Game Theoretical Framework for Analyzing Blockchains Robustness

Paolo Zappalà
Orange Labs, 92320 Chatillon, France
LIA, Avignon Université, 84029 Avignon, France

Marianna Belotti
BDTD 60, Caisse des Dépôts, 75013 Paris, France
Cedric, Cnam, 75003 Paris, France

Maria Potop-Butucaru
Lip6, CNRS UMR 7606, Sorbonne University, 75005 Paris, France

Stefano Secci
Cedric, Cnam, 75003 Paris, France

Abstract

In this paper we propose a game theoretical framework in order to formally characterize the robustness of blockchains systems in terms of resilience to rational deviations and immunity to Byzantine behaviors. Our framework includes necessary and sufficient conditions for checking the immunity and resilience of games and an original technique for composing games that preserves the robustness of individual games. We prove the practical interest of our formal framework by characterizing the robustness of various blockchain protocols: Bitcoin (the most popular permissionless blockchain), Tendermint (the first permissioned blockchain used by the practitioners), Lightning Network, a side-chain protocol and a cross-chain swap protocol. For each one of the studied protocols we identify upper and lower bounds with respect to their resilience and immunity (expressed as no worse payoff than the initial state) face to rational and Byzantine behaviors.

2012 ACM Subject Classification Networks

Keywords and phrases Blockchain protocols, Distributed algorithms, Game-theoretical modeling, Fault tolerance, Failure robustness

Digital Object Identifier 10.4230/LIPIcs.DISC.2021.42

Related Version *Full Version*: `https://hal.archives-ouvertes.fr/hal-02634752/document`

1 Introduction

Distributed Ledger Technologies (DLTs) allow sharing a ledger of transactions among multiple users forming a peer-to-peer (P2P) network. DLTs characterized by a block architecture are called "blockchains". They enable users to transfer cryptoassets in a decentralized manner by means of modular protocols adopted by the users themselves. Beyond the traditional blockchain architectures (*layer-1 protocols*), the literature proposes other protocols that respectively define and regulate interactions in an overlaying network (*layer-2 protocols*) and interactions between different blockchains (*cross-chain protocols*). Each of these protocols establishes the instructions users must follow in order to interact with or through a blockchain. In a blockchain system players can be classified, as proposed in [2] for classical distributed systems, in three different categories: (i) players who follow the prescribed protocol i.e., *altruistic*, (ii) those who act in order to maximise their own benefit i.e., *rational*, and (iii) players who may deviate arbitrarily from the prescribed protocol, i.e. *Byzantine* (cf. [18]). Interactions among users are usually modeled with game theory which analyzes the decision-making process in presence of multiple rational agents, called *players* or *agents*.

In the context of blockchain systems, game theoretical frameworks were introduced in [36, 37] to analyze security aspects and incentive compatibility of Nakamoto's consensus protocol (i.e., Proof-of-Work [25]) characterizing the very first blockchain implementation known as Bitcoin. Users participating in the consensus mechanism (i.e., miners) are considered as individually rational moved by the mere intention to increase their revenues i.e., the rewards earned form the mining activities [7, 32]. Authors in [8, 13, 35, 12] adopt different utility functions for miners and pools that consider costs and relative rewards. Concerning layer-2 and cross-chain protocols, game theoretical analysis are carried out by [5, 6, 15]. These analyses are strictly specific to the particular deployment context rather than to a generic blockchain. Most of the game theoretical models adopted to design secure and robust blockchain protocols, surveyed in [23], (i) address protocols characterizing specific blockchain implementations, (ii) analyze miners' behaviours in the consensus phase and (iii) adopt Nash Equilibria as solution concept.

Concerning *rational agents*, the existing analyses include the study of equilibria and the evaluation of their properties. The most studied and adopted solution concept in literature is the Nash Equilibrium, i.e., a strategy profile in which no player has interest in individually deviating from her own strategy. A first approach to the analysis of robustness is to compare Nash Equilibria, through indices such *Price of Byzantine Anarchy* [24], *Price of Malice* [24] and *Price of Anarchy* [20]. This approach summarizes the outcomes of the games representing protocols, but it does not show explicitly the implementation risks of such systems. A second approach is to analyze peculiar Nash Equilibria. Authors of [28] take probability into account and extend the concept of Nash Equilibrium. In [18], *virtual utility* – alternative to the classical game utility – is introduced to capture the blockchain agreement structure. The analysis of robustness with respect to *Byzantine agents* was modeled in [3] with a Bayesian game. The authors provide the analysis of Tendermint protocol [22]. This method allows making forecasts on the expected outcomes of a game, but it does not provide a comprehensive analysis of the risks. It should be noted that none of the previous works is generic enough to propose a methodology for analyzing the robustness of blockchain protocols to both rational and Byzantine players.

The first generic framework for analyzing the *robustness of distributed protocols* with respect to the behavior of rational and Byzantine players was proposed by the authors of [1] who introduced the concept of *mechanism* (i.e., a pair game-prescribed strategy). Moreover in [1] authors introduced the notions of (i) k-resilience, (ii) practicality and (iii) t-immunity. A strategy profile is defined as *k-resilient* if there is no coalition with at most k players having an incentive to deviate from the prescribed protocol. The category of *practical* strategy profiles is defined when equilibria with weakly dominated strategies are excluded. Finally, *t-immunity* denotes a situation where no player gets a lower outcome if there are at most t Byzantine players that can play any possible strategy. Interestingly, despite its mathematical beauty this framework was never used to analyze the robustness of blockchain protocols.

Our contribution. In this paper we follow the line of work opened in [1] and present a game theoretical framework aiming at characterizing the *robustness of blockchain protocols*. Our contributions can be summarized as follows: (a) we prove that t-immunity property defined in [1] is not verified by a large class of blockchain protocols (cf. Table 1). It should be noted that the authors of [1] already observed that "t-immunity is often impossible to be satisfied by practical systems" and left open the definition of a weaker property; (b) we introduce the new concept of *t-weak-immunity*; a mechanism is t-weak-immune if any altruistic player receives no worse payoff than the initial state, no matter how any set of

t players deviate from the prescribed protocol. This new concept is sufficiently strong to capture the robustness of a large class of blockchain protocols (cf. Table 1); (c) we identify and prove necessary and sufficient conditions for a mechanism to be k-resilient and t-weak-immune; (d) we define a new operator for game composition and prove that it preserves the robustness properties of the individual games; (e) we use our generic framework and the composition operator we study the robustness of a representative set of layer-1, layer-2 and cross-chain protocols: Tendermint [22], Bitcoin [25], Lightning Network protocol [30], the side-chain protocol [29] and the very first implementation of a cross-chain swap protocol proposed in [27] and formalized in [15].

For each one of the analyzed protocols we provide bounds on the number of Byzantine processes in order to verify t-weak immunity. Furthermore, for the same class of protocols we compute bounds on the number of rational processes in order to achieve k-resilience. Our results are reported in Table 1. Interestingly, our analysis allowed us to spot the weakness of the Lightning Network protocol [30] to Byzantine behaviour. Therefore, we propose and further analyze an alternative version of the protocol.

The paper is structured as follows. Section 2 is devoted to the definition of mechanism, (k, t)-robustness, necessary and sufficient conditions for optimal resilience and weak immunity and, composition of mechanisms. We apply in Section 3 the methodology developed in Section 2 to prove the robustness of the protocols presented in [25, 22, 30, 29, 27]. Section 4 concludes the paper. All the illustrations of the models as well as all the proofs for the results presented in this paper are available at [38].

Table 1 Immunity and resilience properties for Tendermint [22], Bitcoin [25], Lightning Network [30], a side-chain protocol [29] and a cross-chain swap protocol [27, 15] with respect to the number of rational deviating agents (k) and the number of Byzantine deviating agents (t) where n is the total number of players in the game.

