mathematics

MDPI

Review

Various Blockchain Governance Games: A Review

Song-Kyoo (Amang) Kim

Faculty of Applied Sciences, Macao Polytechnic University, R. de Luis Gonzaga Gomes, Macao, China; amang@mpu.edu.mo; Tel.: +853-8599-6455

Abstract: Blockchain Governance Game (BGG) is the stochastic model for describing the innovative security enhancement of decentralized network architectures. This hybrid model provides the best strategies to prepare for preventing network failures by attackers. Strategic alliance for the Blockchain Governance Game, a successor to BGG, adds the strategic alliance to prevent attacks by adapting the concept from business domains. The multi-layered Blockchain Governance Game was developed by combining these two basic models to defend complex networks. This paper not only provides a brief summary of each game model, but also identifies two key network security applications based on BGGs that could serve as a guide for actual implementation. This review is intended to encourage people to be inspired for their future research. The author hopes to encourage readers working in related research areas who might integrate BGG models into their research.

Keywords: Blockchain Governance Game; strategic alliance BGG; multi-layered BGG; Internet of Things; fluctuation theory; 51 percent attack; IoT security; connected car; drone swarm; cybersecurity

MSC: 60C55; 60K10; 90B15; 90B50; 91A35; 91A55; 93A30

check for updates

Citation: Kim, S.-K. Various Blockchain Governance Games: A Review. *Mathematics* **2023**, *11*, 2273. https://doi.org/10.3390/math11102273

Academic Editor: Konstantin Kozlov

Received: 19 April 2023
Revised: 8 May 2023
Accepted: 9 May 2023
Published: 12 May 2023

1. Introduction

The blockchain is a distributed public digital ledger that is maintained by consensus among a network of peer-to-peer nodes and blockchain networks, which has been widely used in a wide range of services and applications other than cryptocurrencies [1–3]. A conventional decentralized network has benefits that eliminate most of the security risks from centralized networks. The basic blockchain data structure is designed to store genuine transacted information in a logical chain which grows in an add-only fashion with all new confirmed blocks [4,5]. The federated consensus of miners ensures a blockchain security and this consensus is only reliable if no miner owns more than 50 percent of the computing power in the network and more than half of the nodes are not controlled by a single entity [6–9]. If this assumption is violated, the distributed consensus is rendered invalid [1,4,5]. As a result, decentralized networks (i.e., blockchain networks) require a certain level of security to defend against attackers. Verifying transactions, distributing blocks, and adding blocks to the blockchain are all part of blockchain security. Although blockchain records are not immutable, they are considered a secure network [10,11]. Some researchers have improved the protocol security levels, whilst other studies have proposed a new protocol to prevent the 51 percent attack [2,5]. Although these conventional protocol enhancements aim to prevent the 51 percent attack, their implementations are limited because the critical values and boundaries of solutions are arbitrarily chosen [12,13]. Hence, three theoretical models of Blockchain Governance Games (BGGs) have been developed by Kim [14–16].

- **Blockchain Governance Game (BGG)** [14] is a theoretical model that provides a stochastic game framework for determining the best strategies to prevent network failures. The combination of a mixed strategy game and fluctuation theories yields analytically tractable results for enhancing decentralized network securities.

- **Strategic Alliance for Blockchain Governance Game (SABGG)** provides an alternative method for reserving real nodes [15]. A novel secure blockchain network framework has been suggested for preventing damages. From a strategic management standpoint, the alliance concept is applied on top of a general BGG. This hybrid mathematical model aims to determine the strategies for protecting a network via strategic alliances with other nodes. This model is a combination of a strategic management framework on top of a conventional BGG.
- **Multi-Layered Blockchain Governance Game (MLBGG)** [16] is a complex model which is an analytical stochastic model for performing a security operation in order to protect entire multi-layered networks from attackers. This study thoroughly analyzes the set of networks using explicit mathematical forms for predicting when a security operation should be performed.

The BGG and its variants are widely referenced for related studies on blockchain securities [17–21] and game theory applications [22]. Although game theories are very classical mathematical modeling techniques, game theories have been widely applied, even for conventional artificial intelligence decision-making systems [23,24]. Additionally, various applications are have been developed by adapting BGGs since the first model was invented in the last couple of years [25,26]. Architectural approaches such as BGG, SABGG, and MLBGG could be used as a defense mechanism on the securities of machine learning training [27]. Data poisoning (DP) attacks are designed to undermine the integrity of a target model by modifying the required dataset used by the model during the training phase [28]. A thorough examination of the most recent advances in defense schemes against poisoning attacks is expected to serve as a guideline for developing a novel approach that achieves a certain level of immunization against DP on smart devices feeding data to smart city systems [29]. Although data poisoning defense mechanisms in the testing phase are very rare [27], a trusted execution environment (TEE) has been alternatively suggested for a secure execution environment to protect that which is targets confidentiality and integrity [30,31]. The mechanisms for constructing secured environments are broader than just AI-dedicated ones [32–37]. BGG-based defense mechanisms could be an alternative to TEEs by ensuring the integrity within connected nodes [14–16]. Any security mechanisms that improve integrity and confidentiality could be regarded as AI training system TEEs. The main contribution is that this paper offers a comprehensive review of BGGs and their flagship applications, which have been actively studied recently. This review paper is targeted to encourage other researchers to develop new applications based on BGG models. The BGGs are the first analytical and mathematically proven stochastic models for the blockchain-based network architecture that combine the game and fluctuation theory. The analytical functionals of the BGG models are the explicit formula forms for determining the decision-making parameters to avoid major attacks by executing preliminary operations beforehand.

The paper is organized as follows: Section 2 summarizes the stochastic models of BGG and SABGG. This section also explains how MLBGG is linked to two fundamental models. All mathematical models are also fully listed along with the condensed proofs in this section. In Section 3, two major BGG applications which were adapted into connected cars and a smart drone swarm are included. The BGG optimizations for these application are also demonstrated in this section. Finally, Section 4 contains the conclusion which indicates the direction of future research based on Blockchain Governance Games.

2. Stochastic Models of Blockchain Governance Game Variants

All BGGs are mathematically designed to ensure the feasibility of the model. This section briefly explains the theories of BGG models, which are essentially stochastic models that provide analytically tractable results using fluctuation theory and mixed-strategy game theory. The results allow us to predict the uptime and determine the optimal number of backup nodes to protect the blockchain network. The governance in a blockchain network follows decision parameters such as the time that elapses before an attacker has captured more than half of all nodes. Any action must not be taken until one step before the first pass time expires. Even if an attacker captures less than half the nodes, there is still a possibility that all nodes will be dominated by an attacker.

2.1. Blockchain Governance Game

The BGG is an antagonistic game model with the fluctuation model to analyze the network to enable decision making for preliminary security measures before attacks. The model aims to prevent blockchain-based attacks and keep the network decentralized. In the BGG, an attacker is trying to build an alternative blockchain faster than regular miners [1] and a defender only keeps a small percentage of the nodes that are released before the attack (see Figure 1).

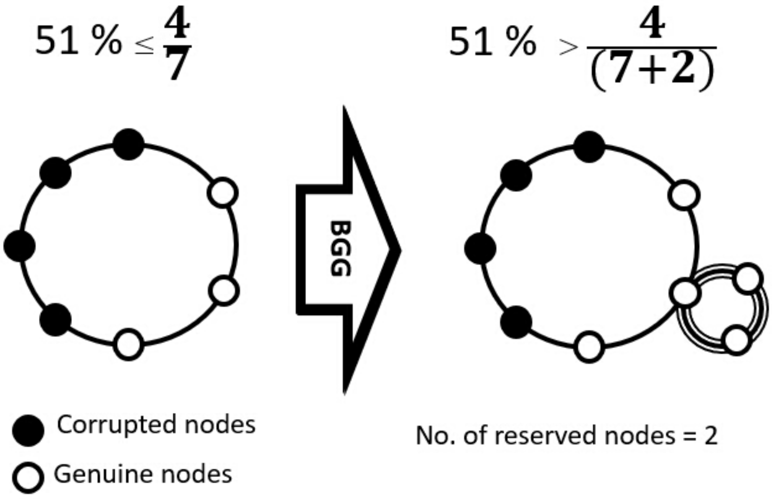

Figure 1. Blockchain Governance Game [14].

The two-player antagonistic stochastic game describes the blockchain network between a defender (player H) and an attacker (player A). The BGG aims to prevent the 51 percent attack and keep the network decentralized. Both players compete to build the blocks either for honest or false ones. Let us assign that

$$\mathcal{A} := \sum_{k\geq 0} X_k \varepsilon_{s_k}, s_0(=0) < s_1 < s_2 < \cdots, \text{a.s.} \tag{1}$$

$$\mathcal{H} := \sum_{j\geq 0} Y_j \varepsilon_{t_j}, t_0(=0) < t_1 < t_2 < \cdots, \text{a.s.} \tag{2}$$

are measurable marked Poisson processes (ε_a is a point mass at a) with respective intensities λ_a and λ_h. The computing performances for generating blocks by corrupted and honest nodes in the network are related with these two random variables. Although the traffic data processing and transmission delay could be modeled with the general input process, the Markovian input process, which is basically the Poisson process, is also widely applied into stochastic modeling for network analysis [38,39]. Player A builds the blocks with false transactions (e.g., double spend) at times $s_1, s_2, \ldots$ and sustains the respective building blocks of magnitudes $X_1, X_{2,\ldots}$ formalized by the process $\mathcal{A}$. The building blocks to player H who generates the honest blocks are similarly described by the process $\mathcal{H}$. The processes $\mathcal{A}$ and $\mathcal{H}$ could be specified by their transforms

$$\mathbb{E}\left[g^{\mathcal{A}(s)}\right] = e^{\lambda_a(s)(g-1)}, \mathbb{E}\left[z^{\mathcal{H}(t)}\right] = e^{\lambda_h(t)(z-1)}. \tag{3}$$

