International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

# Semi-Markov Based Dependability Modeling of Bitcoin Nodes Under Eclipse Attacks and State-Dependent Mitigation

**Chencheng Zhou**
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: czhou@umassd.edu

**Liudong Xing**
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
*Corresponding author*: lxing@umassd.edu

**Qisi Liu**
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: qliu1@umassd.edu

**Honggang Wang**
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: hwang1@umassd.edu

(Received October 30, 2020; Accepted December 21, 2020)

**Abstract**
The block chain technology has immense potential in many different applications, including but not limited to cryptocurrencies, financial services, smart contracts, supply chains, healthcare services, and energy trading. Due to the critical nature of these applications, it is pivotal to model and evaluate dependability of the block chain-based systems, contributing to their reliable and robust operation. This paper models and analyzes the dependability of Bitcoin nodes subject to Eclipse attacks and state-dependent mitigation activities. Built upon the block chain technology, the Bitcoin is a peer-to-peer cryptocurrency system enabling an individual user to trade freely without the involvement of banks or any other types of intermediate agents. However, a node in the Bitcoin is vulnerable to the Eclipse attack, which aims to monopolize the information flow of the victim node. A semi-Markov process (SMP) based approach is proposed to model the Eclipse attack behavior and possible mitigation activities that may prevent the attack from being successful during the attack process. The SMP model is then evaluated to determine the steady-state dependability of the Bitcoin node. Numerical examples are provided to demonstrate the influence of the time to restart the Bitcoin software and time to detect and delete the malicious message on the Bitcoin node dependability.

**Keywords**- Bitcoin, Block chain, Dependability, Eclipse attack, Semi-Markov process (SMP).

## 1. Introduction

The block chain technology has received lots of attentions from academia, governments, and industries in the last decade (Akbari et al., 2017; Atzei et al., 2017; Dai et al., 2019; Ferrag et al., 2018; Kang et al., 2018; Li et al., 2020). As a revolutionary invention in computer science, it has immense potential in many critical applications, including for example, cryptocurrencies, financial services, smart contracts, supply chains, energy trading, and the Internet of Things (Frizzo-Barker et al., 2020; Garay et al., 2017; Xing, 2020, 2021). This paper focuses on Bitcoin (Satoshi, 2008;

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

Wingreen et al., 2020), a peer-to-peer cryptocurrency network system based on the block chain with a market cap of 199 billion (Rudden, 2020). Different from the fiat currency, the Bitcoin is a decentralized network system that enables an individual to trade freely without the involvement of banks.

Despite being built on the secure block chain technology; the Bitcoin network has vulnerability to various cyberattacks or threats. For example, exploiting the open network of the Bitcoin, an attacker may track correspondence of IP addresses and the Bitcoin addresses to gain control of the block chain data (creating incorrect data or gain illegal access to the data) (Koshy et al., 2014). Further, the privacy of users (e.g., personal information) can be in danger as an attacker may track relationships between addresses based on the Bitcoin transactions (Reid and Harrigan, 2013). In addition, the block chain data may be tempered by an attacker through attacking the consensus mechanism of the block chain (Bag et al., 2016). The Bitcoin is also subject to the Crypto Locker-based attack, where a ransomware encrypts files of a victim until a ransom can be paid (Liao et al., 2016). The Bitcoin network is vulnerable to many other types of cyberattacks, including but not limited to the Eclipse attacks (Heilman et al., 2015), Sybil attacks (Zhang and Lee, 2019), mining pool attacks (Bahack, 2013), miner attacks (Rosenfeld, 2011), selfish mining (Eyal and Sirer, 2014), and re-identification attacks (Meiklejohn et al., 2013).

Extensive research efforts have been dedicated to defending the Bitcoin system against diverse security attacks. For example, Gervais et al. (2015) showed that an attacker may delay the propagation of a Bitcoin transaction to a specific node and proposed several countermeasures (dynamic timeouts, penalizing non-responding nodes, and updating block advertisements) to enhance the Bitcoin security. Eyal and Sirer (2014) suggested a mitigation scheme based on practical revision of the Bitcoin protocol to defend Bitcoin against colluding selfish mining attacks. Biryukov and Pustogarov (2015a, 2015b) investigated the Bitcoin over Tor system and showed that such a system is not effective in solving the security problem. Bamert et al. (2014) proposed a hardware token for securing Bitcoin transactions. Sasson et al. (2014) and Monaco (2015) investigated the weakness of Bitcoin in the privacy protection and suggested a decentralized anonymous payment mechanism to improve the privacy protection. Kroll et al. (2013) showed that many Nash equilibria exist for mining strategies and discussed the governance structure requirements. Joux (2004) showed that in multiple hash functions the difficulty of identifying simultaneous collisions is not higher than identifying individual ones. Bastiaan (2015) studied the pool mining threat and performed the stochastic analysis of the Bitcoin using Markov chains. Göbel et al. (2016) applied Markov Chains for detecting block-hiding attacks based on the monitoring of production rate of orphan blocks.

As exemplified above, existing works have mostly focused on detecting potential threats and investigating impacts from the malicious behavior. Only little work has been expended in the dependability modeling and analysis of Bitcoin. Particularly in Zhou et al. (2020), a continuous-time Markov chain-based method was proposed to assess the dependability of a Bitcoin node and effects of several parameters relevant to a miner's habits were examined. However, the model of Zhou et al. (2020) is limited to the exponential state transition time distribution; it is not applicable to the practical cases with other distribution types, where the memoryless property (the past history has no influence on the system's future behavior) does not hold.