Protocol	k-resilience	t-immunity	t-weak immunity	Results
Tendermint	**Yes**, k $< n/3$	**No**	**Yes**, t $< n/3$	Thm. 8
Bitcoin	**Yes**, k $< 3n/20$	**No**	**No**	Thm. 10
Lightning Network	**Yes**, k $< 3n/20$	**No**	**No**	Thm. 12
Closing module	Yes	No	No	Thm. 17
(Alternative closing module)	(Yes)	(No)	(Yes)	Thm. 18
Other modules	Yes	No	Yes	Thm. 14, 15, 20, 22, 23
Side-chain (Platypus)	**Yes**, k $< n/3$	**No**	**Yes**, t $< n/3$	Thm. 25
Cross-chain Swap	**Yes**	**No**	**Yes**	Thm. 28

2 Games theoretical framework for Analyzing protocols robustness

2.1 Preliminaries on Game Theory

The basic idea of a game is to capture a set of players which may act sequentially or simultaneously (cf. [21] for more details). The theoretical concept adopted in this paper is the one of *extensive form game*. A game of this type is represented formally by a tuple $\Gamma = \langle N, T, P, (A_h)_{h\in V}, (u_i)_{i\in N}\rangle$ where N is the set of players, $T = (V, E)$ is a directed rooted tree, $Z \subset V$ is the set of terminal nodes, $P : V \setminus Z \to N$ is a function assigning to each non-end node a player in N, $A_h = \{(x_h, x_i) \in E\}$ for each node $h \in V \setminus Z$ is the set of edges going from node h to some other nodes and represents the set of actions at node h of the tree T, $\Omega_i = \{s_i : V \setminus Z \to A_1 \times A_2 \times \ldots A_h \times \cdots \times A_H, h : P(h) = i\}$ is the set of pure strategies

of player i, $\mathcal{S}_i = \{\sigma_i : \Omega_i \to [0,1], \sum_{s\in\Omega_i} \sigma_i(s) = 1\}$ is the set of mixed strategies of player i and $u_i : Z \to \mathbb{R}$ is the utility function for player $i \in N$. Every pure strategy of player i is a function that assigns an action $a \in A_h$ to every node $h \in V \setminus Z$ in which player i is involved (formally, $h : P(h) = i$). A mixed strategy is a probability distribution over the set of pure strategies of player i. For the sake of simplicity in the notation, analyses and proofs will involve pure strategies only, as the results can be easily generalized then for general mixed strategies. Every game in extensive form can be reformulated in a more compact way (i.e., *normal form*) with a tuple $\Gamma = \langle N, \mathcal{S}, u\rangle$, in which the set of players $N = \{1, \ldots, n\}$ denotes the players involved in the protocol, $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_n$ where $\mathcal{S}_i$ is the set of strategies of player i and $u : \mathcal{S} \to \mathbb{R}^n$ is the utility function of the players. Each player can pick her own strategy $\sigma_i \in \mathcal{S}_i$ generating a strategy profile $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_i, \ldots, \sigma_n) \in \mathcal{S}$ and a utility vector $u(\sigma)$ whose i-th component denotes the utility for player i.

A *solution concept* $\sigma \in \mathcal{S}$ is a strategy profile such that the outcome $u(\sigma)$ pleases every player so that they have no incentive in changing their strategy σ_i. The most known solution concept is the *Nash Equilibrium*, where no player has an incentive to unilaterally change strategy [26]. Formally, a strategy profile σ is a Nash Equilibrium if $u_i(\sigma_1, \sigma_2, \ldots, \sigma_i, \ldots, \sigma_n) \geq u_i(\sigma_1, \sigma_2, \ldots, \tau_i, \ldots, \sigma_n)$ for every player i and for every $\tau_i \in \mathcal{S}_i$. Nash [26] proved that every game in normal form admits at least one Nash Equilibrium. A Nash Equilibrium $\sigma \in \mathcal{S}$ is said to be *strong* if and only if for all $C \subseteq N$ and all $\tau_C \in \mathcal{S}_C$, there exists $i \in C$ such that $u_i(\sigma_C, \sigma_{-C}) \geq u_i(\tau_C, \sigma_{-C})$ i.e., given the strategy of its complements as given, no coalition can deviate in a way that benefits all of its members [9]. Strong Nash Equilibria are easy to be identified but they do not always exist. A Nash Equilibrium is said to be *stable* if it is still a Nash Equilibrium after small deviations in the game [16]. Moreover, as stated in [19], there always exists a stable Nash Equilibrium. Stable Nash Equilibria survive after the iterated deletion of *weakly dominated strategies*, i.e., those strategies $\sigma_i \in \mathcal{S}_i$ that perform as well as or worse than another strategy $\sigma'_i \in \mathcal{S}_i$ no matter which strategy the other players choose (formally, we have that $u_i(\sigma_i, \tau_{-i}) \leq u_i(\sigma'_i, \tau_{-i})$ for all $\tau_{-i} \in \mathcal{S}_{-i}$). In the process of *iterated deletion*, weakly dominated strategies are excluded from the set of strategies available to players and the set of Nash Equilibria is recomputed.

In the paper we analyze protocols modeled as games by studying the strategy profile associated to rational and Byzantine players; we identify the Nash Equilibria of the games and assign some properties to the respective strategy profiles.

2.2 Mechanisms and Robustness

The paper analyzes blockchain protocols in which players can either decide to follow or not the prescribed instructions. The aim of the paper is to model these problems and understand whether the players are incentivized to follow or deviate from the prescribed protocol being respectively altruistic or Byzantine agents. In the following (i) we recall and extend the game theoretical framework based on the concept of mechanism and its properties, (ii) we define new properties on protocol robustness and (iii) we study properties interdependence.

Let us consider a game in normal form $\Gamma = \langle N, \mathcal{S}, u\rangle$ where players find themselves in an initial state, i.e., before starting the application of the protocol. We assign $u_i(\sigma) = 0$ for every $\sigma \in \mathcal{S}$ when the player i is indifferent between the outcome of the strategy profile σ and the initial state one. Analogously, we assign positive utility, $u_i(\sigma) > 0$, when the outcome of σ corresponds to the final state provided by the protocol and negative utility, $u_i(\sigma) < 0$, when the outcome of σ is worse than the initial state one. The values of u_i, for all $i \in N$, correspond to the marginal utility with respect to the initial state. Every decision-making problem is modeled by a game $\Gamma = \langle N, \mathcal{S}, u\rangle$, which shows all the possible strategies available

to the players, including following the prescribed protocol and all its possible deviations. A specific protocol consists of a strategy profile $\sigma = (\sigma_1, \ldots, \sigma_n) \in \mathcal{S}$ and it is denoted by a pair (Γ, σ), called *mechanism* [1]. Every player i is advised to play strategy $\sigma_i \in \mathcal{S}_i$ i.e., the recommended strategy σ is the prescribed protocol. Evaluating the robustness to deviations of a distributed protocol corresponds to identifying the properties of the mechanism (Γ, σ). Players can decide to deviate for two different reasons. On one hand, they can cooperate in order to find a strategy profile that provides a better outcome than the one given by the protocol. On the other hand, some players can behave maliciously for no specific reason and harm the altruistic ones. These two behaviours are prevented, according to [1], if prescribed distributed protocols are respectively (i) practical and k-resilient and/or (ii) t-immune.

A mechanism (Γ, σ) is *practical* if σ is a Nash Equilibrium of the game Γ after the iterated deletion of weakly dominated strategies. Players have a very low incentive to play weakly dominated strategies since they have available a different strategy providing no lower outcome in any scenario. If a mechanism is practical, these strategies are not played.

A mechanism (Γ, σ) is *k-resilient* if there is no coalition of at most k players having an incentive to simultaneously change strategy to get a better outcome. Formally, a strategy profile $\sigma \in \mathcal{S}$ is a *k-resilient equilibrium* if for all $C \subseteq N$ with $1 \leq |C| \leq k$, all $\tau_C \in \mathcal{S}_C$ and all $i \in C$, we have $u_i(\sigma_C, \sigma_{-C}) \geq u_i(\tau_C, \sigma_{-C})$. The concept of k-resilience denotes the tendency of a set of k players to cooperate to move to an equilibrium that differs from the prescribed one. Hence k-resilience generalizes the concept of Nash Equilibrium.

A mechanism (Γ, σ) is *t-immune* if, given at most t players choosing any strategy different from the prescribed one, the rest of the players receive at least the utility they would get if everyone followed the protocol. Formally, a strategy profile $\sigma \in \mathcal{S}$ is *t-immune* if for all $T \subseteq N$ with $|T| \leq t$, all $\tau_T \in \mathcal{S}_T$ and all $i \in N \setminus T$, we have $u_i(\sigma_{-T}, \tau_T) \geq u_i(\sigma)$. The property of t-immunity is very strong and difficult to satisfy since it requires that the protocol provides the best outcome no matter how a set of t players deviates. We therefore introduce a weaker version of the property – *t-weak-immunity* – guaranteeing that non deviating players receive at least the utility value of the initial state (i.e., players receive a positive outcome).

