# A Game-Theoretic Analysis of Cross-Ledger Atomic Swap Protocols

...

***Abstract*—With an increasing adoption rate of the blockchain technology, there is a strong need for achieving interoperability between multiple non-connected ledgers. To ensure atomicity of cross-ledger transactions, several approaches have been recently developed including relays, hash time lock contracts, and packetized payments. We propose a game-theoretical framework to study the outcomes of different cross-ledger transaction protocols with strategic and possibly malicious agents. We estimate the economical cost of being an honest agent, and derive transaction failure rates. We also illustrate that packetized payments are likely to be incomplete, at least without disciplinary mechanisms. Our model confirms that collateral deposits can prevent malicious agents to take advantage of the protocols, and we infer that the deposit amount should depend on the underlying asset price volatility or that it should be dynamically adjusted as the price changes.**

## I. Introduction

An atomic swap is a coordination task where two parties directly exchange assets such that either both parties receive the assets, or none in the event of failure [1]. Atomic swaps are easily achievable on a single ledger by implementing smart-contracts. For example, one can easily implement a contract to swap a token developed on the Ethereum platform against Ethers. Creation of multiple ledgers (altchains) with their specific coins, tokenization of different assets, from company shares to physical goods "digital twins", are evidences of an increasing development and adoption of the blockchain technology. There is, therefore, more and more interest to exchange, or swap, digital assets but also to exchange fiat currency against digital currency.

The common approach is to use a centralised exchange, this is certainly efficient and fast. However, it requires intermediary fees, trust in the exchange (in terms of privacy and transparency of its matching mechanisms), and is vulnerable to different kinds of attacks [2], from wallet hacking [3], to DDoS attacks [4]. Over-the-counter (OTC) operations [5], [6] remain frequent for financial transactions. They allow to execute larger orders with negotiated price, hence possibly avoiding price impact on financial markets. In the financial industry, it is common to use a trusted third party for OTC settlements, such as a central clearing counterparty or a broker-dealer, that have similar disadvantages as centralized exchanges.

In this work, we focus on transactions in the digital era involving disconnected blockchains and without relying on any trusted third party thus, eliminating the intermediary fees and choosing more privacy-preserving mechanisms. However, in such peer-to-peer (P2P) environment, the transacting agents typically don't know each other and, without middle man, they are exposed to malicious behaviors from their counterparty. Therefore, the major challenge in this setting is to achieve atomicity of the cross-ledger transaction.

### A. Contributions

In this paper, we present a framework to analyze the outcome of cross-ledger transactions with strategic agents. In this framework, the two agents agree to enter a transaction at the initial time, their actions can delay or complete the transaction, and the asset price may change if the transaction is not completed promptly. The agents aim to maximize their utility functions which depends 1) on the transaction outcome (success or failure), 2) on the asset price variation (trading profits), and 3) on the duration of the transaction (locked in the game). For simplicity we define two types of agents, honest and malicious, study their strategies and derive preference parameter conditions consistent with their actions. We derive the transaction failure rate as a function of the percentages of honest and malicious agents. We can also quantify, in "dollar value", the economical cost of being an honest agent. Our framework builds on finite extensive-form games with imperfect information, see [7]. The only known unknown information for the agents is the type, honest or malicious, of the other agent they entered the transaction with. We study two types of transactions, hash time lock contracts[1] (HTLCs) and packetized payments, yet our approach can be extended to other transaction protocols.

Multiple findings can be drawn from our analysis that are relevant for real-world applications. For HTLCs, it has been mentioned that the agent completing the transaction receives a free *American option* [8], meaning that she has the choice to complete the transaction, or not, if the asset price changes at her advantage. However, in Section III-B, we show that the other agent can also behave maliciously in order to increase his financial profit which has previously been ignored in the literature. For packetized payments, we show that it is impossible to enforce malicious agents to complete the transaction without an additional disciplinary mechanism. We illustrate that the "biased" preferences of agents for completed transactions have to be economically large, which motivates the necessity of alternative contracting mechanisms such as collateral deposits. Still, we infer that the initial collateral

[1]The full name may be found with the suffix "ed" after hash or lock in research papers. There is no official convention as far as we know.

amount should depend on the asset price volatility, or that it should be dynamically adjusted as the asset price fluctuates.

This paper also provides an up-to-date review of protocols aiming to achieve cross-ledger atomic swaps, and discusses the use of collateral an reputation mechanisms in practice.

### B. Paper structure

The rest of the paper is structured as follows. In Section II we review existing approaches aiming to execute cross-ledger swaps without trusted third party. We present the game-theoretic framework and the analysis of the hash time lock contract and of the packetized payment in Section III. In Section IV we discuss and suggest disciplinary mechanisms that could increase the transaction success rate. Section V concludes and indicates future extensions. The proofs are collected in Appendix A.

## II. Background and related work

In this section, we first briefly review existing approaches to achieve cross-ledger interoperability in distributed manner. Second, we describe in details hash time lock contracts (HTLCs), a coordination mechanism that can be employed to achieve cross-chain atomic swap. Finally, we describe the recently proposed packetized payment (PP), which aims to address some of the weaknesses of HTLCs.

### A. Cross-ledger interoperability

To address some of the issues with centralized exchanges, and transactions with an intermediary in general, distributed exchanges (DEXs) have recently become a popular tool for cross-chain assets exchange [9], [10], [11], [12]. However, the DEXs generally provide solely match-making services and then require either P2P execution that is governed by coordination mechanism such as HTLCs, possibly with collateral deposits. For instance, Bisq [9], an information platform for quotes and P2P transactions with arbitrators, uses a postage of collateral with possible intervention of an arbitrator, thus providing only a limited level of "distributiveness" and still requiring some trust in the arbitration system. Note that transactions executed via Bisq require a collateral deposit and an arbitrator fee. Discussion with community members on slack revealed that 3-5% of transactions fail and go to arbitration, and that this percentage increases during periods of higher market volatility. These observations are perfectly in line with the predictions of our model.

Wanchain [13] and Interledger [14] are other examples of decentralized medium for achieving cross-ledger interoperability. Wanchain enables interfacing and assets conversion to the native Wanchain token (Wancoins) in order, then, to perform a cross-ledger asset exchange. Wanchain implements privacy protection mechanism through a ring signature scheme [15] and a one-time account mechanism via one-time use wallets created for each transaction. Interledger [14] uses Byzantine notaries to construct a payment chain from sender to recipient over multiple ledgers.