In this paper we make advancement in the state of the art by proposing a semi-Markov process (SMP)-based method to model and analyze the dependability of Bitcoin nodes subject to the Eclipse

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

attack. The attack aims to control the information flow (including reception and transmission) of a victim node so that the node loses its connection with other legitimate nodes in the Bitcoin network. Possible mitigation activities that may prevent the attack from being successful during the attack process are considered. The transition time between different states appearing during the attack process can follow any arbitrary type of distributions. Influences of several parameters (related to the time to restart the Bitcoin software and time to detect and delete malicious messages containing forged IP addresses) on the Bitcoin node dependability are examined using examples.

The remainder of the paper is structured as follows: Section 2 presents how an Eclipse attack works and the state transition diagram in the SMP-based method modeling the attack behavior and the possible mitigation activities that may be conducted under each state. Section 3 presents the SMP-based method to evaluate the dependability of Bitcoin nodes. Section 4 investigates impacts of several parameters on the Bitcoin node dependability. Section 5 concludes our study and points out future research directions.

## 2. Modeling the Eclipse Attack

To launch an Eclipse attack, an attacker node floods the victim node with its own IP address to which the victim node will likely connect when restarting the Bitcoin software. In other words, the attacker node tries to fill the routing table of the victim node before the victim node restarts its software. The restart can be forced to happen, or the attacker node can just wait for the restart to happen. Once the restart happens, the victim node builds an outgoing connection with the attacker's IP address in the routing table. At the same time, the attacker node continuously builds an incoming connection to the victim node. Consequently, the victim node's information flow channel is controlled or monopolized by the malicious node; the victim node can be fed incorrect or fake data by the attacker node (Heilman et al., 2015).

In the case of the attacker node being able to implement the Eclipse attack to more nodes, information flows of more nearby nodes may be controlled, and gradually the block chain network may be compromised. Therefore, a successful Eclipse attack may lead to other more sever cyberattacks like selfish mining, double-spending, and block withholding (Heilman et al., 2015).

To model the dependability of a Bitcoin node, we build a state transition diagram as shown in Figure 1. During the Eclipse attack process five major states can be differentiated, which are the initial good state 0, state 1 where the routing table has been hacked, state 2 where the node has restarted its software, state 3 where the node is connected to the attacker node, and state 4 where the node is monopolized by the attacker node. Mitigation actions under some states may be performed to prevent or alleviate the consequence of the Eclipse attack.

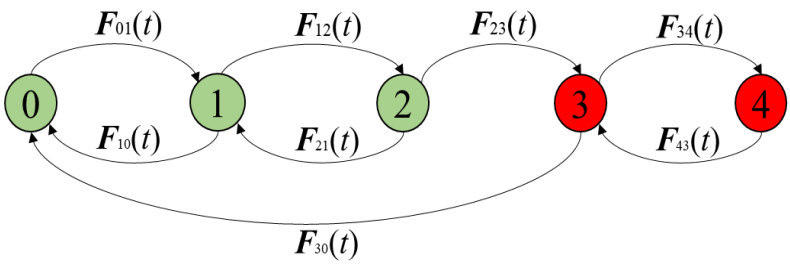


**Figure 1.** State transition diagram of a Bitcoin node under the Eclipse attack.

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

Specifically, under the initial state 0, the Bitcoin node operates correctly without being compromised by any attack. During this state 0, a malicious attacker node sends mmessages containing many forged IP addresses, which may gradually overwrite all the legal IP addresses in the routing table of the node causing the transition from state 0 to state 1. Under state 1, if the victim node executes the restart, then the node transits its state to state 2; if the suspicious message containing the forged IP addresses is detected and deleted by the user, then the node transits its state back to the initial good state 0. Under state 2, if the victim node is connected to the attacker's IP address, then the node transits to state 3; if the user cleans the routing table of the victim node with a certain tool, then the node transits back to state 1. Under state 3, if the victim node selects an IP address from the hacked routing table and establishes an outgoing connection, then the node transits to state 4; if the user restores the healthy connection through a certain maintenance action successfully, then the node transits back to state 0. Under state 4, the attacker node establishes and controls all the incoming connections to the victim node, fully monopolizing the information flow (incoming and outgoing) of the victim node, i.e., the Eclipse attack is successful. Under state 4, if the user detects the malicious connection from the attacker node and re-establishes connections with legitimate nodes, then the node transits its state back to state 3.

## 3. Evaluating Bitcoin Node Dependability

In this section, based on the state transition diagram in Figure 1, the SMP-based method is applied to assess the dependability of a Bitcoin node undergoing the Eclipse attack and mitigation actions. Let $F_{ij}(t)$ represent the cumulative distribution function (CDF) of the transition time from state $i$ to state $j$ ($i, j = 0, 1, 2, 3, 4$). There is no limitation to the distribution type of the state transition time. For illustration purpose, the Weibull distribution is selected due to its flexibility in representing different types of transition rates (decreasing, constant, and increasing) (Trivedi, 1982; Dohi et al., 2001; Xing and Amari, 2015; Xing et al., 2019). The CDF of the Weibull distribution is $F_{ij}\,(t\,;\,\alpha_{ij},\,\beta_{ij}) = 1 - e^{-(\frac{t}{\alpha_{ij}})^{\beta_{ij}}}$ with $(\alpha_{ij}, \beta_{ij})$ denoting the scale and shape parameters, respectively. The exponential distribution and the Rayleigh distribution are special cases of the Weibull distribution when the shape parameter is 1 and 2 respectively.

There are two major steps in the steady state probability analysis of the SMP (Kharoufeh et al., 2010; Kumar et al., 2013; Liu et al., 2019).

