution of Orphan Block Generation Time

To analyze the distribution of orphan block generation time, we set up a new Markov process with an absorption state Δ_2, $\Delta_2 = \{\tilde{\Delta}_k, 0 \le k \le 2n\}$, and

$$\tilde{\Delta}_k = \{(k,i,j) : 0 \le i,j \le n+1, i+j = n+1\}.$$

Then the Markov process { (*N*(*t*), *M*(*t*), *K*(*t*)) : $t \ge 0$} operates on a new state space

$$\{\Delta_2\} \cup \{(k,i,j) : 0 \le k \le 2n, 0 \le i+j \le n\}.$$

At the same time, Figures 5 and 6 depict the state transition relations of the Markov process { (*N*(*t*), *M*(*t*), *K*(*t*)) : $t \ge 0$} with the absorption state Δ_2. Additionally, we denote Θ as the infinitesimal generator of the Markov process { (*N*(*t*), *M*(*t*), *K*(*t*)) : $t \ge 0$} with the absorption state Δ_2, where,

$$\Theta = \begin{pmatrix} 0 & 0 \\ S^0 & S \end{pmatrix},$$

$S^0 + S\mathbf{e} = 0$. For the non-zero matrix elements of Θ, readers may refer to Part B of the Appendix.

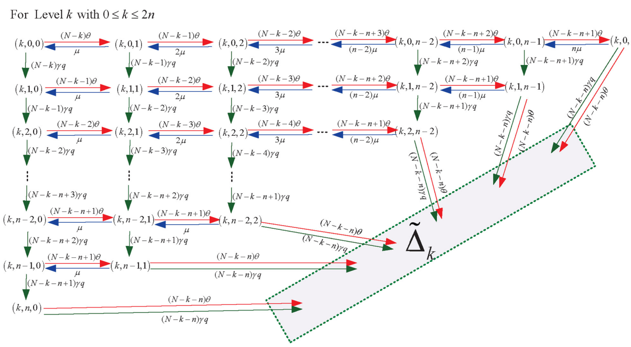


Fig. 5. The state transition relations of the Markov process Θ: Part one.

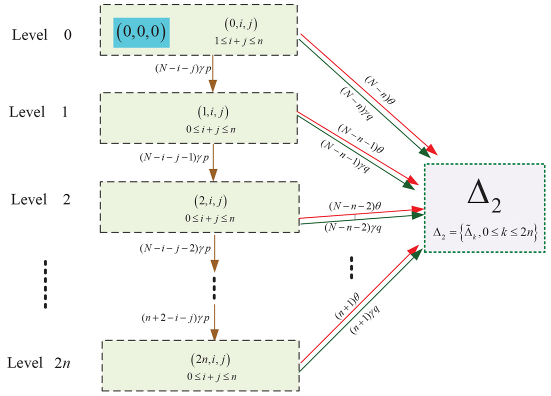


Fig. 6. The state transition relations of the Markov process Θ: Part Two.

In a PBFT-based blockchain system with repairable voting nodes, we denote W_O as the orphan block generation time, which refers to the interval between when the primary node proposes a new transaction package to vote on and the determination moment when a transaction package becomes an orphan block. During this interval, the transaction package cannot obtain sufficient approval votes. As a result, this orphan block must be rolled back to the transaction pool and wait for further processing. Let (ω_0, ω) denote the initial probability distribution of the Markov process Θ with absorption state Δ_2 at time $t = 0$ for $\omega_0 = 0$. The vector $\omega = (1, 0, \ldots, 0)$ shows that the Markov process S is at the state (0, 0, 0) at time 0.

The following theorem provides an expression for the probability distribution of the orphan block generation time W_O by means of the first passage times and the phase-type distributions of finite sizes. Here, we state it in the form of a theorem without proof since it is similar to that given in Theorem 1.

Theorem 2: If the initial probability distribution of the Markov process Θ with absorption state Δ_2 is (ω_0, ω) for $\omega_0 = 0$, then the probability distribution of the orphan block generation time W_O is of phase-type with an irreducible matrix representation (ω, S) of finite sizes, and

$$F_{W_O}(t) = P\{W_O \le t\} = 1 - \omega \exp\{St\}e, \quad t \ge 0.$$

Also, the average orphan block generation time is given by

$$E[W_O] = -\omega S^{-1}e.$$

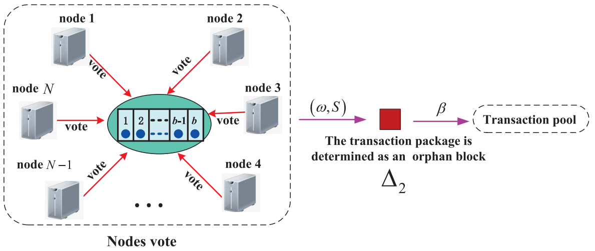


Fig. 7. The process that the orphan block is rolled back to the transaction pool.

IV. Queueing Analysis for the PBFT-Based Blockchain System

In this section, we set up an $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ queue to study the PBFT-based blockchain system with repairable voting nodes. Based on this, we first give the stationary probability vector of the queueing model and then provide performance analysis for the PBFT-based blockchain system with repairable voting nodes.

A. An $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ Queue

Starting from the generation processes of a block and an orphan block, as described in Section III, we provide the model description of the $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ queue as follows:

(1) Transaction arrivals at the transaction pool: In the PBFT-based blockchain system with repairable voting nodes, the arrival process of transactions contains two parts:

(a) The external transactions arrive at the transaction pool: We assume that the external transactions arrive at the transaction pool according to a Poisson process with an arrival rate $\lambda > 0$.

(b) The orphan blocks are rolled back to the transaction pool: We subdivide the rollback of every orphan block into two stages: The first stage is to determine the transaction package as an orphan block by the voting nodes; The second stage is to roll back the orphan block to the transaction pool through the network propagation. As seen from Section III-B, the orphan block generation time W_O follows a PH distribution with irreducible matrix representation (ω, S) of finite sizes, where the size of the orphan block is b. In addition, we assume that the rollback time of the orphan block follows an exponential distribution with a rollback rate β. See Fig. 7 for more details.

Combining the above Assumptions **(a)** with **(b)**, it is seen that the total transaction arrivals at the PBFT-based blockchain system with repairable voting nodes are two processes: One is of phase type with irreducible matrix representation (ω, S), another is Poisson with arrival rate λ.

(2) The service times: In the PBFT-based blockchain system with repairable voting nodes, we consider the block generation process and block-pegged process as a two-stage service process. In other words, the service process consists of two stages: The first stage is that the PBFT-based blockchain system with repairable voting nodes randomly selects b transactions from the transaction pool with equal probability to form a transaction package, and then this transaction package can be successfully determined as a block by the voting nodes;

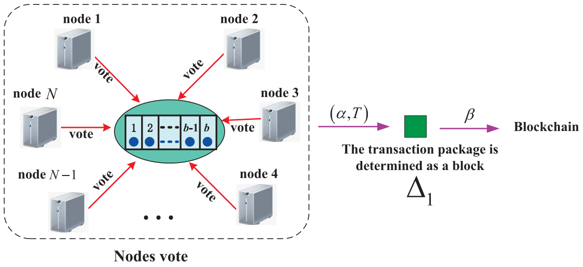


Fig. 8. Two different stages that the block enters the blockchain.

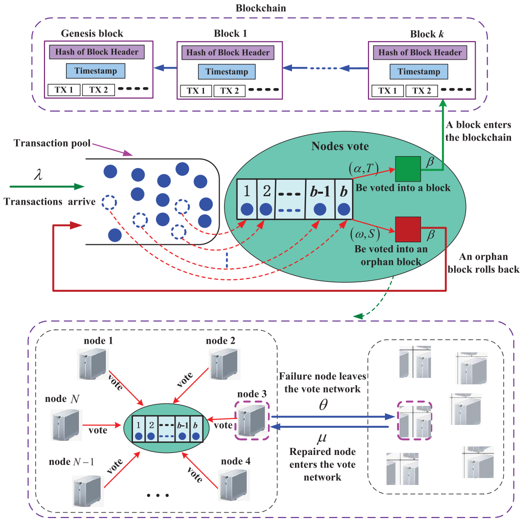


Fig. 9. The $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ queue.

the second stage is that this block is pegged on the blockchain through the network propagation. Referring to Section III-A, the block generation time W_B of every transaction package at the PBFT-based blockchain system follows a PH distribution with irreducible matrix representation (α, T) of finite sizes, where the size of the orphan block is b. Also, we assume that the block-pegged time in the network follows an exponential distribution with a block-pegged rate β. See Fig. 8 for more details.

(3) Independence: We assume that all random variables defined above are independent of each other.

From the above model assumptions, it is easy to see that the PBFT-based blockchain system with repairable nodes can be described as an $\mathrm{M}\oplus\mathrm{PH}^b/\mathrm{PH}^b/1$ queue, which is depicted in Fig. 9. To compute easily, we need to express the service time. Note that a block is pegged to the blockchain through two-stage processes: a PH distribution and an exponential distribution, the total time of these two is written as a Markov process whose infinitesimal generator is given by:

$$\begin{pmatrix} T & T^0 & 0 \\ 0 & -\beta & \beta \\ 0 & 0 & 0 \end{pmatrix},$$

where the final state is an absorbing state.

Let

$$\tilde{T} = \begin{pmatrix} T & T^0 \\ 0 & -\beta \end{pmatrix}, \quad \tilde{T}^0 = \begin{pmatrix} 0 \\ \beta \end{pmatrix},$$
$$\tilde{\alpha} = (\alpha, 0).$$

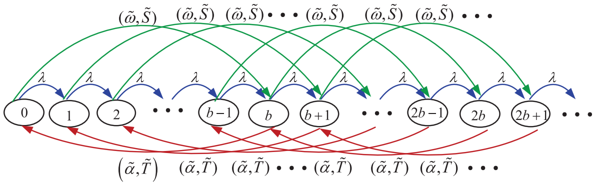


Fig. 10. The state transition relations of the Markov process $\{(I(t), C(t), D(t)) : t \geq 0\}$.

It is clear that the time length that the block is generated and pegged to the blockchain follows a continuous-time PH distribution with irreducible matrix representation $(\tilde{\alpha}, \tilde{T})$.

