Towards faster settlement in HTLC-based Cross-Chain Atomic Swaps

Subhra Mazumdar
TU Wien, Christian Doppler Laboratory
Blockchain Technologies for the Internet of Things
Vienna, Austria
subhra.mazumdar@tuwien.ac.at

Abstract—Hashed Timelock (HTLC)-based atomic swap protocols enable the exchange of coins between two or more parties without relying on a trusted entity. This protocol is like the American call option without premium. It allows the finalization of a deal within a certain period. This puts the swap initiator at liberty to delay before deciding to proceed with the deal. If she finds the deal unprofitable, she just waits for the time-period of the contract to elapse. However, the counterparty is at a loss since his assets remain locked in the contract. The best he can do is to predict the initiator’s behavior based on the asset’s price fluctuation in the future. But it is difficult to predict as cryptocurrencies are quite volatile, and their price fluctuates abruptly. We perform a game theoretic analysis of HTLC-based atomic cross-chain swap to predict whether a swap will succeed or not. From the strategic behavior of the players, we infer that this model lacks fairness. We propose Quick Swap, a two-party protocol based on hashlock and timelock that fosters faster settlement of the swap. The parties are required to lock griefing-premium along with the principal amount. If a party finds a deal unfavorable, he has the provision to cancel the swap. We prove that Quick Swap is more participant-friendly than HTLC-based atomic swap. Our work is the first to propose a protocol to ensure fairness of atomic-swap in a cyclic multi-party setting.

Index Terms—Cryptocurrencies, Atomic Swap, Hashed Timelock (HTLC), Griefing Attack, Game Theory, Griefing-Premium, Faster Settlement

I. INTRODUCTION

Centralized exchange enabled users to trade one cryptocurrency for the other. For example, Alice wants to exchange \( x_a \) coins for Bob’s \( y_b \) coins. The exchange can be done by involving a third party Carol, where both Alice can deposit \( x_a + x \) coins and Bob deposits \( y_b \) coins with Carol respectively. \( x \) is the service fee charged by Carol for offering the swap service. If Carol is honest, she will handover \( x_a \) coins to Bob, \( y_b \) coins to Alice and keep the service charge.

If she is malicious, she can just run away with Alice’s and Bob’s money. Thus, it become very important to have decentralized exchange of cryptocurrencies. With the introduction of Blockchain [Nakamoto(2008)], it is now possible to realize decentralized protocols for atomic swaps without relying on any trusted third party [TN1(2013)], [Thomas and Schwartz(2015)], [Herlihy(2018)], [Zamyatin et al.(2019)], [Zakhary et al.(2020)], [Tai et al.(2021)], [Moreno-Sanchez et al.(2020)], [Lys et al.(2021)], [Thyagarajan et al.(2022)], [Narayananam et al.(2022)]. Alice and Bob can now safely lock their coins and have rules encoded in the smart contract. Exchange of assets leads to change of ownership, it either succeeds or fails in entirety. Bitcoin-based blockchains primarily leverage on Hashed Timelock Contracts or HTLC [Herlihy(2018)], [Borkowski et al.(2019)], [Dai et al.(2020)], [Narayananam et al.(2022)] for exchanging Bitcoins (BTC) with other cryptocurrencies like Litecoins (LTC), Ether (ETH), ERC Tokens.

Fig. 1: HTLC-based atomic swap. We assume that there is no waiting involved when parties lock their coins and are willing to exchange it as well.

We explain a two-party HTLC-based atomic swap protocol
with an example shown in Fig. 1. Alice wants to exchange \( x_a \) coins for \( y_b \) coins of Bob. Both of them have accounts in two different blockchains Chain-a and Chain-b. Alice samples a secret \( s \), generates a hash \( H = \mathcal{H}(s) \) and shares it with Bob at time \( t_0 \). The protocol comprises two phases: Lock and Claim. Lock phase defines the time interval within which the parties have locked their assets in the contracts instantiated in the respective blockchains. Alice locks \( x_a \) coins in Chain-a at time \( t_1 \). The coins are locked in a contract where the spending conditions are as follows: if Bob provides the secret \( s \) within \( t_a \) units of time, then he claims \( x_a \) coins else Bob initiates a refund after \( t_a \) elapses. Once the transaction is confirmed in Chain-a in the next \( t_a \) units of time, Bob locks \( y_b \) coins in Chain-b at time \( t_2 \). He uses the same hash value \( H \) in the contract deployed in Chain-b for locking his coins. The spending conditions are different: if Alice provides the secret \( s \) within \( t_b \) units of time, then she claims \( y_b \) coins else Alice initiates a refund after \( t_b \) elapses, where \( t_a > t_b \). It takes \( t_b \) units for the lock transaction to be confirmed in Chain-b. Claim phase signals the period within which the parties claim their assets. In the best case involving zero waiting, Alice broadcasts the claim transaction at \( t_3 \), releases the preimage \( s \) and claims \( y_b \) coins from Bob. The latter uses the preimage \( s \) at broadcasts a transaction to claim \( x_a \) coins at time \( t_4 \), where \( t_4 - t_3 = t_v \) is the time taken by Bob to observe Alice’s transaction in Chain-b. Once the ownership of the assets changes successfully, an instance of the protocol succeeds.