The BGG is randomly observed in accordance with the point process which is equivalent with the duration of the proof-of-work (PoW) completion (approximately 10 min in the Bitcoin) in the blockchain network [14]. The observation process is assumed to be a delayed renewal process:

$$\mathcal{T} := \sum_{i \geq 0} \varepsilon_{\tau_i}, \tau_0(> 0)), \tau_1, \ldots, \tag{4}$$

and let us consider the combined process as follows:

$$(A(t), H(t)) := \mathcal{A} \otimes \mathcal{H}([0, \tau_k]), k = 0, 1, \ldots. \tag{5}$$

with respective increments upon $\mathcal{A} \otimes \mathcal{H}$. Therefore, this process embedded over $\mathcal{T}$ could be defined as follows:

$$(X_k, Y_k) := \mathcal{A} \otimes \mathcal{H}([\tau_{k-1}, \tau_k]), k = 1, 2, \ldots, X_0 = A_0, Y_0 = H_0, \tag{6}$$

then the observation process is formalized as

$$\mathcal{A}_\tau \otimes \mathcal{H}_\tau := \sum_{k \geq 0} (X_k, Y_k) \varepsilon_{\tau_k}, \tag{7}$$

where

$$\mathcal{A}_\tau = \sum_{i \geq 0} X_i \varepsilon_{\tau_i}, \mathcal{H}_\tau = \sum_{i \geq 0} Y_i \varepsilon_{\tau_i}, \tag{8}$$

where X_k and Y_k are dependent on the notation $\Delta_k := \tau_k - \tau_{k-1}, k = 0, 1, \ldots, \tau_{-1} = 0$, and the magical transforms of increments are as follows:

$$\gamma(g, z) = \mathbb{E}\Big[g^{X_k} \cdot z^{Y_k}\Big] = \delta(\lambda_a(1-g) + \lambda_h(1-z)), \tag{9}$$

$$\gamma_0(g, z) = \mathbb{E}\Big[g^{A_0} z^{H_0}\Big] = \delta_0(\lambda_a(1-g) + \lambda_h(1-z)), \tag{10}$$

where

$$|g| < 1, |z| < 1, \ \delta(\theta) = \mathbb{E}\Big[e^{-\theta \Delta_1}\Big], \ \delta_0(\theta) = \mathbb{E}\Big[e^{-\theta \tau_0}\Big]. \tag{11}$$

The stochastic process $\mathcal{A}_\tau \otimes \mathcal{H}_\tau$ describes the evolution of a conflict between players with an observation process $\mathcal{T}$=$\{\tau_0, \tau_1, \ldots\}$. The game is over when, on the k-th observation epoch τ_k, the collateral building blocks to player A exceeds more than the half of the total nodes M. To further formalize the game, the exit index is introduced:

$$\nu := inf\bigg\{k : A_k = A_0 + X_1 + \cdots + X_k \geq \bigg(\frac{M}{2}\bigg)\bigg\}, \tag{12}$$

$$\mu := inf\bigg\{j : H_j = H_0 + Y_1 + \cdots + Y_j \geq \bigg(\frac{M}{2}\bigg)\bigg\}. \tag{13}$$

The first passage time τ_ν is the associated exit time from the confined game and the Formula (7) is modified as follows:

$$\overline{\mathcal{A}_\tau} \otimes \overline{\mathcal{H}_\tau} := \sum_{k \geq 0}^{\nu} (X_k, Y_k) \varepsilon_{\tau_k}, \tag{14}$$

which the path of the game from $\mathcal{F}(\Omega) \cap \{\nu < \mu\}$, which gives an exact definition of the model observed until τ_ν. We shall be targeting the confined game in the view point of player A (an attacker) because an attacker beats a defender at time τ_ν, otherwise, an honest node generates correct blocks. The joint functional of the blockchain network model is as follows:

$$\Phi_{\lceil \frac{M}{2} \rceil} = \Phi_{\lceil \frac{M}{2} \rceil}(\xi, g_0, g_1, z_0, z_1) = \mathbb{E}\Big[\xi^\nu \cdot g_0^{A_{\nu-1}} \cdot g_1^{A_\nu} \cdot z_0^{H_{\nu-1}} \cdot z_1^{H_\nu} \mathbf{1}_{\{\nu < \mu\}}\Big], \tag{15}$$

The above functional represents the status of genuine and corrupted nodes upon the exit moment τ_{ν}. The latter is of particular interest as we are interested in not only the prediction of catching up the blocks by attackers but also one observation prior to this. The **Theorem BGG-1** establishes an explicit formula $\Phi_{\frac{M}{2}}$ from (14) and (15) which is based on the first exceed model by Dshahalow [40,41]. The operators of the first exceed model are defined as follows:

$$\mathcal{D}_{(x,y)}\left[f(x,y)\right](u,v) := (1-u)(1-v)\sum_{x\geq 0}\sum_{y\geq 0} f(x,y)u^x v^y, \tag{16}$$

then

$$f(x,y) = \mathfrak{D}^{(x,y)}_{(u,v)}\left[\mathcal{D}_{(x,y)}\left\{f(x,y)\right\}\right], \tag{17}$$

where $\{f(x,y)\}$ is a sequence, with the inverse

$$\mathfrak{D}^{(m,n)}_{(u,v)}(\bullet) = \begin{cases} \left(\frac{1}{m!\cdot n!}\right) \lim_{(u,v)\to 0} \frac{\partial^m \partial^n}{\partial u^m \partial v^n} \frac{1}{(1-u)(1-v)}(\bullet), & m \geq 0, n \geq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

Theorem BGG-1. *The functional $\Phi_{\frac{M}{2}}$ of the process of (15) satisfies following expression:*

$$\Phi_{\lceil \frac{M}{2} \rceil} = \mathfrak{D}^{(\lceil \frac{M}{2} \rceil, \lceil \frac{M}{2} \rceil)}_{(u,v)}\left\{\Gamma_0^1 - \Gamma_0 + \frac{\xi\cdot\gamma_0}{1-\xi\gamma}\left(\Gamma^1 - \Gamma\right)\right\}. \tag{19}$$

where

$$\gamma := \gamma(g_0 g_1 u, z_0 z_1 v), \tag{20}$$
$$\gamma_0 := \gamma_0(g_0 g_1 u, z_0 z_1 v), \tag{21}$$
$$\Gamma := \gamma(g_1 u, z_1 v), \tag{22}$$
$$\Gamma_0 := \gamma_0(g_1 u, z_1 v), \tag{23}$$
$$\Gamma^1 := \gamma(g_1, z_1 v), \tag{24}$$
$$\Gamma_0^1 := \gamma_0(g_1, z_1 v). \tag{25}$$

The probability generating functions (PGFs) of $A_{\nu-1}$, A_ν and the exit index could be found as follows:

$$\mathbb{E}[\xi^\nu] = \Phi_{\lceil \frac{M}{2} \rceil}(\xi, 1, 1, 1, 1), \tag{26}$$
$$\mathbb{E}\left[g_0^{A_{\nu-1}}\right] = \Phi_{\lceil \frac{M}{2} \rceil}(1, g_0, 1, 1, 1), \tag{27}$$
$$\mathbb{E}\left[g_1^{A_\nu}\right] = \Phi_{\lceil \frac{M}{2} \rceil}(1, 1, g_1, 1, 1), \tag{28}$$

and the marginal mean of the first exceed index is as follows from (26):

$$\mathbb{E}[\nu] = \frac{\partial}{\partial \xi}\Phi_{\lceil \frac{M}{2} \rceil}(\xi, 1, 1, 1, 1)\Big|_{\xi=1}. \tag{29}$$

The special case of the observation process which has the memoryless characteristic is adapted for demonstrating an analytical solution of the BGG model on Appendix A.1. Memoryless observation processes are useful for describing BGG networks because it implicates that there is no additional cost for remembering past information.

2.2. Strategic Alliance for Blockchain Governance Game

SABGG, which is one of BGG variants, provides an alternative way to reserve genuine nodes for defending the network from attackers. The concept of alliance from the strategic management perspectives is applied on the top of a typical stochastic game framework [15]. The strategic alliance in the business is an agreement within multiple parties to pursue a set of agreed upon objectives [42]. This enhanced hybrid theoretical model finds the best strategies towards the preparation for preventing a network failure through allied genuine nodes (see Figure 2).

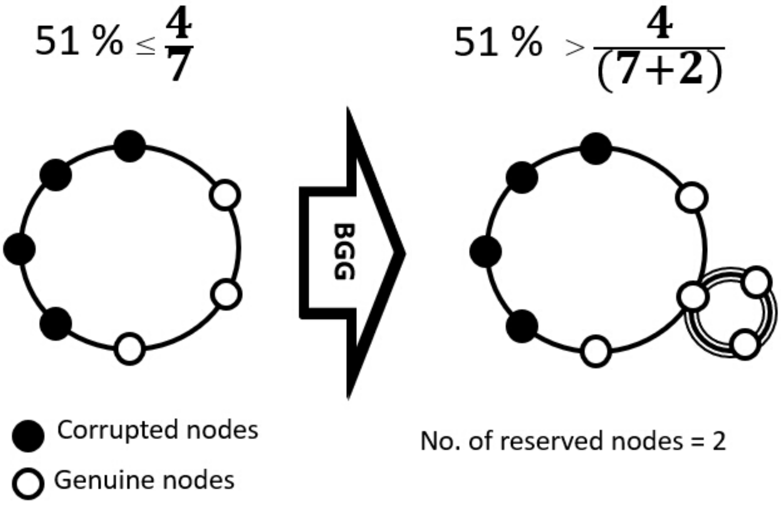

Figure 2. Strategic Alliance For Blockchain Governance Game [15].

The security operations are determined by the cost of adding genuine nodes, the number of nodes actively added to a blockchain network, and the total mining power of an attacker. The setup of the basic stochastic model is the same as the BGG model [14] except for managing backup nodes. The competitive non-cooperative game is considered. Two players (called "corrupted" and "genuine") compete with each other to complete their blocks first. From (1) and (2), measurable marked Poisson processes with respective intensities λ_c (an attacker) and λ_g (a defender) are as follows:

$$\mathcal{C} := \sum_{j\geq 0} J_j \varepsilon_{u_j}, u_0(=0) < u_1 < u_2 < \cdots, \text{a.s.} \tag{30}$$

$$\mathcal{G} := \sum_{k\geq 0} K_k \varepsilon_{v_k}, v_0(=0) < v_1 < v_2 < \cdots, \text{a.s.} \tag{31}$$

and a third-party observation point process [14,16] is also defined as follows:

$$\mathcal{U} := \sum_{i\geq 0} \varepsilon_{t_i}, t_0(>0), t_1, t_2, \ldots. \tag{32}$$

Player C (i.e., an attacker) build the blocks which contain false transactions, including double spending at the times u_1, $u_2, \ldots$. These blocks are built with the magnitudes J_1, J_2, …, formalized by the process $\mathcal{C}$. Similarly, player G generates the blocks which contain the correct transactions with the block of magnitudes K_1, $K_2, \ldots$. Both players are competing with each other to build their blocks. The processes $\mathcal{C}$ and $\mathcal{G}$ are specified by their transforms:

$$\mathbb{E}\left[y^{\mathcal{C}(u)}\right] = e^{\lambda_c(u)(y-1)}, \mathbb{E}\left[z^{\mathcal{G}(v)}\right] = e^{\lambda_g(v)(z-1)}, \tag{33}$$

and

$$(C_i, G_i) := \mathcal{C} \otimes \mathcal{G}([0, t_i]), i = 0, 1, \ldots, \tag{34}$$

as the forms of an observation process upon $\mathcal{C} \otimes \mathcal{G}$ embedded over t, with respective increments

$$(J_i, K_i) := \mathcal{C} \otimes \mathcal{G}([t_{i-1}, t_i]), i = 1, 2, \ldots, J_0 = C_0, K_0 = G_0. \tag{35}$$