Similarly, the time length that the orphan block is generated and rolled back to the transaction pool follows a continuous-time PH distribution with irreducible matrix representation $(\tilde{\omega}, \tilde{S})$, where,

$$\tilde{S} = \begin{pmatrix} S & S^0 \\ 0 & -\beta \end{pmatrix}, \quad \tilde{S}^0 = \begin{pmatrix} 0 \\ \beta \end{pmatrix},$$
$$\tilde{\omega} = (\omega, 0).$$

B. Analysis of the $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ *Queue*

Let $I(t)$ be the number of transactions in the transactions pool at time t. Then

$$I(t) \in \{0, 1, 2, \ldots, b-1, b, b+1, b+2, \ldots\}.$$

We denote by $C(t)$ and $D(t)$ the phases of the block generation time and the orphan-block generation time at time t, respectively. It is clear that $\{(I(t), C(t), D(t)) : t \geq 0\}$ is a continuous-time Markov process.

Based on Fig. 10, the infinitesimal generator Υ of the Markov process $\{(I(t), C(t), D(t)) : t \geq 0\}$ is given by

$$\Upsilon = \begin{pmatrix} B_1^{(0)} & A_0^{(0)} & & & \\ A_2^{(1)} & A_1 & A_0 & & \\ & A_2 & A_1 & A_0 & \\ & & \ddots & \ddots & \ddots \end{pmatrix},$$

where

$$B_1^{(0)} = \begin{pmatrix} \tilde{S} - \lambda I & \lambda I & & \\ & \ddots & \ddots & \\ & & \tilde{S} - \lambda I & \lambda I \\ & & & \tilde{S} - \lambda I \end{pmatrix},$$

$$A_0^{(0)} = \begin{pmatrix} \left(\tilde{S}^0\tilde{\omega}\right) \otimes \tilde{\alpha} & & & \\ & \left(\tilde{S}^0\tilde{\omega}\right) \otimes \tilde{\alpha} & & \\ & & \ddots & \\ \lambda(I \otimes \tilde{\alpha}) & & & \left(\tilde{S}^0\tilde{\omega}\right) \otimes \tilde{\alpha} \end{pmatrix},$$

$$A_2^{(1)} = \begin{pmatrix} I \otimes \tilde{T}^0 & & & \\ & I \otimes \tilde{T}^0 & & \\ & & \ddots & \\ & & & I \otimes \tilde{T}^0 \end{pmatrix},$$

$$A_2 = \begin{pmatrix} I \otimes \left(\tilde{T}^0 \tilde{\alpha}\right) & & & \\ & I \otimes \left(\tilde{T}^0 \tilde{\alpha}\right) & & \\ & & \ddots & \\ & & & I \otimes \left(\tilde{T}^0 \tilde{\alpha}\right) \end{pmatrix},$$

$$A_1 = \begin{pmatrix} \tilde{S} \oplus \tilde{T} - \lambda I & \lambda I & & \\ & \ddots & \ddots & \\ & & \tilde{S} \oplus \tilde{T} - \lambda I & \lambda I \\ & & & \tilde{S} \oplus \tilde{T} - \lambda I \end{pmatrix},$$

$$A_0 = \begin{pmatrix} \left(\tilde{S}^0 \tilde{\omega}\right) \otimes I & & & \\ & \left(\tilde{S}^0 \tilde{\omega}\right) \otimes I & & \\ & & \ddots & \\ \lambda I & & & \left(\tilde{S}^0 \tilde{\omega}\right) \otimes I \end{pmatrix},$$

and I is the identity matrix of appropriate dimensions.

The continuous-time Markov process Υ is a level-independent QBD process. Thus, we can apply the matrix-geometric solution given in Neuts [59] to analyze the QBD process Υ. As stated in the following theorem, we give the condition that the Markov process Υ is positive recurrent. Based on this, we can further analyze the performance of the PBFT-based blockchain system with repairable voting nodes.

Theorem 3: The level-independent QBD Υ is positive recurrent if and only if

$$\lambda + b\delta_1 \tilde{S}^0 < b\delta_2 \tilde{T}^0,$$

where δ_1 and δ_2 are the stationary probability vectors of the two Markov processes $\tilde{T} + \tilde{T}^0\tilde{\alpha}$ and $\tilde{S} + \tilde{S}^0\tilde{\omega}$, respectively.

Proof: Let $W = (\tilde{S} + \tilde{S}^0\tilde{\omega}) \oplus (\tilde{T} + \tilde{T}^0\tilde{\alpha})$. Since δ_1 satisfies $\delta_1(\tilde{T} + \tilde{T}^0\tilde{\alpha}) = 0$, $\delta_1 e = 1$; and δ_2 satisfies $\delta_2(\tilde{S} + \tilde{S}^0\tilde{\omega}) = 0$, $\delta_2 e = 1$. It is easy to see that $(\delta_1 \otimes \delta_2)[(\tilde{S} + \tilde{S}^0\tilde{\omega}) \oplus (\tilde{T} + \tilde{T}^0\tilde{\alpha})] = 0$.

For the continuous-time QBD process Υ, we use the mean-drift method to provide its stability condition. Readers may refer to [59, Ch. 1] or [60, Ch. 3] for more details. We write

$$\begin{aligned} A &= A_2 + A_1 + A_0 \\ &= \begin{pmatrix} W - \lambda I & \lambda I & & \\ & \ddots & \ddots & \\ & & W - \lambda I & \lambda I \\ \lambda I & & & W - \lambda I \end{pmatrix}. \end{aligned}$$

Clearly, the Markov process A is irreducible, aperiodic and positive recurrent. Let $\upsilon = (\upsilon_1, \upsilon_2, \ldots, \upsilon_b)$ be the stationary probability vector of Markov process A, where $\upsilon_k = \gamma_k(\delta_1 \otimes \delta_2), k = 1, 2, \ldots, b$. Then from the system of linear equations: $\upsilon A = 0$ and $\upsilon e = 1$, we easily get that $\gamma_1 = \gamma_2 = \cdots \gamma_b = 1/b$, thus, we obtain $\upsilon_k = (\delta_1 \otimes \delta_2)/b, k = 1, 2, \ldots, b$.

Using the mean-drift method, it is easy to check that the QBD process Υ is positive recurrent if and only if $\upsilon A_0 e < \upsilon A_2 e$, i.e.,

$$\lambda + b\delta_1 \tilde{S}^0 < b\delta_2 \tilde{T}^0.$$

This completes the proof. ■

When the QBD process Υ is positive recurrent, we write its stationary probability vector as $\psi = (\psi_0, \psi_1, \psi_2, \ldots)$, where

$$\psi_k = \left(\tilde{\psi}_{kb}, \tilde{\psi}_{kb+1}, \ldots, \tilde{\psi}_{(k+1)b-1}\right), k \geq 0.$$

Note that such a stationary probability vector ψ generally has no explicit expression, and we can only develop its numerical solution. To this end, according to [59, Ch. 3], we need to numerically compute the rate matrix R, which is the minimal nonnegative solution to the nonlinear matrix equation $R^2A_2 + RA_1 + A_0 = 0$. Based on this, we give an iterative algorithm (see [59, Ch. 3]) to numerically compute the rate matrix R as follows:

$$\begin{aligned} R_0 &= 0, \\ R_{n+1} &= \left(R_n^2 A_2 + A_0\right)\left(-A_1^{-1}\right), n = 1, 2, 3, \ldots. \end{aligned} \tag{4}$$

For the matrix sequence $\{R_n, n \geq 0\}$, using [59, Ch. 3], it is easy to see that as $n \to \infty$, $R_n \uparrow R$. Thus, for any sufficiently small positive number ε, there exists a positive integer $\mathbf{n}$ such that $\|R_{\mathbf{n}+1} - R_{\mathbf{n}}\| < \varepsilon$. In this case, we take $R \approx R_{\mathbf{n}}$, which gives an approximate solution to the nonlinear matrix equation $R^2A_2 + RA_1 + A_0 = 0$.

The following theorem provides an expression for the stationary probability vector ψ using the rate matrix R. This conclusion is directly derived from [59, Ch. 1, Th. 1.2.1]. Here, we restate it without providing proof.

Theorem 4: If the QBD process Υ is positive recurrent, then its stationary probability vector $\psi = (\psi_0, \psi_1, \psi_2, \ldots)$ is given by

$$\psi_k = \psi_1 R^{k-1}, k \geq 1, \tag{5}$$

where ψ_0 and ψ_1 are the unique solution to the following system of linear equations:

$$\begin{cases} \psi_0 B_1^{(0)} + \psi_1 A_2^{(1)} = 0, \\ \psi_0 A_0^{(0)} + \psi_1 (A_1 + RA_2) = 0, \\ \psi_0 e + \psi_1 (I - R)^{-1} e = 1. \end{cases} \tag{6}$$

C. Performance Measures of the PBFT System

Based on the established $\mathrm{M} \oplus \mathrm{PH}^b/\mathrm{PH}^b/1$ queue and the obtained stationary probability vector ψ, we can provide some key performance measures of the PBFT-based blockchain system with repairable voting nodes as follows:

(a1) The stationary probability of having no transaction package in the PBFT-based blockchain system is given by

$$\eta_1 = \psi_0 e.$$

(a2) The stationary probability of having a transaction package in the PBFT-based blockchain system is given by

$$\eta_2 = 1 - \eta_1 = 1 - \psi_0 e.$$

(b1) The stationary rate of pegging a block on the blockchain in the PBFT-based blockchain system is given by

$$r_1 = (1 - \psi_0 e)\frac{1}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)} = \eta_2 \frac{1}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)}.$$

(b2) The stationary rate of rolling back an orphan block to the transaction pool is given by

$$r_2 = (1 - \psi_0 e)\frac{1}{\left(-\tilde{\omega}\tilde{S}^{-1}e\right)} = \eta_2\frac{1}{\left(-\tilde{\omega}\tilde{S}^{-1}e\right)}.$$

(c) The block throughput of the PBFT-based blockchain system is given by

$$\mathrm{TH(block)} = (1 - \psi_0 e)\frac{1}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)} = \eta_2\frac{1}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)}.$$

In this paper, we define block throughput as the number of blocks per second and transaction throughput as the number of transactions per second. The latter is a general definition of throughput, and it is our focus. In what follows, we provide an effective method to compute the transaction throughput of the PBFT-based blockchain system with repairable voting nodes.

Theorem 5: The transaction throughput of the PBFT-based blockchain system is given by

$$\mathrm{TH} = (1 - \psi_0 e)\frac{b}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)} = \eta_2\frac{b}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)}.$$

Proof: From Section IV-A, we can see that the time length that a block is generated and pegged to the blockchain follows a PH distribution with irreducible matrix representation $(\tilde{\alpha}, \tilde{T})$. This finding shows that the average time that the PBFT-based blockchain system pegs a block with b transactions on the blockchain is $(-\tilde{\alpha}\tilde{T}^{-1}e)$. Therefore, the number of transactions dealt with by the PBFT-based blockchain system per unit time is $b/(-\tilde{\alpha}\tilde{T}^{-1}e)$.