▶ **Definition 1** (t-weak-immunity). *A mechanism (Γ, σ) is* t-weak-immune *if for all $T \subseteq N$: $|T| \leq t$, all $\tau_T \in \mathcal{S}_T$ and all $i \in N \setminus T$, we have $u_i(\sigma_{-T}, \tau_T) \geq 0$.*

A player that joins a t-weak-immune mechanism will not suffer any loss (i.e., outcome with negative utility) if there are at most t deviating players in the game. We say that a mechanism is *weak immune* if it is t-weak-immune for all $t \in N$ and that a mechanism is *(k, t)-robust* if it is k-resilient and t-weak-immune.

Following the terminology introduced in [1], if every strict subset of players has no incentive to change their strategy σ, we say that the mechanism (Γ, σ) is *strongly resilient*. The concepts of k-resiliency and practicality are strictly connected with the properties of Nash Equilibria, such as *strength* [9] and *stability* [16, 19], which have been fully studied in [11]. Indeed, it is possible to prove that (i) if a mechanism (Γ, σ) is strongly resilient, then σ is a strong equilibrium of Γ and that (ii) if σ is a stable equilibrium of Γ, then the mechanism (Γ, σ) is practical (cf. [38] for details). In [16], authors prove that there always exists at least one stable Nash Equilibrium, therefore as a corollary there is always at least one practical mechanism. We know from [19] that the properties of strength and stability are independent. This means that we cannot draw conclusions about a property knowing whether the other property is fulfilled or not. Thus, we can state the following theorem.

▶ **Theorem 2** (independence). *The property of strongly resiliency and practicality are independent.*

We say that a mechanism (Γ, σ) is *optimal resilient* if it is practical and strongly resilient. In the sequel we verify whether protocols can be modeled with strongly resilient and/or practical mechanisms. Both properties have to be verified since they are independent. If a protocol does not provide a mechanism with a strong Equilibrium, it is necessary to compute k such that k-resiliency is fulfilled. On the other hand, given a generic game Γ, it is always possible to easily identify which are the practical mechanisms that always exist.

2.3 Composition of Games and Mechanisms

Blockchains systems are complex protocols designed in a modular way. In order to study the robustness of such complex protocols, we need to analyze the individual modules and infer the properties of the system by composition. For this scope we introduce the new notion of *composition of games* that, to the best of our knowledge, has never been defined in the literature. Given two different games A and B, the composition of games is defined by the operator $\odot$, hence $A \odot B$ denotes the composition of game A and B. Given two games that are played separately and independently, the composition corresponds to players picking a strategy from each game and receiving as utility the sum of the utilities of the two games.

▶ **Definition 3** (games composition). *Given $A = \langle N, \mathcal{S}_A, u_A \rangle$ and $B = \langle N, \mathcal{S}_B, u_B \rangle$ two games in normal form with the same set of players N, two different sets of strategies $\mathcal{S}_A = \{\mathcal{S}_{Ai} : i \in N\}$ and $\mathcal{S}_B = \{\mathcal{S}_{Bi} : i \in N\}$ and two different utility functions: $u_A : \mathcal{S}_A \to \mathbb{R}^N$ and $u_B : \mathcal{S}_B \to \mathbb{R}^N$ then, it is possible to define a new game $C = A \odot B$, called composition of A and B, characterized as follows: $C = \langle N, \mathcal{S}_C, u_C \rangle$, where N is the set of the players, $\mathcal{S}_C := \{(s_{Ai}, s_{Bi}), s_{Ai} \in \mathcal{S}_{Ai}, s_{Bi} \in \mathcal{S}_{Bi}, \forall i \in N\}$ is the set of strategies and $u_C(\{(\sigma_{Ai}, \sigma_{Bi})\}) := u_A(\{\sigma_{Ai}\}) + u_B(\{\sigma_{Bi}\})$ is the utility function.*

In the context of non-cooperative games linear transformations of utility functions are considered invariant transformations since they preserve the main properties of the game [14]. Therefore, we define the utility function of the composition of games as the sum of the utility functions of the composed games. It is possible to extend the definition of games composition to pairs of games in which different sets of players are involved. Indeed, if a player i is involved in game A but not in game B, it is possible to extend game $B = \langle N, \mathcal{S}_B, u_B \rangle$ to $B' = \langle N', \mathcal{S}'_B, u'_B \rangle$ in which player i is added ($N' = N \cup \{i\}$) and she is assigned a "null" strategy ($\mathcal{S}'_B = \mathcal{S}_B \times \{\sigma_\emptyset\}$) not influencing the utilities of the outcomes. Formally, for all $s \in \mathcal{S}_B$ and for all $j \in N' \setminus \{i\}$ we have that $u'_j(s, \sigma_\emptyset) = u_j(s)$, while for $i \in N'$ we have that $u_i(s, \sigma_\emptyset) = 0$. Intuitively it is possible to extend the definition of games composition to more than two games. In Section 3.3.4 we use the notation $A \odot B \odot C$ to represent either game $A \odot (B \odot C)$ or $(A \odot B) \odot C$. The following propositions allow us to (i) model the building blocks of complex protocols, (ii) study the properties of the subsequent mechanisms and (iii) deduce the properties of the composed protocol through the composition of mechanisms.

Concerning the solutions of the composition of games, we prove that Nash Equilibria can be identified by selecting equilibria within the single games. It is not possible to create or destroy Nash equilibrium strategies by composing independent games.

▶ **Theorem 4** (Nash Equilibria composition). *Let $A = \langle N, \mathcal{S}_A, u_A \rangle$ and $B = \langle N, \mathcal{S}_B, u_B \rangle$ be two games in normal form representation. Then, $\{(\sigma_{Ai}, \sigma_{Bi})\}$ is a Nash Equilibrium for $A \odot B$ if and only if $\{\sigma_{Ai}\}$ and $\{\sigma_{Bi}\}$ are Nash Equilibria respectively for A and B.*

Moreover, the operator composition is not only closed with respect to Nash Equilibria, but also closed with respect to the property of practicality. Let $A = \langle N, \mathcal{S}_A, u_A \rangle$ and $B = \langle N, \mathcal{S}_B, u_B \rangle$ be two games and let (A, σ_A) and (B, σ_B) be two practical mechanisms. Then, $(A \odot B, \{\sigma_{Ai}, \sigma_{Bi}\})$ is a practical mechanism.

Concerning robustness properties for composition of games, we can state the following results on resiliency and weak immunity for two composed games. The results can be generalized for the composition of multiple games.

▶ **Theorem 5** (resiliency). *Let $A = \langle N, \mathcal{S}_A, u_A \rangle$ and $B = \langle N, \mathcal{S}_B, u_B \rangle$ be two games and let (A, σ_A) and (B, σ_B) be two mechanisms respectively k-resilient and k'-resilient. Then, $(A \odot B, \{\sigma_{Ai}, \sigma_{Bi}\})$ is a $\min(k, k')$-resilient mechanism.*

▶ **Theorem 6** (weak immunity). *Let $A = \langle N, \mathcal{S}_A, u_A \rangle$ and $B = \langle N, \mathcal{S}_B, u_B \rangle$ be two games and let (A, σ_A) and (B, σ_B) be two mechanisms respectively t-weak-immune and t'-weak-immune. Then, $(A \odot B, \{\sigma_{Ai}, \sigma_{Bi}\})$ is a $\min(t, t')$-weak-immune mechanism.*

The first result states that given two k, k'-resilient mechanisms, the threshold on the maximum number of rational players allowed in the composition of games is the minimum among the number of rational players in the individual mechanisms. According to the second theorem, given two t, t'-weak immune mechanisms, the threshold on the maximum number of Byzantine players allowed in the composition of games is the minimum among the number of Byzantine players in the individual, as well. The proofs make use of the definition of k-resilience and t-weak immunity respectively; if a mechanism is k-resilient, then the protocol is followed if there are at most k rational players while if a mechanism is t-weak immune it provides non-negative outcomes if there are at most t Byzantine players.