A. Griefing Attack in Timelocked Contracts

One of the main disadvantages of HTLC-based atomic swap is that parties are not enforced to settle the transaction. It has already been shown in [Han et al.(2019), zm(2018)] that atomic swap is equivalent to American call option without premium. In American call option, buyer is allowed to exercise the contract no later than the strike time. We illustrate the situation in Fig. 2 After Alice has locked \( x_a \) coins at \( t_1 \), the time taken for the transaction to be confirmed in Chain-a is \( t_a \). Ideally, Bob should begin locking his coins at \( t_2 = t_1 + t_a \). However, he may choose to delay speculate within duration \( t_2 \) to \( t_4 \) - \( \epsilon \) and delay or he may choose to not lock his coins at all. If Bob does not lock his coins, then Alice’s coins get locked unnecessarily and she loses her coins in paying fees to miners for successful mining of refund transaction. This is termed as draining attack [Eizinger et al.(2021)]. Let us assume that Bob locks his coins at \( t'_2 \) = \( t_2 + t_b \). The time taken for the transaction to be confirmed in Chain-b is \( t_b \). If Alice chooses not to delay, then she will claim the coins at time \( t'_4 = t'_2 + t_a \). However, Alice may delay in claiming the coins or just abort. If she chooses to claim the coins at time \( t''_4 = t'_4 - \epsilon \) where \( t''_4 = t'_2 + t_b \) then Bob gets to claim \( x_a \) coins at \( t''_4 = t'_4 + \epsilon \). Bob has to wait a duration of \( t''_4 - t'_4 \) where \( t''_4 = t'_4 + \epsilon \). Upon simplification, we observe that he waits \( t''_4 - t''_3 = t''_4 - \epsilon - t'_3 \), which is the delay due to Alice’s speculation. It may so happen that Alice does not want to claim \( y_b \) coins, she waits for \( t_b \) units to elapse, and Bob broadcasts her refund transaction at time \( t''_5 = t'_2 + t_b \). Alice broadcasts her refund transaction after \( t_a \) units elapse, i.e., at time \( t''_5 = t_1 + t_a \). If Alice chooses not to respond, then it leads to a Griefing Attack [Robinson(2019)]. Coins remain unutilized in either of the blockchains leading to substantial rise in opportunity cost. We define the attack formally.

**Definition 1:** (Griefing Attacks in Atomic Swap) Given two parties A and B, such that A is required to forward an HTLC of \( x_a \) coins to B for a certain timeperiod \( t_a \), and in turn, B must forward an HTLC of \( y_b \) coins to A for a timeperiod \( t_b : t_a > t_b \), a griefing attack can happen in the following situations:

- **A locks \( x_a \) coins and B doesn’t lock \( y_b \) coins:** It leads to A’s \( x_a \) coins being locked for \( t_a \) units, the loss in terms of collateral cost being \( O(x_a t_a) \).

- **A locks \( x_a \) coins, B has locked \( y_b \) coins, and A aborts:** In such a situation, A grieves B at the cost of locking his coins for \( t_a \) units of time. B’s coins remains locked for \( t_b \) units, the loss in terms of collateral cost being \( O(y_b t_b) \).