In addition, the stationary probability of having a transaction package in the PBFT-based blockchain system is given by η_2. Thus, the transaction throughput of the PBFT-based blockchain system is the product of the stationary probability of having a transaction package in the PBFT-based blockchain system and the number of transactions dealt with per unit time in the PBFT-based blockchain system. In terms of means, we have

$$\mathrm{TH} = (1 - \psi_0 e)\frac{b}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)} = \eta_2\frac{b}{\left(-\tilde{\alpha}\tilde{T}^{-1}e\right)}.$$

This completes the proof. ■

V. Reliability Analysis of the PBFT-Based Blockchain System

In this section, we set up two new Markov processes to analyze the reliability of the PBFT-based blockchain system with repairable voting nodes. Here, we treat the inability of blockchain systems to generate blocks as unavailable. In the following discussion, we will examine two different scenarios.

A. Unavailability Due to Failed Nodes

In this subsection, we consider the case where the PBFT-based blockchain system becomes unavailable due to the number of failed nodes reaching $n+1$. In this case, the PBFT-based blockchain system cannot proceed with the voting process, resulting in the absence of new block generation.

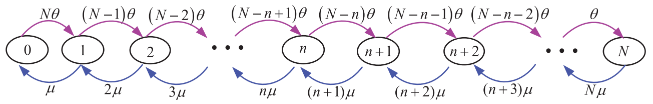


Fig. 11. The state transition relations of the process $\{K(t), t \geq 0\}$.

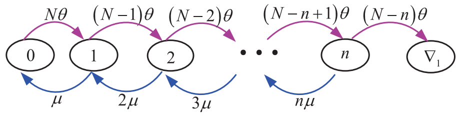


Fig. 12. The state transition relations of the Markov process $\{K(t), t \geq 0\}$ with the absorption state ∇_1.

In the case of a failure rate of θ and a repair rate of μ, let $K(t)$ be the number of failed nodes in the PBFT-based blockchain system at time t. Then, $\{K(t), t \geq 0\}$ is a continuous-time birth-death process with a state space $\Phi = \{0, 1, 2, \ldots, N\}$, and its state transition relations are depicted in Fig. 11.

Based on Fig. 11, we write the infinitesimal generator of the birth-death process $\{K(t), t \geq 0\}$ as Q_A, whose expression is given by equation (7), at the bottom of the next page.

Obviously, the birth-death process $K(t)$ is aperiodic, irreducible, and positive recurrent. Let $\zeta = (\zeta_0, \zeta_1, \ldots, \zeta_N)$ be the stationary probability vector of the birth-death process $\{K(t), t \geq 0\}$. Then from the system of linear equations: $\zeta Q_A = 0$ and $\zeta e = 1$, we have

$$\zeta_k = \frac{(N-k+1)\theta}{k\mu}\zeta_{k-1}, \quad 1 \leq k \leq N,$$

and

$$\zeta_0 = \frac{1}{1 + \sum_{k=1}^{N}\frac{N!}{(N-k)!k!}\rho^k},$$

where $\rho = \theta/\mu$.

Let A_1 be the inherent stationary availability of the PBFT-based blockchain system with repairable voting nodes. Then,

$$A_1 = \zeta_0 + \zeta_1 + \zeta_2 + \cdots + \zeta_n = \frac{1 + \sum_{k=1}^{n}\frac{N!}{(N-k)!k!}\rho^k}{1 + \sum_{k=1}^{N}\frac{N!}{(N-k)!k!}\rho^k}.$$

To analyze the inherent reliability $R_1(t)$ of the PBFT-based blockchain system with repairable voting nodes, we let all the states in the set $\{n+1, n+2, \ldots, N\}$ be absorption state ∇_1. Then, the Markov process $\{K(t), t \geq 0\}$ operates on a new state space $\{0, 1, 2, \ldots, n\} \cup \{\nabla_1\}$, and its state transition relations are depicted in Fig. 12. We write the infinitesimal generator of the Markov process $\{K(t), t \geq 0\}$ with the absorption state ∇_1 as

$$Q_{\nabla_1} = \begin{pmatrix} T_\nabla & T_\nabla^0 \\ 0 & 0 \end{pmatrix},$$

where, $T_\nabla^0 = (0; 0; \ldots; (N-n)\theta)$ and T_∇ is shown in equation (8) at the bottom of the next page.

Let $(\varphi(0), \varphi_\nabla(0))$ be the initial probability distribution of the Markov process Q_{∇_1}, where $\varphi_\nabla(0) = 0$, $\varphi(0) = (1, 0, 0, \ldots, 0)$, $\tau_1 = \inf\{t \geq 0 : \tilde{K}(t) = n+1, \tilde{K}(0) = 0\}$, and $R_1(t) = P\{\tau_1 > t\}$, then following theorem provides an expression for the inherent reliability $R_1(t)$ of the PBFT-based blockchain system.

Theorem 6: If the initial probability distribution of the Markov process Q_{∇_1} is $(\varphi(0), \varphi_{\nabla}(0))$, then the inherent reliability $R_1(t)$ of the PBFT-based blockchain system is given by

$$R_1(t) = \sum_{k=0}^{n} \varphi_k(t),$$

where $\varphi(t) = (\varphi_0(t), \varphi_1(t), \varphi_2(t), \ldots, \varphi_n(t))$ satisfies the Chapman-Kolmogorov forward differential equation $\varphi'(t) = \varphi(t)\, T_{\nabla}$ with the initial condition $\varphi(0) = (1, 0, 0, \ldots, 0)$.

Proof: It is easy to see that $R_1(t) = \varphi(t)e$. Let

$$\varphi^*(s) = \int_0^{\infty} e^{-st}\varphi(t)\,dt, \quad s > 0 \quad i = 0, 1, 2, \ldots, n,$$

be the Laplace transform of $\varphi(t)$, then

$$\int_0^{\infty} e^{-st}\varphi'(t)\,dt = \int_0^{\infty} e^{-st}\varphi(t)\,dt \cdot T_{\nabla}, \quad s > 0,$$

thus, we obtain

$$\varphi^*(s) = \int_0^{\infty} e^{-st}\varphi(t)\,dt = \varphi(0)(sI - T_{\nabla})^{-1}, \quad s > 0.$$

Also, we have

$$R_1^*(s) = \varphi^*(s)e = \varphi(0)(sI - T_{\nabla})^{-1}e, \quad s > 0.$$

Therefore, by inverting $R_1^*(s)$, we have

$$R_1(t) = P\{\tau_1 > t\} = \sum_{k=0}^{n} \varphi_k(t).$$

This completes the proof. ■

Based on Theorem 6, the average time before the first failure of the PBFT-based blockchain system is given by

$$\mathrm{MTTFF}_1 = \int_0^{\infty} R_1(t)\,dt = R_1^*(0) = -\varphi(0)\, T_{\nabla}^{-1} e.$$

B. Unavailability Due to Both Failed Nodes and Disapproval Votes

In this subsection, we consider the case where the PBFT-based blockchain system becomes unavailable due to the sum of failed nodes and disapproval votes reaching $n + 1$. To this end, we use a three-dimensional Markov process to analyze the availability and reliability of the PBFT voting process with repairable voting nodes.

According to Figures 1 and 2, we write the infinitesimal generator of the Markov process $\{(N(t), M(t), K(t)) : t \geq 0\}$ as $\boldsymbol{Q}$, where

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & & & \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & & \\ \vdots & & \ddots & \ddots & \\ Q_{2n,0} & & & Q_{2n,2n} & Q_{2n,2n+1} \\ Q_{2n+1,0} & & & & Q_{2n+1,2n+1} \end{pmatrix}.$$

For the non-zero matrix elements of the infinitesimal generator $\boldsymbol{Q}$, readers may refer to Part C of the Appendix.

The Markov process { (*N*(*t*), *M*(*t*), *K*(*t*)) : $t \geq 0$} is irreducible and contains only a finite number of states, thus it is positive recurrent. By corresponding to its state space Ω, we define the probabilities

$$\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_{2n}, \boldsymbol{\pi}_{2n+1}),$$

where

$$\begin{aligned} \boldsymbol{\pi}_k = \big(&\pi_{k,0,0}, \pi_{k,0,1}, \ldots, \pi_{k,0,n+1}; \pi_{k,1,0}, \pi_{k,1,1}, \ldots, \\ &\pi_{k,1,n}; \ldots; \pi_{k,n,0}, \pi_{k,n,1}; \pi_{k,n+1,0}\big), 0 \leq k \leq 2n, \\ \boldsymbol{\pi}_{2n+1} = \big(&\pi_{2n+1,0,0}, \pi_{2n+1,0,1}, \ldots, \pi_{2n+1,0,n}; \pi_{2n+1,1,0}, \\ &\pi_{2n+1,1,1}, \ldots, \pi_{2n+1,1,n-1}; \ldots; \pi_{2n+1,n,0}\big). \end{aligned}$$

In order to drive an expression for the stationary probability vector $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_{2n}, \boldsymbol{\pi}_{2n+1})$, we need to compute the inverse matrices for the matrices $Q_{k,k}$ in infinitesimal generator $\boldsymbol{Q}$ for $1 \leq k \leq 2n + 1$. To do this, we observe that the structure of $Q_{k,k}$ is the same as that of matrix $T_{i,i}$, the computation of the inverse matrix $Q_{k,k}^{-1}$ by RG-factorizations is similar to the inverse matrix $T_{i,i}^{-1}$. Thus, we omit the details

$$Q_A = \begin{pmatrix} -N\theta & N\theta & & & & \\ \mu & -[\mu + (N-1)\theta] & (N-1)\theta & & & \\ & 2\mu & -[2\mu + (N-2)\theta] & (N-2)\theta & & \\ & & \ddots & \ddots & \ddots & \\ & & & (N-1)\mu & -[(N-1)\mu + \theta] & \theta \\ & & & & N\mu & -N\mu \end{pmatrix}. \quad (7)$$

$$T_{\nabla} = \begin{pmatrix} -N\theta & N\theta & & & \\ \mu & -[\mu + (N-1)\theta] & (N-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-1)\mu & -[(n-1)\mu + (N-n+1)\theta] & (N-n+1)\theta \\ & & & n\mu & -[n\mu + (N-n)\theta] \end{pmatrix} \quad (8)$$

of computing the inverse matrix $Q_{k,k}^{-1}$. Readers may refer to Section III-A.