3 Applications

In this section we prove the effectiveness of our framework by analyzing the robustness of different blockchain protocols. Section 3.1 and 3.2 analyze layer-1 protocols (Tendermint [22] and Bitcoin [25]) while Section 3.3 and 3.4 address layer-2 protocols (Lightning Network [30], a protocol on top of the Bitcoin blockchain and the side-chain protocol Platypus [29]). Finally, Section 3.5 analyzes a cross-chain swap protocol [27] allowing two users to exchange cryptoassets belonging to two different blockchains. Names of the variables in the following sections are consistent with the notation used in the papers where protocols are introduced.

3.1 Tendermint

This section addresses the Tendermint consensus (i.e., Tendermint-core [22, 4]) which is characterized by three rounds: the Pre-Propose round, the Propose round and the Vote round. During the Pre-Propose round, the proposer presents a block to the other participants. During the Propose round, each participant chooses whether to accept or not the block and broadcasts her decision. If the votes for the proposal exceed a predetermined threshold ν then participants start the Vote phase. If the block receives more than ν votes, it is validated. Tendermint's consensus algorithm sets $\nu = n - f = \frac{2}{3}n$; the threshold representing the number of non-faulty actors (as n denotes the total number of nodes and f the total number of faulty nodes) is set to $\frac{2}{3}$ of the network participants.

▶ **Definition 7.** *The Tendermint game is a mechanism $(\Gamma^{tc}, \sigma^{tc})$ such that the game Γ^{tc} represents the decision-making problem and the strategy σ^{tc} is the prescribed consensus protocol. Once a proposal v is received, N players choose either to check or not to check the validity of the value, then they can choose either to Vote or Not to Vote for it. At the very first stage of the game (stage a) a player can choose either to check (C) the validity or not check (NC). If she checks it, she can choose to Vote or Not Vote for it, in case value v is valid (stage b) or not (stage c). If she does not check it (stage d), she can choose to*

Vote (V) or Not Vote (NV) for it. Every strategy τ is represented by a vector (a, b, c, d) in which $a \in \{C, NC\}$, $b, c, d \in \{V, NV\}$. The utility for player i is $u_i(\tau) = 1$ if a valid block is approved or a non-valid block is not approved, $u_i(\tau) = 0$ if a valid block is not approved and $u_i(\tau) < 0$ if a non-valid block is approved.

The strategy prescribed by Tendermint's consensus protocol is $\sigma^{tc} = (C, V, NV, NV)$ i.e., to check for the validity of the proposal and then if the block is valid to vote for it, otherwise not vote for it. If the number of rational or Byzantine players allowed is $f < \frac{1}{3}n$, the other players have the necessary threshold to validate a block. Indeed, they can veto any validation of blocks proposed by malicious nodes. The mechanism $(\Gamma^{tc}, \sigma^{tc})$ is thus not f-weak-immune for any $f \geq \frac{1}{3}n$ and we can state the following results.

▶ **Theorem 8.** *The mechanism $(\Gamma^{tc}, \sigma^{tc})$ is (f, f)-robust for any $f < \frac{1}{3}n$.*

3.2 Bitcoin

Bitcoin is a permissionless blockchain based on the Proof-of-Work mechanism [25] where every user has a chance to publish a new block in the distributed ledger. The user probability to mine a new block is proportional to her computational power α. Bitcoin's protocol requires that once a block is mined, it should be broadcast to every other user. In case two or more blocks are mined at the same moment, the players split their effort to mine from any of the blocks (i.e., a *fork* is generated). Hence, published blocks are not automatically validated; they are considered as valid when belonging to the *longest chain* i.e., the longest branch of the ledger called *main chain*. A valid block generates a reward to the users who mined it.

As for Tendermint, Bitcoin's protocol can be represented by a mechanism $(\Gamma^{btc}, \sigma^{btc})$. We take into account the worst-case scenario, in which the Byzantine users coordinate, thus they are represented by a single player i. The altruistic users act in the same way and can therefore be represented by a second player j. The strategies of the players correspond to choosing (i) where in the chain add a new block and (ii) when to publish the mined blocks. Player j plays only one strategy defined by σ^{btc} i.e., she follows the protocol by mining on the main chain or splitting her effort if there is more than one chain of the same length available. Since the game is stochastic, we group all the equivalent states of the game in the same class; we consider two states as equivalent if they have the same configuration independently from the precise position in the chain (i.e., the difference between the number of mined blocks by the i and j is the same). In the Bitcoin blockchain a best practice is to consider a block as valid if belonging to a chain where at least B (usually, $B = 6$) blocks have been published afterwards, because it is presumably considered impossible to create a longer chain that does not include it. This block is invalidated if a fork is made at the previous block and more than $B + 1$ blocks are published starting from it. In this way, the block does not belong to the longest chain anymore and it is not considered as valid.

▶ **Definition 9.** *The Bitcoin game is a mechanism $(\Gamma^{btc}, \sigma^{btc})$ such that the game Γ^{btc} represents the decision-making problem and the strategy σ^{btc} is the prescribed protocol. The game Γ^{btc} is characterized by two players i and j, who have respectively mining power α and $1 - \alpha$ and every state of the game can be represented by the state class $\{x_k\}_{k \in \{0,1,\dots,B+1\}}$, where x_k is the number of blocks mined, yet not published, at level k by player i. The block at level $k = 0$ is the only one to be published. The initial state of the game is $\{x_k = 0\}$ $\forall k \in \{0, 1, \dots, B + 1\}$, while the final state of the game is represented by the state class with value $x_{B+1} \geq 1$. While player j has only one possible strategy σ^{btc}, player i can choose which branches to mine from (i.e. at which level k add the block). The utility of the players is the number of bitcoins they own according to the published blocks on the longest chain.*

The game theoretical framework let us state the following results on Bitcoin's mechanism robustness. Any subset of players T with $|T| = t$ having mining power $\alpha > 0$ have a small probability, not negligible, to perform a successful attack, by building a longer chain which does not include a block which was already considered valid (Theorem 10).

▶ **Theorem 10.** *The Bitcoin mechanism* $(\Gamma^{btc}, \sigma^{btc})$ *is not t-weak-immune for any t.*

▶ **Theorem 11.** *The Bitcoin mechanism* $(\Gamma^{btc}, \sigma^{btc})$ *is k-resilient if k players have at most* $\alpha \leq \frac{3}{20}$ *as total mining power.*

On the long run the majority of users ($\alpha \geq \frac{1}{2}$) produce the longer chain. However, on the short run a minority of users ($\alpha < \frac{1}{2}$) can make a fork on the longer chain with positive probability. The following theorem provides the value of this probability.

▶ **Theorem 12.** *The probability for a Byzantine player with computation power* α*, with* $\alpha < \frac{1}{2}$*, to prevent a transaction to be published within* $\Delta > 0$ *blocks is:*

$$\Phi_\Delta(\alpha) = \frac{\alpha}{1-\alpha} - \sum_{k=1}^{\Delta-1} (1 - \Phi_{\Delta-k}(\alpha)) \cdot \alpha^k \cdot (1-\alpha)^k \cdot M(k) \quad \textit{where,}$$

$M(k)$ *is a function defined in [17] mapping natural numbers to the sequence* $1, 1, 2, 5, 13, 42 \ldots$.

3.3 Lightning Network

In the Bitcoin blockchain transactions are collected in blocks, validated and published on the ledger. Bitcoin faces a problem of scalability, in terms of speed, volume and value of the transactions. In order to overcome these issues authors in [30] introduce a layer-2 class of protocols called Lightning Network. The latter allows users to create bidirectional payment *channels* to handle unlimited transactions in a private manner i.e., off-chain without involving the blockchain. Two users A and B open a channel by publishing on the Bitcoin blockchain two transactions towards a fund F. The amounts of the two transactions constitute the initial balance of the channel. In Section 3.3.1 we analyze the module to open a channel. The fund F can send or receive cryptoassets via blockchain transactions only if both users sign them. Once the channel is opened, users can exchange by simply privately updating the balance of the channel (cf. Section 3.3.2). The protocol to update the balance is discussed in Section 3.3.3. A further construction allowing users to create transactions within the channel that can be triggered at will is adopted in the protocol to update the balance (cf. Section 3.3.4). When the users decide to close the channel, two transactions are published on the Bitcoin blockchain: one from F to A and another from F to B. The value of the transactions corresponds to the ones of the latest balance. The protocol to close the channel is presented in Section 3.3.2. Lightning Network allows transactions also between users who have not opened a common channel (i.e., *routed payment*). Indeed, two users can perform a transaction through a path of open channels, using other users as intermediate nodes. This protocol is analyzed in Section 3.3.4.