**Motivation for Griefing in Atomic Swap**

A party may grieve intentionally or decides to abort when the situation is not favourable for exchanging coins. We consider our parties to be either genuinely interested in exchanging coins or malicious. We define characteristic of each type:

- **Interested to Exchange:** A party who is willing to exchange coins but might end up grieving depending on whether she finds a favourable exchange rate.
Malicious: A party whose only motive is to mount Denial-of-Service (DoS) attack on the counterparty. Such a party will not take any actions after the counterparty has locked coins for exchange. Gain of malicious party is the lost opportunity cost of counterparty’s locked coins.

Enforcing the parties not to back out from the deal is a major challenge. Additionally, we observe HTLC lacks flexibility as it does not provide an option to cancel the contract when the situation turns out to be unfavorable. Is it possible to propose an atomic swap that allows cancellation but at the same time penalizes malicious behavior?

B. Contributions

- We model HTLC-based atomic cross-chain swap as a two-player sequential game and analyze the success rate of the protocol as a function of exchange rate and delay.
- We observe HTLC-based atomic swap is not participant-friendly by estimating the success rate of such a protocol.
- We propose Quick Swap, a hashlock and timelock-based protocol compatible with Bitcoin script. Our protocol is more robust and efficient, allowing easy cancellation of trade and penalizing a party if it grieves.
- Quick Swap can also be generalized to multi-party cyclic atomic swap for countering griefing attacks.

II. GAME THEORETIC ANALYSIS OF HTLC-BASED ATOMIC SWAP

A. System Model & Assumptions

The atomic swap protocol comprises two phase: Lock Phase - for the duration \( t_0 \) to just before \( t_3 \), and Claim Phase - from time \( t_3 \) onward. It proceeds sequentially, with the assets being locked first and then the assets being claimed in the next phase. Given two parties Alice and Bob, their strategy space consists of the actions continue and stop. In this paper, we will alternate use the term stop and abort, both denoting that a party chooses not to take any action. After Alice locks \( x_a \) coins in Chain-a, Bob can choose to either continue, i.e., lock \( y_b \) coins in Chain-b, or abort from the process. There is no way by which Alice can abort before timeout period \( t_a \) elapses, if Bob aborts. If Bob locks his coins before that, Alice can choose continue with the swap by claiming \( y_b \) coins and releasing the preimage of \( H \) before \( t_b \) elapses. If she choose to stop, she will wait for \( t_b \) to elapse and Bob initiates a refund after that. If either of the party is malicious, then the sole motive would be to keep the coins locked. So if Alice is malicious, she will not initiate the Claim Phase, and if Bob is malicious, he will simply abort in the Lock Phase without locking his coins.

B. Basic Setup

We denominate coins locked by Bob as a function of value of Alice’s coin at a given time \( t \). Thus we express \( y_b \) coins as \( x(y_b,t) \), where the price is decided based on the exchange rate prevailing at time \( t \). At time \( t_0 \approx t_1 \), \( x(y_b,t_1) \) is the price of Bob’s coins decided to be exchanged for \( x_a \) coins of Alice. Price of Bob’s coins at any time \( t \) follow a geometric Brownian motion [Dipple et al.(2020)]. Coins locked by Bob (or B), i.e., \( y_b \), is denoted as \( x(y_b,t) \), when denominated in terms of Alice’s coins at time \( t \). \( x(y_b,t) \) follows a geometric Brownian motion:

\[
\ln \frac{x(y_b,t + \tau)}{x(y_b,t)} = \left( \mu - \frac{\sigma^2}{2} \right) \tau + \sigma (W_{t+\tau} - W_t)
\]

(1)

where \( W \) follows a Wiener Process with drift \( \mu \) and variance \( \sigma^2 \) [Malliaris(1990)].

Given Eq[1], the expected price of \( x(y_b,t) \) at time \( t + \lambda \), \( \mathcal{E}(x(y_b,t),\lambda) \), Probability density function of \( x(y_b,t) \) at time \( t + \lambda \), \( P(x,x(y_b,t),\lambda) \), and Cumulative density function of \( x(y_b,t) \) coins at time \( t + \lambda \), \( C(x,x(y_b,t),\lambda) \) is expressed as follows:

\[
\mathcal{E}(x(y_b,t),\lambda) = \mathbb{E}[x(y_b,t + \lambda)|x(y_b,t)] = x(y_b,t)e^{\mu \lambda}
\]

\[
P(x,x(y_b,t),\lambda) = \mathbb{P}[x(y_b,t + \lambda) = x|x(y_b,t)]
\]

\[
= e^{-\frac{\left( \ln \left( \frac{x}{x_0} \right) - \mu \lambda \right)^2}{2\sigma^2 \lambda}}
\]