(1) Evaluate the one-step transition probability matrix of the embedded Markov chain (EMC) of the SMP (refer to appendices (A1-A3) of Kumar et al. (2013)).
(2) Calculate the sojourn time $T_i$ in each state $i$.

The steady-state probability $P_i$ of each state $i \in \{0,1,2,3,4\}$ can thus be evaluated using (1), where $v_i$ is the steady-state probability of state $i$ in the EMC.

$$P_i = \frac{v_i T_i}{\sum_{j\in\{0,1,2,3,4\}} v_j T_j}. \quad (1)$$

Specifically, expression (2) presents the kernel matrix K($t$) of the SMP model in Figure 1, where $k_{ij}(t)$ is the probability that the SMP has entered state $i$, the next state transition takes place within time $t$ and the next state is state $j$ (Kumar et al., 2013). The sum of elements on the same row of $\mathbf{K}(t)$ is always 1.

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

$$\mathbf{K}(t)=\begin{bmatrix} 0 & k_{01}(t) & 0 & 0 & 0 \\ k_{10}(t) & 0 & k_{12}(t) & 0 & 0 \\ 0 & k_{21}(t) & 0 & k_{23}(t) & 0 \\ k_{30}(t) & 0 & 0 & 0 & k_{34}(t) \\ 0 & 0 & 0 & k_{43}(t) & 0 \end{bmatrix} \tag{2}$$

In the case of the Weibull state transition time, the non-zero elements in K(*t*) are defined by (3)-(10). We use MATLAB and Wolfram Mathematics to calculate the integrals involved in those expressions.

$$k_{01}(t)=F_{01}(t)=1-e^{-\left(\frac{t}{\alpha_{01}}\right)^{\beta_{01}}} \tag{3}$$

$$k_{43}(t)=F_{43}(t)=1-e^{-\left(\frac{t}{\alpha_{43}}\right)^{\beta_{43}}} \tag{4}$$

$$k_{10}(t)=\int_0^t \bar{F}_{12}(x)dF_{10}(x)=\frac{\beta_{10}}{{\alpha_{10}}^{\beta_{10}}}\int_0^t x^{\beta_{10}-1}\,e^{-\left[\left(\frac{x}{\alpha_{10}}\right)^{\beta_{10}}+\left(\frac{x}{\alpha_{12}}\right)^{\beta_{12}}\right]}dx \tag{5}$$

$$k_{12}(t)=\int_0^t \bar{F}_{10}(x)dF_{12}(x)=\frac{\beta_{12}}{{\alpha_{12}}^{\beta_{12}}}\int_0^t x^{\beta_{12}-1}\,e^{-\left[\left(\frac{x}{\alpha_{10}}\right)^{\beta_{10}}+\left(\frac{x}{\alpha_{12}}\right)^{\beta_{12}}\right]}dx \tag{6}$$

$$k_{21}(t)=\int_0^t \bar{F}_{23}(x)dF_{21}(x)=\frac{\beta_{21}}{{\alpha_{21}}^{\beta_{21}}}\int_0^t x^{\beta_{21}-1}\,e^{-\left[\left(\frac{x}{\alpha_{21}}\right)^{\beta_{21}}+\left(\frac{x}{\alpha_{23}}\right)^{\beta_{23}}\right]}dx \tag{7}$$

$$k_{23}(t)=\int_0^t \bar{F}_{21}(x)dF_{23}(x)=\frac{\beta_{23}}{{\alpha_{23}}^{\beta_{23}}}\int_0^t x^{\beta_{23}-1}\,e^{-\left[\left(\frac{x}{\alpha_{21}}\right)^{\beta_{21}}+\left(\frac{x}{\alpha_{23}}\right)^{\beta_{23}}\right]}dx \tag{8}$$

$$k_{30}(t)=\int_0^t \bar{F}_{34}(x)dF_{30}(x)=\frac{\beta_{30}}{{\alpha_{30}}^{\beta_{30}}}\int_0^t x^{\beta_{30}-1}\,e^{-\left[\left(\frac{x}{\alpha_{30}}\right)^{\beta_{30}}+\left(\frac{x}{\alpha_{34}}\right)^{\beta_{34}}\right]}dx \tag{9}$$

$$k_{34}(t)=\int_0^t \bar{F}_{30}(x)dF_{34}(x)=\frac{\beta_{34}}{{\alpha_{34}}^{\beta_{34}}}\int_0^t x^{\beta_{34}-1}\,e^{-\left[\left(\frac{x}{\alpha_{30}}\right)^{\beta_{30}}+\left(\frac{x}{\alpha_{34}}\right)^{\beta_{34}}\right]}dx \tag{10}$$

The one-step transition probability matrix of the EMC in step (1) is thus determined as $\mathbf{K}(\infty)$, as shown in (11) (Kulkarni, 2016). Since the sum of elements on the same row of $\mathbf{K}(\infty)$ is always 1, $k_{01}(\infty)=1$ and $k_{43}(\infty)=1$ (they are the only element in row 1 and row 5, respectively).

$$\mathbf{K}(t)=\begin{bmatrix} 0 & k_{01}(\infty) & 0 & 0 & 0 \\ k_{10}(\infty) & 0 & k_{12}(\infty) & 0 & 0 \\ 0 & k_{21}(\infty) & 0 & k_{23}(\infty) & 0 \\ k_{30}(\infty) & 0 & 0 & 0 & k_{34}(\infty) \\ 0 & 0 & 0 & k_{43}(\infty) & 0 \end{bmatrix} \tag{11}$$

In this study, in order to estimate the steady-state probabilities of the EMC for the SMP, we apply Wolfram Mathematics to solve the EMC steady-state equations $\mathbf{v}=\mathbf{v}\cdot\mathbf{K}(\infty)$ and $\mathbf{v}\cdot\mathbf{e}^{\boldsymbol{T}}=\mathbf{1}$, where row vectors $\mathbf{v}=[\,v_0\ v_1\ v_2\ v_3\ v_4]$ and $\mathbf{e}=[1\ 1\ 1\ 1\ 1]$.