The observation process is formalized as

$$\mathcal{C}_t \otimes \mathcal{G}_t := \sum_{i\geq 0}(J_i, K_i)\varepsilon_{t_i}, \tag{36}$$

where

$$\mathcal{C}_t = \sum_{i\geq 0} J_i\varepsilon_{t_i}, \mathcal{G}_t = \sum_{i\geq 0} K_i\varepsilon_{t_i}, \tag{37}$$

with position-dependent marking. The functional could be found as follows:

$$\alpha(y,z) = \mathbb{E}\Big[y^{J_i} \cdot z^{K_i}\Big], |y| \leq 1, |z| \leq 1, \tag{38}$$

with the notation $U_i := t_i - t_{i-1},\ i = 0, 1, \ldots, t_{-1} = 0$. By using the double expectation [3], we have

$$\alpha(y,z) = \alpha\big(\lambda_c(1-y) + \lambda_g(1-z)\big), \tag{39}$$

and

$$\alpha_0(y,z) = \mathbb{E}\Big[y^{C_0} z^{G_0}\Big] = \alpha_0\big(\lambda_c(1-y) + \lambda_g(1-z)\big), \tag{40}$$

where

$$\alpha(\theta) = \mathbb{E}\Big[e^{-\theta U_1}\Big], \alpha_0(\theta) = \mathbb{E}\Big[e^{-\theta t_0}\Big], |x| < 1, |z| < 1, \tag{41}$$

The game is ended when the total number of corrupted nodes C_i in the network becomes more than half of the total nodes by player C (an attacker) or G_l to player G (a defender) exceeds more than half of the total nodes, respectively. To further formalize the game, the exit indexes are defined as follows:

$$\nu := inf\bigg\{j : C_j\,(= C_0 + J_1 + \cdots + J_j) \geq \bigg(\frac{M}{2}\bigg)\bigg\}, \tag{42}$$

$$\nu_2 := inf\bigg\{j : C_j\,(= C_0 + J_1 + \cdots + J_j) - B_\eta \geq \bigg(\frac{M}{2}\bigg)\bigg\}, \tag{43}$$

$$\mu_1 := inf\bigg\{i : G_i(= G_0 + K_1 + \cdots + K_i) + B_\eta \geq \bigg(\frac{M}{2}\bigg)\bigg\}, \tag{44}$$

$$\mu := inf\bigg\{l : G_l(= G_0 + K_1 + \cdots + K_l) \geq \bigg(\frac{M}{2}\bigg)\bigg\}, \tag{45}$$

where B_η is the number of available nodes and η is the maximum fixed number of allied nodes in the network system (i.e., $B_\eta \leq \eta$). If player G (a defender) has the allies, player C (an attacker) could only win the game at moment t_{ν_2}, instead of moment t_ν. The game is over at $min\{\nu, \nu_2, (\mu_1), \mu\}$. The first passage time t_ν is the associated exit time from the confined game and the Formula (36) is modified as

$$\overline{\mathcal{C}_t} \otimes \overline{\mathcal{G}_t} := \sum_{n\geq 0}^{\nu}(J_n, K_n)\varepsilon_{t_n}, \tag{46}$$

which gives an exact definition of the model observed until t_ν without the strategic alliance action. The joint functional of the blockchain network model with the strategic alliance is as follows:

$$\Theta_{\frac{M}{2}} = \mathbb{E}\Big[\zeta^{\nu} \cdot y_0^{C_{\nu-1}} \cdot y_1^{C_\nu} \cdot b^{C_\nu - B_\eta} \cdot z_0^{G_{\mu-1}} \cdot z_1^{G_\mu}\mathbf{1}_{\{\nu<\nu_2<\mu\}}\Big], \tag{47}$$

where M indicates the total number of nodes in a blockchain network and this functional represents the status of attackers and defenders upon the exit time t_ν. It is noted that we are interested in not only the prediction of catching up the blocks by attackers but also one observation prior to this. The **Theorem BGG-2** establishes an explicit formula for $\Theta_{\frac{M}{2}}$

with (46) and (47). Additionally, $\mathcal{D}$- and $\mathfrak{D}$-operators from (16) and (18) are extended as follows:

$$\mathcal{D}^{(q,r,s)}_{(a,b,c)}\left[g(a,b,c)\right] := (1-q)(1-r)(1-s)\left\{\sum_{a\geq 0}\sum_{b\geq 0}\sum_{c\geq 0} g(a,b,c)q^a r^b s^c\right\}, \tag{48}$$

where $|q| < 1, |r| < 1, |s| < 1$, then we have

$$g(a,b,c) = \mathfrak{D}^{(a,b,c)}_{(q,r,s)}\left[\mathcal{D}_{(a,b,c)}\left\{g(a,b,c)\right\}(q,r,s)\right], \tag{49}$$

where $\{g(a,b,c)\}$ is a sequence, with the inverse

$$\mathfrak{D}^{(a,b,c)}_{(q,r,s)}(\bullet) = \begin{cases} \left(\frac{1}{a!\cdot b!\cdot c!}\right)\lim_{(q,r,s)\to 0}\frac{\partial^a\partial^b\partial^c}{\partial q^a\partial r^b\partial s^c}\frac{1}{(1-q)(1-r)(1-s)}(\bullet), & a,b,c\geq 0\\ 0, & \text{otherwise.}\end{cases} \tag{50}$$

Theorem BGG-2. *The functional $\Theta_{\frac{M}{2}}$ from (47), satisfies the following expression:*

$$\Theta_{\lceil\frac{M}{2}\rceil} = \mathfrak{D}^{(\lceil\frac{M}{2}\rceil,\lceil\frac{M}{2}\rceil,\lceil\frac{M}{2}\rceil)}_{(q,r,s)}\left\{\sigma_\eta\cdot\beta\left[\frac{1-\beta^1}{1-\beta}\right]\left[\alpha_0^1-\alpha_0+\frac{\zeta\alpha_0}{1-\zeta\alpha}\left(\alpha^1-\alpha\right)\right]\right\}, \tag{51}$$

where

$$\Phi := \alpha(y_0y_1bqr, z_0z_1s), \tag{52}$$

$$\Phi_0 := \alpha_0(y_0y_1bqr, z_0z_1s), \tag{53}$$

$$\alpha := \alpha(y_1bq, z_1), \tag{54}$$

$$\alpha_0 := \alpha_0(y_1bq, z_1), \tag{55}$$

$$\alpha^1 := \alpha(y_1b, z_1), \tag{56}$$

$$\alpha_0^1 := \alpha_0(y_1b, z_1), \tag{57}$$

$$\beta := \alpha(br, s), \tag{58}$$

$$\beta^1 := \alpha(r, 1), \tag{59}$$

$$\sigma_\eta := \mathbb{E}\left[b^{-B_\eta}\right]. \tag{60}$$

From (51), the probability generating functions of $C_{\nu-1}$, C_ν, and the exit index ζ could be found as follows:

$$\mathbb{E}[\zeta^\nu] = \Theta_{\lceil\frac{M}{2}\rceil}(\zeta,1,1,1,1,1), \tag{61}$$

$$\mathbb{E}\left[y_0^{C_{\nu-1}}\right] = \Theta_{\lceil\frac{M}{2}\rceil}(1,y_0,1,1,1,1), \tag{62}$$

$$\mathbb{E}\left[y_1^{C_\nu}\right] = \Theta_{\lceil\frac{M}{2}\rceil}(1,1,y_1,1,1,1), \tag{63}$$

$$\mathbb{E}\left[b^{C_\nu-B_\eta}\right] = \Theta_{\lceil\frac{M}{2}\rceil}(1,1,1,b,1,1). \tag{64}$$

The marginal mean of the first exceed index $\mathbb{E}[\nu]$ and the moment of security operation $\mathbb{E}[\tau_{\nu-1}]$ are found from (51) and (61):

$$\mathbb{E}[\nu] = \frac{\partial}{\partial\zeta}\Theta_{\lceil\frac{M}{2}\rceil}(\zeta,1,1,1,1,1)\Big|_{\zeta=1}, \tag{65}$$

$$\mathbb{E}[t_{\nu-1}] = \mathbb{E}[t_0] + \mathbb{E}[U_1](\mathbb{E}[\nu]-1). \tag{66}$$

Although the original BGG in Section 2.1 is an innovative idea in itself for the decentralized network security enhancement, certain numbers of real nodes must be reserved in advance. This is one of the reasons why the strategic alliance concept is being adapted for a

BGG successor. Additionally, it is noted that the memoryless observation process has been applied for SABGG networks on Appendix A.2.

2.3. Multi-Layered Blockchain Governance Game

The MLBGG is a combined stochastic game model based on the two-layered BGG network [16]. Layer-0 is a single SABGG-based network and Layer-1 is a set of multiple BGG-based networks (see Figure 3). This innovative multi-layer networking framework makes it easy for BGGs to apply various hierarchical system architectures, including IoT server networks [16], edge fog computing [43,44] and hierarchical network systems [45,46].

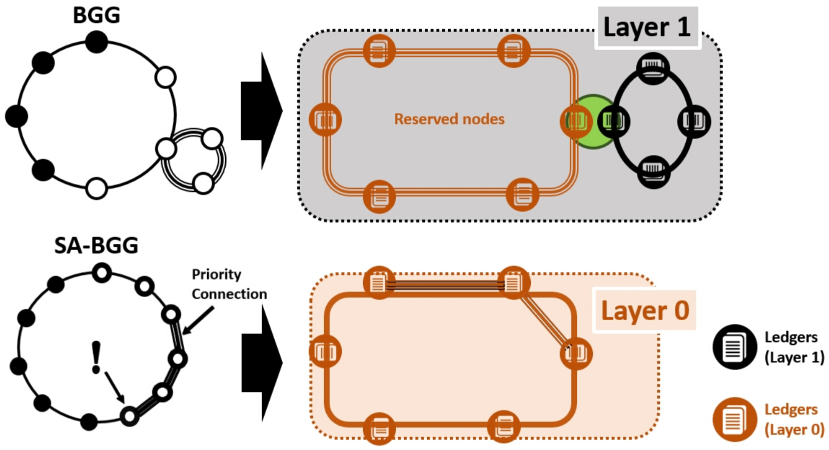

Figure 3. Multi-Layered Blockchain Governance Game [16].