Relays [16], sidechains [17], [18], off-chain payment channels [19], [20], [21], and solutions based on chain relays [22] require smart-contract functionality and building interfaces to such systems, similar to the case of blockchain-based medium like Wanchain.

HTLCs have been recently proposed [23], aiming to achieving atomicity of a cross-ledger transaction without any connections between the ledgers. They are often employed in decentralized exchanges to complete P2P exchange [10], [11]. Therefore, studying and improving HTLCs is of a high interest [24], [25], [26], as well as addressing the limitations of HTLCs that have already been underlined in [22], [27], including strong assumptions required to maintain security, interactiveness, need for synchronizing clocks between blockchains, and temporal locking of assets.

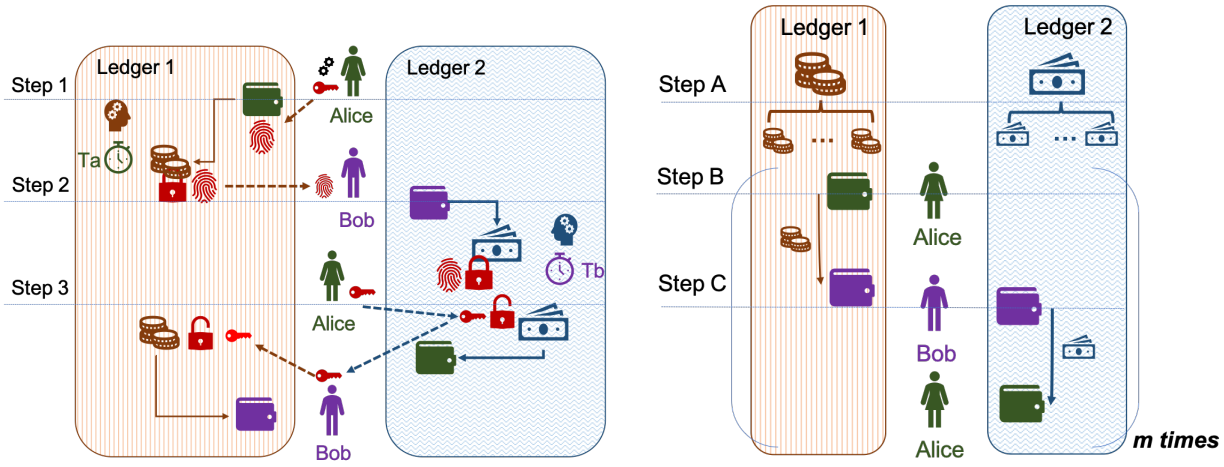
### B. Hash time lock contracts (HTLCs)

An hash time lock contract requires the two agents separately locking their assets on the respective blockchains, using the hash of a secret, generated by one of the user. The assets then can be unlocked upon revealing the preimage of a hash. This approach was first proposed on the Bitcoin forum [23] by a user with a pseudonym TierNolan. HTLCs have recently been studied by the distributed ledger community: Herlihy [28] provided a first extensive analysis of the scheme, Borkowski et al. [29] surveyed atomic swaps for distributed ledgers.

The HTLC protocol consists of the following steps. The users have accounts (wallets) on two disconnected ledgers executing smart contracts. Consider that Alice wants to send assets on Ledger 1 to Bob, in exchange for assets on Ledger 2 from Bob. At Step 1, Alice initiates the transaction by generating a secret (a key) that will be used to unlock the asset transfers later on. She then deploys a smart contract on Ledger 1, that will lock her assets until time $Ta$, this contract will transfer to Bob the assets only if the secret generated by Alice is revealed, or input in the smart contract. To verify the secret, Alice reveals its hash as a part of the smart contract (cf. Figure 1(a) Step 1). One important functionality of this contract is that after time $Ta$, if the secret has not been revealed, then the smart contract expires and the assets will be unlocked and will return to Alice's wallet.

Next, Bob can verify the contract deployed by Alice on Ledger 1 (assets, delivery address, etc) and use the hash submitted by Alice in order to deploy a similar contract on Ledger 2 (cf. Figure 1(a) Step 2), specifying the amount he is willing to transfer to Alice and expiry time $Tb$, with $Tb < Ta$, until which the assets are locked on Ledger 2. At Step 3, Alice can verify the contract deployed on Ledger 2, unlock the assets, and initiate their transfer to her wallet by revealing the secret on Ledger 2. Now, since the secret was revealed on Ledger 2, Bob can use it to unlock the assets on Ledger 1 and complete the cross-ledger transaction.

Note that it is important that the date until which the assets are locked on Ledger 2 is smaller than the date until which they are locked on Ledger 1. This ensures that after the assets on Ledger 2 are unlocked and the secret is revealed, Bob still




(a) Hash time lock contract (HTLC)

(b) Packetized payment (PP)

Fig. 1. Cross-ledger atomic swap protocols.

has time to unlock the assets on Ledger 1 and complete the transaction.

In the best case scenario, this mechanism enables atomic cross-ledger exchange of the assets without relying on a trusted party and without connection between ledgers. If Alice does not unlock the assets on Ledger 2 before $Tb$, then the assets are transferred back to Bob, thus, she has no incentive to reveal the secret as that would allow Bob to execute the smart contract on Ledger 1 and transfer the assets to his wallet while keeping his assets on Ledge 2. In turn, once the secret is revealed, Bob's assets are transferred to Alice and he should execute the smart contract on Ledger 1 in order to complete the transaction, otherwise he transferred his assets without receiving Alice's assets.

However some attacks, such as DDoS or secret hack, are still possible. Moreover, the asset price volatility and malicious behaviour from the agents driven by an attempt to maximize financial profits could negatively affect the transaction counterpart. For instance, if Alice becomes inactive before completion of the transaction, then the assets will be blocked on both ledgers. While it can be desirable and can be tolerated by an agent, this can incur significant losses to a counterpart.