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

In step (2), based on Kumar et al. (2013) we use (12)-(16) to evaluate the sojourn time $T_i$ in each state $i$.

$$T_0 = \int_0^\infty \bar{F}_{01}(t)\,dt = \int_0^\infty e^{-(\frac{t}{\alpha_{01}})^{\beta_{01}}} dt \tag{12}$$

$$T_1 = \int_0^\infty \bar{F}_{10}\bar{F}_{12} dt = \int_0^\infty e^{-\left[\left(\frac{t}{\alpha_{10}}\right)^{\beta_{10}}+\left(\frac{t}{\alpha_{12}}\right)^{\beta_{12}}\right]} dt \tag{13}$$

$$T_2 = \int_0^\infty \bar{F}_{21}\bar{F}_{23} dt = \int_0^\infty e^{-\left[\left(\frac{t}{\alpha_{21}}\right)^{\beta_{21}}+\left(\frac{t}{\alpha_{23}}\right)^{\beta_{23}}\right]} dt \tag{14}$$

$$T_3 = \int_0^\infty \bar{F}_{30}\bar{F}_{34} dt = \int_0^\infty e^{-\left[\left(\frac{t}{\alpha_{30}}\right)^{\beta_{30}}+\left(\frac{t}{\alpha_{34}}\right)^{\beta_{34}}\right]} dt \tag{15}$$

$$T_4 = \int_0^\infty \bar{F}_{43}(t)\,dt = \int_0^\infty e^{-(\frac{t}{\alpha_{43}})^{\beta_{43}}} dt \tag{16}$$

With $v_i$ and $T_i$ evaluated, based on (1) the steady state probabilities $P_i$ ($i$ =0, 1, 2, 3, 4) can be obtained. Further, $D = P_0 + P_1 + P_2$ gives the probability that the Bitcoin node can operate correctly, i.e., the dependability of the Bitcoin node; $\bar{D} = P_3 + P_4$ gives the probability that the node is compromised or fully controlled by an attacker node, that is, the Bitcoin system is not dependable any more.

## 4. Example Analysis and Discussions

This section illustrates the SMP-based method for the Bitcoin node dependability analysis. We also examine influences of parameters modeling the time to restart and time to detect and delete the malicious messages on the Bitcoin node dependability by varying the Weibull distribution parameters of $F_{12}$ and $F_{10}$ in the dependability analysis. Table 1 shows the baseline parameters used in the numerical analysis (based on data of Thoman et al., 1969; Bailey and Dell, 1973; Kumar et al., 2013).

**Table 1.** Baseline model parameters.

| CDF | Distribution | Parameter Values |
|---|---|---|
| $F_{01}$ | Weibull | $\alpha_{01}$ = 6, $\beta_{01}$=1.54 |
| $F_{10}$ | Rayleigh | $\alpha_{10}$ = 10, $\beta_{10}$=2 |
| $F_{12}$ | Rayleigh | $\alpha_{12}$ = 12, $\beta_{12}$=2 |
| $F_{21}$ | Rayleigh | $\alpha_{21}$ = 14, $\beta_{21}$=2 |
| $F_{23}$ | Rayleigh | $\alpha_{23}$ = 12, $\beta_{23}$=2 |
| $F_{30}$ | Rayleigh | $\alpha_{30}$ = 10, $\beta_{30}$=2 |
| $F_{34}$ | Weibull | $\alpha_{34}$ = 11, $\beta_{34}$=1.32 |
| $F_{43}$ | Weibull | $\alpha_{43}$ = 11, $\beta_{43}$=0.85 |

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

### 4.1 Effects of Scale Parameters ($\alpha_{12}$, $\alpha_{10}$)

We vary the value of scale parameter ($\alpha_{12}$ and $\alpha_{10}$) from 1 hour to 96 hours and collect the system state probabilities and the final dependability $D$ (evaluated as $P_0$+$P_1$+$P_2$) in Tables 2-3. All other unchanging parameters use the values from Table 1. Figures 2 and 3 show the results of the node dependability graphically.

**Table 2.** State probabilities and dependability with changing $\alpha_{12}$.

| $\alpha_{12}$ (hour) | 1 | 6 | 12 | 24 | 48 | 96 |
|---|---|---|---|---|---|---|
| $P_0$ | 0.129931 | 0.187512 | 0.258956 | 0.329244 | 0.361635 | 0.371511 |
| $P_1$ | 0.051837 | 0.284053 | 0.423381 | 0.554982 | 0.604907 | 0.619781 |
| $P_2$ | 0.396252 | 0.308642 | 0.179237 | 0.068325 | 0.019745 | 0.005138 |
| $P_3$ | 0.275175 | 0.216571 | 0.124472 | 0.047447 | 0.013712 | 0.003568 |
| $P_4$ | 0.146802 | 0.115538 | 0.066403 | 0.025312 | 0.007315 | 0.001903 |
| $D$ | 0.578022 | 0.783429 | 0.875532 | 0.952552 | 0.986287 | 0.996431 |