3.3.1 Opening module

In order to open a channel, the Bitcoin users create a transaction Tx towards F and two different commitments ($C1a$ for A and $C1b$ for B) letting them close the channel unilaterally. The protocol [30] specifies in which order the commitments Tx, $C1a$ and $C1b$ have to be signed by the users. We formalize the protocol with a game in extensive form Γ^{op} (cf.

Definition 13) where at every node of the tree (i.e., decision step) the player involved in the protocol has two actions available: either following it by signing the commitment required or not following it. The *initial state* corresponds to having no channel opened, while the final state corresponds to having the channel opened. We assign "null" utility to the initial state and positive utility (by convention fixed to 1) to the final state. If at any step the players do not follow the protocol, they get back to the initial state with outcome $(0,0)$. If they do follow it at every step, they are able to open the channel having as an outcome $(1,1)$. We denote by $\sigma^{op} = (\{C1b_{A\cdot}, Tx_{A\cdot}\}, \{C1a_{\cdot B}, Tx_{AB}\})$ the strategy profile recommended by the protocol in which the actions are played respectively at nodes $(\{1,3\},\{2,4\})$.

▶ **Definition 13.** *The opening game Γ^{op} is a game in extensive form, with two players $\{A,B\}$ and 4 nodes, labeled by a number (1 is the root):*

1. *A has two actions available: $C1b_{\cdot\cdot}$ provides outcome $(0,0)$; $C1b_{A\cdot}$ leads to node 2.*
2. *B has two actions available: $C1a_{\cdot\cdot}$ provides outcome $(0,0)$; $C1a_{\cdot B}$ leads to node 3.*
3. *A has two actions available: $Tx_{\cdot\cdot}$ provides outcome $(0,0)$; $Tx_{A\cdot}$ leads to node 4.*
4. *B has two actions available: $Tx_{A\cdot}$ provides outcome $(0,0)$; Tx_{AB} provides outcome $(1,1)$.*

The protocol is thus represented by the mechanism $(\Gamma^{op}, \sigma^{op})$, whose properties we analyze in the sequel.

▶ **Theorem 14.** *The mechanism $(\Gamma^{op}, \sigma^{op})$ is not immune.*

The mechanism would be immune if both players receive no lower payoff than $u(\sigma^{op}) = (1,1)$, no matter what the other player chooses. A counterexample is B deviating from $\sigma_B^{op} = \{C1a_{\cdot B}, Tx_{AB}\}$ to $\tau_B = \{C1a_{\cdot\cdot}, Tx_{AB}\}$, i.e. B refusing to signing $C1a$ at step 2. For player A the outcome of $u_A(\sigma_A^{op}, \tau_B) = 0 < 1 = u(\sigma^{op})$.

▶ **Theorem 15.** *The mechanism $(\Gamma^{op}, \sigma^{op})$ is optimal resilient and weak immune.*

3.3.2 Classical and alternative closing modules

As described in Section 3.3.1 both users A and B can unilaterally close the channel by publishing respectively on the blockchain commitment $C1a$ and $C1b$. If a user decides to unilaterally close the channel, she receives her part of the fund after that a given number Δ of blocks are validated on the Bitcoin blockchain, while the other user receives it immediately. The protocol recommends to close the channel by creating a new transaction, namely ES, that let the players receive their cryptoasset immediately. We model the problem with the following game in normal form.

▶ **Definition 16.** *The closing game $\Gamma^{cl} = \langle N, \mathcal{S}, u\rangle$ of the channel with balance (x_A, x_B) with $x_A, x_B > 0$ is a game in normal form, with two players $\{A,B\}$ who have available three different pure strategies each: $\mathcal{S}_A = \{C1a_{AB}, DN, ES\}$ and $\mathcal{S}_B = \{C1b_{AB}, DN, ES\}$. The value of the utility can be found in the following payoff table.*

		B		
		$C1b_{AB}$	DN	ES
A	$C1a_{AB}$	$(\frac{1}{2}, \frac{1}{2})$	$(0,1)$	$(0,1)$
	DN	$(1,0)$	$(-1,-1)$	$(-1,-1)$
	ES	$(1,0)$	$(-1,-1)$	$(1,1)$

First, we assume that the channel (x_A, x_B) is funded by both players i.e., $x_A, x_B > 0$. If one of the two players has no asset involved in the channel, we have to model the problem with a degenerate game, in which she can arbitrarily play any possible strategy. We recommend

users to never unilaterally fund the channel. The players have three different strategies: publishing their commitment, seeking a deal to create a new transaction ES or just doing nothing DN. We assign null utility to players who receive their asset after Δ blocks, positive utility (normalized to 1) if they receive it immediately, negative utility if they cannot redeem their cryptoassets. If they both try and publish their commitment $(C1a_{AB}, C1b_{AB})$ we assume they have equal probability to get their commitment published first. The protocol recommends the strategy profile $\sigma^{cl} = (ES, ES)$ i.e., both players seek a deal. In the following we analyze the properties of the mechanism $(\Gamma^{cl}, \sigma^{cl})$.

▶ **Theorem 17.** *Under the assumption $x_A > 0$ or $x_B > 0$, the mechanism $(\Gamma^{cl}, \sigma^{cl})$ is optimal resilient, but not weak immune.*

To prove that the mechanism is not weak immune it is sufficient to show a counterexample. Indeed, if A chooses ES as required by the protocol and B chooses the Byzantine strategy N, player A receives a negative outcome $u_A(\sigma_A^{cl}, DN) = u_A(ES, DN) = -1$. Since the mechanism is not weak immune, it is not immune either. We thus provide an alternative protocol that satisfies the property of weak immunity.

▶ **Theorem 18.** *Under the assumption $x_A > 0$ or $x_B > 0$, the only weak immune mechanism is (Γ^{cl}, σ^*) with $\sigma^* = (C1a_{AB}, C1b_{AB})$.*

If users play this strategy, they never get a negative utility if the other player deviates. It is easy to prove that this is the only strategy profile with this property.

3.3.3 Updating module

Performing a transaction within a channel consists in updating its balance. Technically, the previous commitments ($C1a$ and $C1b$) with balance (x_A, x_B) are replaced by two new commitments ($C2a$ and $C2b$) with different balance (x'_A, x'_B). In order to prevent players from publishing old commitments, they sign two Breach Remedy Transactions ($BR1a$ and $BR1b$), that can invalidate $C1a$ and $C2b$. Indeed, if any party publishes an outdated commitment the other one can retrieve all the cryptoassets in the fund. If, for instance, A publishes the outdated commitment $C1a$, she can retrieve her fund x_A unless B publishes $BR1a$ before Δ blocks are validated. The protocol to update the balance requires the players to sign the commitments in a specific order [30]. We formalize the protocol with a game in extensive form Γ^{up} (cf. Definition 19). The initial state corresponds to the previous balance (with null utility), the final state to the updated balance (with utility equal to 1). We assign a negative value to the states in which players lose their cryptoassets or part of them.