### C. Packetized payments (PPs)

Aiming to address the problem of locked assets and the complexity of the HTLC's implementation, the packetized payment (PP) approach was recently proposed [30]. According to this approach, the total asset amounts to be traded are split into $m$ "economically-insignificant" amounts (cf. Figure 1(b) Step A). Next, as shown on Figure 1(b), Steps B-C, these small portions of assets will be sent on one and then on another blockchain sequentially: Steps B and C are to be repeated $m$ times in order to complete the transaction. Note that, at each iteration, the protocol may require the agent to match and extend the previous transfer such that the agents are alternatively exposed to counterparty risk. If not, then there is a transfer leader who agrees to solely expose herself or himself to a run by the other agent. If one agent behaves maliciously and does not execute the transfer when it is her or his turn, the counterpart agent loses *only* a fraction of the assets he was willing to trade. Therefore, PP upper bounds the amount of assets that can be lost by a fraction of the assets determined at Step A and prevents the whole amount of assets being blocked for a long period of time. It also prevents a potential loss of the whole amount of assets, while requiring only simple transfer transactions.

Lightening networks can be employed for micropayments [19] to avoid transaction fees multiplied by the number of the splits. However, this reintroduces the problem of the assets being locked and, in this case, in the form of collateral deposit on the escrow accounts of each agent on each blockchain: Alice and Bob will need to create two micropayment channels, one on top of each blockchain, and lock the collateral on each channel. In addition, if an honest agent does not receive a payment from a counterpart, and is willing to close the micropayment channel, the funds on the escrow account will be blocked for a certain blockchain-specific period of time [20].

Importantly, and as already noted in [30], PPs cannot be used to exchange non-fungible non-liquid assets such as CryptoKitties[2] or "digital twins" of physical goods.

## III. A Game-Theoretic Analysis

We first describe our game-theoretic framework and its assumptions. Then, we specialize it to the HTLC and PP protocols, and report the main results from our analysis.

### A. Framework

Two agents, Alice and Bob or $a$ and $b$, want to exchange one unit of asset 1, say one Altcoin, from $a$ for some units

[2]https://www.cryptokitties.co/

asset 2, say Ethers, from $b$. We assume that the asset 2 is the reference asset in which the agents value their goods. We denote $P_t$ the time-$t$ price of asset 1 expressed in units of asset 2, for example the price of one Altcoin in Ethers. We assume for simplicity that there is no interest rate or coin staking, meaning that the asset quantities do not increase by themselves whenever locked in a special wallet or account. Therefore, only the price of asset 1 is stochastic in our framework.

There are three possible times at which the agents may take actions: 0, 1, and 2. The price dynamics of asset 1 is given by

$$P_t = P_{t-1} \pm \delta \tag{1}$$

with equal probability of up and down moves, for some initial price $P_0 > 0$ and some constant $\delta > 0$ such that $\delta \leq P_0/2$ so that the price remains nonnegative during the game. Note that the asset price is a martingale, that is the expected value of next period's price is equal to the current price, $\mathbb{E}[P_t \mid P_{t-1}] = P_{t-1}$ for $t = 1, 2$.

There are three types of actions that the agents may take: continue $c$, wait $w$, and stop $s$. If an agent plays $s$ then the game is over and the transaction fails. If an agent plays $w$ then one time period passes and the price changes. If an agent plays $c$ then either it is the other player turn, or the transaction is completed. The agents take actions sequentially and the set of possible actions at a particular instant depends on the history of previous actions.

We assume that the agents are strategic and aim to maximize their interests which is a function of three terms: the transaction success, the financial profit resulting from the asset price change, and the total time spent, or locked, in the transaction. We assume that there are two types of agents: the honest or high type $h$, and the malicious or low type $l$. We model the agent incentives using a utility function as follows:

$$\mathcal{U}(i, j) = \alpha_{i,j} X + \beta_i XY - \gamma_{i,j} Z_i \tag{2}$$

for any agent $i \in \{a, b\}$ of type $\in \{h, l\}$, and where $X = 1$ indicates transaction success and $X = -1$ transaction failure, $Y$ is the profit and loss resulting from the asset price change and transfer, and $Z_i$ measures how long agent $i$ was engaged in the game. The constant $\alpha_{i,j} \geq 0$ measures the extend to which an agent is willing to complete the transaction. For example, if $\alpha_{i,j}$ is large then the agent will most likely prefer to complete the transaction despite an adverse price change. We set $\beta_b = 1$ and $\beta_a = -1$ modeling the agent opposite exposures to price changes. The constant $\gamma_{i,j} > 0$ is a measure of how impatient the agent is to terminate transaction. Note that, if the transaction fails $X = -1$ then Alice is positively exposed to $Y$ as asset 1 was not transferred to Bob since $\beta_a X = 1$ in this case.

In this work, we assume that a malicious agent only cares about profits resulting from a price change while being locked-in the transaction as shortly as possible. On the other extreme, we assume that an honest agent always acts to complete the transaction as rapidly as possible. We formalize the two types in the following definition.

TABLE I
NOTATIONS SUMMARY.

| Notation | Description |
|---|---|
| $a$ and $b$ | Alice and Bob |
| $h$ and $l$ | honest and malicious |
| $c$, $w$, and $s$ | actions: continue, wait, and stop |
| $w_\pm$ | price moves up/down after $w$ |
| $\mathcal{T}_i$ | agent $i$ type |
| $\mu_i$ | honest agent $i$ percentage, $\mathbb{P}[\mathcal{T}_i = h]$ |
| $\mathcal{A}(j, \mathcal{H})$ | agent type $j$ action after $\mathcal{H}$ |
| $X$ | swap success (1) or failure (-1) |
| $Y$ | financial profit and loss |
| $Z_i$ | agent $i$ lock-in time |
| $\alpha_{i,j}$ | agent preference param. for swap success |
| $\gamma_{i,j}$ | agent impatience param. |

**Definition III.1** (Agent types)**.** *An agent of type $h$, namely* honest*, always chooses to play continue $c$. An agent of type $l$, namely* malicious*, satisfies the parameter condition $\alpha_{i,l} = 0$ for $i = a, b$.*

In Sections III-B and III-C we derive the optimal strategy of the malicious agent, and the conditions on $\alpha_{i,h}$ such that an agent is willingly honest.

We denote $\mu_i$ the fraction of honest agents $i$ and, thus, $1 - \mu_i$ the fraction of malicious agents $i$ for $i \in \{a, b\}$. The agents meet at random and they do not know whether the other agent is malicious or not. Besides that, the agents have full information about their environments.