This functional represents the status of an attacker and honest nodes upon the exit time $\tau_v^l, l = \{0, 1, \ldots, \eta\}$. Let us consider the matrix of a function $\boldsymbol{f}_{(x,y)}$ and $\boldsymbol{G}_{(u,v)}$ as follows:

$$\boldsymbol{f}_{(x,y)} := \begin{bmatrix} f_0(x,y) \\ f_1(x,y) \\ \vdots \\ \vdots \\ f_n(x,y) \end{bmatrix}, \boldsymbol{G}_{(u,v)} := \begin{bmatrix} G_0(u,v) \\ \vdots \\ G_l(u,v) \\ \vdots \\ G_\eta(u,v) \end{bmatrix}, \tag{67}$$

and the matrix operators based on the first exceed model from **Theorem BGG-1** are adapted as follows:

$$\mathbb{D} \odot \boldsymbol{f}_{(x,y)} := \mathcal{D}_{(x,y)}\{\boldsymbol{f}\} = \begin{bmatrix} (1-u)(1-v)\sum\sum f_0(x,y)u^x v^y \\ (1-u)(1-v)\sum\sum f_1(x,y)u^x v^y \\ \vdots \\ (1-u)(1-v)\sum\sum f_l(x,y)u^x v^y \\ \vdots \\ (1-u)(1-v)\sum\sum f_n(x,y)u^x v^y \end{bmatrix}, \tag{68}$$

and

$$\mathbb{D}^{\boldsymbol{-1}}_{\boldsymbol{M_r}} \odot \boldsymbol{G}_{(u,v)} := \begin{bmatrix} \mathfrak{D}^{(m_0,n_0)}_{(u,v)}\{G_0(u,v)\} \\ \mathfrak{D}^{(m_1,n_0)}_{(u,v)}\{G_1(u,v)\} \\ \vdots \\ \mathfrak{D}^{(m_l,n_l)}_{(u,v)}\{G_l(u,v)\} \\ \vdots \\ \mathfrak{D}^{(m_n,n_n)}_{(u,v)}\{G_r(u,v)\} \end{bmatrix}, \boldsymbol{M_r} = \begin{bmatrix} m_0 & n_0 \\ m_1 & n_1 \\ \vdots & \vdots \\ \vdots & \vdots \\ m_r & n_r \end{bmatrix}. \tag{69}$$

The new matrix operators for matrix calculations were additionally introduced in this research [16] and the functional matrix for all blockchain networks in Layer-1 is as follows:

$$\mathbf{\Phi}^{\mathbf{1}}_{\boldsymbol{M}_\eta} = \begin{Bmatrix} \Phi^0_{\lceil \frac{M_0}{2} \rceil} \\ \vdots \\ \Phi^l_{\lceil \frac{M_l}{2} \rceil} \\ \vdots \\ \Phi^\eta_{\lceil \frac{M_\eta}{2} \rceil} \end{Bmatrix} = \mathbb{D}^{-\mathbf{1}}_{\boldsymbol{M}_\eta} \odot \boldsymbol{G}_{(u,v)},\ \boldsymbol{M}_\eta = \begin{Bmatrix} \lceil \frac{M_0}{2} \rceil & \lceil \frac{M_0}{2} \rceil \\ \vdots & \vdots \\ \lceil \frac{M_l}{2} \rceil & \lceil \frac{M_l}{2} \rceil \\ \vdots & \vdots \\ \lceil \frac{M_\eta}{2} \rceil & \lceil \frac{M_\eta}{2} \rceil \end{Bmatrix}, \tag{70}$$

and

$$\Phi^l_{\lceil \frac{M_l}{2} \rceil} = \mathfrak{D}^{\left(\lceil \frac{M_l}{2} \rceil, \lceil \frac{M_l}{2} \rceil\right)}_{(u,v)} \left\{ \Gamma^1_0 - \Gamma_0 + \frac{\xi \cdot \gamma_0}{1 - \xi\gamma} \left(\Gamma^1 - \Gamma \right) \right\}, l = \{0, \ldots, \eta\}, \tag{71}$$

where

$$\gamma \ := \ \gamma^l(g_0 g_1 u, z_0 z_1 v), \tag{72}$$

$$\gamma_0 \ := \ \gamma^l_0(g_0 g_1 u, z_0 z_1 v), \tag{73}$$

$$\Gamma \ := \ \gamma^l(g_1 u, z_1 v), \tag{74}$$

$$\Gamma_0 \ := \ \gamma^l_0(g_1 u, z_1 v), \tag{75}$$

$$\Gamma^1 \ := \ \gamma^l(g_1, z_1 v), \tag{76}$$

$$\Gamma^1_0 \ := \ \gamma^l_0(g_1, z_1 v). \tag{77}$$

From (71), the probability-generating functions (PGFs) for $A^l_{\nu^l - 1}$, $A^l_{\nu^l}$, and the exit index ν^l of the l-th BGG network in the Layer-1 are determined as follows:

$$\mathbb{E}\left[\xi^{\nu^l}\right] = \Phi^l_{\lceil \frac{M_l}{2} \rceil}(\xi, 1, 1, 1, 1), \tag{78}$$

$$\mathbb{E}\left[g_0^{A^l_{\nu^l - 1}}\right] = \Phi^l_{\lceil \frac{M_l}{2} \rceil}(1, g_0, 1, 1, 1), \tag{79}$$

$$\mathbb{E}\left[g_1^{A^l_{\nu^l}}\right] = \Phi^l_{\lceil \frac{M_l}{2} \rceil}(1, 1, g_1, 1, 1), l = \{0, \ldots, \eta\}. \tag{80}$$

From (71) and (78), the marginal mean of the decision-making moment for the l-th network in the Layer-1 (i.e., $\tau_{\nu^l - 1}$) could be found as follows:

$$\mathbb{E}\left[\tau_{\nu^l - 1}\right] = \mathbb{E}[\tau_0] + \mathbb{E}[\Delta_1]\left(\mathbb{E}\left[\nu^l\right] - 1\right), l = \{1, \ldots, \eta\}. \tag{81}$$

where

$$\mathbb{E}\left[\nu^l\right] = \frac{\partial}{\partial \xi} \Phi^l_{\lceil \frac{M}{2} \rceil}(\xi, 1, 1, 1, 1)\Big|_{\xi = 1}. \tag{82}$$

Alternatively, Layer-0 is mapped with the SABGG and the exit indices are formalized as follows:

$$\nu := inf\left\{ j : C_j \,(= C_0 + J_1 + \cdots + J_j) \geq \left(\frac{\eta}{2}\right) \right\}, \tag{83}$$

$$\nu_2 := inf\left\{ j : C_j \,(= C_0 + J_1 + \cdots + J_j) - B^0 \geq \left(\frac{\eta}{2}\right) \right\}, \tag{84}$$

$$\mu := inf\left\{ l : G_l (= G_0 + K_1 + \cdots + K_l) \geq \left(\frac{\eta}{2}\right) \right\}, \tag{85}$$

where B^0 $(\leq \eta)$ is the number of available nodes within the ally. The game in Layer-0 is over at $min\{\nu, \nu_2, \mu\}$. The first passage time t_ν is the associated exit time from the confined game from (51) which gives an exact definition of the model observed until t_ν without the strategic alliance action. The explicit formula of the SABGG [15] is as follows:

$$\Theta_{\frac{\eta}{2}} = \mathbb{E}\Big[\zeta^{\nu} \cdot y_0^{C_{\nu-1}} \cdot y_1^{C_\nu} \cdot b^{C_\nu - B_\eta} \cdot z_0^{G_{\mu-1}} \cdot z_1^{G_\mu} \mathbf{1}_{\{\nu<\nu_2<\mu\}}\Big], \tag{86}$$

from the **Theorem BGG-2** on Section 2.2,

$$\Theta_{\lceil \frac{\eta}{2} \rceil} = \mathfrak{D}_{(q,r,s)}^{(\lceil \frac{\eta}{2} \rceil, \lceil \frac{\eta}{2} \rceil, \lceil \frac{\eta}{2} \rceil)} \Big\{\sigma_\eta \cdot \beta\left(\frac{1-\beta^1}{1-\beta}\right) \cdot \left(\alpha_0^1 - \alpha_0 + \frac{\zeta \Phi_0}{1-\zeta\Phi}\left(\alpha^1 - \alpha\right)\right)\Big\}, \tag{87}$$

where

$$\Phi := \alpha(y_0 y_1 bqr, z_0 z_1 s), \tag{88}$$
$$\Phi_0 := \alpha_0(y_0 y_1 bqr, z_0 z_1 s), \tag{89}$$
$$\alpha := \alpha(y_1 bq, z_1), \tag{90}$$
$$\alpha_0 := \alpha_0(y_1 bq, z_1), \tag{91}$$
$$\alpha^1 := \alpha(y_1 b, z_1), \tag{92}$$
$$\alpha_0^1 := \alpha_0(y_1 b, z_1), \tag{93}$$
$$\beta := \alpha(br, s), \tag{94}$$
$$\beta^1 := \alpha(r, 1), \tag{95}$$
$$\sigma_\eta := \mathbb{E}\Big[b^{-B_\eta}\Big]. \tag{96}$$

The moment of making a decision (i.e., $t_{\nu-1}$) of the Layer-0 could be found from (66). Additionally, the probability of bursting the Layer-0 of the blockchain network $q^0(s_h)$ is determined as follows:

$$q^0(s_g) = \begin{cases} \mathbb{E}\Big[\mathbf{1}_{\{C_\nu \geq \frac{\eta}{2}\}}\Big], & s_g = \{DoNothing\}, \\ \mathbb{E}\Big[\mathbf{1}_{\left\{C_\nu \geq \frac{\eta(1+\alpha)}{2}\right\}}\Big], & s_g = \{Action\}. \end{cases} \tag{97}$$

where α is an overhead portion for protecting Layer-0 (i.e., $B^0 = \frac{\alpha \cdot \eta}{2}$). The probability of bursting a SABGG network by an attacker could be as follows:

$$q(s_g) = \begin{cases} \sum_{k>\frac{\eta}{2}} \mathbb{E}\Big[\mathbf{1}_{\{C_\nu = k\}}\Big], & s_g = \{\text{DoNothing}\}, \\ \mathbb{E}\Big[\sum_{k>\frac{\eta}{2}+B^0} \mathbb{E}\Big[\mathbf{1}_{\{C_\nu = k\}}\Big]\Big], & s_g = \{\text{Action}\}, \end{cases} \tag{98}$$

where

$$\mathbb{E}\Big[\mathbf{1}_{\{C_\nu = k\}}\Big] = \mathbb{E}\left[\mathbb{E}\left[\frac{(\lambda_c t_\nu)^k}{k!} \cdot e^{-\lambda_c t_\nu} \middle| t_\nu\right]\right]. \tag{99}$$

3. Blockchain Governance Game Applications

This section focuses on two important applications for BGGs. The first application is to implement the BGG into a connected car to track the spare parts of connected cars back through the supply chain to their original manufacturers and preventing counterfeits. A connected car transfers data to others based on an automotive vehicle network (AVN) and cars are equipped for the in-vehicle networks. These car are usually unmanned aerial vehicles (UAVs) which drive artificial intelligence [47,48]. The IoV security is intended to improve fleet management and accident avoidance [25]. The second application is the application of the SABGG to intelligent drone swarms [26]. An intelligent drone is a drone equipped with artificial intelligence (AI) that operates autonomously and without a command center. The SABGG is used to improve the safety of an intelligent drone swarm by estimating the timing of interim actions by ensuring optimal drone accountability. It is noted that verifiable random functions (VRFs) are incorporated into blockchain-based security applications to eliminate the need for heavy computational power for mining [49,50]. The VRF can choose a miner at random, and each node has the same chance of being a miner [51,52]. This method has been modified to select a miner who will map inputs to verifiable pseudo-random outputs. The VRF is assumed to be fully supported for implementing the BGG on real systems.