▶ **Definition 19.** *The updating game Γ^{up} is a game in extensive form, with two players $\{A, B\}$ and 5 nodes, labeled by a number (1 is the root):*

1. *A plays. $C2b_{\cdot\cdot}$ provides outcome $(0, 0)$; $C2b_{A\cdot}$ leads to node 2.*
2. *B plays. $C2a_{\cdot\cdot}$ provides outcome $(0, 0)$; $C2b_{AB}$ provides outcome $(1, 1)$; $C2a_{\cdot B}$ leads to node 3.*
3. *A plays. $BR1a_{\cdot\cdot}$ provides outcome $(0, 0)$; $C2a_{AB}$ provides outcome $(1, 1)$; $BR1a_{A\cdot}$ leads to node 4.*
4. *B plays. $BR1b_{\cdot B}$ provides outcome $(1, 1)$; $BR1b_{\cdot\cdot}$ leads to node 5.*
5. *A plays. $C1a_{AB}$ provides outcome $(-1, 1)$; $C2a_{AB}$ provides outcome $(1, 1)$.*

The protocol recommends to sign all the commitments and it is thus represented by the strategy profile $\sigma^{up} = (\{C2b_{A\cdot}, BR1a_{A\cdot}, C2a_{AB}\}, \{C2a_{\cdot B}, BR1b_{\cdot B}\})$. We analyze the mechanism $(\Gamma^{up}, \sigma^{up})$ under the assumption that it is always possible to publish a transaction

within Δ blocks, otherwise it is not possible to validate the breach remedy transactions in time. The probability that this happens when a Byzantine agent with computational power α attacks the Bitcoin blockchain is $1 - \Phi_\Delta(\alpha)$ (cf. Theorem 12).

▶ **Theorem 20.** *The mechanism* $(\Gamma^{up}, \sigma^{up})$ *is optimal resilient and weak immune with probability* $1 - \Phi_\Delta(\alpha)$*, but it is not immune.*

3.3.4 Routing module

Lightning Network provides a protocol, called *Hash time Locked Contract* (HTLC), that allows to create transactions that can be triggered at will. The protocol for the HTLC works as follows: (i) user A creates a pair (H, R), where H is public and R is its private key; (ii) she shares with user B a commitment together with the string H; (iii) once this commitment is published on the Bitcoin blockchain, user B can receive the transaction only if she can provide the private key R within Δ blocks. It is easy to check that R is the private key of H, but it is almost impossible to retrieve R, given H. In this way, user A can trigger the transaction whenever she wants by disclosing R to user B. The protocol can be represented by a mechanism $(\Gamma^{htlc}, \sigma^{htlc})$, that has the very same structure of the updating module (cf. Section 3.3.3) and thus satisfies optimal resilience and weak immunity, but not immunity.

The HTLC is implicated in the protocol allowing users to perform transactions also if they do not share a common channel. Indeed, it is sufficient that among the two users there is a path of channels i.e., a sequence of users who two-by-two share a channel. For instance, let us suppose that users A and C have both opened a separate channel with a third user B. In the *routed payment* user B is the intermediate node. The model can be easily generalised to any number of intermediate nodes. Routing fees are not included, but they would not change the solution of the game. The protocol for routed payment works as follows: (i) user C creates a pair of strings (H, R) and then discloses H to user A; (ii) user A creates an HTLC with user B locked with the public key H then, (iii) user B creates an HTLC with user C locked with H; (iv) finally, user C discloses R with user B and triggers the transaction, and so does user B with user A. In this way, user C receives the payment, user A sends it and user B gains from a channel with A what she loses from the channel with C. In practice, the value of the two transactions do not coincide, so that the difference consists in the fee to be provided to user B. We formalize the protocol with a game in extensive form Γ^{rout}. The strategy profile recommended by the protocol is denoted by $\sigma^{rout} = (\{H_A^{AB}\}, \{H_B^{BC}, Y\}, \{Y, Y\})$.

▶ **Definition 21.** *The routing game* Γ^{rout} *is a game in extensive form, with three players* $\{A, B, C\}$ *and 5 nodes, labeled by a number (*1 *is the root):*

1. *C has two actions available: either N, not sending H to A, which provides outcome* $(0, 0, 0)$*, or Y, sending H to A, which leads to node* 2*.*
2. *A has two actions available: either* $H_\cdot^{AB}$*, which provides outcome* $(0, 0, 0)$*, or* H_A^{AB}*, which leads to node* 3*.*
3. *B has two actions available: either* $H_\cdot^{BC}$*, which provides outcome* $(0, 0, 0)$*, or* H_B^{BC}*, which leads to node* 4*.*
4. *C has two actions available: either N, not disclosing R to B, which provides outcome* $(0, 0, 0)$*, or Y, disclosing R to B, which leads to node* 5*.*
5. *B has two actions available: either N, not disclosing R to A, which provides outcome* $(1, -1, 1)$ *or Y, disclosing R to A, which provides outcome* $(1, 1, 1)$*.*

▶ **Theorem 22.** *Under the assumption that in both HTLCs the transactions can be triggered,* $(\Gamma^{rout}, \sigma^{rout})$ *is optimal resilient and weak immune, but it is not immune.*

The HTLCs introduced in the protocol work independently from the routing protocol. We can model them with two different mechanisms: $(\Gamma^{AB}, \sigma^{AB})$ for H^{AB} and $(\Gamma^{BC}, \sigma^{BC})$ for H^{BC}. The mechanism $(\Gamma^{AB}, \sigma^{AB})$ represents the HTLC deployed on the channel A-B, while the mechanism $(\Gamma^{BC}, \sigma^{BC})$ refers to the HTLC implemented on the channel B-C. The HTLCs belong to two different channels, so they are independent one from another. The assumption from the routing protocol is that in both HTLCs the transactions can be triggered, but this is true only if every transaction can be published within Δ blocks. Under this assumption, the routed payment is represented by three independent protocols $(\Gamma^{rout}, \sigma^{rout})$, $(\Gamma^{AB}, \sigma^{AB})$, and $(\Gamma^{BC}, \sigma^{BC})$. Therefore, we analyze the properties of its mechanism by defining and analyzing the composition of the three games $(\Gamma^{rout} \odot \Gamma^{AB} \odot \Gamma^{BC}, \{\sigma_i^{rout}, \sigma_i^{AB}, \sigma_i^{BC}\})$.

▶ **Theorem 23.** *The mechanism* $(\Gamma^{rout} \odot \Gamma^{AB} \odot \Gamma^{BC}, \{\sigma_i^{rout}, \sigma_i^{AB}, \sigma_i^{BC}\})$ *is optimal resilient and weak immune with probability* $1 - \Phi_\Delta(\alpha)$ *(cf. Theorem 12).*

Proof. The operator composition (cf. Definition 3) is invariant with respect the properties of the mechanisms. Thanks to Theorems 20, 22 we have that $(\Gamma^{rout}, \sigma^{rout})$, $(\Gamma^{AB}, \sigma^{AB})$ and $(\Gamma^{BC}, \sigma^{BC})$ are practical hence their composition $(\Gamma^{rout} \odot \Gamma^{AB} \odot \Gamma^{BC}, \{\sigma_i^{rout}, \sigma_i^{AB}, \sigma_i^{BC}\})$ is practical. Analogously, thanks to Theorems 20, 22 we have that every single mechanism is k-resilient for all k and t-weak-immune for all t. Theorems 5, 6 allow us to say that the composition $(\Gamma^{rout} \odot \Gamma^{AB} \odot \Gamma^{BC}, \{\sigma_i^{rout}, \sigma_i^{AB}, \sigma_i^{BC}\})$ is k-resilient for all k and t-weak-immune for all t, i.e., it is strongly resilient and weak immune. ◀

Recap. All the results of the Lightning Network are available in Table 1. The Lightning Network is built on top of Bitcoin blockchain therefore its properties depend highly on Bitcoin blockchain's ones. If we exclude the closing protocol, the Lightning Network satisfies optimal resilience and weak immunity. Hence, we can compose (cf. Definition 3) Lightning Network protocols' games with Bitcoin mechanism's (that provide weaker results, cf. Section 3.2) and prove that the Lightning Network satisfies the same properties of the Bitcoin mechanism.

3.4 Side-chain

A different solution to overcome the scalability and privacy problems of permissionless blockchains is offered by Platypus [29], a protocol that allows a group of users to create a childchain (sidechain) that can handle off-chain transactions without the need of synchrony among peers. This section analyzes the protocol to create a Platypus chain proposed in [29]. In this section we would like to extend the analysis performed in [29] proving new properties which fit our framework. The protocol lets the childchain validators broadcast transactions to the peers until the number of validators who have confirmed the transactions overcome a defined threshold. The protocol is divided into phases consisting of players acting at the same time, indeed it is possible to model this protocol with a game in extensive form Γ^{cr}, in which players are split into two categories: normal users (set U) and the validators (set V). Users' utility is positive if their transactions are successfully published and it is negative if different transactions are validated instead.