We now describe some notations, they are also summarize in Table I. We write $\mathbb{E}[\mathcal{X} \mid \mathcal{Y}]$ the expected value of the variable $\mathcal{X}$ given the history of actions and possibly other refinements $\mathcal{Y}$. We use brackets to denote the history of actions, for examples $\{\emptyset\}$ for no action taken and $\{c, w, c\}$ for continue–wait–continue actions. Which agent played a particular action and whose turn it is to play next will be clear from the game descriptions. As the price of asset 1 is stochastic, we sometimes emphasize the price change after an agent played $w$, for example if the action is $w$ we may write the history $w_+$, or $w_-$, if we consider only the scenario in which the price went up, or down respectively. We write $\mathcal{T}(i)$ the type of agent $i$, for example $\mathcal{T}_a = h$ means that Alice is honest. We denote $\mathcal{A}(j, \mathcal{H})$ the best response, or action taken, by an agent of type $j$ following the history $\mathcal{H}$ and is defined as the action maximizing her or his expected utility.

**Remark III.2.** *The term $\gamma_{i,j} Z_i$ is assumed to be small in comparison to the others terms in the agent's utility function. An economic justification is that we think of the temporal immediacy of the transaction as a lesser concern than its success or value change. Still, the existence of this term remains important as, the agents being risk-neutral with respect to the price change and the price being a martingale, it gives them some incentive to complete the transaction sooner rather than later.*

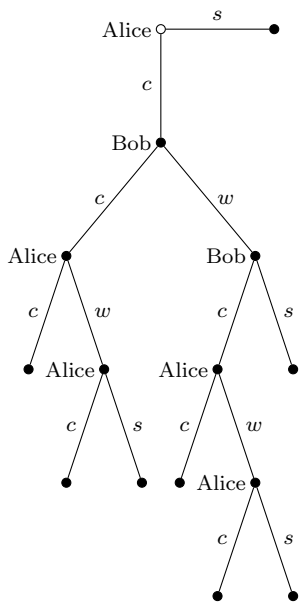



Fig. 2. HTLC sequence of actions (The empty circle indicates the root node).

TABLE II
VALUES OF $Z_a$, $Z_b$, GIVEN FINAL HISTORY OF ACTIONS.

| Actions History | $Z_a$ | $Z_b$ |
|---|---|---|
| $\{s\}$, $\{c,c,c\}$ | 0 | 0 |
| $\{c,w,c,c\}$, $\{c,c,w,c\}$ | 1 | 1 |
| $\{c,w,s\}$ | $2+\epsilon$ | 1 |
| $\{c,w,c,w,c\}$ | 2 | 2 |
| $\{c,c,w,s\}$ | $2+\epsilon$ | $1+\epsilon$ |
| $\{c,w,c,w,s\}$ | $2+\epsilon$ | $2+\epsilon$ |

### *B. An HTLC game*

The sequence of actions for the HTLC are described in Figure 2. The first action is taken by Alice as she should lock-in her asset for a period of length $2+\epsilon$ for some infinitesimal $\epsilon$. Then Bob should similarly lock-in his asset, for a smaller period of length $1+\epsilon$, and has the possibility to wait one period. Then, if Bob locked-in his contract, Alice is left to complete the transaction but also has the possibility to wait one period. We assume that the price does not change between time $t$ and $t+\epsilon$ for $t=0,1,2$. The small $\epsilon$ term guarantees that the assets are still locked on the action time.

The variable $Y$ is given by the profit and loss on asset 1, that is $Y = P_\tau - P_0$ where $\tau$ is the time at which the transaction ends. The possible values taken by variable $Z_i$ for $i=a,b$ are summarized in Table II. Although stylized, the HTLC setup has 18 possible outcomes taking into consideration the asset price scenarios.

Alice has full control on whether the transaction succeeds or fails in the last part of the transaction. We start by characterizing the behavior of malicious Alice in this situation.

**Proposition III.3** (Malicious Alice – final actions)**.** *We have that* $\mathcal{A}(l,\{c,c\}) = \{w_+,s\}\cup\{w_-,c\}$, $\mathcal{A}(l,\{c,w_-,c\}) = c$, *and* $\mathcal{A}(l,\{c,w_+,c\}) = \{w_+,s\}\cup\{w_-,c\}$ *if and only if*

$$\delta > \frac{1}{2}\gamma_j(3+\epsilon).$$

This shows that a malicious Alice who is not impatient, or who expects reasonably large price movement, will always wait and complete the transaction only if the price change moves at her advantage, or will complete it immediately if the asset price already moved at her advantage when Bob waited beforehand.

In the following proposition we show that Bob can also act maliciously.

**Proposition III.4** (Malicious Bob)**.** *We have that* $\mathcal{A}(l,\{c\}) = \{w_+,c\}\cup\{w_-,s\}$ *if and only if*

$$\mu_a > \frac{\gamma_{b,l}(1+\epsilon/2)}{2\delta+\gamma_{b,l}(1+\epsilon/2)}.$$

Having in mind that $\gamma_{b,l}$ is presumably small, if the price change $\delta$ is large, then a malicious Bob does not even require a large fraction of honest agents $\mu_a$ to exploit the protocol at his profit. Indeed, by waiting Bob can ensure that the transaction will continue only when the price moves in the direction that benefits him. We now characterize existence conditions for an honest Bob.

**Proposition III.5** (Honest Bob)**.** *We have that* $\mathcal{A}(h,\{c\}) = c$ *if and only if*

$$\max\left(\delta,\ \frac{(1-\mu_a)\gamma_{b,h}(1+\epsilon/2)-2\mu_a\delta}{1+\mu_a}\right) < \alpha_{b,h}$$

*and*

$$\alpha_{b,h} < 2\delta - \gamma_{b,h}(1+\epsilon/2) + \frac{\gamma_{b,h}}{(1-\mu_a)}.$$

As expected, the first inequality indicates that the preference parameter $\alpha_{b,h}$ of honest Bob must be large enough so that he plays continue despite possible predatory actions from Alice. Interestingly, the second inequality indicates that $\alpha_{b,h}$ should not be too large when the probability of Alice being honest $\mu_a$ is small. This is because then Bob can increase the probability of transaction success by playing $\mathcal{A}(h,\{c\}) = \{w,c\}$. Indeed, at time 0 the probability that malicious Alice plays $\{w,c\}$ is 50% whereas at time 1 it is 75%.