3.1. Automotive Vehicle Network Security for Connected Cars

Automotive vehicle network (AVN) is a network that connects a number of vehicles and sensors by wireless communication which enables inter-vehicle information sharing as well as network-to-vehicle communications [53]. Blockchain technology might apply to safety monitoring including the replacement of smart car parts. The blockchain-based Internet-of-Vehicles (BIoV) network structure has been designed [25] to handle such situations. This application adapts the BGG into data sharing security and tractability using consensus schemes. The components in a connected car, the equipment of a service center, and a headquarter database are hooked up as one blockchain network (see Figure 4). Each smart component could mechanically or electronically generate random values and share these values with other smart components. Tires, brakes, an engine, a transmitter in a car could be the smart components which shall be capable to communicate with other components and to construct ledgers. A service center could also generate unique values based on registered car databases.

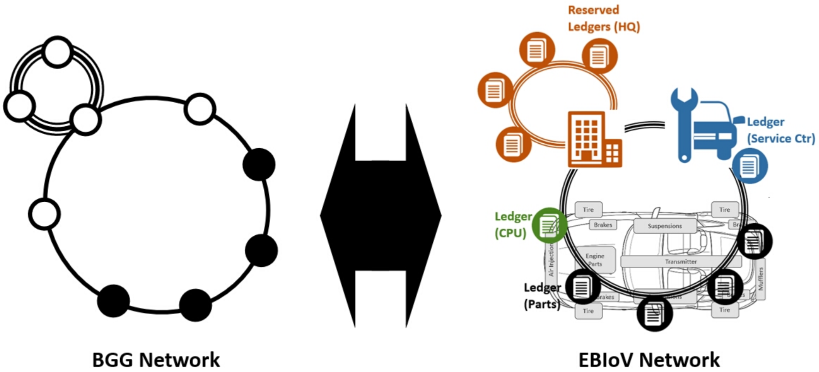

Figure 4. BGG Application for the EBIoV architecture [25].

The IoT-enabled components in a connected car generate values based on their mechanical actions and these generated values are shared with all other components including assigned service centers and company headquarters. It is noted that the values from connected car nodes are unique and randomly generated. The enhanced BIoV network (also known as EBIoV) does not have a reward system that requires a high computing power to create ledgers. All nodes in the EBIoV network, including those in service centers and headquarters, have an equal chance of becoming a miner that generates ledgers without requiring a large amount of computing power using the VRF [51,52]. The mechanism for the EBIoV network protection is identical to that of the BGG and the EBIoV network and is secured by adding nodes to reduce the possibility of an attacker intercepting blocks with false control requests. The optimal number of EBIoV nodes and the estimated car value could be analytically solvable and the corresponding total cost function $\mathfrak{S}(n,\rho)_{Total}$ is formulated as follows [25]:

$$\mathfrak{S}(n,\rho)_{Total} = \left\{ c(n,\rho)\left(1 - q^{1}_{(n,\rho)}\right) + \left(c(n,\rho) + V\right) q^{1}_{(n,\rho)} \right\} p_{A_{-1}} + V q^{0} \cdot (1 - p_{A_{-1}}), \quad (100)$$

where

$$p_{A_{-1}} = \sum_{k=0}^{\left\{\frac{M}{2}-\lambda_a\widetilde{\delta}\right\}} \left(\frac{\left\{\lambda_a\left(\widetilde{\delta_0}+\mathbb{E}[\nu-1]\widetilde{\delta}\right)\right\}^k}{k!} \cdot e^{-\lambda_a\left(\widetilde{\delta_0}+\mathbb{E}[\nu-1]\widetilde{\delta}\right)} \right), \tag{101}$$

$$q^0 \simeq 1 - \sum_{k=0}^{\frac{M}{2}} \left(\frac{\left\{\lambda_a\left(\widetilde{\delta_0}+\mathbb{E}[\nu-1]\widetilde{\delta}\right)\right\}^k}{k!} \cdot e^{-\lambda_a\left(\widetilde{\delta_0}+\mathbb{E}[\nu-1]\widetilde{\delta}\right)} \right), \tag{102}$$

$$P_j = \binom{n}{j}\rho^j(1-\rho)^{n-j}. \tag{103}$$

The total cost could be minimized by finding a proper parameter set (n, ρ), constituting a combination of an acceptance rate and the number of total backup nodes from headquarters. The optimal parameter set (n^*, ρ^*) minimizes the cost function from (100). The demonstration for optimizing the cost function is shown in Figure 5.

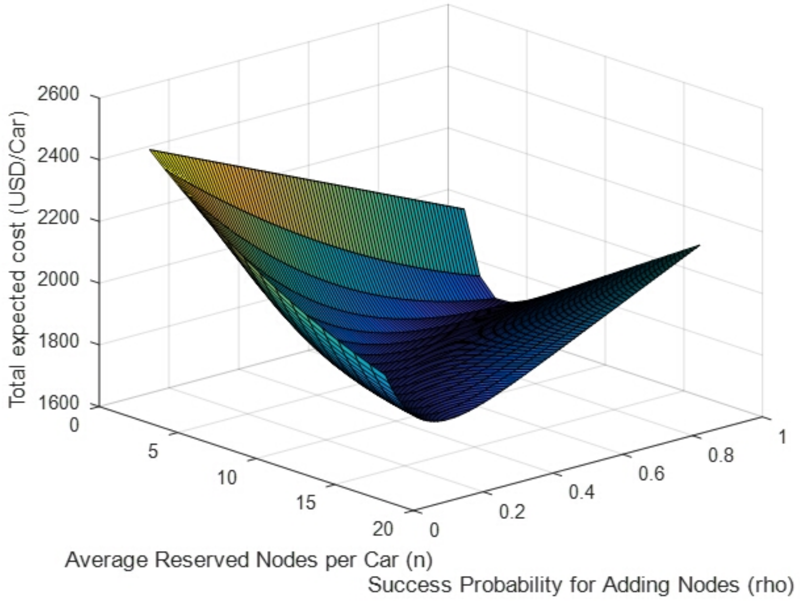

Figure 5. Optimization Example for the EBIoV [25].

The time for requesting the additional nodes shall be the moment $\tau_{\nu-1}$, which is one step prior to the moment in which an attacker catches more than half of the whole nodes. This application has established the enhanced blockchain-based IoV network architecture by bringing a theoretical stochastic model. The EBIoV aims to design an advanced secure IoT network architecture for protecting a connected car by adapting the BGG.

3.2. *Security Architecture of Smart Drone Swarm*

The advanced secured drone swarm network structure is introduced when the drones in a swarm are connected to one another and the swarm is hooked up as a single blockchain network [26]. Such a drone swarm (i.e., of smart drones) could operate their tasks artificially and independently despite their disconnection with the command center. It is noted that each drone randomly generates unique data and broadcasts these data to others. These random generations are equivalent to transactions in a typical blockchain network (see Figure 6).

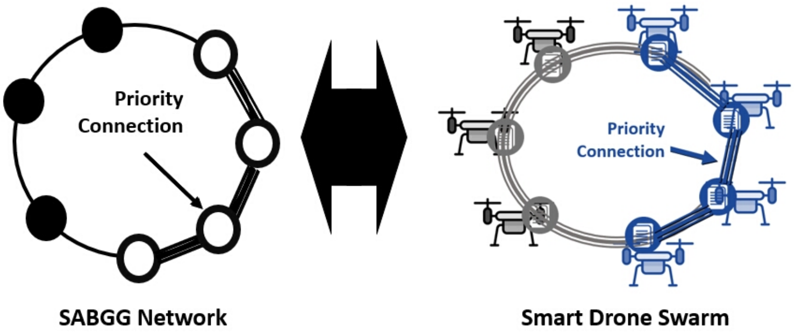

Figure 6. SABGG application for the drone swarm network architecture [26].

All drones in the swarm network shall have nearly equal chances of generating the blocks without or only with minimal computational power by applying the VRF on the Ethereum virtual machine (EVM). Although the VRF could be implemented on other blockchain network environments, the EVM has more flexibility in terms of implementing a new consensus mechanism including VRF and WMSR (the weighted-mean-subsequence-reduced) algorithms. Instead of VRF, the WMSR algorithm might be an alternative choice for achieving resilient consensus in decentralized sensor networks and smart robots [43–46]. The mechanism for protecting a smart drone swarm network is identical to the SABGG. The governance in a swarm network is driven by the decision-making parameters which include a prior time before catching more than half of the total drones by an attacker and the cost function could be defined as follows:

$$\mathfrak{S}(\varrho) = \Big(c(\varrho)\Big(1 - q^1_\eta\Big) + (c(\varrho) + V)q^1(\varrho)\Big)p_{A_{-1}} + V \cdot q^0(1 - p_{A_{-1}}), \tag{104}$$

where

$$p_{A_{-1}} \simeq 1 - \sum_{k=0}^{\frac{M}{2}} \left(\frac{\{\lambda_a(\widetilde{\gamma_0} + \mathbb{E}[\nu - 1]\widetilde{\gamma})\}^k}{k!} \cdot e^{-\lambda_a(\widetilde{\gamma_0} + \mathbb{E}[\nu-1]\widetilde{\gamma})} \right), \tag{105}$$

$$q^1(\varrho) = \sum_{j=0}^{\frac{M}{2}-1} \sum_{\{k \geq \frac{M}{2}+j\}} \left(\frac{\lambda_a\Big(\widetilde{\delta_0} + \mathbb{E}[\nu - 1]\widetilde{\delta}\Big)}{k!} \cdot e^{-\lambda_a\left(\widetilde{\delta_0} + \mathbb{E}[\nu-1]\widetilde{\delta}\right)} \right) P_j, \tag{106}$$

$$P_j = \binom{\frac{M}{2} - 1}{j} \varrho^j (1 - \varrho)^{\frac{M}{2} - 1 - j}. \tag{107}$$

The total cost $\mathfrak{S}(\varrho)$ shall be minimized by the given ϱ, which is the optimal value of alliance accountability (i.e., the acceptance rate) from (104). After the demonstration from previous research [26], the minimum cost of performing an operation of the intelligent drone swarm with a certain proportion of the acceptance rate for the alliance request to other drones could be analytically calculated. The moment of alliance request $\tau_{\nu-1}$ shall be one step prior to the time when an attacker catches more than half of the total drones. This practice aims to simulate smart drone management and its visualization of the results (see Figure 7).