▶ **Definition 24.** *The creation game is a game* Γ^{cr} *in extensive form, where* $U \cup V$ *is the set of players, with* $m_v = |U \cup V|$*. Every phase corresponds to a node of the tree, at which players play at the same time.*

- ***Phase 1****; only the player* p_0 *is involved. The player* p_0 *has two actions: either complete the transaction* Y *or not* N*. If she does not, the outcome is* 0 *for all players.*

- ***Phase 2****; every player within normal users play at the same time. Everyone has available the same two actions: broadcasting their transaction* Y *or not* N*. If the transaction is not broadcast for player* i*, her utility is always* 0*.*
- ***Phase 3****; the validators can choose within a set of actions* a_u *with* $u \subseteq U$ *i.e., they can validate all the transaction for the users within the set* u*. The cardinality of the set of their actions is equal to* $2^{|U|}$*. The utility for the validators corresponds to the number of valid transactions which are broadcast.*
- ***Phase 4****; the validators can choose within a set of actions in the form* $(b_t, s_{t'})$*, where* t *and* t' *are any subset of transactions broadcast in Phase 3. The action* b_t *consists in broadcasting the transactions belonging to the set* t *until* $\lfloor 2m_v/3 \rfloor + 1$ *validators receive it, while* $s_{t'}$ *means to send the transactions in* t'*.*

We define the mechanism $(\Gamma^{cr}, \sigma^{cr})$, where $\sigma^{cr} \in \mathcal{S}$ is the strategy of following the protocol i.e., for normal users u the strategy is $\sigma_u^{cr} = Y$, while for validators v the strategy is $\sigma_v^{cr} = (a_{u^*}, b_{t^*}, s_{t^*})$, where u^* is the set of users who send a message and t^* is the set of transactions broadcast in Phase 3. We thus analyze the properties of the mechanism.

▶ **Theorem 25.** *The mechanism* $(\Gamma^{cr}, \sigma^{cr})$ *is optimal resilient and* $\lfloor \frac{m_v}{3} \rfloor$*-weak-immune, but not* t*-immune for any* t*.*

In [29] it is proved that no wrong transaction can be validated if there are at most $\lfloor \frac{m_v}{3} \rfloor$ corrupted players. This property cannot be expressed with the concept of immunity, which is too strong. Hence, to capture this information we exploit the definition of t-weak-immunity (cf. Definition 1). Within our model, the upper bound on the number of corrupted players means that no negative payoff is given to the players under the hypothesis that there are at most $\lfloor \frac{m_v}{3} \rfloor$ Byzantine nodes i.e., that the mechanism is $\lfloor \frac{m_v}{3} \rfloor$-weak-immune.

3.5 Cross-chain swap

In this section we analyze the protocol introduced in [27] allowing two users to swap assets that belong to two different blockchains which do not communicate with each other. In [15] the authors introduce a theoretical framework proving that the protocol is correct for those players who are altruistic, no matter what the others do. In the following we prove that the Cross-chain swap protocol [27] satisfies the (k, t)-weak-robustness. In this protocol users publish two different transactions on two different blockchains (e.g., Altcoin and Bitcoin) that can be triggered with the disclosure of a single private key x by means of hashed time lock contracts (HTLCs, cf. Section 3.3). The transactions have to be published within two different time intervals, Δ_1 and Δ_2 (where $\Delta_1 \geq 2\Delta_2$), depending on the corresponding blockchain. In a 2-player context authors in [27, 15] assume that the transactions can be published within the time interval $[0, \min(\Delta_1, \Delta_2)] = [0, \Delta_2]$. Since the two blockchains are independent we model the protocol with two different mechanisms $(\mathcal{G}_1, \sigma_1)$ and $(\mathcal{G}_2, \sigma_2)$ (cf. Definitions 26 and 27), representing the actions that players perform in each blockchain (i.e., $(\mathcal{G}_1, \sigma_1)$ for the Bitcoin blockchain and $(\mathcal{G}_2, \sigma_2)$ for the Altcoin blockchain).

▶ **Definition 26.** *The Bitcoin game is an extensive form game* $\mathcal{G}_1$ *with 2 players* $\{A, B\}$ *and 5 nodes (*1 *is the root):*

1. *A can either (*Y*) create TX1 and TX2, that leads to node* 2 *or (*N*) not create them, with outcome* $(0, 0)$*.*
2. *B can either (*Y*) sign TX2, that leads to node* 3*, or (*N*) refuse to do it, with outcome* $(0, 0)$*.*

3. *A can either (N) do nothing, with thus outcome* $(0,0)$*, or (Y) publish TX1 on the Bitcoin blockchain, that leads to node* 4*.*
4. *Both A and B have available two actions: either (Y) publish TX2 before secret x is revealed or (N) not publish it. If any of the two users does so, the outcome is* $(0,0)$*. Otherwise, A reveals secret x and* (N,N) *leads to node* 5*.*
5. *B can either (Y) publish secret x on the Bitcoin blockchain or (N) not publish it. If she does, the outcome is* $(1,1)$*. If she does not, the outcome is* $(1,-1)$*.*

The strategy profile that corresponds to following the protocol is $\sigma_1 = (\{Y,Y,N\},\{Y,N,Y\})$*.*

▶ **Definition 27.** *The Altcoin game is an extensive form game* $\mathcal{G}_2$ *with 2 players* $\{A,B\}$ *and 5 nodes (*1 *is the root):*

1. *B can either (Y) create TX3 and TX4, or (N) do nothing. The action Y leads to node* 2*, while the action N leads to the outcome* $(0,0)$*.*
2. *A can either (Y) sign TX4, that leads to node* 3*, or (N) refuse to do it, with outcome* $(0,0)$*.*
3. *B can either (N) do nothing, with thus outcome* $(0,0)$*, or (Y) publish TX3 on the Altcoin blockchain, that leads to node* 4*.*
4. *Both A and B have available two actions: either (Y) publish TX4 before secret x is revealed or (N) not publish it. If any of the two does so, the outcome is* $(0,0)$*. Otherwise, A reveals secret x and* (N,N) *leads to node* 5*.*
5. *A can either (Y) publish secret x on the Altcoin blockhain or (N) not publish it. If she does, the outcome is* $(1,0)$*. If she does not, the outcome is* $(0,0)$*.*

The strategy profile that corresponds to following the protocol is $\sigma_2 = (\{Y,N,Y\},\{Y,Y,N\})$*.*

Since the two blockchains are independent, we consider the composition of the two games $(\mathcal{G}_1 \odot \mathcal{G}_2, \{\sigma_{1i}, \sigma_{2i}\})$ representing the full protocol and analyze it.We can easily see that the mechanism is not immune, indeed it is sufficient that one player does not create or publish a transaction to stop the protocol. However, we have the following important result.

▶ **Theorem 28.** *Under the assumption that any transaction can be published within a time interval* $[0,\Delta_2]$*, the mechanism* $(\mathcal{G}_1 \odot \mathcal{G}_2, \{\sigma_{1i}, \sigma_{2i}\})$ *is optimal resilient and weak immune, but it is not immune.*

4 Conclusions

We proposed the first generic game theoretical framework that models the robustness of blockchains towards rational and Byzantine behaviors. In this paper we identified the necessary and sufficient conditions for a protocol to be robust (defined as the conjunction of two properties: k-resilience and t-weak immunity) and developed a methodology to characterize the robustness of complex protocols via the composition of simpler robust building blocks. The effectiveness of our framework was demonstrated by its capability to capture the robustness of various blockchain protocols such as Bitcoin, Tendermint, lightning networks (original and alternative closing modules), side-chain and cross-chain protocols. Our work continues the work of [1] that introduced the notion of robustness defined in terms of t-immunity and k-resilience. The framework of [1] was never used till our study in the context of blockchain protocols. Using the framework of [1] we proved that a large class of blockchain protocols (cf. Table 1) does not satisfy the t-immunity property. It should be noted that our negative result related to the t-immunity property does not depend on the

specific choice of a utility function. Therefore, we proposed a relaxation of this property i.e., t-weak immunity. We analysed the k-resilience and the t-weak immunity of a large class of blockchain protocols, providing bounds on respectively the number of rational and Byzantine processes (cf. results in Table 1).