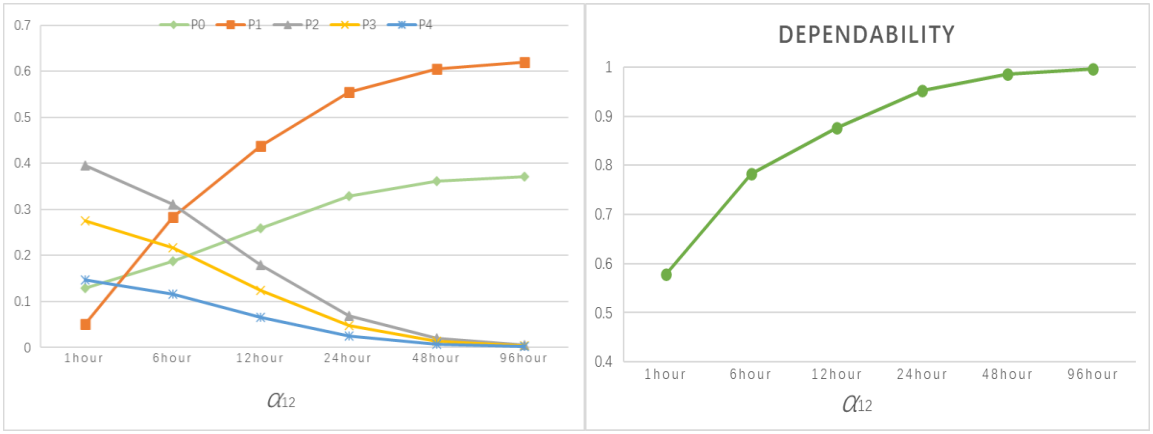


**Figure 2.** Steady-state probabilities and dependability with changing $\alpha_{12}$.

It can be observed from Figure 2 that when $\alpha_{12}$ (reflecting the time to restart the system) varies from 1 hour to 96 hours, the system dependability increases. This is intuitive since as the system restart rate decreases, the steady-state probabilities of being in state 2 (restart), and thus the subsequent states 3 (connected) and 4 (monopolized) decrease (as shown in Figure 2) while the steady-state probability of being in state 1 (the origin of the transition) and the initial state 0 have an increasing trend. Overall, the system dependability that is calculated as ($P_0$+$P_1$+$P_2$) shows an increasing trend.

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

**Table 3.** State probabilities and dependability with changing $\alpha_{10}$.

| $\alpha_{10}$ (hour) | 1 | 6 | 12 | 24 | 48 | 96 |
|---|---|---|---|---|---|---|
| $P_0$ | 0.843826 | 0.399027 | 0.220021 | 0.130262 | 0.100424 | 0.092280 |
| $P_1$ | 0.140724 | 0.404664 | 0.441405 | 0.441438 | 0.4383116 | 0.42164 |
| $P_2$ | 0.007481 | 0.115855 | 0.199815 | 0.252766 | 0.2722216 | 0.277704 |
| $P_3$ | 0.005195 | 0.080455 | 0.138762 | 0.175531 | 0.170423 | 0.192850 |
| $P_4$ | 0.002771 | 0.042922 | 0.074027 | 0.093644 | 0.1008519 | 0.102881 |
| $D$ | 0.992032 | 0.919545 | 0.861241 | 0.824468 | 0.8109576 | 0.807149 |

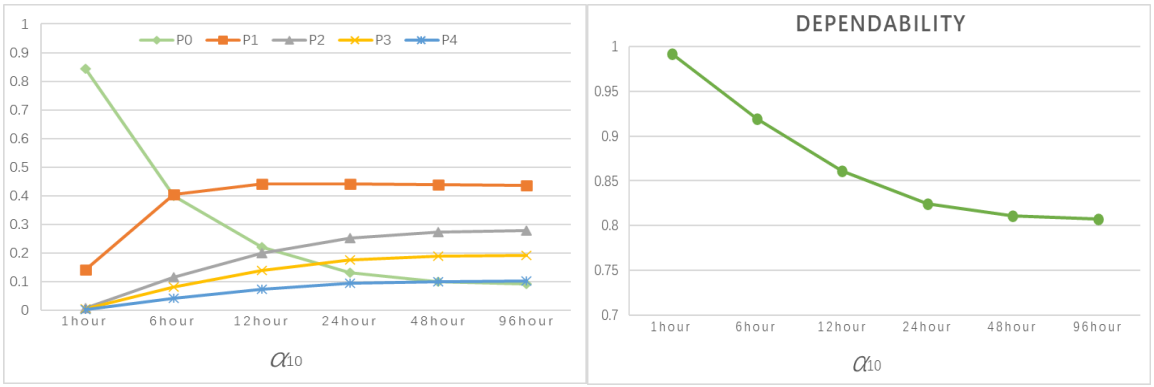


**Figure 3.** Steady-state probabilities and dependability with changing $\alpha_{10}$.

Figure 3 shows that when $\alpha_{10}$ (time to detect and delete the malicious message) varies from 1 hour to 96 hours, the node dependability decreases. The numerical results support the intuition that the node is more likely to get compromised when the time of detecting the malicious message increases. As the detection time increases, the probabilities of being in state 1 (the origin of the transition), and subsequent state 2 (restart), state 3 (connected), and state 4 (monopolized) all increase while the steady-state probability of the initial state 0 has a decreasing trend (as shown in Figure 3). Overall, the node dependability shows a decreasing trend as the value of $\alpha_{10}$ increases.

### 4.2 Effects of Shape Parameter ($\beta_{12}$, $\beta_{10}$)

The value of $\beta$ has a distinct effect on the state transition rate. Specifically, $\beta$<1corresponds to a decreasing transition rate, $\beta$=1 corresponds to a constant transition rate, and $\beta$>1 corresponds to an increasing transition rate. In order to reflect all these characteristics in this case study, we vary $\beta$ from 0.3 to 8. Tables 4 and 5 and Figures 4 and 5 show the numerical and graphic results of the different state probabilities and the final node dependability, respectively. All other unchanging parameters use values from Table 1.