(2)

\[
C(x,x(y_b,t),\lambda) = \mathbb{C}[x(y_b,t + \lambda) \leq x|x(y_b,t)]
\]

\[
= \text{erfc}\left( \frac{\ln \left( \frac{x}{x_0} \right) - \mu \lambda}{\sqrt{2\sigma^2 \lambda}} \right)
\]

where \( \text{erfc} \) is the complementary error function. \( \text{erfc}(x) \) is defined as:

\[
\text{erfc}(x) = \frac{2}{\pi} \int_x^\infty e^{-t^2} dt
\]

(3)

The expressions are taken from [Xu et al.(2021)].

We assume both parties know each other’s parameters. Transaction fees are assumed to be negligible compared to the amounts involved in the transactions. The rest of the notations are defined in Table I.

| Notation | Description |
|----------|-------------|
| \( x_a \) | Coins possessed by Alice or A |
| \( y_b \) | Coins of Bob or B that A decides to buy for \( x_a \) coins at time \( t_0 \) |
| \( t_0 \) | Time taken for a transaction to get confirmed in Chain-a |
| \( t_1 \) | Time taken for a transaction to get confirmed in Chain-b |
| \( t_3 \) | HTLC forwarded by A to B in Chain-a, locking \( x_a \) expires |
| \( t_b \) | HTLC forwarded by B to A in Chain-b, locking \( y_b \) expires |
| \( \epsilon \) | Propagation delay |
| \( \theta_b \) | Short time gap |
| \( \theta_a \) | Belief of A regarding type of B being interested to swap |
| \( \sigma_{\text{pre}} \) | Belief of A regarding type of B being interested to swap |
| \( \rho \) | Success premium of A, if swap succeeds |
| \( \rho \) | Success premium of B, if swap succeeds |
| \( x(y_b,t) \) | Price of \( y_b \) coins in terms of \( x_a \) at time a given time \( t \) |
| \( \rho_a \) | Time discounting factor of A |
| \( \rho_b \) | Time discounting factor of B |
| \( f_a \) | Transaction fee in Chain-a |
| \( f_b \) | Transaction fee in Chain-b |
| \( \epsilon \) | Wiener process drift |
| \( \lambda \) | Wiener process variance |

\( \mathcal{E}(x(y_b,t),\lambda) \) | Expected price of \( x(y_b,t) \) at time \( t + \lambda \), also expressed as \( \mathbb{E}[x(y_b,t + \lambda) = x(y_b,t)e^{\mu \lambda}] \)

\( P(x,x(y_b,t),\lambda) \) | Probability density function at time \( t + \lambda \) given that price at time \( t \) is \( x(y_b,t) \)

\( C(x,x(y_b,t),\lambda) \) | Cumulative density function at time \( t + \lambda \) given that price at time \( t \) is \( x(y_b,t) \)

TABLE I: Notations used in the paper
C. Game Model

We model the interaction between two parties Alice, denoted as A, and Bob, denoted as B, as a sequential game $\Gamma_{\text{swap}}$. We assume that all the miners in the blockchains Chain-a and Chain-b are honest. In our model, B considers A to be interested in exchanging coins with probability $\theta_1$ and malicious with probability $1 - \theta_1$. A considers B to be interested with probability $\theta_2$ and malicious with probability $1 - \theta_2$. Given the belief A has regarding B's type, if she has chosen to initiate the lock phase at time $t_1$, then B chooses either to lock $y_B$ coins or abort based on his belief of A's type at time $t_2$. If B doesn't lock his coins before $t_1 + t_a$ elapses then the swap stands canceled. If B locks his coins, A makes the next move at $t_3$, choosing either to claim $y_B$ coins or abort. Once A has chosen an action, B follows that.