Alice can also behave honestly, but as shown in the next proposition her preference for complete transactions must be large.

**Proposition III.6** (Honest Alice)**.** *We have that* $\mathcal{A}(h,\{c,c\}) = c$, *and* $\mathcal{A}(h,\{c,w,c\}) = c$ *if and only if*

$$\alpha_{a,h} > \max\left(\delta - \frac{1}{2}\gamma_{a,h}(1+\epsilon),\ 2\delta - \frac{1}{2}\epsilon\right).$$

Still having in mind that $\gamma_{a,h}$ is small and that $0 \le \mu_b \le 1$, this suggests that the commitment to play $c$ requires a preference parameter $\alpha_{a,h}$ of value larger than $\approx 2\delta$ which may be economically large.

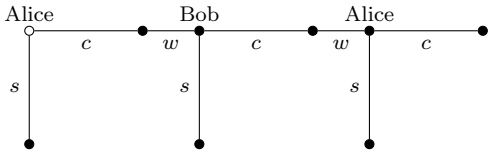



Fig. 3. Packetized payment short sequence of actions (The empty circle indicates the root node).

A malicious Alice may always be willing to play $c$ at the initial time as shown in next proposition.

**Proposition III.7** (Malicious Alice – first action)**.** *We have that $\mathcal{A}(l, \{\emptyset\}) = c$ if and only if*

$$\mu_b > \frac{-\delta + (3/2)\gamma_{a,l}(1+\epsilon)}{3\delta + (3/2)\gamma_{a,l}}$$

This shows that a malicious Alice will in general be willing to enter the transaction as long as her impatience parameter $\gamma_{a,l}$ remains small.

Finally, we can give an estimation on the failure rate of this type of transaction as a function of the honest agent percentages.

**Corollary III.8** (Transaction failure probability)**.** *Assuming that agents are defined as in Propositions III.3 to III.7, then the percentage of incomplete transactions is* $1 - (1 + \mu_a + \mu_b + \mu_a\mu_b)/4$.

It is immediate that if there is no malicious agent then the failure rate is 0% but, interestingly, it upper bounded by 75% in our model. This is because the malicious agents may decide to trade at zero price profit in order to unlock their asset sooner.

### C. A short packetized payment game

Packetized payments split the transaction into small transfers where each agent exposed herself or himself to a one-way transfer alternatively. At any point in time, one agent may decide not to transfer furthermore and stop the transaction. As a consequence, the variable $Y$ depends on the exit time and is given by $Y_0 = 0$ , $Y_{N+1} = P_{t_N} - P_0$, and

$$Y_n = \begin{cases} \frac{n}{N}P_{t_n} - \frac{n-1}{N}P_0 & \text{if } n \text{ is odd} \\ \frac{n-1}{N}P_{t_n} - \frac{n}{N}P_0 & \text{if } n \text{ is even} \end{cases} \quad \text{for } n = 1, \dots, N \tag{3}$$

where the subscript $n$ indicates the current step of the transaction, and $t_n$ indicates the time at which the $n$-th step takes place. For clarity of exposition, we study an atomic swap split in 3 payments so that $t_n = n$ for $n = 0, 1, 2$. Still, this setup is sufficient to illustrate the functioning of packetized payments.

The sequence of actions for the packetized payment game are described in Figure 3. In summary, Alice transfers half of the asset to Bob, then Bob makes the whole $P_0$ payment, and finally Alice transfers the remaining half of the asset. The agents can only decide to continue $c$ or stop $s$ the transaction. However, when an agent plays $c$ we assume that the transaction also waits $w$, or paused, for a short time period over which the asset price changes. It is here implicitly assumed that the parameter $\delta$ is smaller than in Section III-B. Since time is mechanic in this setup and does not dependent on the agents' actions, we simplify further the setup as follows.

**Assumption III.9** (Time irrelevance)**.** *We assume that $\gamma_{i,j} = 0$ for $i \in \{a, b\}$ and $j \in \{h, l\}$.*

The first and striking result is that malicious agents, either Bob or Alice, will never complete the transaction. Indeed, there is no incentive for an agent who only cares about its financial profit to complete the transaction, as shown in the following proposition.

**Proposition III.10** (Malcious Alice and Bob)**.** *We have that $\mathcal{A}(l, \{c, w\}) = s$ and $\mathcal{A}(l, \{c, w, c, w\}) = s$.*

From this result we can also infer the percentage of failed transactions.

**Corollary III.11** (Transaction failure probability)**.** *Assuming that both malicious and honest agents participate in the transaction, then the percentage of incomplete transactions is* $1 - \mu_b\mu_a$.

We derive a condition such that Bob is honest.

**Proposition III.12** (Honest Bob)**.** *Bob is honest if and only if*

$$\mu_a > \frac{2P_0}{4\alpha_{b,h} + P_0 - \delta}$$

We see that if the price is more volatile, larger $\delta$, then a larger fraction of honest $a$ agents is required. Interestingly, even with no price movement $\delta = 0$ and only honest $a$ agents, we see that Bob preference parameter for completed transaction must be fairly large in value, $\alpha_{b,h} > P_0/4$. We now derive conditions such that Alice is honest.

**Proposition III.13** (Honest Alice)**.** *Alice is honest if and only if*

$$\alpha_{a,h} > \frac{1}{4}(P_0 + 2\delta) \quad \textit{and} \quad \mu_b > \frac{P_0}{4\alpha_{a,h} + P_0}. \tag{4}$$

We see that the conditions for Alice to be honest are more stringent than for Bob even in a setup when $\mu_a = \mu_b = 1$. Notably if the percentage of honest Bob becomes very small, $\mu_b \to 0$, then there cannot exist any honest Alice unless $\alpha_{a,h} \to +\infty$.

## IV. Discussion

In this section, we summarize the main results from the game-theorethic approach and we discuss mechanisms that could be used to prevent malicious behaviour in atomic swap

### A. HTLC

We illustrated that Bob can also behave maliciously in the HTLC protocol. This indicates that, at the initial time, Alice holds a contingent American option at time zero, where the contingency is a function of Bob's type and of the asset price. In our framework, Alice actually holds an American knock-in barrier option if Bob is malicious, and an American option

if Bob is honest. In addition, we showed that the economic capital to match Bob's and, in particular, Alice's preferences $\alpha_{i,h}$ for successful transaction are large. Indeed, it is of the order $\delta$ with a majority of honest agents $\mu_a \approx \mu_b \approx 1$ and therefore depends on the asset price volatility. Finally, if the proportion of honest agents becomes small, then it becomes absurd for other agents to behave honestly.