Let $R_k = Q_{k,k+1}(-Q_{k+1,k+1}^{-1})$ for $0 \le k \le 2n$. Next, we provide an expression for the stationary probability vector $\boldsymbol{\pi}$ using the following theorem.

Theorem 7: The stationary probability vector $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_{2n+1})$ of the Markov process Q is matrix-product, and it is given by

$$\boldsymbol{\pi}_{k+1} = \boldsymbol{\pi}_0 R_0 R_1 \cdots R_k, \quad 0 \le k \le 2n, \tag{9}$$

where $\boldsymbol{\pi}_0$ is uniquely determined by means of solving the following system of linear equations

$$\boldsymbol{\pi}_0 \left(Q_{0,0} + R_0 Q_{1,0} + \cdots + \prod_{i=0}^{2n} R_i Q_{2n+1,0} \right) = 0, \tag{10}$$

with the normalized condition

$$\boldsymbol{\pi}_0 = 1 \Bigg/ \left(e + \sum_{k=0}^{2n} R_0 R_1 \ldots R_k e \right). \tag{11}$$

Proof: From $\boldsymbol{\pi} Q = 0$ and $\boldsymbol{\pi} e = 1$, we can obtain

$$\begin{cases} \boldsymbol{\pi}_k Q_{k,k+1} + \boldsymbol{\pi}_{k+1} Q_{k+1,k+1} = 0, \quad 0 \le k \le 2n, \\ \sum_{k=0}^{2n+1} \boldsymbol{\pi}_k Q_{k,0} = 0, \\ \sum_{k=0}^{2n+1} \boldsymbol{\pi}_k e = 1. \end{cases} \tag{12}$$

Using equation (12), we obtain

$$\boldsymbol{\pi}_1 = \boldsymbol{\pi}_0 Q_{0,1} \left(-Q_{1,1}^{-1} \right) = \boldsymbol{\pi}_0 R_0,$$
$$\boldsymbol{\pi}_2 = \boldsymbol{\pi}_1 Q_{1,2} \left(-Q_{2,2}^{-1} \right) = \boldsymbol{\pi}_1 R_1 = \boldsymbol{\pi}_0 R_0 R_1,$$
$$\vdots$$
$$\boldsymbol{\pi}_{2n+1} = \boldsymbol{\pi}_{2n} Q_{2n,2n+1} \left(-Q_{2n+1,2n+1}^{-1} \right) = \boldsymbol{\pi}_0 R_0 R_1 \cdots R_{2n}.$$

Therefore, we obtain

$$\boldsymbol{\pi}_{k+1} = \boldsymbol{\pi}_0 R_0 R_1 \cdots R_k, \quad 0 \le k \le 2n.$$

Further, using equations (9) and (12), we can obtain the boundary equation (10) and the normalized condition (11). This proof is completed. ■

Using the stationary probability vector $\boldsymbol{\pi}$, we can provide availability analysis of the PBFT-based blockchain system with repairable voting nodes as follows:

(a) The inherent stationary availability of the PBFT-based blockchain system with repairable voting nodes is given by

$$\begin{aligned} A_2 &= 1 - \left(\pi_{0,0,n+1} + \pi_{1,0,n+1} + \cdots + \pi_{2n,0,n+1} \right) \\ &= 1 - \sum_{k=0}^{2n} \pi_{k,0,n+1}. \end{aligned}$$

(b) The operational stationary availability of the PBFT-based blockchain system with repairable voting nodes is given by

$$A_3 = 1 - P_O,$$

where P_O represents the stationary probability that the transaction package becomes an orphan block, and

$$P_O = \sum_{k=0}^{2n} \sum_{i=0}^{n+1} \sum_{j=n+1-i} \pi_{k,i,j}.$$

Finally, we provide expression for the operational reliability $R_2(t)$ of the PBFT-based blockchain system. To this end, we let all the states in the set

$$\{(0,0,n+1), (0,1,n), \ldots, (2n,n+1,0)\}$$

be absorption state ∇_2. Then the Markov process $\{(N(t), M(t), K(t)) : t \ge 0\}$ operates on a new state space $\Omega_\nabla \cup \{\nabla_2\}$, where,

$$\Omega_\nabla = \bigcup_{k=0}^{2n+1} \text{Level } k,$$

for $0 \le k \le 2n+1$,

$$\begin{aligned} \text{Level } k = \{ & (k,0,0), (k,0,1), \ldots, (k,0,n-1), (k,0,n); \\ & (k,1,0), (k,1,1), \ldots, (k,1,n-2), (k,1,n-1); \\ & (k,2,0), (k,2,1), \ldots, (k,2,n-2); \ldots; \\ & (k,n,0) \}. \end{aligned}$$

We write the infinitesimal generator of the Markov process $\{(N(t), M(t), K(t)) : t \ge 0\}$ with the absorption state ∇_2 as

$$Q_{\nabla_2} = \begin{pmatrix} S_\nabla & S_\nabla^0 \\ 0 & 0 \end{pmatrix}.$$

Let $(\phi(0), \phi_\nabla(0))$ be the initial probability distribution of the Markov process Q_{∇_2}, where $\phi(0) = (1,0,0,\ldots,0)$ and $\phi_\nabla(0) = 0$,

$$\begin{aligned} \tau_2 = \inf\{ & t \ge 0 : M(t) + K(t) = n+1, \\ & (N(0), M(0), K(0)) = (0,0,0)\}, \end{aligned}$$

and $R_2(t) = P\{\tau_2 > t\}$, then following theorem provides expression for the operational reliability $R_2(t)$ of the PBFT-based blockchain system.

Theorem 8: If the initial probability distribution of the Markov process Q_{∇_2} is $(\phi(0), \phi_\nabla(0))$, then the operational reliability $R_2(t)$ of the PBFT-based blockchain system is given by

$$R_2(t) = P\{\tau_2 > t\} = \sum_{k=0}^{2n+1} \phi_k(t).$$

where $\phi(t) = (\phi_0(t), \phi_1(t), \phi_2(t), \ldots, \phi_{2n+1}(t))$ satisfies the Chapman-Kolmogorov forward differential equation $\phi'(t) = \phi(t) S_\nabla$ with the initial condition $\phi(0) = (1,0,0,\ldots,0)$.

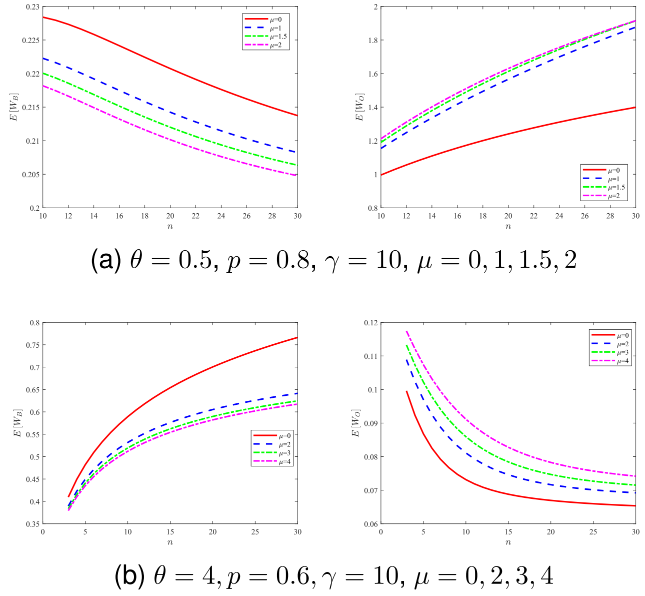
Based on Theorem 8, the average time before the first failure of the PBFT-based blockchain system is

$$\text{MTTFF}_2 = \int_0^\infty R_2(t)\,dt = -\phi(0) S_\nabla^{-1} e.$$

VI. Simulation and Numerical Experiments

In this section, we conducted three groups of experiments to verify the validity of our theoretical results and to show how some key system parameters influence performance measures of the PBFT-based blockchain system with repairable voting nodes.

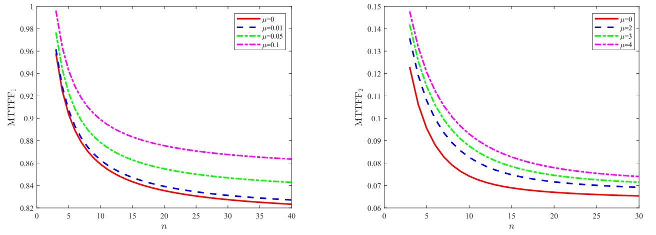
Group One: The impact of introducing the repair process on theoretical performance measures: In this part, we use numerical examples to compare the theoretical



(a) $\theta = 0.5$, $p = 0.8$, $\gamma = 10$, $\mu = 0, 1, 1.5, 2$

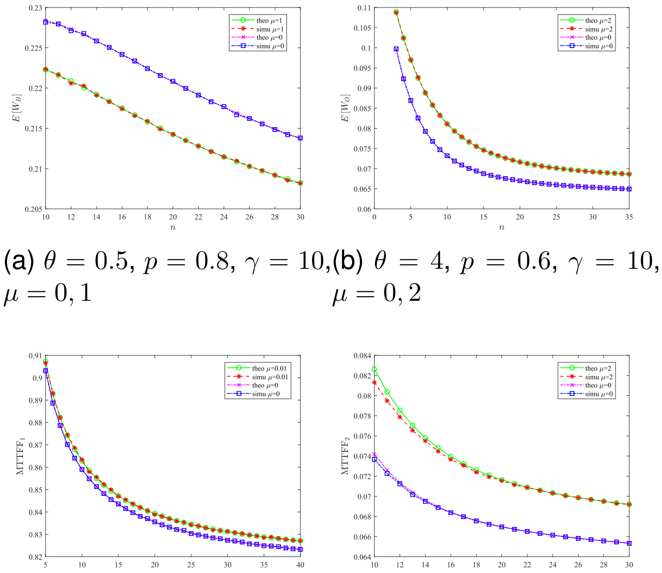
(b) $\theta = 4, p = 0.6, \gamma = 10$, $\mu = 0, 2, 3, 4$

Fig. 13. Theoretical values of $E[W_B]$ and $E[W_O]$ with and without repairable voting nodes.



(a) MTTFF_1 vs. n and μ (b) MTTFF_2 vs. n and μ

Fig. 14. MTTFF_1 and MTTFF_2 vs. n and μ.