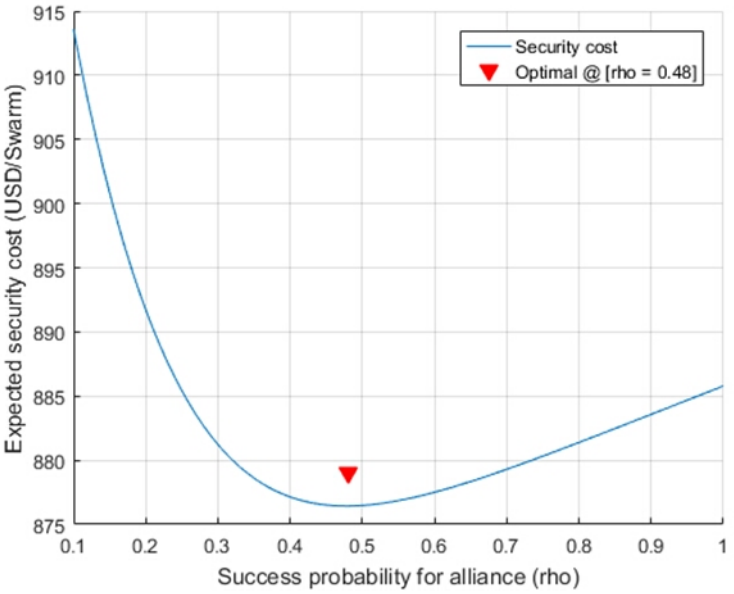

Figure 7. Optimization practices for the drone swarm security [26].

An advanced secure drone swarm network architecture protects a drone swarm from an attacker by adapting a Blockchain Governance Game variant. The SABGG, which has been analytically proven [15], was adapted for a decentralized network to improve the drone swarm security [26].

4. Conclusions and Future Research

This paper offers a comprehensive review on the BGGs and their flagship applications which have been actively studied recently. All BGG models are mathematically fully proven without any numerical approach. The great strength of the BGG models is that they provide analytically tractable solutions as explicit formula forms for determining the decision-making parameters to avoid major attacks by executing preliminary operations beforehand. The BGG models shall be extended to various blockchain-based cybersecurity areas including IoT security and a secured decentralized service network design. Their network architecture design could be widely applied in various application domains including cybersecurity, network architecture, service design and IT business models as long as a blockchain network is considered as their security enhancement. The innovative mathematical models are targeted to improve the security based on the network's architectural perspectives. Although all theoretically BGG models and major applications are mathematically proven, some challenges are as follows:

- **AI-enabled BGG model:** predicting the moment of attacks is always challenging and adapting machine learning techniques for forecasting could be considered to improve the BGG models.
- **Developing the applications for MLBGG:** the direct applications for MLBGG (multi-layered BGG) have not been found to date.
- **Actual implementations of BGG models:** implementing BGG models with the VRF on real blockchain networks is a challenging task. It is noted that the VRF shall be implemented on the Ethereum virtual machine before implementing the BGGs to see how these theorical models actually work.

All the above challenges shall be considered future research topics and anyone can freely invest in the above challenges as their research topics. This review is expected to be helpful in launching a new blockchain-based service with the improvement of network security based on BGG. It is predicted that more and more complex security threats will emerge, which will in turn require the development of more advanced defense techniques to detect and combat such threats. Therefore, it is foreseeable that ensuring the robustness of defenses against network attacks will become a priority and an industry standard.

Funding: This work has been supported in part by the Macao Polytechnic University (MPU) under Grant RP/FCA-04/2023.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: There are no available data to be stated.

Acknowledgments: Special thanks to all reviewers who provided valuable comments to make this paper complete.

Conflicts of Interest: The author declares no conflict of interest.

Appendix A. The Marginal Mean of the First Exceed Index under Memoryless Observation Process

Let us consider that the observation process has the memoryless property. This observation process is really practical for the actual implementations of BGG and SABGG models because it indicates that a defender (or a service provider) does not spend the additional cost of storing past information.

Appendix A.1. Memoryless BGG Model

To build the cost function of the BGG, we need to find the marginal mean of the first exceed index from (29) to determine the decision-making moment $\tau_{\nu-1}$, which could be found as follows:

$$\mathbb{E}[\tau_{\nu-1}] = \mathbb{E}[\tau_0] + \mathbb{E}[\Delta_1](\mathbb{E}[\nu] - 1). \tag{A1}$$

Recalling from (18), the operator $\mathcal{D}_a^q$ is determined as follows:

$$G(u) = (1-u)\sum_{x\geq 0} f(x)u^x, \tag{A2}$$

and

$$\mathcal{D}_{(x,y)}\left[f_1(x)f_2(y)\right](u,v) = \mathcal{D}_x[f_1(x)]\mathcal{D}_y\left[f_2(y)\right], \tag{A3}$$

then

$$f(x,y) = \mathfrak{D}_{(u,v)}^{(x,y)}\left[\mathcal{D}_{(x,y)}\left[f(x,y)\}\right], \tag{A4}$$

$$f_1(x)f_2(y) = \mathfrak{D}_u^x[\mathcal{D}_x\{f_1(x)\}]\mathfrak{D}_v^y[\mathcal{D}_y\{f_2(y)\}], \tag{A5}$$

where $\{f(x), (f_1(x)f_2(y))\}$ are a sequence, with the inverse (18) and

$$\mathfrak{D}_u^m(\bullet) = \begin{cases} \frac{1}{m!}\lim_{u\to 0}\frac{\partial^m}{\partial u^m}\frac{1}{(1-u)}(\bullet), & m\geq 0, \\ 0, & \text{otherwise}, \end{cases} \tag{A6}$$

and

$$\mathfrak{D}_{(u,v)}^{(m,n)}[G_1(u)G_2(v)] = \mathfrak{D}_u^m[G_1(u)]\mathfrak{D}_v^n[G_2(v)]. \tag{A7}$$

The functional $\mathfrak{D}$ is defined on the space of all analytic functions at 0 and it has the following properties:

- $\mathfrak{D}_u^m$ is a linear functional with fixed points at constant functions;
- $\mathfrak{D}_u^m\sum_{k=0}^{\infty}a_k u^k = \sum_{k=0}^{m}a_k$.

It is also noted that Formulas (20)–(25) could be rewritten as follows:

$$\gamma = \gamma_a \cdot \gamma_h := \gamma_a(g_0 g_1 u)\gamma_h(z_0 z_1 v), \tag{A8}$$

$$\gamma_0 = \gamma_a^0 \cdot \gamma_h^0 := \gamma_a^0(g_0 g_1 u)\gamma_h^0(z_0 z_1 v), \tag{A9}$$

$$\mathit{\Gamma} := \gamma_a(g_1 u)\gamma_h(z_1 v), \tag{A10}$$

$$\mathit{\Gamma}_0 := \gamma_a^0(g_1 u)\gamma_h^0(z_1 v), \tag{A11}$$

$$\mathit{\Gamma}^1 := \gamma_a(g_1)\gamma_h(z_1 v), \tag{A12}$$

$$\mathit{\Gamma}_0^1 := \gamma_a^0(g_1)\gamma_h^0(z_1 v). \tag{A13}$$

Recalling from (26), the formula is assigned as follows:

$$\mathbb{E}[\xi^\nu] = \mathit{\Phi}_{\lceil \frac{M}{2} \rceil}(\xi, 1, 1, 1, 1) = L^1 + L^2 - L^3, \tag{A14}$$

where

$$L^1 = \mathfrak{D}_{(u,v)}^{\left(\frac{M}{2}, \frac{M}{2}\right)}\left[\gamma_h^0(v) - \gamma_a^0(u)\gamma_h^0(v)\right], \tag{A15}$$

$$L^2 = \mathfrak{D}_{(u,v)}^{\left(\frac{M}{2}, \frac{M}{2}\right)}\left[\frac{\xi \cdot \gamma_a^0(u)\gamma_h^0(v)\gamma_h(v)}{1 - \xi\gamma_a(u)\gamma_h(v)}\right], \tag{A16}$$

$$L^3 = \mathfrak{D}_{(u,v)}^{\left(\frac{M}{2}, \frac{M}{2}\right)}\left[\frac{\xi \cdot \gamma_a^0(u)\gamma_h^0(v)\gamma_a(u)\gamma_h(v)}{1 - \xi\gamma_a(u)\gamma_h(v)}\right]. \tag{A17}$$

Since the observation process has memoryless properties, the process is exponentially distributed and the functionals from (9)–(11) are as follows:

$$\gamma_a^0(u) = \frac{1}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_a\right) - \widetilde{\delta}_0 \cdot \lambda_a u} = \frac{\beta_a^0}{1 - \alpha_a^0 \cdot u}, \tag{A18}$$

$$\gamma_a(u) = \frac{1}{\left(1 + \widetilde{\delta} \cdot \lambda_a\right) - \widetilde{\delta} \cdot \lambda_a u} = \frac{\beta_a}{1 - \alpha_a \cdot u}, \tag{A19}$$

$$\gamma_h^0(v) = \frac{1}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_h\right) - \widetilde{\delta}_0 \cdot \lambda_h v} = \frac{\beta_h^0}{1 - \alpha_h^0 \cdot v}, \tag{A20}$$

$$\gamma_h(v) = \frac{1}{\left(1 + \widetilde{\delta} \cdot \lambda_h\right) - \widetilde{\delta} \cdot \lambda_h v} = \frac{\beta_h}{1 - \alpha_h \cdot v}, \tag{A21}$$

$$\beta_a^0 = \frac{1}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_a\right)}, \; \alpha_a^0 = \frac{\widetilde{\delta}_0 \cdot \lambda_a}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_a\right)}, \tag{A22}$$

$$\beta_a = \frac{1}{\left(1 + \widetilde{\delta} \cdot \lambda_a\right)}, \; \alpha_a = \frac{\widetilde{\delta}_0 \cdot \lambda_a}{\left(1 + \widetilde{\delta} \cdot \lambda_a\right)}, \tag{A23}$$

$$\beta_h^0 = \frac{1}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_h\right)}, \; \alpha_h^0 = \frac{\widetilde{\delta}_0 \cdot \lambda_h}{\left(1 + \widetilde{\delta}_0 \cdot \lambda_h\right)}, \tag{A24}$$