The extensive-form game $\Gamma_{\text{swap}}$, is defined for players A and B. The payoff function $u_{k, \theta, b} : S \times N \to \mathbb{R}$ for any player $k \in \{A, B\}$ and $\theta \in \{\text{interested} (\text{int}), \text{malicious}\}$, where $S = \{\text{continue}, \text{stop}\}$, is denoted as $u_{k, \theta, b}(s, t)$ where $s \in S$ and $t \in N$. $S$ denotes the set of actions for players A and B. $u_{k, \theta, b}(s, t)$ specifies the payoff the player $k$ of type $\theta$ would get at time $t \in N$, if the players chose their action $s \in S$. The game begins with Nature (N) choosing the type of players A and B, where the probability of picking both players who are interested to exchange coins is $\theta_1 \theta_2$ (probability of picking malicious A (B) is $1 - \theta_1 (1 - \theta_2)$). Since selecting type of players are mutually independent, all possible combinations probability is the product of individual probability. In the next step, interested A will select her strategy of either continue or stop, based on the belief of B's type, where as malicious A will always choose to continue. Next, interested B chooses his strategy based on his belief of A's type, however malicious B will choose stop. Finally, interested A will choose to claim the coins if the exchange rates are in her favour else she aborts. Malicious A will always abort.

1) Preference Structure: To calculate the payoff of each party for type interested[1] we use backward induction, starting from $t_3$. The claim phase starts at time $t_3$, when an interested A decides whether to continue and reveal preimage of $H$, or stop for $t_3$ to elapse. If interested A decides to claim $y_B$ coins, then an interested B claims $y_B$ coins. We define the utility or payoff for each strategy. If interested A decides to continue at $t_3$, then the time taken for the redeem transaction to get confirmed in Chain-b is $\tau_B$. We multiply payoff of A upon continuing with a factor $(1 + sp_a)$, where $u_a$ is the success premium (or $u_a$ for B) to emphasize that rational parties gain higher utility by swapping their assets rather than abort. The utility of interested A is expressed with their time discounted for duration $\tau_B$. Similarly, when interested B claims coins in Chain-a at time $t_3 + t_a$, time taken for the transaction to be confirmed is $\tau_a$, hence the utility is expressed with their time discounted for duration $t_a + \tau_a$.

$$u_{A,\text{int}}(\text{cont}, t_3) = (1 + sp_a)E(x(y_B, t_3), \tau_B)e^{-r_B\tau_B} - f_b$$
$$u_{B,\text{int}}(\text{cont}, t_3) = (1 + sp_B)x_a\tau_a e^{-r_a\tau_a} - f_a$$

where $t_3 = t_2 + \tau_B + T$, and $T \in [0, t_b - \theta - \epsilon]$ being the delay by interested A before she decides to claim the coins. If interested A decides to stop, then interested B has to abort as well. The coins remain locke uselessly for the entire duration. The utility is expressed as: $u_{A,\text{int}}(\text{stop}, t_3) = x_a e^{-r_a\tau_a} - f_a$ and $u_{B,\text{int}}(\text{stop}, t_3) = \mathcal{E}(x(y_B, t_3), t_a)e^{-r_a\tau_a} - f_b$. An interested A will decide on continue over stop at $t_3$, if the following condition holds:

$$(1 + sp_a)E(x(y_B, t_3), \tau_B)e^{-r_B\tau_B} - f_b \geq x_a e^{-r_a\tau_a} - f_a$$
$$or, x(y_B, t_3)e^{\mu \tau_B} \geq x_a e^{-r_a\tau_a} - (f_a - f_b)e^{-r_a\tau_a}$$
$$or, x(y_B, t_3) \geq \frac{(x_a e^{-r_a\tau_a} - (f_a - f_b)e^{-r_a\tau_a})e^{\mu \tau_B}}{(1 + sp_a)}$$

We derive $x(y_B, t_3)^*$ in terms of $x_a$ for which the above inequality holds. If $x(y_B, t_3) \geq x(y_B, t_3)^*$ then A claims the coins.

In the second round of lock phase, interested B has to decide whether he will continue (i.e., lock $y_B$ coins) or stop after A has locked $x_a$ coins. Since B is unaware of A's type, he calculates the expected payoff where with probability $1 - \theta_1$ he has to stop at time $t_2 + \tau_B$ and with probability $\theta_1$, he can either continue, if the price of coins rise to $x(y_B, t_3)^*$, or stop at time $t_2 + \tau_B + T$, if the price drops. The payoff is expressed as time discounted, expected utility for duration $\tau_B + T$. An interested B calculates what will be the probability that price of his coins will rise to $x(y_B, t_3)^*$ within time $\tau_B + T$.