(a) $\theta = 0.5$, $p = 0.8$, $\gamma = 10$, $\mu = 0, 1$

(b) $\theta = 4$, $p = 0.6$, $\gamma = 10$, $\mu = 0, 2$

(c) $\theta = 0.5$, $\mu = 0, 0.01$

(d) $\theta = 4, p = 0.6, \gamma = 10$, $\beta = 3$, $\mu = 0, 2$

Fig. 15. Comparison of simulation and theoretical values of four performance measures.

performance measures of PBFT-based blockchain systems with and without repairable voting nodes. Among them, the performance measures without repairable voting nodes obtained by the theory provided in this paper are represented by $\mu = 0$.

For a PBFT-based blockchain system, the block generation time and the orphan block generation time are important factors that affect its performance. Therefore, we first compared the average block generation time $E[W_B]$ and the average orphan block generation time $E[W_O]$ in different cases.

As shown in Fig. 13(a) or Fig. 13(b), under different failure rates θ, the average block generation time $E[W_B]$ is shorter in systems with repairable voting nodes than in systems without them. Conversely, the average orphan block generation time $E[W_O]$ is longer in systems with repairable voting nodes than in systems without them. Additionally, we can observe that in the PBFT-based blockchain system without repairable voting nodes, the average block generation time and the average orphan block generation time have opposite trends as n increases in different network environments. This opposite trend is consistent with a system that has repairable nodes and may have a negative impact on the performance of the system during a round of voting. To validate this effect, we conducted a subsequent experiment on throughput, and the results confirmed the validity of the inference.

Next, we compared the average time before the first failure under different assumption scenarios. For Fig. 14(a), we take $\theta = 0.5$, $n \in [3, 40]$, and $\mu = 0, 0.01, 0.05, 0.1$; For Fig. 14(b), we take $\theta = 4$, $p = 0.6$, $\gamma = 10$, $\beta = 3$, $n \in [3, 30]$, and $\mu = 0, 2, 3, 4$. As shown in Fig. 14(a) or Fig. 14(b), we can observe that systems with repairable voting nodes have a longer average time before the first failure compared to those without, and the average time before the first failure with repairable voting nodes increases as the repair rate increases. These findings suggest that introducing a repair process to the PBFT-based blockchain system can increase the average time before the first failure. More importantly, these results prove once again that the introduction of the repair process is beneficial to the performance of the PBFT-based blockchain system.

Group Two: Comparison and verification of simulation and theory: To validate the method proposed in this paper, we performed Monte Carlo simulation experiments and compared the simulation results with the theoretical results using the same parameter settings.

In Fig. 15, we compared the simulation and theoretical values with and without repaired voting nodes. Each of our simulations, consisting of 10000 rounds, is repeated 100 times to calculate the average values of $E[W_B]$, $E[W_O]$, MTTFF_1, and MTTFF_2. From Fig. 15, we can observe that there is little difference between the theoretical and the simulation values, which proves the correctness of our theory well.

Group Three: Mixed verification of throughput and availability: In this part, we mainly analyzed the TH, A_1, A_2 and A_3 of the PBFT-based blockchain system.

TABLE II
THE ORDERS OF IMPORTANT SUB-MATRICES WITH DIFFERENT b AND n

values of n	$\tilde{S}$ or $\tilde{T}$	$\tilde{S} \oplus \tilde{T} - \lambda I$	A_1
2	31×31	961×961	$961b \times 961b$
3	71×71	5041×5041	$5041b \times 5041b$
4	136×136	18496×18496	$18496b \times 18496b$
5	231×231	53361×53361	$53361b \times 53361b$

```
Algorithm 1: Approximately Computing Transaction Throughput TH
Input: The key parameters: $\mu$, $\theta$, $\gamma$, $\beta$, $p$, $\lambda$, $b$, $n$;
       A controllable accuracy $\varepsilon$
Output: Transaction throughput TH of the PBFT-based
        blockchain system
1 Determine transition blocks: $\tilde{T}$, $\tilde{S}$ and initial probability
  vectors: $\tilde{\alpha}$, $\tilde{\omega}$ ;
2 Compute the average rates by using $r_B = 1/(-\tilde{\alpha}\tilde{T}^{-1}e)$
  and $r_O = 1/(-\tilde{\omega}\tilde{S}^{-1}e)$ ;
3 Based on the obtained rates $r_B$ and $r_O$, determine the
  transition blocks of the QBD process
      $\left\{B_1^{(0)}, A_2^{(1)}, A_0^{(0)}, A_2, A_1, A_0\right\}$;
4 Use equation (4) to compute the rate matrix $R$, stop the
  iteration if
      $\|R_{\mathbf{n}+1} - R_{\mathbf{n}}\| < \varepsilon$,
  and let $R \approx R_{\mathbf{n}}$ ;
5 Solve $\psi_0$ and $\psi_1$ by the system equations (6) ;
6 Compute $\eta_2$ by equation $\eta_2 = 1 - \psi_0 e$ ;
7 Compute the transaction throughput TH by equation
  TH $= b r_B \eta_2$ ;
8 Output the transaction throughput TH.
```

Before analyzing the impact of specific parameters on the transaction throughput of the PBFT-based blockchain system with repairable voting nodes, it is necessary to determine the orders of various significant sub-matrices with different values of b and n. Table II provides valuable information about the rapid growth of sub-matrix orders due to the influence of Kronecker operators and n. These higher order sub-matrices limit the ability to calculate the transaction throughput TH using MATLAB software. Therefore, our throughput analysis relies mainly on simulation. Meanwhile, to demonstrate the effectiveness of our throughput simulation, we also provide an approximate algorithm, Algorithm 1, for calculating transaction throughput. In this approximate algorithm, we assume that the average block generation time and the average orphan block generation time follow exponential distributions, similar to the algorithm proposed by Chang et al. [49]. Based on this approximate theoretical method and simulation experiments, we adopted the same parameter settings, taking $\theta = 2$, $p = 0.8$, $\gamma = 5$, $\lambda = 5$, $b = 150$, $\mu = 0, 0.5, 1, 1.5$, and presented the calculation results in Fig. 16.

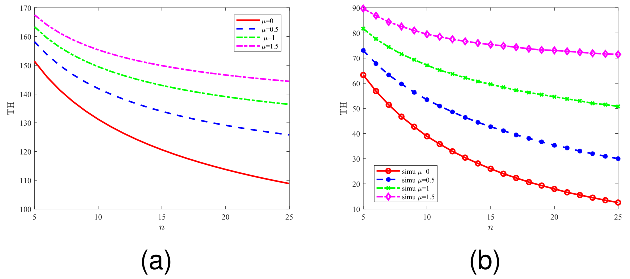


Fig. 16. Simulation and approximation theoretical values of TH.

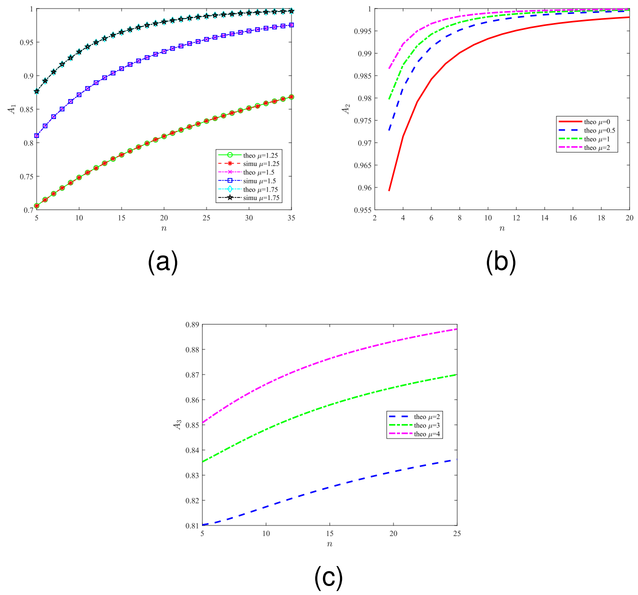


Fig. 17. A_1, A_2 and A_3 vs. n and μ.

As shown in Fig. 16, we can observe that the throughput trends of the theoretical approximation and the simulation are similar. Moreover, the throughput with repairable voting nodes is higher than the throughput without repairable voting nodes in both the simulation and theoretical approximation. This suggests that introducing a repair process to the PBFT-based blockchain system can improve the throughput. Furthermore, the larger the value of μ, the higher the throughput.

In order to explore the impact of μ and n on A_1, A_2, and A_3, we conducted three different experiments. For Fig. 17(a), we take $\theta = 0.5$, $n \in [5, 35]$, $\mu = 1.25, 1.5, 1.75$; For Fig. 17(b), we take $\theta = 2$, $p = 0.65$, $\gamma = 10$, $\beta = 3$, $n \in [3, 20]$, $\mu = 0, 0.5, 1, 2$; For Fig. 17(c), we take $\theta = 0.5$, $p = 0.85$, $\gamma = 10, \beta = 3$, $n \in [5, 25]$, $\mu = 2, 3, 4$. As depicted in Fig. 17, we can see that the stationary availabilities A_1, A_2 and A_3 increase as μ increases. In other words, a higher repair rate in the PBFT-based blockchain system results in higher stationary availabilities. These numerical results indicate that introducing a repair process to the PBFT-based blockchain system can enhance its availability.

Finally, it is worth mentioning that we cannot ignore the impact of the total number of voting nodes, $N = 3n+1$, on the performance of the PBFT-based blockchain system. Combining Fig. 16 and Fig. 17, we can observe that the transaction throughput decreases as the value of n increases, while the stationary availabilities increase as the value of n increases. Therefore, we can conclude that if our goal is to achieve a high transaction throughput in a PBFT-based blockchain system, we need to have a smaller number of voting nodes. However, if the majority of these voting nodes are Byzantine, it could compromise the security of a PBFT-based blockchain system. This finding suggests that administrators of voting-based blockchain systems may face a significant challenge in determining the optimal number of voting nodes, as they may need to balance system stability, security, and performance. Here, the selection of corresponding validators when adding a new parachain to the Polkadot system [61] can serve as a typical example.