$$\beta_h = \frac{1}{\left(1 + \widetilde{\delta} \cdot \lambda_h\right)}, \; \alpha_h = \frac{\widetilde{\delta}_0 \cdot \lambda_h}{\left(1 + \widetilde{\delta} \cdot \lambda_h\right)}, \tag{A25}$$

where $\widetilde{\delta}_0 = \mathbb{E}[\tau_0]$ and $\widetilde{\delta} = \mathbb{E}[\Delta_k]$. From (A15), we have

$$L^1 = \beta_h^0\left[\frac{1 - \left(\alpha_h^0\right)^{\frac{M}{2}+1}}{1 - \left(\alpha_h^0\right)}\right]\left(1 - \beta_a^0\left[\frac{1 - \left(\alpha_a^0\right)^{\frac{M}{2}+1}}{1 - \left(\alpha_a^0\right)}\right]\right),$$

and from (A16),

$$L^2 = \sum_{n\geq 0} \xi^{n+1} \left\{ \left(\beta_a^0 \cdot (\beta_a)^n \right) \cdot \sum_{j=0}^{\frac{M}{2}} \left\{ (-1)^j \left(\alpha_a \alpha_a^0 \right)^j \psi_{n-1}^a(j) \right\} \right\}$$
$$\cdot \left\{ \left(\beta_h^0 \cdot (\beta_h)^{n+1} \right) \cdot \sum_{k=0}^{\frac{M}{2}} \left\{ (-1)^k \left(\alpha_h \alpha_H^0 \right)^k \psi_n^h(k) \right\} \right\},$$

and from (A17),

$$L^3 = \sum_{n\geq 0} \xi^{n+1} \left\{ \left(\beta_a^0 \cdot (\beta_a)^{n+1} \right) \cdot \sum_{j=0}^{\frac{M}{2}} \left\{ (-1)^j \left(\alpha_a \alpha_a^0 \right)^j \psi_n^a(j) \right\} \right\}$$
$$\cdot \left\{ \left(\beta_h^0 \cdot (\beta_h)^{n+1} \right) \cdot \sum_{k=0}^{\frac{M}{2}} \left\{ (-1)^k \left(\alpha_h \alpha_h^0 \right)^k \psi_n^h(k) \right\} \right\},$$

where

$$\psi_n^a(j) = \left(\sum_{i=0}^{j} \binom{n+i}{i} (-1)^i \left(\frac{\alpha_a}{\alpha_a^0} \right)^i \right), \tag{A26}$$

$$\psi_n^h(k) = \left(\sum_{i=0}^{k} \binom{n+i}{i} (-1)^i \left(\frac{\alpha_h}{\alpha_h^0} \right)^i \right). \tag{A27}$$

From (29) and (A14)–(A17), the mean of the first exceed index is determined as follows:

$$\mathbb{E}[\nu] = \left(\beta_a^0 \beta_h^0 \beta_h \right) \sum_{n\geq 0} (n) \left[(\beta_a \beta_h)^n \Xi_{n-1}^h \left(\frac{M}{2} \right) \left\{ \Xi_{n-2}^a \left(\frac{M}{2} \right) - \Xi_{n-1}^a \left(\frac{M}{2} \right) \beta_a \right\} \right], \tag{A28}$$

where

$$\Xi_n^a(m) = \sum_{j=0}^{m} \left\{ (-1)^j \left(\alpha_a \alpha_a^0 \right)^j \psi_n^a(j) \right\}, \tag{A29}$$

$$\Xi_n^h(m) = \sum_{k=0}^{m} \left\{ (-1)^k \left(\alpha_h \alpha_h^0 \right)^k \psi_n^h(k) \right\}. \tag{A30}$$

Appendix A.2. Memoryless SABGG Model

To build the cost function of the SABGG, we need to find the marginal mean of the first exceed index from (65) to determine the marginal mean of $t_{\nu-1}$, which could be found as follows:

$$\mathbb{E}[t_{\nu-1}] = \mathbb{E}[t_0] + \mathbb{E}[U_1](\mathbb{E}[\nu] - 1). \tag{A31}$$

From (A2), the $\mathcal{D}$- and $\mathfrak{D}$-operators could be operated as follows:

$$\mathcal{D}_{(x,y,z)} \left[f_1(x) f_2(y) f_3(z) \right](u,v) = \mathcal{D}_x[f_1(x)] \mathcal{D}_y \left[f_2(y) \right] \mathcal{D}_z \left[f_3(z) \right], \tag{A32}$$

and

$$\mathfrak{D}_{(u,v,w)}^{(m,n,r)} [G_1(u) G_2(v) G_3(w)] = \mathfrak{D}_u^m [G_1(u)] \mathfrak{D}_v^n [G_2(v)] \mathfrak{D}_w^r [G_3(w)]. \tag{A33}$$

then we have

$$f(x,y,z) = \mathfrak{D}^{(x,y,z)}_{(u,v,w)}\left[\mathcal{D}_{(x,y,z)}\left[f(x,y,z)\right\}\right], \tag{A34}$$

$$f_1(x)f_2(y)f_3(z) = \mathfrak{D}^{x}_{u}[\mathcal{D}_x\{f_1(x)\}]\mathfrak{D}^{y}_{v}[\mathcal{D}_y\{f_2(y)\}]\mathfrak{D}^{z}_{w}[\mathcal{D}_z\{f_3(z)\}], \tag{A35}$$

where $\{f(x),(f_1(x)f_2(y)f_3(z))\}$ are a sequence, with the inverse (18). Recalling from (65), the formula is assigned as follows:

$$\mathbb{E}[\zeta^{\nu}] = \Theta_{\lceil \frac{M}{2} \rceil}(\zeta,1,1,1,1,1) = R^1 + R^2 - R^3, \tag{A36}$$

where

$$R^1 = \left\{\frac{a_g b_c b_g}{1-b_c b_g}\right\}\left\{\Xi_{\frac{M}{2}}(0)\right\}\left(1 - b_c + b_c\left(\sum_{l\geq 0}^{\frac{M}{2}}\left(a^0_c\right)^l\right)\right)$$
$$-\left\{\Xi_{\frac{M}{2}}(0)\right\}\left(\frac{b^0_c}{a_g}\right)\left\{\sum_{k\geq 0}^{\frac{M}{2}}\left(1+\left(\frac{a_g b_c b_g}{1-b_c b_g}\right)\right)^{k+1}\right\}\left(\sum_{l\geq 0}^{\frac{M}{2}}\left(a^0_c\right)^l\right),$$

$$R^2 = \left(\frac{\zeta b^0_c b_c b^0_g}{a^0_c\{1-b_c b_g - a_c\} - a_g(a_c+1)}\right)$$
$$\cdot\sum_{l\geq 0}\left[\left(a^0_g\right)^l\left\{\Xi^{\frac{M}{2}-l}(\zeta) - (a_g)\Xi^{\frac{M}{2}-l-1}(\zeta)\right\}\left\{\sum_{j=l}\binom{j}{l}\left(\frac{1}{a^0_g}\right)^{j+1}\right\}\right],$$

$$R^3 = \left\{\frac{\zeta b_c\left(b^0_c\right)^2}{\{1-b_c b_g - a_c\} - a_g(a_c+1)}\right\}$$
$$\cdot\sum_{h\geq 0}\binom{k}{h}\left(a^0_g\right)^{h-1}\left\{\sum_{k=h}\left(\frac{a^0_c}{a^0_g}\right)^k\right\}\left[\Xi^{\frac{M}{2}-h}(\zeta) - (a_g)\Xi^{\frac{M}{2}-h-1}(\zeta)\right]$$
$$+\left(\frac{\zeta b^0_c b^0_g (b_c)^2}{a^0_c(1-a_c)}\right)\left(\frac{1}{(1-b_c b_g) - a_c - a_g(1-a_c)}\right)$$
$$\cdot\sum_{h\geq 0}\binom{k}{h}\left(a^0_g\right)^h\sum_{k\geq h}\left(\frac{a^0_c}{a^0_g}\right)^{k+1}\left[\Xi^{\frac{M}{2}-h}(\zeta) - (a_g)^k\Xi^{\frac{M}{2}-h-1}(\zeta)\right],$$

and

$$\Xi_{\frac{M}{2}}(0) := \left\{\sum_{u=0}^{m}\left(\frac{\left(\frac{M}{2}\right)!}{\left(\left(\frac{M}{2}\right)-u\right)!}\right)\prod_{h=1}^{\left(\frac{M}{2}\right)}\left(\frac{h!}{1-\left(\frac{a_c}{1-b_c b_g}\right)}\right)\right\}, \tag{A37}$$

$$\Xi^m(\zeta) = \left\{\sum_{u=0}^{m}\left(\frac{m!}{(m-u)!}\right)\prod_{l=1}^{m}\left(\frac{l!}{1-\left(\frac{a^0_g}{1-\zeta b^0_c b^0_g}\right)}\right)\right\}, \tag{A38}$$

and

$$b_c^0 = \frac{1}{(1+\widetilde{\alpha_0}\cdot\lambda_c)},\ a_c^0 = \frac{\widetilde{\alpha_0}\cdot\lambda_c}{(1+\widetilde{\alpha_0}\cdot\lambda_c)}, \tag{A39}$$

$$b_c = \frac{1}{(1+\widetilde{\alpha}\cdot\lambda_c)},\ a_c = \frac{\widetilde{\alpha}\cdot\lambda_c}{(1+\widetilde{\alpha}\cdot\lambda_c)}, \tag{A40}$$

$$b_g^0 = \frac{1}{(1+\widetilde{\alpha_0}\cdot\lambda_g)},\ a_g^0 = \frac{\widetilde{\alpha_0}\cdot\lambda_g}{(1+\widetilde{\alpha_0}\cdot\lambda_g)}, \tag{A41}$$

$$b_g = \frac{1}{(1+\widetilde{\alpha}\cdot\lambda_g)},\ a_g = \frac{\widetilde{\alpha}\cdot\lambda_g}{(1+\widetilde{\alpha}\cdot\lambda_g)}, \tag{A42}$$

where $\widetilde{\alpha_0} = \mathbb{E}[t_0]$ and $\widetilde{\alpha} = \mathbb{E}[U_k]$ from (41). The marginal mean of the first exceed index $\mathbb{E}[\nu]$ could be followed from (65) and (A36)–(A38).