These results are based on strict hypotheses under which the model we introduced takes into account all the possible alternatives to the protocol. As future work we plan to relax these hypotheses and provide more accurate estimation of the robustness indices. Moreover, we plan to investigate the resilience of other blockchain protocols such as Algorand [10] or DAG-based blockchains (e.g., Spectre [33], Phantom [34] or IOTA [31]). A further possible direction of research is an extension of our framework in order to analyse repeated consensus protocols (e.g., protocols presented in [4]).

References

1 Ittai Abraham, Danny Dolev, Rica Gonen, and Joe Halpern. Distributed computing meets game theory: Robust mechanisms for rational secret sharing and multiparty computation. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*, PODC '06, pages 53–62, New York, NY, USA, 2006. Association for Computing Machinery. doi:10.1145/1146381.1146393.

2 Amitanand S. Aiyer, Lorenzo Alvisi, Allen Clement, Michael Dahlin, Jean-Philippe Martin, and Carl Porth. Bar fault tolerance for cooperative services. In *SOSP '05*, 2005.

3 Yackolley Amoussou-Guenou, Bruno Biais, Maria Potop-Butucaru, and Sara Tucci Piergiovanni. Rational vs byzantine players in consensus-based blockchains. In Amal El Fallah Seghrouchni, Gita Sukthankar, Bo An, and Neil Yorke-Smith, editors, *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020*, pages 43–51. International Foundation for Autonomous Agents and Multiagent Systems, 2020. URL: https://dl.acm.org/doi/abs/10.5555/3398761.3398772.

4 Yackolley Amoussou-Guenou, Antonella Del Pozzo, Maria Potop-Butucaru, and Sara Tucci-Piergiovanni. Dissecting tendermint. In Mohamed Faouzi Atig and Alexander A. Schwarzmann, editors, *Networked Systems*, pages 166–182, Cham, 2019. Springer International Publishing.

5 Zeta Avarikioti, Eleftherios Kokoris-Kogias, Roger Wattenhofer, and Dionysis Zindros. Brick: Asynchronous incentive-compatible payment channels. In *International Conference on Financial Cryptography and Data Security*, 2021.

6 Marianna Belotti, Stefano Moretti, Maria Potop-Butucaru, and Stefano Secci. Game theoretical analysis of Atomic Cross-Chain Swaps. In *40th IEEE International Conference on Distributed Computing Systems (ICDCS)*, Singapore, Singapore, 2020. URL: https://hal.archives-ouvertes.fr/hal-02414356.

7 Marianna Belotti, Stefano Moretti, and Paolo Zappalà. Rewarding miners: bankruptcy situations and pooling strategies. In *17th European Conference on Multi-Agent Systems (EUMAS)*, Tessaloniki, Greece, 2020. URL: https://hal.archives-ouvertes.fr/hal-02481155.

8 Iddo Bentov, Pavel Hubácek, Tal Moran, and Asaf Nadler. Tortoise and hares consensus: the meshcash framework for incentive-compatible, scalable cryptocurrencies. *IACR Cryptology ePrint Archive*, 2017:300, 2017.

9 B.Douglas Bernheim, Bezalel Peleg, and Michael D Whinston. Coalition-proof nash equilibria i. concepts. *Journal of Economic Theory*, 42(1):1–12, 1987. doi:10.1016/0022-0531(87)90099-8.

10 Jing Chen and Silvio Micali. Algorand: A secure and efficient distributed ledger. *Theor. Comput. Sci.*, 777:155–183, 2019. doi:10.1016/j.tcs.2019.02.001.

11 Altannar Chinchuluun, Panos Pardalos, Athanasios Migdalas, and Leonidas Pitsoulis. *Pareto Optimality, Game Theory And Equilibria*, volume 17. Springer, 2008. doi:10.1007/978-0-387-77247-9.

12 Christian Ewerhart. Finite blockchain games. *Economics Letters*, 197:109614, 2020. doi:10.1016/j.econlet.2020.109614.

13 Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security*, pages 436–454. Springer, 2014.

14 Peter Hammond. *Utility Invariance in Non-Cooperative Games*, volume 38, pages 31–50. Springer, June 2006. doi:10.1007/0-387-25706-3.

15 Maurice Herlihy. Atomic cross-chain swaps. In *Proceedings of the 2018 ACM symposium on principles of distributed computing*, pages 245–254, 2018.

16 John Hillas. On the definition of the strategic stability of equilibria. *Econometrica*, 58(6):1365–1390, 1990. URL: http://www.jstor.org/stable/2938320.

17 OEIS Foundation Inc. The on-line encyclopedia of integer sequences, 2021. URL: https://oeis.org/A178682.

18 Aggelos Kiayias and Aikaterini-Panagiota Stouka. Coalition-safe equilibria with virtual payoffs. *arXiv preprint*, 2019. arXiv:2001.00047.

19 Elon Kohlberg and Jean-Francois Mertens. On the strategic stability of equilibria. *Econometrica: Journal of the Econometric Society*, pages 1003–1037, 1986.

20 Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In Christoph Meinel and Sophie Tison, editors, *STACS 99*, pages 404–413, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

21 Harold William Kuhn and Albert William Tucker. *Contributions to the Theory of Games*, volume 2. Princeton University Press, 1953.

22 Jae Kwon. Tendermint: Consensus without mining. *Draft v. 0.6, fall*, 1(11), 2014.

23 Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y. Liang, and D. I. Kim. A survey on blockchain: A game theoretical perspective. *IEEE Access*, 7:47615–47643, 2019.

24 Thomas Moscibroda, Stefan Schmid, and Roger Wattenhofer. When selfish meets evil: Byzantine players in a virus inoculation game. In *Proceedings of the Annual ACM Symposium on Principles of Distributed Computing*, volume 2006, pages 35–44, January 2006. doi:10.1145/1146381.1146391.

25 Satoshi Nakamoto. A peer-to-peer electronic cash system, 2008.

26 John F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950. doi:10.1073/pnas.36.1.48.

27 Tier Nolan. Re: Alt chains and atomic transfers. accessed on January 10, 2020. https://bitcointalk.org/index.php?topic=193281.msg2224949#msg2224949.

28 Rafael Pass and Elaine Shi. Fruitchains: A fair blockchain. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pages 315–324, 2017.

29 Alejandro Ranchal Pedrosa and Vincent Gramoli. Platypus: Offchain protocol without synchrony. In Aris Gkoulalas-Divanis, Mirco Marchetti, and Dimiter R. Avresky, editors, *18th IEEE International Symposium on Network Computing and Applications, NCA 2019, Cambridge, MA, USA, September 26-28, 2019*, pages 1–8. IEEE, 2019. doi:10.1109/NCA.2019.8935037.

30 Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments, 2016.

31 Serguei Popov, Olivia Saa, and Paulo Finardi. Equilibria in the tangle. *Comput. Ind. Eng.*, 136:160–172, 2019. doi:10.1016/j.cie.2019.07.025.

32 Okke Schrijvers et al. Incentive compatibility of bitcoin mining pool reward functions. In *International Conference on Financial Cryptography and Data Security*, pages 477–498. Springer, 2016.

33 Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. Spectre: A fast and scalable cryptocurrency protocol. *IACR Cryptol. ePrint Arch.*, 2016:1159, 2016.

34 Yonatan Sompolinsky and Aviv Zohar. Phantom. *IACR Cryptology ePrint Archive, Report 2018/104*, 2018.

35 Itay Tsabary and Ittay Eyal. The gap game. In *Proceedings of the 2018 ACM SIGSAC conference on Computer and Communications Security*, pages 713–728, 2018.

36 Florian Tschorsch and Björn Scheuermann. Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys & Tutorials*, 18(3):2084–2123, 2016.

37 Wenbo Wang, Dinh Thai Hoang, Zehui Xiong, Dusit Niyato, Ping Wang, Peizhao Hu, and Yonggang Wen. A survey on consensus mechanisms and mining management in blockchain networks. *arXiv preprint*, pages 1–33, 2018. `arXiv:1805.02707`.

38 Paolo Zappalà, Marianna Belotti, Maria Potop-Butucaru, and Stefano Secci. Game theoretical framework for analyzing blockchains robustness. Technical report, Sorbonne Université, CNRS, Laboratoire d'Informatique de Paris 6, LIP6, 2020. URL: `https://hal.archives-ouvertes.fr/hal-02634752/document`.