From a practical perspective, HTLCs are relatively complex and apply only to ledger supporting smart contracts. Hence, they may be costly to run and possible challenging to combine with additional disciplinary mechanisms.

### B. Packetized payments

We showed that there is no incentive for malicious agents to complete the transaction in packetized payments. Indeed, it is always optimal for them to leave the transaction with an additional "free" small piece of assets. As a consequence, the transaction failure rate is typically large and the economic incentive for agents to behave honestly are also large. Also, it is important to note that no matter how small the packet transfers are, a malicious agent can enter many transactions in parallel in order to make large profits.

From a practical perspective, PPs are relatively simple, but require many transfers, whose total cost is therefore uncertain. They do not require ledgers supporting smart contracts but do not apply to swap with indivisible tokens. It is also worth noticing that there may be small delays between transfers for network validation, which in turn lead to price fluctuations, as described in the PP game.

### C. Insentivizing honest behavior

*a) Collateral deposit:* Using collateral deposits to reduce the risk of agents exposed to adverse behaviour of other agents is not new. Zamyatin et al. [22] suggested using a collateral at least equal to the assets locked on the blockchain for a trade. They also proposed overcollateralization and a liquidation mechanism to mitigate extreme price fluctuations for short and long term cross-ledger transaction. While this ensures that economically rational agents have no incentive to misbehave, a disadvantage of this solution is that if an agent would like to transfer all his assets of one kind, he will be obliged to execute multiple transactions, each with an amount (approximately) equal to a half of the amount of the assets he currently possesses.

Based on the proposed game-theoretic model, it can be shown that a marginal amount of collateral is sufficient to prevent agents to behave maliciously. We modify the frameworks of Section III so as to require agents to put collateral which is lost if they exit the transaction without completing it. The following Propositions that it extinguishes malicious behaviors in our framework.

**Proposition IV.1** (HTLC with collateral)**.** *Assume that Alice and Bob put $\delta$ of collateral, then it is optimal for malicious players to always continue the transaction without waiting in the HTLC game described in Section III-B.*

**Proposition IV.2** (PP with collateral)**.** *Assume that Bob puts a collateral larger than $(P_0+\delta)/2$, and Alice puts a collateral larger than $(P_0+2\delta)/2$, then it is optimal for malicious players to continue the transaction in the packetized payment game described in Section III-C.*

Two relevant observations can be made for real-world applications. First, in both cases the minimum collateral requirement involve the term $\delta$ which suggests that collateral demand should be a function of the asset price volatility, which is known to be time varying. Second, for the packetized payment, the initial collateral involves the fractional transfer value $P_0/2$, or $P_{t_n}/N$ in general, which suggests that the collateral requirement can be small, with $N$ large, but should be adjusted dynamically as the asset price changes. Indeed, the price can varies up or down to $P_0 \pm N\delta$ in extreme scenarios, but will in general fluctuate significantly less.

*b) Reputation mechanism:* We always assumed that an agent cannot predict the strategy of his or her counterpart ex-ante as the agent types, malicious or honest, are not observable. However, in reality, if an agent trades regularly with another agent that it can identify, or if an agent has some information on the previous behavior of another agent, then a self selected agent matching can occurs instead of a random one. In principle, as all the transactions executed on a ledger can generally be seen, analysis of a transaction history of an agent can be analyzed to build his reputation. However, computation of such reputation value is problematic in a case of permissionless blockcahins for several reasons. First, an agent can create multiple accounts and attempt to preserve his anonymity. Even though, it has been shown that deanonymization is possible [31], one cannot guarantee a perfect mapping between one user and all his transactions, in case of multiple accounts. Second, it may not always be possible to distinguish a cross-ledger transaction from a single-chain transaction. However, if these two challenges are addressed, thanks to the book-keeping property and immutability of a ledger, using reputation mechanism to complement existing protocols.

## V. Conclusion and future work

We introduced a game-theoretic approach to model agent behaviors in cross-ledger transactions such as hash time lock contracts and packetized payments. We derive conditions for agents to behave honestly or maliciously, as well as different measures of economic and of transaction success. We proposed to dynamically compute and adjust the collateral amounts in order to enforce honest behaviors among agents, and we discussed the implementation challenges of reputation systems as a disciplinary mechanism.

An important observation is that cross-ledger atomic swap trustless protocols should use disciplinary mechanisms such as collateral deposit. Future research will therefore study the implementation, cost, performance, and complexity of various protocols on permissionless blockchains supporting smart-contract functionality, such as Ethereum and Neo[32]. From

a theoretical perspective, it will be important to model the agent strategies in more complex protocols in order to ensure their robustness to misbehavior. Indeed, our framework is only a first step to a consistent comparative analysis of different protocols. For example, which protocol agents would select and why if they were given the choice between multiple ones. In addition, more realistic features can be brought into our framework. For example, blockchain transaction fess or coin stacking (which is similar to earning dividends or interests on a locked-in asset) may have an impact on the agents' actions.

## REFERENCES


[1] H. Garcia-Molina, "Using semantic knowledge for transaction processing in a distributed database," *ACM Transactions on Database Systems (TODS)*, vol. 8, no. 2, pp. 186–213, 1983.

[2] T. Moore and N. Christin, "Beware the middleman: Empirical analysis of bitcoin-exchange risk," in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 25–33.

[3] "Details of $5 million bitstamp hack revealed," accessed 2018-04-17. [Online]. Available: https://www.coindesk.com/unconfirmed-report-5-million-bitstamp-bitcoin-exchange

[4] "Crypto exchange bitfinex bounces back after a ddos attack," accessed 2018-04-17. [Online]. Available: https://www.ccn.com/crypto-exchange-bitfinex-bounces-back-after-a-ddos-attack

[5] "itbit," accessed 2018-04-17. [Online]. Available: https://www.itbit.com/otc

[6] "Hiveex-large volume cryptocurrency otc brokerage," accessed 2018-04-17. [Online]. Available: https://www.hiveex.com

[7] M. J. Osborne and A. Rubinstein, *A course in game theory*. MIT press, 1994.