International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 480-492, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.029

**Table 4.** State probabilities and dependability with changing $\beta_{12}$.

| $\beta_{12}$ | 0.3 | 0.7 | 1 | 1.5 | 2 | 5 | 6 | 8 |
|---|---|---|---|---|---|---|---|---|
| $P_0$ | 0.259741 | 0.211342 | 0.209831 | 0.208039 | 0.207821 | 0.201314 | 0.199813 | 0.189987 |
| $P_1$ | 0.171022 | 0.133373 | 0.128987 | 0.121131 | 0.121081 | 0.123878 | 0.125134 | 0.123413 |
| $P_2$ | 0.151092 | 0.181213 | 0.181534 | 0.181721 | 0.182143 | 0.183987 | 0.184012 | 0.187234 |
| $P_3$ | 0.217572 | 0.259226 | 0.252931 | 0.252707 | 0.241050 | 0.241296 | 0.241687 | 0.240535 |
| $P_4$ | 0.200571 | 0.238971 | 0.233168 | 0.232961 | 0.228336 | 0.222442 | 0.222802 | 0.221740 |
| $D$ | 0.581855 | 0.525928 | 0.520352 | 0.510891 | 0.511045 | 0.509179 | 0.508959 | 0.500634 |

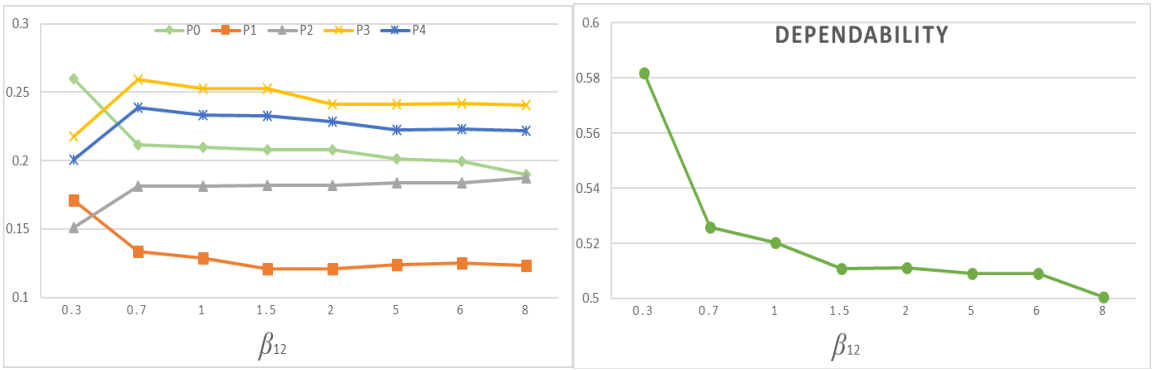


**Figure 4.** Steady-state probabilities and dependability with changing $\beta_{12}$.

**Table 5.** State probabilities and dependability with changing $\beta_{10}$.

| $\beta_{10}$ | 0.3 | 0.7 | 1 | 1.5 | 2 | 5 | 6 | 8 |
|---|---|---|---|---|---|---|---|---|
| $P_0$ | 0.264125 | 0.317062 | 0.310035 | 0.302540 | 0.30254 | 0.289604 | 0.287598 | 0.286883 |
| $P_1$ | 0.157029 | 0.232915 | 0.249917 | 0.268048 | 0.268048 | 0.299345 | 0.304197 | 0.305928 |
| $P_2$ | 0.153642 | 0.184436 | 0.180348 | 0.175988 | 0.175988 | 0.168463 | 0.167296 | 0.166880 |
| $P_3$ | 0.221245 | 0.265587 | 0.259701 | 0.253422 | 0.253422 | 0.242587 | 0.240907 | 0.240307 |
| $P_4$ | 0.203957 | 0.244835 | 0.239409 | 0.233621 | 0.233621 | 0.223632 | 0.222083 | 0.221531 |
| $D$ | 0.574797 | 0.734413 | 0.740299 | 0.746577 | 0.746577 | 0.758012 | 0.759092 | 0.759692 |

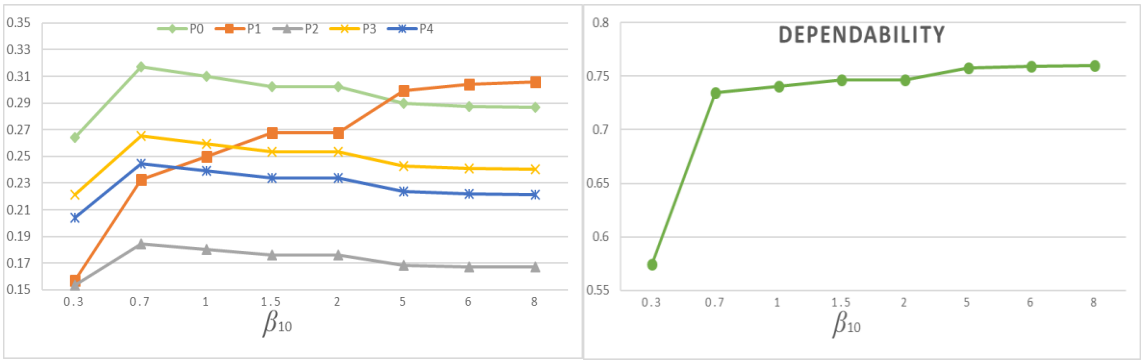


**Figure 5.** Steady-state probabilities and dependability with changing $\beta_{10}$.

Figure 4 shows that as the value of $\beta_{12}$ varies from 0.3 to 8, the overall trend of the node dependability decreases. This is because as the value of $\beta_{12}$ increases, the probability of occupying state 2 increases and the probability of staying in state 1 decreases significantly, which further causes the decrease in the probability of state 0 (as demonstrated in Figure 4). Overall, the node dependability shows a decreasing trend. It is also intuitive that the node dependability has an overall increasing trend in Figure 5 as the value of $\beta_{10}$ varies from 0.3 to 8.