VII. Concluding Remarks

In this paper, we analyzed a new dynamic and adaptive PBFT algorithm, which extends the ordinary PBFT by introducing a repair process for failed nodes. Based on this extension, we set up a large-scale Markov process to analyze the performance and reliability of the PBFT-based blockchain system with repairable voting nodes. On the one hand, we have extended the static PBFT consensus algorithm to achieve self-healing and restart of failed nodes, ensuring that the PBFT-based blockchain system can effectively adapt to changes in the network and nodes. On the other hand, we have demonstrated that the introduction of a repair process can improve the performance and reliability of the PBFT-based blockchain system. Therefore, we are optimistic that the methodology and results presented in this paper can be applied

$$F_{i,i}^{(k)} = \begin{pmatrix} (N-k-i)\gamma p & & & \\ & (N-k-i-1)\gamma p & & \\ & & \ddots & \\ & & & (N-k-n)\gamma p \end{pmatrix}_{(n+1-i)\times(n+1-i)};$$

for $0 \le k \le 2n$,

$$T_{k.k} = \begin{pmatrix} K_{0,0}^{(k)} & K_{0,1}^{(k)} & & & \\ & K_{1,1}^{(k)} & K_{1,2}^{(k)} & & \\ & & \ddots & \ddots & \\ & & & K_{n-1,n-1}^{(k)} & K_{n-1,n}^{(k)} \\ & & & & K_{n,n}^{(k)} \end{pmatrix},$$

for $0 \le i \le n-1$,

$$K_{i,i+1}^{(k)} = \begin{pmatrix} (N-k-i)\gamma q & & & \\ & (N-k-i-1)\gamma q & & \\ & & \ddots & \\ & & & (N-k-n+1)\gamma q \\ & & & 0 \end{pmatrix}_{(n+1-i)\times(n-i)},$$

$$K_{i,i}^{(k)} = \begin{pmatrix} c_{k,i,0} & (N-k-i)\theta & & & \\ \mu & c_{k,i,1} & (N-k-i-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i-1)\mu & c_{k,i,n-i-1} & (N-k-n+1)\theta \\ & & & (n-i)\mu & c_{k,i,n-i} \end{pmatrix}_{(n+1-i)\times(n+1-i)},$$

where,

$$\begin{aligned} c_{k,i,m} &= -[(N-k-i-m)(\theta+\gamma)+m\mu], 0 \le m \le n-i-1, \\ c_{k,i,n-i} &= -[(N-k-n)\gamma p+(n-i)\mu], \\ K_{n,n}^{(k)} &= -(N-k-n)\gamma p. \end{aligned}$$

$$F_{i,i}^{(k)} = \begin{pmatrix} (N-k-i)\gamma p & & & \\ & (N-k-i-1)\gamma p & & \\ & & \ddots & \\ & & & (N-k-n)\gamma p \end{pmatrix}_{(n+1-i)\times(n+1-i)};$$

for $0 \le k \le 2n-1$,

to more general practical scenarios of PBFT-based blockchain systems, leading to the development of efficient algorithms aimed at improving the throughput, security, and reliability of PBFT-based blockchain systems from the purpose of many actual uses. Along these lines, we will continue our future research in the following directions:

—Search for efficient algorithms to deal with the multi-dimensional Markov processes with a block structure corresponding to PBFT-based blockchain systems when the lifetime and/or repair time of a voting node is of phase type;

—Identify repaired nodes that rejoin the network and consider the time required to distribute workloads or transaction packages to newly joined nodes.

—Set up reward functions that are related to the cost structure, transaction fees, block rewards, security, reliability, throughput, and other relevant aspects, to develop stochastic

$$S_{k.k} = \begin{pmatrix} G_{0,0}^{(k)} & G_{0,1}^{(k)} & & & \\ & G_{1,1}^{(k)} & G_{1,2}^{(k)} & & \\ & & \ddots & \ddots & \\ & & & G_{n-1,n-1}^{(k)} & G_{n-1,n}^{(k)} \\ & & & & G_{n,n}^{(k)} \end{pmatrix},$$

for $0 \leq i \leq n-1$,

$$G_{i,i+1}^{(k)} = \begin{pmatrix} (N-k-i)\gamma q & & & \\ & (N-k-i-1)\gamma q & & \\ & & \ddots & \\ & & & (N-k-n+1)\gamma q \\ & & & 0 \end{pmatrix}_{(n+1-i)\times(n-i)},$$

$$G_{i,i}^{(k)} = \begin{pmatrix} c_{k,i,0} & (N-k-i)\theta & & & \\ \mu & c_{k,i,1} & (N-k-i-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i-1)\mu & c_{k,i,n-i-1} & (N-k-n+1)\theta \\ & & & (n-i)\mu & c_{k,i,n-i} \end{pmatrix}_{(n+1-i)\times(n+1-i)},$$

$$c_{k,i,m} = -[(N-k-i-m)(\theta+\gamma)+m\mu], 0 \leq m \leq n-i;$$

$$G_{n,n}^{(k)} = -(N-k-n)(\theta+\gamma);$$

$$S_{2n,2n} = \begin{pmatrix} H_{0,0} & H_{0,1} & & & \\ & H_{1,1} & H_{1,2} & & \\ & & \ddots & \ddots & \\ & & & H_{n-1,n-1} & H_{n-1,n} \\ & & & & H_{n,n} \end{pmatrix},$$

for $0 \leq i \leq n-1$,

$$H_{i,i+1} = \begin{pmatrix} (n+1-i)\gamma q & & & \\ & (n-i)\gamma q & & \\ & & \ddots & \\ & & & 2\gamma q \\ & & & 0 \end{pmatrix}_{(n+1-i)\ \times(n-i)},$$

$$H_{i,i} = \begin{pmatrix} f_{i,0} & (n+1-i)\theta & & & \\ \mu & f_{i,1} & (n-i)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i-1)\mu & f_{i,n-i-1} & 2\theta \\ & & & (n-i)\mu & f_{i,n-i} \end{pmatrix}_{(n+1-i)\ \times(n+1-i)},$$

$$f_{i,m} = -[(n+1-i-m)(\theta+\gamma q)\ +m\mu], 0 \leq m \leq n-i-1,$$

$$f_{i,n-i} = -[(N-k-n)\gamma p+(n-i)\mu];$$

$$H_{n,n} = -(\theta+\gamma q).$$

optimization, Markov decision processes, and stochastic game models to study PBFT-based blockchain systems.

—In the direction of stochastic optimization, dynamic control, and Markov decision processes of PBFT-based blockchain systems, develop efficient algorithms to find optimal policies that take into account throughput, security, reliability, and node management.

Appendix

This appendix provides the non-zero matrix elements or blocks in three important infinitesimal generators: Ψ, as defined in Section III-A; Θ, as defined in Section III-B; and Q, as defined in Section V-B. Our purpose is to increase the readability of the main paper.

Part A: The infinitesimal generator Ψ

$$\Psi = \begin{pmatrix} 0 & 0 \\ T^0 & T \end{pmatrix},$$

where, $T^0 + T\mathbf{e} = 0$,

$$T^0 = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ T_{2n}^0 \end{pmatrix}, T_{2n}^0 = \begin{pmatrix} L_0 \\ L_1 \\ \vdots \\ L_n \end{pmatrix},$$

Part C: The infinitesimal generator *Q*

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & & & \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & & \\ \vdots & & \ddots & \ddots & \\ Q_{2n,0} & & & Q_{2n,2n} & Q_{2n,2n+1} \\ Q_{2n+1,0} & & & & Q_{2n+1,2n+1} \end{pmatrix},$$

where,

$$Q_{2n+1,0} = \begin{pmatrix} J_{0,0} \\ J_{1,0} \\ \vdots \\ J_{n,0} \end{pmatrix}_{(\sum_{i=1}^{n+1} i)\times(\sum_{i=1}^{n+2} i)},$$

$$J_{i,0} = \begin{pmatrix} \beta \\ \beta \\ \vdots \\ \beta \end{pmatrix}_{(n+1-i)\times(n+2)}, \quad 0 \le i \le n;$$

$$Q_{2n+1,2n+1} = \begin{pmatrix} K_{0,0} & K_{0,1} & & & \\ & K_{1,1} & K_{1,2} & & \\ & & \ddots & \ddots & \\ & & & K_{n-1,n-1} & K_{n-1,n} \\ & & & & K_{n,n} \end{pmatrix},$$

for $0 \le i \le n-1$,

$$K_{i,i+1} = \begin{pmatrix} (n-i)\gamma q & & & \\ & (n-i-1)\gamma q & & \\ & & \ddots & \\ & & & \gamma q \\ & & & 0 \end{pmatrix}_{(n+1-i)\ \times(n-i)},$$

$$K_{i,i} = \begin{pmatrix} a_{i,0} & (n-i)\theta & & & \\ \mu & a_{i,1} & (n-i-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i-1)\mu & a_{i,n-i-1} & \theta \\ & & & (n-i)\mu & a_{i,n-i} \end{pmatrix}_{(n+1-i)\times(n+1-i)};$$

$$a_{i,m} = -[(n-i-m)(\theta+\gamma q) + m\mu + \beta], \quad 0 \le m \le n-i,$$
$$K_{n,n} = -\beta;$$

for $1 \le k \le 2n$,

$$L_i = \begin{pmatrix} (n+1-i)\gamma p \\ (n-i)\gamma p \\ \vdots \\ \gamma p \end{pmatrix}_{(n+1-i)\times 1}, 0 \le i \le n; \qquad T = \begin{pmatrix} T_{0,0} & T_{0,1} & & & & \\ & T_{1,1} & T_{1,2} & & & \\ & & \ddots & \ddots & & \\ & & & & T_{2n-1,2n-1} & T_{2n-1,2n} \\ & & & & & T_{2n,2n} \end{pmatrix},$$

$$Q_{k,0} = \begin{pmatrix} A_{0,0} \\ A_{1,0} \\ \vdots \\ A_{n+1,0} \end{pmatrix}_{(\sum_{i=1}^{n+2} i)\times(\sum_{i=1}^{n+2} i)},$$

$$A_{i,0} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \beta \end{pmatrix}_{(n+2-i)\times(n+2)}, \quad 0 \le i \le n,$$

$$A_{n+1,0} = (\beta, 0, 0, \ldots, 0)_{1\times(n+2)};$$

for $0 \le k \le 2n-1$,

$$Q_{k,k+1} = \begin{pmatrix} B_{0,0}^{(k)} & & & & \\ & B_{1,1}^{(k)} & & & \\ & & \ddots & & \\ & & & B_{n,n}^{(k)} & \\ & & & & 0 \end{pmatrix},$$

for $0 \le i \le n$,

$$B_{i,i}^{(k)} = \begin{pmatrix} (N-k-i)\gamma p & & & & \\ & (N-k-i-1)\gamma p & & & \\ & & \ddots & & \\ & & & (N-k-n)\gamma p & \\ & & & & 0 \end{pmatrix}_{(n+2-i)\times(n+2-i)};$$