[8] A. Etheridge and M. Baxter, *A course in financial calculus*. Cambridge University Press, 2002.

[9] "Bisq white paper," accessed 2019-04-15. [Online]. Available: https://docs.bisq.network/exchange/whitepaper.html

[10] "Decred-compatible cross-chain atomic swapping," https://github.com/decred/atomicswap/, 2018, last accessed 2019-04-07.

[11] "Komodo white paper," accessed 2019-04-15. [Online]. Available: https://komodoplatform.com/wp-content/uploads/2018/06/Komodo-Whitepaper-June-3.pdf

[12] "0x white paper," accessed 2018-04-17. [Online]. Available: https://0x.org/pdfs/0x_white_paper.pdf

[13] "Building super financial markets for the new economy," https://wanchain.org/files/Wanchain-Whitepaper-EN-version.pdf, accessed 2019-02-05.

[14] S. Thomas and E. Schwartz, "A protocol for interledger payments," *URL https://interledger. org/interledger.pdf*, 2015.

[15] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret: Theory and applications of ring signatures," in *Theoretical Computer Science*. Springer, 2006, pp. 164–186.

[16] "Btc relay," accessed 2018-04-17. [Online]. Available: https://github.com/ethereum/btcrelay

[17] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," *URL: http://www.opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains*, 2014.

[18] S. Johnson, P. Robinson, and J. Brainard, "Sidechains and interoperability," *arXiv preprint arXiv:1903.04077*, 2019.

[19] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," 2016.

[20] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 17–30.

[21] A. Miller, I. Bentov, R. Kumaresan, and P. McCorry, "Sprites: Payment channels that go faster than lightning," *arXiv preprint arXiv:1702.05812*, 2017.

[22] A. Zamyatin, D. Harz, J. Lind, P. Panayiotou, A. Gervais, and W. J. Knottenbelt, "Xclaim: Interoperability with cryptocurrency-backed tokens," Tech. Rep., 2018.

[23] "Bitcoin Wiki: Atomic cross-chain trading," https://en.bitcoin.it/wiki/, last accessed 2019-02-05.

[24] J. Kirsten and H. Davarpanah, "Anonymous atomic swaps using homomorphic hashing," *Available at SSRN 3235955*, 2018.

[25] G. Zyskind, C. Kisagun, and C. Fromknecht, "Enigma catalyst: A machine-based investing platform and infrastructure for crypto-assets," 2018.

[26] J. A. Liu, "Atomic swaptions: Cryptocurrency derivatives," *arXiv preprint arXiv:1807.08644*, 2018.

[27] T. Koensa and E. Polla, "Assessing interoperability solutions for distributed ledgers," accessed 2018-04-17. [Online]. Available: https://www.ingwb.com/media/2667864/assessing-interoperability-solutions-for-distributed-ledgers.pdf

[28] M. Herlihy, "Atomic cross-chain swaps," in *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*. ACM, 2018, pp. 245–254.

[29] M. Borkowski, D. McDonald, C. Ritzer, and S. Schulte, "Towards atomic cross-chain token transfers: State of the art and open questions within tast," *Distributed Syste ms Group, TU Wien (Technische Universität Wien), Vienna, Austria, Tech. Rep*, 2018.

[30] D. Robinson, "HTLCs considered harmful, presented at Stanford Blockchain Conference (SBC) '19," http://diyhpl.us/wiki/transcripts/stanford-blockchain-conference/2019/htlcs-considered-harmful/, 2019, last accessed 2019-02-05.

[31] A. Biryukov, D. Khovratovich, and I. Pustogarov, "Deanonymisation of clients in bitcoin p2p network," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 15–29.

[32] "Neo white paper," accessed 2019-04-15. [Online]. Available: https://docs.neo.org/en-us/whitepaper.html

## APPENDIX A
## PROOFS

Most of the arguments in the proofs here-below follow from the hypothesis that an agent always takes the actions which maximize her or his expected utility, taking into account future and possibly adversarial actions from the other agent. We always describe the key conditions (inequalities) to be verified but provide limited details on the derivations as they can be long and tedious. We sometimes refer to agent $i$ of type $j$ by $(i,j)$ for brevety. Note that Alice and Bob must take into account the likelihood that they are trading with a malicious or with a honest agent. For example, this means that $(b,j)$'s expected utility given the history of actions $\mathcal{H}$ is given by

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(b,j) \mid \mathcal{H}\right] = \mu_a \mathbb{E}\left[\mathcal{U}(b,j) \mid \mathcal{H} \cap \{\mathcal{T}_a = h\}\right] \\ + (1-\mu_a)\mathbb{E}\left[\mathcal{U}(b,j) \mid \mathcal{H} \cap \{\mathcal{T}_a = l\}\right]\end{aligned}$$

where $\mathbb{E}\left[\mathcal{U}(b,j) \mid \mathcal{H} \cap \{\mathcal{T}_a = l\}\right]$ denotes the expected utility of $(b,j)$ under the assumption that $a$ is malicious, and so on.

*a) Proof of Proposition III.3:* We have to show that the described final actions are indeed optimal for the agent $(a,l)$,

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,w_+,s\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,w_+,c\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,w_-,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,w_-,s\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,w_+,s\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,w_+,c\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,w_-,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,w_-,s\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w_+,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w_+,s\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w_-,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w_-,s\}\right]\end{aligned}$$

Then, we have to show that it is optimal for $(a,l)$ to wait, or not, given the above terminal actions, namely that the following inequalities hold

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,w\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,c,c\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,w\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_+,c,c\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w\}\right]\end{aligned}$$

Note that, for any penultimate history $\mathcal{H}_*$, we have

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(a,l) \mid \{\mathcal{H}_*,c\}\right] &= P_0 - P_\tau - \gamma_{a,l}\tau \\ \mathbb{E}\left[\mathcal{U}(a,l) \mid \{\mathcal{H}_*,s\}\right] &= P_\tau - P_0 - \gamma_{a,l}(2+\epsilon)\end{aligned}$$

where $\tau = 0,1,2$ is the possible final time. The expected utility of $(a,l)$ if she waits is in turn given by

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,x,c,w\}\right] = 0.5\,\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,x,c,w_+,s\}\right] \\ + 0.5\,\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,x,c,w_-,c\}\right]\end{aligned}$$

for $x \in \{\emptyset, w_+\}$, and by

$$\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w\}\right] = \mathbb{E}\left[\mathcal{U}(a,l) \mid \{c,w_-,c,w,c\}\right]$$

Writing explicitly the above expressions and taking the most stringent inequalities leads to the desired parameter conditions.