## 5. Conclusion and Future Work

An Eclipse attack to a Bitcoin system aims to block a node's view of the block chain so that the victim node is at the mercy of the attacker node. The existing model on the Bitcoin node dependability analysis has the limitation of the memoryless property on the state transition time (i.e., the exponentially-distributed transition time). This paper contributes by proposing an SMP-based modeling method for dependability analysis of Bitcoin nodes subject to the eclipse attack and related mitigation actions during the attack process. The method is applicable to arbitrary types of state transition time distributions. Using numerical examples, we examine the influences of parameters modeling the miner's habits in restart and malicious message detection on the Bitcoin node dependability.

This work has focused on the steady-state Bitcoin node dependability analysis. In the future we plan to extend the SMP-based method for time-dependent dependability analysis of the Bitcoin node and network under the Eclipse attack. We are also interested in modeling other types of cyberattacks such as block withholding mining (Qin et al., 2020) and selfish mining (Yang et al., 2020). In addition, we may investigate resilience strategies to enhance the robustness of Bitcoin operation and improve its immunity to different types of threats.

**Conflict of Interest**

The authors confirm that there is no conflict of interest to declare for this publication.

**Acknowledgments**

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that help improve the quality of this work.

## References

Akbari, E., Wu, Q., Zhao, W., Arabnia, H.R., & Yang, M.Q. (2017, December). From block chain to internet-based Voting. In *2017 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 218-221). IEEE. Las Vegas, United States.

Atzei, N., Bartoletti, M., & Cimoli, T. (2017, April). A survey of attacks on ethereum smart contracts SOK. In *International Conference on Principles of Security and Trust* (pp. 164-186). Springer. Berlin, Heidelberg.

Bag, S., Ruj, S., & Sakurai, K. (2016). Bitcoin block withholding attack: analysis and mitigation. *IEEE Transactions on Information Forensics and Security*, *12*(8), 1967-1978.

Bahack, L. (2013). Theoretical bitcoin attacks with less than half of the computational power. *arXiv preprint arXiv:1312.7013*, https://eprint.iacr.org/2013/868.pdf.

Bailey, R.L., & Dell, T.R. (1973). Quantifying diameter distributions with the Weibull function. *Forest Science*, *19*(2), 97-104.

Bamert, T., Decker, C., Wattenhofer, R., & Welten, S. (2014, September). Bluewallet: the secure bitcoin wallet. In *International Workshop on Security and Trust Management* (pp. 65-80). Springer. Cham. Switzerland.

Bastiaan, M. (2015, January). Preventing the 51%-attack: a stochastic analysis of two phase proof of work in Bitcoin. https://fmt.ewi.utwente.nl/media/175.pdf. Accessed in December 2020.

Biryukov, A., & Pustogarov, I. (2015a, January). Proof-of-work as anonymous micropayment: rewarding a tor relay. In *International Conference on Financial Cryptography and Data Security* (pp. 445-455). Springer. Berlin, Heidelberg.

Biryukov, A., & Pustogarov, I. (2015b, May). Bitcoin over tor isn't a good idea. In *IEEE Symposium on Security and Privacy* (pp. 122-134). IEEE. San Jose, United States.

Dai, H.N., Zheng, Z., & Zhang, Y. (2019). Blockchain for Internet of things: a survey. *IEEE Internet of Things Journal*, *6*(5), 8076-8094.

Dohi, T., Goševa-Popstojanova, K., & Trivedi, K. (2001). Estimating software rejuvenation schedules in high-assurance systems. *The Computer Journal*, *44*(6), 473-485.

Eyal, I., & Sirer, E.G. (2014, March). Majority is not enough: bitcoin mining is vulnerable. In *International Conference on Financial Cryptography and Data Security* (pp. 436-454). Springer. Berlin, Heidelberg.

Ferrag, M.A., Derdour, M., Mukherjee, M., Derhab, A., Maglaras, L., & Janicke, H. (2018). Blockchain technologies for the internet of things: research issues and challenges. *IEEE Internet of Things Journal*, *6*(2), 2188-2204.

Frizzo-Barker, J., Chow-White, P.A., Adams, P.R., Mentanko, J., Ha, D., & Green, S. (2020). Blockchain as a disruptive technology for business: a systematic review. *International Journal of Information Management*, *51*, 102029.

Garay, J., Kiayias, A., & Leonardos, N. (2017, August). The bitcoin backbone protocol with chains of variable difficulty. In *Annual International Cryptology Conference* (pp. 291-323). Springer. Cham, Switzerland.

Gervais, A., Ritzdorf, H., Karame, G.O., & Capkun, S. (2015, October). Tampering with the delivery of blocks and transactions in bitcoin. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 692-705). Denver, United States.

Göbel, J., Keeler, H.P., Krzesinski, A.E., & Taylor, P.G. (2016). Bitcoin blockchain dynamics: the selfish-mine strategy in the presence of propagation delay. *Performance Evaluation*, *104*, 23-41.