$$Q_{2n,2n+1} = \begin{pmatrix} C_{0,0} & & & \\ & C_{1,1} & & \\ & & \ddots & \\ & & & C_{n,n} \\ & & & 0 \end{pmatrix},$$

where,

$$C_{i,i} = \begin{pmatrix} (n+1-i)\gamma p & & & \\ & (n-i)\gamma p & & \\ & & \ddots & \\ & & & \gamma p \\ & & & 0 \end{pmatrix}_{(n+2-i)\ \times(n+1-i)}, 0 \le i \le n;$$

$$Q_{0,0} = \begin{pmatrix} D_{0,0} & D_{0,1} & & & \\ D_{1,0} & D_{1,1} & D_{1,2} & & \\ \vdots & & \ddots & \ddots & \\ D_{n,0} & & & D_{n,n} & D_{n,n+1} \\ D_{n+1,0} & & & & D_{n+1,n+1} \end{pmatrix},$$

$$D_{n+1,0} = (\beta, 0, \ldots, 0)_{1\times(n+2)}, D_{n+1,n+1} = -\beta,$$

for $0 \leq k \leq 2n-1$,

$$T_{k,k+1} = \begin{pmatrix} F_{0,0}^{(k)} & & & \\ & F_{1,1}^{(k)} & & \\ & & \ddots & \\ & & & F_{n,n}^{(k)} \end{pmatrix},$$

for $0 \leq i \leq n$,

Part B: The infinitesimal generator Θ

$$\Theta = \begin{pmatrix} 0 & 0 \\ S^0 & S \end{pmatrix},$$

where, $S^0 + S\mathbf{e} = 0$,

$$S^0 = \begin{pmatrix} S_0^0 \\ S_1^0 \\ \vdots \\ S_{2n-1}^0 \\ S_{2n}^0 \end{pmatrix}, S_k^0 = \begin{pmatrix} S_{k,0}^0 \\ S_{k,1}^0 \\ \vdots \\ S_{k,n-1}^0 \\ S_{k,n}^0 \end{pmatrix}, 0 \leq k \leq 2n;$$

$$S_{k,i}^0 = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ (N-k-n)(\theta+\gamma q) \end{pmatrix}_{(n+1-i)\times 1}, 0 \leq i \leq n-1,$$

$$S_{k,n}^0 = (N-k-n)(\theta+\gamma q);$$

$$D_{i,0} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \beta \end{pmatrix}_{(n+2-i)\times(n+2)}, 1 \leq i \leq n,$$

for $0 \leq i \leq n$,

$$D_{i,i+1} = \begin{pmatrix} (N-i)\gamma q & & & \\ & (N-i-1)\gamma q & & \\ & & \ddots & \\ & & & (N-n)\gamma q \\ & & & 0 \end{pmatrix}_{(n+2-i)\times(n+1-i)};$$

$$D_{0,0} = \begin{pmatrix} -b_{0,0} & N\theta & & & \\ \mu & -b_{0,1} & (N-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & n\mu & -b_{0,n} & (N-n)\theta \\ \beta & & & (n+1)\mu & -d_0-\beta \end{pmatrix}_{(n+2)\times(n+2)},$$

$$\begin{aligned} b_{i,m} &= -[(N-i-m)(\theta+\gamma)+m\mu], 0 \leq m \leq n-i, \\ d_i &= (n+1-i)\mu, \quad 0 \leq i \leq n, \end{aligned}$$

for $1 \leq i \leq n$,

$$D_{i,i} = \begin{pmatrix} -b_{i,0} & (N-i)\theta & & & \\ \mu & -b_{i,1} & (N-i-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i)\mu & -b_{i,n-i} & (N-n)\theta \\ & & & d_i & -d_i-\beta \end{pmatrix}_{(n+2-i)\times(n+2-i)};$$

for $1 \leq k \leq 2n$,

$$Q_{k,k} = \begin{pmatrix} E_{0,0}^{(k)} & E_{0,1}^{(k)} & & & \\ & E_{1,1}^{(k)} & E_{1,2}^{(k)} & & \\ & & \ddots & \ddots & \\ & & & E_{n,n}^{(k)} & E_{n,n+1}^{(k)} \\ & & & & E_{n+1,n+1}^{(k)} \end{pmatrix},$$

$$E_{n+1,n+1}^{(k)} = -\beta,$$

for $0 \leq i \leq n$,

$$E_{i,i+1}^{(k)} = \begin{pmatrix} (N-k-i)\gamma q & & & \\ & (N-k-i-1)\gamma q & & \\ & & \ddots & \\ & & & (N-k-n)\gamma q \\ & & & 0 \end{pmatrix}_{(n+2-i)\times(n+1-i)},$$

$$E_{i,i}^{(k)} = \begin{pmatrix} -c_{k,i,0} & (N-k-i)\theta & & & \\ \mu & -c_{k,i,1} & (N-k-i-1)\theta & & \\ & \ddots & \ddots & \ddots & \\ & & (n-i)\mu & -c_{k,i,n-i} & (N-k-n)\theta \\ & & & d_i & -d_i-\beta \end{pmatrix}_{(n+2-i)\ \times(n+2-i)},$$

$$c_{k,i,m} = -[(N-k-i-m)(\theta+\gamma)+m\mu], 0 \leq m \leq n-i.$$

$$S = \begin{pmatrix} S_{0,0} & S_{0,1} & & \\ & \ddots & \ddots & \\ & & S_{2n-1,2n-1} & S_{2n-1,2n} \\ & & & S_{2n,2n} \end{pmatrix},$$

for $0 \leq k \leq 2n-1$,

$$S_{k,k+1} = \begin{pmatrix} F_{0,0}^{(k)} & & & \\ & F_{1,1}^{(k)} & & \\ & & \ddots & \\ & & & F_{n,n}^{(k)} \end{pmatrix},$$

for $0 \leq i \leq n$,

References

[1] M. Pease, R. Shostak, and L. Lamport, "Reaching agreement in the presence of faults," *J. ACM*, vol. 27, no. 2, pp. 228–234, 1980.

[2] V. Buterin and V. Griffith, "Casper the friendly finality gadget," 2017, *arXiv:1710.09437*.

[3] E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," 2018, *arXiv:1801.10228*.

[4] M. Baudet et al., *State Machine Replication in the Libra Blockchain*, Libra Assoc., Geneva, Switzerland, 2019.

[5] D. Cason, E. Fynn, N. Milosevic, Z. Milosevic, E. Buchman, and F. Pedone, "The design, architecture and performance of the tendermint blockchain network," in *Proc. 40th Int. Symp. Rel. Distrib. Syst. (SRDS)*, Chicago, IL, USA, 2021, pp. 23–33.

[6] G. Yu, B. Wu, and X. Niu, "Improved blockchain consensus mechanism based on PBFT algorithm," in *Proc. 2nd Int. Conf. Adv. Comput. Technol., Inf. Sci. Commun. (CTISC)*, 2020, pp. 14–21.

[7] Q. T. Thai, J. C. Yim, T. W. Yoo, H. K. Yoo, J. Y. Kwak, and S. M. Kim, "Hierarchical Byzantine fault-tolerance protocol for permissioned blockchain systems," *J. Supercomput.*, vol. 75, no. 11, pp. 7337–7365, 2019.

[8] A. A. Sajo, X. Huang, M. Saif, and Q. Chen, "Latency minimization in blockchain-enabled fog computing networks: A novel byzantine fault tolerance approach," in *Proc. IEEE 23rd Int Conf High Perform. Comput. Commun., 7th Int. Conf. Data Sci. Syst., 19th Int. Conf. Smart City, 7th Int. Conf. Dependability Sensor, Cloud Big Data Syst. Appl. (HPCC/DSS/SmartCity/DependSys)*, 2021, pp. 1529–1536.

[9] W. Jaafar, K. J. R. Beyara, I. Aouini, J. Ben Abderrazak, and H. Yanikomeroglu, "On the deployment of blockchain in edge computing wireless networks," in *Proc. IEEE 11th Int. Conf. Cloud Netw. (CloudNet)*, 2022, pp. 168–176.

[10] T. Crain, V. Gramoli, M. Larrea, and M. Raynal, "DBFT: Efficient leaderless Byzantine consensus and its application to blockchains," in *Proc. IEEE 17th Int. Symp. Netw. Comput. Appl. (NCA)*, 2018, pp. 1–8.

[11] J. Chen, X. Zhang, and P. Shangguan, "Improved PBFT algorithm based on reputation and voting mechanism," *J. Phys., Conf. Ser.*, vol. 1486, no. 3, 2020, Art. no. 032023, doi: 10.1088/1742-6596/1486/3/032023.

[12] P. Li, G. Wang, X. Chen, F. Long, and W. Xu, "Gosig: A scalable and high-performance Byzantine consensus for consortium blockchains," in *Proc. 11th ACM Symp. Cloud Comput.*, 2020, pp. 223–237.

[13] Y. Li et al., "An extensible consensus algorithm based on PBFT," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. (CyberC)*, 2019, pp. 17–23.

[14] F. Gai, J. Niu, S. Ali Tabatabaee, C. Feng, and M. Jalalzai, "Cumulus: A secure BFT-based sidechain for Off-chain scaling," in *Proc. IEEE/ACM 29th Int. Symp. Qual. Service (IWQOS)*, 2021, pp. 1–6.

[15] H. Wang, K. Guo, and Q. Pan, "Byzantine fault tolerance consensus algorithm based on voting mechanism," *J. Comput. Appl.*, vol. 39, no. 6, p. 1766, 2019.

[16] Y. Wang, Z. Song, and T. Cheng, "Improvement research of PBFT consensus algorithm based on credit," in *Proc. Int. Conf. Blockchain Trustworthy Syst.*, Guangzhou, China, 2019, pp. 47–59.

[17] W. Tong, X. Dong, and J. Zheng, "Trust-PBFT: A peerTrust-based practical Byzantine consensus algorithm," in *Proc. Int. Conf. Netw. Netw. Appl. (NaNA)*, 2019, pp. 344–349.

[18] S. Gao, T. Yu, J. Zhu, and W. Cai, "T-PBFT: An EigenTrust-based practical Byzantine fault tolerance consensus algorithm," *China Commun.*, vol. 16, no. 12, pp. 111–123, Dec. 2019.

[19] S. Sakho, J. Zhang, F. Essaf, K. Badiss, T. Abide, and J. K. Kiprop, "Research on an improved practical byzantine fault tolerance algorithm," in *Proc. 2nd Int. Conf. Adv. Comput. Technol., Inf. Sci. Commun. (CTISC)*, 2020, pp. 176–181.