References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 1 December 2021).
2. Beikverdi, A.; Song, J. Trend of centralization in Bitcoin's distributed network. In Proceedings of the 2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Takamatsu, Japan, 1–3 June 2015; pp. 1–6.
3. Yli-Huumo, I.; Ko, D.; Choi, S.; Park, S.; Smolander, K. Where Is Current Research on Blockchain Technology? A Systematic Review. *PLoS ONE* **2016**, *11*, e0163477. [CrossRef] [PubMed]
4. Liu, Z., Luong, N.C.; Wang, W.; Niyato, D.; Wang, P.; Liang, Y.-C.; Kim, D.I. A Survey on Applications of Game Theory in Blockchain (2019). Available online: https://arxiv.org/abs/1902.10865 (accessed on 1 May 2019).
5. Decker, C.; Wattenhofer, R. Information propagation in the Bitcoin network. In Proceedings of the IEEE P2P 2013 Proceedings, Trento, Italy, 9–11 September 2013; pp. 1–10.
6. Kim, W. Bitcoin, Blockchain Mechanism and Its Evolution. 2018. Available online: http://www.itfind.or.kr/publication/ (accessed on 1 December 2021). (In Korean)
7. Narayanan, A.; Clar, J. Bitcoin's Academic Pedigree. *Mag. Commun. ACM* **2017**, *60*, 36–45. [CrossRef]
8. Weiss, M.; Corsi, E. Bitfury: Blockchain for Government. *HBP Case* **2018**, *12*, 818-031.
9. Armknecht, F.; Karame, G.O.; Mandal, A.; Youssef, F.; Zenner, E. Ripple: Overview and Outlook. In *Trust and Trustworthy Computing: Proceedings of the 8th International Conference, Heraklion, Greece, 24–26 August 2015*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 9229, pp. 163–180.
10. Bhuiyan, B. An Overview of Game Theory and Some Applications. *Philos. Prog.* **2016**, *59*, 111–128. Available online: https://www.banglajol.info/index.php/PP/article/view/36683 (accessed on 1 May 2019). [CrossRef]
11. Antonopoulos, A.M. *Mastering Bitcoin: Programming the Open Blockchain*, 2nd ed.; O'Reilly: Sebastopol, CA, USA, 2017.
12. Eyal, I.; Sirer, E. *Majority Is Not Enough: Bitcoin Mining Is Vulnerable*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8437, pp. 436–454.
13. Garay, J.; Kiayias, A.; Leonardos, N. *The Bitcoin Backbone Protocol: Analysis and Applications*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2015; Volume 9057, pp. 281–310.
14. Kim, S.-K. Blockchain Governance Game. *Comput. Ind. Eng.* **2019**, *136*, 373–380. [CrossRef]
15. Kim, S.-K. Strategic Alliance for Blockchain Governance Game. *Probab. Eng. Inf. Sci.* **2020**, *36*, 184–200. [CrossRef]
16. Kim, S.-K. Multi-Layered Blockchain Governance Game. *Axioms* **2022**, *11*, 109. [CrossRef]
17. Kang, J.; Xiong, Z.; Niyato, D.; Ye, D.; Kim, D.I.; Zhao, J. Toward Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2906–2920. [CrossRef]
18. Lohachab, A.; Garg, S.; Kang, B.; Amin, M.B.; Lee, J.; Chen, S.; Xu, X. Towards Interconnected Blockchains: A Comprehensive Review of the Role of Interoperability among Disparate Blockchains. *ACM Comput. Surv.* **2021**, *54*, 1–39. [CrossRef]
19. Erfan, F.; Bellaiche, M.; Halabi, T. Game-theoretic Designs for Blockchain-based IoT: Taxonomy and Research Directions. In Proceedings of the 2022 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS), Newark, CA, USA, 15–18 August 2022; pp. 27–37.
20. Liu, Y.; Lu, Q.; Zhu, L.; Paik, H.Y.; Staples, M. A systematic literature review on blockchain governance. *J. Syst. Softw.* **2023**, *197*, 111576. [CrossRef]
21. Alkadi, R.; Alnuaimi, N.; Yeun, C.Y.; Shoufan, A. Blockchain Interoperability in Unmanned Aerial Vehicles Networks: State-of-the-Art and Open Issues. *IEEE Access* **2022**, *10*, 14463–14479. [CrossRef]
22. Liu, Z.; Luong, N.C.; Wang, W.; Niyato, D.; Wang, P.; Liang, Y.C.; Kim, D.I. A Survey on Blockchain: A Game Theoretical Perspective. *IEEE Access* **2019**, *7*, 47615–47643. [CrossRef]

23. Yang, X.; Gong, G.; Tian, Y. Optimal Game Theory in Complicated Virtual-modeling and CGF Decision-making with Multi-granularities. In Proceedings of the 2008 International Conference on Smart Manufacturing Application, Goyangi, Republic of Korea, 9–11 April 2008; pp. 95–99.
24. Yang, X.; Gong, G.; Tian, Y. Generalized Optimal Game Theory in virtual decision-makings. In Proceedings of the 2008 Chinese Control and Decision Conference, Yantai, China, 2–4 July 2008; pp. 196–1964.
25. Kim, S.-K. Enhanced IoV Security Network by Using Blockchain Governance Game. *Mathematics* **2021**, *9*, 109. [CrossRef]
26. Kim, S.-K. Advanced Drone Swarm Security by Using Blockchain Governance Game. *Mathematics* **2022**, *10*, 3338. [CrossRef]
27. Ramirez, M.A.; Kim, S.-K.; Al Hamadi, H.; Damiani, E.; Byon, Y.-J.; Kim, Y.-J.; Cho, C.-S.; Yeun, C.Y. Poisoning Attacks and Defenses on Artificial Intelligence: A Survey. 2022. Available online: https://arxiv.org/abs/2202.10276 (accessed on 1 April 2023).
28. Biggio, B.; Corona, I.; Maiorca, D.; Nelson, B.; Šrndić, N.; Laskov, P.; Giacinto, G.; Roli, F. Evasion attacks against machine learning at test time. *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases* **2013**, *8190*, 387–402.
29. Paudice, A.; Muñoz-González, L.; Gyorgy, A.; Lupu, E.C. Detection of adversarial training examples in poisoning attacks through anomaly detection. *arXiv* **2018**, arXiv:1802.03041.
30. Liu, X.; Xie, L.; Wang, Y.; Zou, J.; Xiong, J.; Ying, Z.; Vasilakos, A.V. Privacy and Security Issues in Deep Learning: A Survey. *IEEE Access* **2020**, *9*, 4566–4593. [CrossRef]
31. Xue, M.; Yuan, C.; Wu, H.; Zhang, Y.; Liu, W. Machine Learning Security: Threats, Countermeasures, and Evaluations. *IEEE Access* **2020**, *8*, 4720–74742. [CrossRef]
32. Qi, X.; Zeyi, T.; Zijiang, H.; Qun, L. FABA: an algorithm for fast aggregation against byzantine attacks in distributed neural networks. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; pp. 4824–4830.
33. Wang, B.; Gong, N.Z. Stealing hyperparameters in machine learning. In Proceedings of the IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 20–24 May 2018; pp. 36–52.
34. Tramer, F.; Zhang, F.; Juels, A.; Reiter, M.K.; Ristenpart, T. Stealing machine learning models via prediction apis. In Proceedings of the 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, 11 August 2016; pp. 601–618.
35. Juuti, M.; Szyller, S.; Marchal, S.; Asokan, N. PRADA: protecting against DNN model stealing attacks. *arXiv* **2018**, arXiv:1805.02628.
36. Chen, Z.; Lv, N.; Liu, P.; Fang, Y.; Chen, K.; Pan, W. Intrusion Detection for Wireless Edge Networks Based on Federated Learning. *IEEE Access* **2020**, *8*, 217463–217472. [CrossRef]
37. Chakarov, A.; Nori, A.; Rajamani, S.; Sen, S.; Vijaykeerthy, D. Debugging machine learning tasks. *arXiv* **2016**, arXiv:1603.07292v1.
38. Kleinrock, L. *Queueing Systems, Volume 1: Theory*; Wiley-Interscience: New York, NY, USA, 1975.
39. Kim, Y.; Lim, H. Multi-Agent Reinforcement Learning-Based Resource Management for End-to-End Network Slicing. *IEEE Access* **2021**, *9*, 56178–56190. [CrossRef]
40. Dshalalow, J.H. *First Excess Level Process, Advances in Queueing*; CRC Press: Boca Raton, FL, USA, 1995; pp. 244–261.
41. Dshalalow, J.H.; Ke, H.-J. Layers of noncooperative games. *Nonlinear Anal.* **2009**, *71*, 283–291. [CrossRef]
42. Whittington, R.; Regnér, P.; Angwin D.; Johnson, G.; Scholes, K. *Exploring Strategy Text and Cases*, 11th ed.; Pearson: Harlow, UK, 2017.
43. Dorri, A.; Steger, M.; Kanhere, S.S.; Jurdak, R. BlockChain: A Distributed Solution to Automotive Security and Privacy. *IEEE Commun. Mag.* **2017**, *55*, 119–125. [CrossRef]
44. Baker, J. Edge Computing–The New Frontier of the Web. 2017. Available online: https://hackernoon.com/edge-computing-a-beginners-guide-8976b6886481 (accessed on 1 December 2021).
45. ERPINNEW. Fog Computing vs. Edge Computing. 2017. Available online: https://erpinnews.com/fog-computing-vs-edge-computing (accessed on 1 May 2019).
46. Cisco Networking Academy. *Connecting Networks Companion Guide*; Cisco Press: Indianapolis, IN, USA, 2014.
47. Rouse, M. Internet of Vehicles. 2018. Available online: https://whatis.techtarget.com/definition/Internet-of-Vehicles (accessed on 1 May 2019).
48. Kim S.; Shrestha, R. Internet of Vehicles, Vehicular Social Networks, and Cybersecurity. In *Automotive Cyber Security*; Springer: Singapore, 2020.
49. Micali, S.; Vadhan, S.; Rabin, M. Verifiable random functions. In Proceedings of the 40th IEEE Symposium on Foundations of Computer Science, New York, NY, USA, 17–19 October 1999; pp. 120–130.
50. Dodis, Y.; Yampolskiy, A. A Verifiable Random Function with Short Proofs and Keys. *Lect. Notes Comput. Sci.* **2005**, *3386*, 416–431.
51. Gorbunov, S. Algorand Releases First Open-Source Code: Verifiable Random Function. 2018. Available online: https://medium.com/algorand/ (accessed on 1 May 2019).
52. Zhao, W. MIT Professor's Blockchain Protocol Nets 62 Million in New Funding. 2018. Available online: https://www.coindesk.com/mit-professors-Blockchain-protocol-nets-62-million-in-new-funding (accessed on 1 May 2019).
53. Kihei, B.; Copeland, J.A.; Chang, Y. Automotive Doppler sensing: The Doppler profile with machine learning in vehicle-to-vehicle networks for road safety. In Proceedings of the 2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Sapporo, Japan, 3–6 July 2017; pp. 1–5.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.