In the following two proofs, we will make use of the fact that $(a,h)$ always plays $c$.

**Lemma A.1.** *For any $x \in \{\emptyset, w_+, w_-\}$ and $j \in \{h,l\}$,*

$$\mathbb{E}\left[\mathcal{U}(b,j) \mid \{c,x,c\} \cap \{\mathcal{T}_a = h\}\right] = \mathbb{E}\left[\mathcal{U}(b,j) \mid \{c,x,c,c\}\right].$$

*b) Proof of Proposition III.4:* We have to verify the optimality of $(b,l)$'s actions given the specification of $(a,l)$ in Proposition III.3 and given that $(a,h)$ always plays $c$, namely that the following inequalities hold

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,w\}\right] &> \mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,c\}\right] \\ \mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,w_+,c\}\right] &> \mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,w_+,s\}\right] \\ \mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,w_-,s\}\right] &> \mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,w_-,c\}\right]\end{aligned}$$

*c) Proof of Proposition III.5:* With the parameter conditions one can show that

$$\mathbb{E}\left[\mathcal{U}(b,h) \mid \{c,c\}\right] > \mathbb{E}\left[\mathcal{U}(b,h) \mid \{c,w\}\right].$$

*d) Proof of Proposition III.6:* With the parameter conditions one can show that

$$\begin{aligned}\mathbb{E}\left[\mathcal{U}(a,h) \mid \{c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,h) \mid \{s\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,h) \mid \{c,c,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,h) \mid \{c,c,w\}\right] \\ \mathbb{E}\left[\mathcal{U}(a,h) \mid \{c,x,c,c\}\right] &> \mathbb{E}\left[\mathcal{U}(a,h) \mid \{c,x,c,s\}\right]\end{aligned}$$

for $x = w_+$ and $x = w_-$.

*e) Proof of Proposition III.7:* With the parameter conditions one can show that

$$\mathbb{E}\left[\mathcal{U}(a,l) \mid \{c\}\right] > \mathbb{E}\left[\mathcal{U}(a,l) \mid \{s\}\right]$$

*f) Proof of Corollary III.8:* The transaction succeeds in four cases with probabilities:

$$\begin{aligned}\mathbb{P}[\mathcal{T}_a = \mathcal{T}_b = h] &= \mu_a\mu_b \\ \mathbb{P}[\{\mathcal{T}_a = h\} \cap \{\mathcal{T}_b = l\} \cap \{c,w_+,c,c\}] &= (1-\mu_b)\mu_a/2 \\ \mathbb{P}[\{\mathcal{T}_a = l\} \cap \{\mathcal{T}_b = h\} \cap \{c,c,w_-,c\}] &= \mu_b(1-\mu_a)/2\end{aligned}$$

and

$$\begin{aligned}\mathbb{P}[\{\mathcal{T}_a = l\} \cap \{\mathcal{T}_b = l\} \cap \{c,w_+,c,w_-,c\}] = \\ (1-\mu_b)(1-\mu_a)/4.\end{aligned}$$

*g) Proof of Proposition III.10:* This is immediate. At time 2 if $\mathcal{T}_a = l$ then Alice loses $P_2/2$ in utility by playing $c$ instead of $s$. Similarly, at time 1 if $\mathcal{T}_b = l$ then Bob gets $P_1/2$ in utility by playing $s$ whereas he expect to receive $\mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,c\}\right] = \mu_a(P_1 - P_0) + (1-\mu_a)(P_1/2 - P_0)$ if he plays $c$. We have $\mathbb{E}\left[\mathcal{U}(b,l) \mid \{c,c\}\right] < P_1/2$ since $\delta < P_0/2$ and $\mu_a \leq 1$, hence a malicious Bob plays $s$.

*h) Proof of Corollary III.11:* The transaction succeeds only if Alice and Bob are honest which happens with probability $\mathbb{P}[\mathcal{T}_a = \mathcal{T}_b = h] = \mu_a\mu_b$.

*i) Proof of Proposition III.12:* We have $\mathbb{E}\left[\mathcal{U}(b,h) \mid \{c,w,c\}\right] = \alpha_{b,h} + \mu_a(P_1 - P_0) + (1-\mu_a)(0.5P_1 - P_0)$ and $\mathbb{E}\left[\mathcal{U}(b,h) \mid \{c,w,s\}\right] = -\alpha_{b,h} + 0.5P_1$. We obtain that $\mathcal{A}(b,\{c,w\}) = c$ by taking $P_1 = P_0 - \delta$.

*j) Proof of Proposition III.13:* We have $\mathcal{A}(h,\{c,w,c,w\}) = c$ if and only if $\alpha_{a,h} + P_0 - P_2 > -\alpha_{a,h} + P_0 - P_2/2$ which is equivalent to $\alpha_{a,h} > (P_0 + 2\delta)/4$. Then, with $\mathcal{A}(h,\{c,w,c,w\}) = c$, we have that $\mathbb{E}\left[\mathcal{U}(a,h) \mid \{c\}\right] = \mu_b\alpha_{a,h} + (1-\mu_b)(-\alpha_{a,h} - P_0/2)$ and $\mathbb{E}\left[\mathcal{U}(a,h) \mid \{s\}\right] = -\alpha_{a,h}$. Therefore, for agent $a$ to be honest it must also be that $\mu_b > P_0/(4\alpha_{a,h} + P_0)$

*k) Proof of Proposition IV.1:* One can show that with the collateral

$$\mathbb{E}\left[\mathcal{U}(i,l) \mid \{\mathcal{H}, w\}\right] < \mathbb{E}\left[\mathcal{U}(i,l) \mid \{\mathcal{H}, c\}\right]$$

for $i = b$ and $\mathcal{H} = \{c\}$, and for $i = a$ and $\mathcal{H} \in \{\{c, c\}, \{c, w, c\}\}$ because the loss of collateral anneals any potential asset price gain resulting from waiting one period. Hence it is optimal to always terminate the transaction as soon as possible.

*l) Proof of Proposition IV.2:* This is immediate as malicious players would never be able to make any profit by exiting prematurely the transaction.