Heilman, E., Kendler, A., Zohar, A., & Goldberg, S. (2015). Eclipse attacks on bitcoin's peer-to-peer network. In *24th USENIX Security Symposium* (pp. 129-144). Washington D.C., United States.

Joux, A. (2004, August). Multicollisions in iterated hash functions. Application to cascaded constructions. In *Annual International Cryptology Conference* (pp. 306-316). Springer. Berlin, Heidelberg.

Kang, J., Yu, R., Huang, X., Wu, M., Maharjan, S., Xie, S., & Zhang, Y. (2018). Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet of Things Journal*, *6*(3), 4660-4670.

Kharoufeh, J.P., Solo, C.J., & Ulukus, M.Y. (2010). Semi-Markov models for degradation-based reliability. *IIE Transactions*, *42*(8), 599-612.

Koshy, P., Koshy, D., & McDaniel, P. (2014, March). An analysis of anonymity in bitcoin using P2P network traffic. In *International Conference on Financial Cryptography and Data Security* (pp. 469-485). Springer. Berlin, Heidelberg.

Kroll, J.A., Davey, I.C., & Felten, E.W. (2013, June). The economics of Bitcoin mining, or bitcoin in the presence of adversaries. In *Proceedings of WEIS* (Vol. 2013, p. 11). Washington D.C., United States.

Kulkarni, V.G. (2016). *Modeling and analysis of stochastic systems*. Taylor & Francis, CRC Press, United States.

Kumar, G., Jain, V., & Gandhi, O.P. (2013). Availability analysis of repairable mechanical systems using analytical semi-Markov approach. *Quality Engineering*, *25*(2), 97-107.

Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2020). A survey on the security of blockchain systems. *Future Generation Computer Systems*, *107*, 841-853.

Liao, K., Zhao, Z., Doupé, A., & Ahn, G.J. (2016, June). Behind closed doors: measurement and analysis of CryptoLocker ransoms in bitcoin. In *2016 APWG Symposium on Electronic Crime Research (eCrime)* (pp. 1-13). IEEE. Toronto, Canada.

Liu, Q., Xing, L., & Zhou, C. (2019). Probabilistic modeling and analysis of sequential cyber-attacks. *Engineering Reports*, *1*(4), e12065.

Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M., & Savage, S. (2013, October). A fistful of bitcoins: characterizing payments among men with no names. In *Proceedings of the 2013 Conference on Internet Measurement Conference* (pp. 127-140). Barcelona, Spain.

Monaco, J.V. (2015, May). Identifying bitcoin users by transaction behavior. In *Biometric and Surveillance Technology for Human and Activity Identification XII* (Vol. 9457, p. 945704). International Society for Optics and Photonics. Baltimore, United States.

Qin, R., Yuan, Y., & Wang, F.Y. (2020). Optimal block withholding strategies for blockchain mining pools. *IEEE Transactions on Computational Social Systems*, *7*(3), 709-717.

Reid, F., & Harrigan, M. (2013). An analysis of anonymity in the bitcoin system. In *Security and Privacy in Social Networks.* Springer, New York, United States, pp. 197-223.

Rosenfeld, M. (2011). Analysis of bitcoin pooled mining reward systems. *arXiv preprint arXiv:1112.4980*.

Rudden, J. (2020). Bitcoin market capitalization quarterly. In 2020 *Statista.* https://www.statista.com/statistics/377382/bitcoin-market-capitalization/. Accessed in October 2020.

Sasson, E.B., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., & Virza, M. (2014, May). Zerocash: decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy* (pp. 459-474). IEEE. San Jose, CA, USA.

Satoshi, N. (2008). Bitcoin: A peer-to-peer electronic cash system. *Consulted*, *1*(2012), 28.

Thoman, D.R., Bain, L.J., & Antle, C.E. (1969). Inferences on the parameters of the Weibull distribution. *Technometrics*, *11*(3), 445-460.

Trivedi, K.S. (1982). *Probability and statistics with reliability, queuing, and computer science applications* (Vol. 13). Englewood Cliffs, NJ: Prentice-Hall.

Wingreen, S.C., Kavanagh, D., John Ennis, P., & Miscione, G. (2020). Sources of cryptocurrency value systems: the case of bitcoin. *International Journal of Electronic Commerce*, *24*(4), 474-496.

Xing, L. (2020). Reliability in internet of things: current status and future perspectives. *IEEE Internet of Things Journal*, *7*(8), 6704-6721.

Xing, L. (2021). Cascading failures in internet of things: review and perspectives on reliability and resilience. *IEEE Internet of Things Journal*, *8*(1), 44-64. doi: 10.1109/JIOT.2020.3018687.

Xing, L., & Amari, S.V (2015). *Binary decision diagrams and extensions for system reliability analysis*. Wiley-Scrivener, MA. United States. ISBN: 978-1-118-54937-7.

Xing, L., Levitin, G., & Wang, C. (2019). *Dynamic system reliability: modeling and analysis of dynamic and dependent behaviors*. John Wiley & Sons.

Yang, R., Chang, X., Mišić, J., & Mišić, V.B. (2020). Assessing block chain selfish mining in an imperfect network: honest and selfish miner views. *Computers & Security*, *97*, 101956.

Zhang, S., & Lee, J.H. (2019). Double-spending with a sybil attack in the bitcoin decentralized network. *IEEE Transactions on Industrial Informatics*, *15*(10), 5715-5722.

Zhou, C., Xing, L., & Liu, Q. (2020). Dependability analysis of Bitcoin subject to Eclipse attacks. *International Journal of Mathematical, Engineering and Management Sciences*, accepted in September 2020 and currently in press.