[20] G. Christofi, "Study of consensus protocols and improvement of the Delegated Byzantine Fault Tolerance (DBFT) algorithm," M.S. thesis, Telecommun. Eng., Universitat Politècnica de Catalunya, Barcelona, Spain, 2019.

[21] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1689–1703, Mar. 2020.

[22] G. M. Diouf, H. Elbiaze, and W. Jaafar, "On Byzantine fault tolerance in multi-master Kubernetes clusters," *Future Gener. Comput. Syst.*, vol. 109, pp. 407–419, Aug. 2020.

[23] G. G. Gueta et al., "SBFT: A scalable and decentralized trust infrastructure," in *Proc. 49th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, 2019, pp. 568–580.

[24] L. Lao, X. Dai, B. Xiao, and S. Guo, "G-PBFT: A location-based and scalable consensus protocol for IOT-blockchain applications," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. (IPDPS)*, 2020, pp. 664–673.

[25] Y. Meshcheryakov, A. Melman, O. Evsutin, V. Morozov, and Y. Koucheryavy, "On performance of PBFT for IoT-applications with constrained devices," 2021, *arXiv:2104.05026*

[26] X. Yuan, F. Luo, M. Z. Haider, Z. Chen, and Y. Li, "Efficient Byzantine consensus mechanism based on reputation in IoT blockchain," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–14, May 2021. [Online]. Available: https://doi.org/10.1155/2021/9952218

[27] W. Hu, Y. Hu, W. Yao, and H. Li, "A blockchain-based Byzantine consensus algorithm for information authentication of the Internet of Vehicles," *IEEE Access*, vol. 7, pp. 139703–139711, 2019.

[28] J. Lim, T. Suh, J. Gil, and H. Yu, "Scalable and leaderless Byzantine consensus in cloud computing environments," *Inf. Syst. Front.*, vol. 16, no. 1, pp. 19–34, 2014.

[29] A. Sheikh, V. Kamuni, A. Urooj, S. Wagh, N. Singh, and D. Patel, "Secured energy trading using byzantine-based blockchain consensus," *IEEE Access*, vol. 8, pp. 8554–8571, 2019.

[30] S. Nakamoto. "Bitcoin: A peer-to-peer electronic cash system." 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[31] V. Buterin, "Ethereum whitepaper," 2013. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper

[32] M. Correia, G. S. Veronese, N. F. Neves, and P. Verissimo, "Byzantine consensus in asynchronous message-passing systems: A survey," *Int. J. Crit. Comput.-Based Syst.*, vol. 2, no. 2, pp. 141–161, 2011.

[33] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication," in *Proc. Int. Workshop Open Probl. Netw. Secur.*, Zurich, Switzerland, 2016, pp. 112–125.

[34] V. Gramoli, "From blockchain consensus back to Byzantine consensus," *Future Gener. Comput. Syst.*, vol. 107, pp. 760–769, Jun. 2020.

[35] C. Berger and H. P. Reiser, "Scaling byzantine consensus: A broad analysis," in *Proc. 2nd Workshop Scalable Resil. Infrastruct. Distrib. Ledgers*, 2018, pp. 13–18.

[36] S. Gupta, J. Hellings, S. Rahnama, and M. Sadoghi, "An in-depth look of BFT consensus in blockchain: Challenges and opportunities," in *Proc. 20th Int. Middlew. Conf. Tutor.*, 2019, pp. 6–10.

[37] N. Stifter, A. Judmayer, and E. Weippl, "Revisiting practical Byzantine fault tolerance through blockchain technologies," in *Proc. Secur. Qual. Cyber-Phys. Syst. Eng.*, 2019, pp. 471–495.

[38] S. Alqahtani and M. Demirbas, "Bottlenecks in blockchain consensus protocols," in *Proc. IEEE Int. Conf. Omni-Layer Intell. Syst. (COINS)*, 2021, pp. 1–8.

[39] X. Zheng and W. Feng, "Research on practical Byzantine fault tolerant consensus algorithm based on blockchain," *J. Phys., Conf. Ser.*, vol. 1802, no. 3, 2021, Art. no. 32022, doi: 10.1088/1742-6596/1802/3/032022.

[40] B. Gan, Q. Wu, X. Li, and Y. Zhou, "Classification of blockchain consensus mechanisms based on PBFT algorithm," in *Proc. Int. Conf. Comput. Eng. Appl. (ICCEA)*, 2021, pp. 26–29.

[41] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *Commun. ACM*, vol. 61, no. 7, pp. 95–102, 2018.

[42] J. Göbel, H. P. Keeler, A. E. Krzesinski, and P. G. Taylor, "Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay," *Perform. Eval.*, vol. 104, pp. 23–41, Oct. 2016.

[43] K. Javier and B. Fralix, "A further study of some Markovian Bitcoin models from Göbel et al.," *Stoch. Models*, vol. 36, no. 2, pp. 223–250, 2020.

[44] Q. L. Li, Y. X. Chang, X. Wu, and G. Zhang, "A new theoretical framework of pyramid Markov processes for blockchain selfish mining," *J. Syst. Sci. Syst. Eng.*, vol. 30, no. 6, pp. 667–711, 2021.

[45] X. S. Song, Q. L. Li, Y. X. Chang, and C. Zhang, "A Markov process theory for network growth processes of DAG-based blockchain systems," 2022, *arXiv:2209.01458*.

[46] F. Q. Ma, Q. L. Li, Y. H. Liu, and Y. X. Chang, "Stochastic performance modeling for practical Byzantine fault tolerance consensus in the blockchain," *Peer-to-Peer Netw. Appl.*, vol. 15, no. 6, pp. 2516–2528, 2022.

[47] Q. L. Li, J. Y. Ma, and Y. X. Chang, "Blockchain queue theory," in *Proc. Int. Conf. Comput. Soc. Netw.*, 2018, pp. 25–40.

[48] Q. L. Li, J. Y. Ma, Y. X. Chang, F. Q. Ma, and H. B. Yu, "Markov processes in blockchain systems," *Comput. Soc. Netw.*, vol. 6, no. 1, pp. 1–28, 2019.

[49] Y. X. Chang, Q. L. Li, Q. Wang, and X. S. Song, "Dynamic practical byzantine fault tolerance and its blockchain system: A large-scale Markov modeling," 2022, *arXiv:2210.14003*.

[50] M. Nischwitz, M. Esche, and F. Tschorsch, "Bernoulli meets PBFT: Modeling BFT protocols in the presence of dynamic failures," in *Proc. 16th Conf. Comput. Sci. Intell. Syst.*, 2021, pp. 291–300.

[51] X. Hao, L. Yu, Z. Liu, L. Zhen, and G. Dawu, "Dynamic practical Byzantine fault tolerance," in *Proc. IEEE Conf. Commun. Netw. Security*, 2018, pp. 1–8.

[52] Y. Hao, Y. Li, X. Dong, L. Fang, and P. Chen, "Performance analysis of consensus algorithm in private blockchain," in *Proc. IEEE Intell. Veh. Sympo. (IV)*, pp. 280–285, 2018.

[53] H. Sukhwani, J. M. Martnez, X. Chang, K. S. Trivedi, and A. Rindos, "Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric)," in *Proc. IEEE 36th Symp. Rel. Distrib. Syst. (SRDS)*, 2017, pp. 253–255.

[54] T. Lorünser, B. Rainer, and F. Wohner, "Towards a performance model for Byzantine fault tolerant services," in *Proc. CLOSER*, 2022, pp. 178–189.

[55] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, "Performance analysis of private blockchain platforms in varying workloads," in *Proc. 26th Int. Conf. Comput. Commun. Netw. (ICCCN)*, 2017, pp. 1–6.

[56] A. A. Monrat, O. Schelén, and K. Andersson, "Performance evaluation of permissioned blockchain platforms," in *Proc. IEEE Asia-Pacific Conf. Comput. Sci. Data Eng. (CSDE)*, 2020, pp. 1–8.

[57] A. Ahmad, M. Saad, J. Kim, D. Nyang, and D. Mohaisen, "Performance evaluation of consensus protocols in blockchain-based audit systems," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2021, pp. 654–656.

[58] K. Zheng, Y. Liu, C. Dai, Y. Duan, and X. Huang, "Model checking PBFT consensus mechanism in healthcare blockchain network," in *Proc. 9th Int. Conf. Inf. Technol. Med. Educ. (ITME)*, 2018, pp. 877–881.

[59] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Baltimore, MD, USA: Johns Hopkins Univ., 1981.

[60] Q.-L. Li, *Constructive Computation in Stochastic Models With Applications: The RG-Factorizations*. New York, NY, USA: Springer, 2010.

[61] J. Burdges et al., "Overview of polkadot and its design considerations," 2020, *arXiv:2005.13456*.

Yan-Xia Chang is currently pursuing the Ph.D. degree with the School of Economics and Management, Beijing University of Technology, Beijing, China. Her research interests include mathematical models of blockchain, resource management of big networks, queueing networks, mean-field theory, and service systems.

Qing Wang is currently pursuing the Ph.D. degree with the Monash Business School, Monash University, Caulfield East, Australia. Her research interests include corporate financing, big data, and machine learning.

Quan-Lin Li received the Ph.D. degree from the Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, China. He is a Full Professor with the School of Economics and Management Sciences, Beijing University of Technology, Beijing. He has published over 60 research papers in a variety of international journals, such as *Advances in Applied Probability*, *Queueing Systems*, *Stochastic Models*, *European Journal of Operational Research*, *Computer Networks*, *Performance Evaluation*, *Discrete Event Dynamic Systems*, *Computers and Operations Research*, *Computers & Mathematics With Applications*, *Annals of Operations Research*, and *International Journal of Production Economics*. His current research interests include stochastic models, computer networks, resource management in big networks, health care systems, mathematical models of blockchain, and sharing economy.

Yaqian Ma is currently pursuing the Ph.D. degree with the School of Economics and Management, Beijing University of Technology, Beijing, China. Her research interests include blockchain systems, machine learning, and Markov decision process.

Chi Zhang received the B.S. degree in industrial engineering and the M.S. degree in management science and engineering from Xi'an Jiaotong University, and the Ph.D. degree in systems engineering from the Stevens Institute of Technology. He is currently an Associate Professor with the School of Economics and Management, Beijing University of Technology. His research efforts focus on complex networked systems reliability analysis and optimization, maintenance optimization, critical infrastructure resilience, and warranty policy optimization.