$$u_{B,\text{int}}(\text{cont}, t_2) = \theta_1 \left[ \left(1 - C \left(x(y_B, t_3)^*, x(y_B, t_2), \tau_B + T \right) \right) u_{B,\text{int}}(\text{cont}, t_3) + \int_0^{x(y_B, t_3)^*} \frac{P \left(p, x(y_B, t_2), \tau_B \right) u_{B,\text{int}}(\text{stop}, t_3) dp}{e^{\mu \tau_B}} \right] + (1 - \theta_1) \left( u_{B,\text{int}}(\text{stop}, t_2) \right)$$

Interested A knows what action was taken at $t_3$ and so there is no need to consider B's type.

$$u_{A,\text{int}}(\text{cont}, t_2) = \int_{x(y_B, t_3)^*}^{\infty} \frac{P \left(p, x(y_B, t_2), \tau_B + T \right) u_{A,\text{int}}(\text{cont}, t_3) dp}{e^{\mu \tau_B}} + C \left(x(y_B, t_3)^*, x(y_B, t_2), \tau_B \right) u_{A,\text{int}}(\text{stop}, t_3)$$

If B stops at $t_2$, then A’s coins remain locked for $t_a$ units of time. The utility is expressed as $u_{A,\text{int}}(\text{stop}, t_2) = x_a e^{-r_a\tau_a} - f_a$ and $u_{B,\text{int}}(\text{stop}, t_2) = x(y_B, t_2)$. B’s decision is dependent on how price of $y_B$ evolves until $t_3$. He will decide to continue over stop, if the following condition holds:

1We do not calculate the payoff for type malicious as we are interested in quantifying the success rate of HTLC-based atomic swap and this is possible only in presence of interested parties.
The HTLC-based atomic swap is not participant-friendly. 

Proof: A’s willingness to participate in the atomic swap is decided by the expected success of the protocol for given set of parameters. The success rate (SR) of a swap is the probability that the swap succeeds after it has been initiated, i.e. after A has locked coins at $t_1$ [Xu et al.(2021)]. For a given pair of $\theta_1$ and $\theta_2$, SR is defined as function of $x_a$ (A’s coins or tokens), delay by A (T) at final step while claiming coins, and delay by B (T’) at second step while locking $y_b$ coins. It is expressed as:

$$SR(x_a, T', T) = \int_{x_1[x_a]} P_A(p, T') P_B(p, T) \, dp$$

where $P_A(p, T') = \theta_2 P(p, x(y_b, t_1), \tau_a + T')$ and $P_B(p, T) = \theta_1 \left( 1 - C(x(y_b, t_3), x_a, p, \tau_0 + T) \right)$.

We plot the success rate of the protocol in Fig. 3(a-c), the parameters used are $t_0 = 1 \, hr$ and $\tau_a = \tau_b = 3 \, hrs$. As per standard practice, $t_a = 48 \, hrs$ and $t_b = 24 \, hrs$ [Han et al.(2019)]. We select $x_a \in [1, 3]$ given $x(y_b, t_1) = 2$. The parameters $T'$ is varied between $[0, 20]$ and $T'$ is varied between $[0, 21]$. The fee $f_a$ and $f_b$ is negligible, so we consider them to be 0. The success premiums $sp_a = sp_b = 0.3$, time-discounting factor $r_a = r_b$ is chosen from $\{0.005, 0.01\}$, $\sigma$ is varied between 0.1 and 0.2, and $\mu$ is selected from $\{-0.002, 0.002\}$. We observe that the success rate is $\geq 0.9$ in Fig. 3(a-b), and around 0.6 for Fig. 3(c), when $T = T' = 0$. When $T$ or $T'$ increases, the success rate drops and beyond certain range, it becomes N.A (not applicable) as $u_{A, init}(cont, t_1) < x_a$. Success rate is function of $T$ and $T'$ that cannot be determined before the swap proceeds to the second round or to the third round. In the worst case, if $T \approx t_b$ and/or $T' \approx t_a - t_b$, then the success rate drops drastically as the payoff upon continuing is too low. In such situation, a party is better off if he does not participate in the swap rather than keep his coins locked for one full day.

III. Quick Swap: A Protocol Based on Ideal Swap Model

The flaw in the HTLC-based atomic swap is that either of the parties can speculate and delay in settling the transaction without losing anything. Our objective is to force the parties to settle the transaction faster, and penalize for delayed action. We provide a high-level overview of our proposed protocol where a party will lock the principal amount for swap provided he or she gets a guarantee of compensation upon suffering from a griefing attack. With respect to the previous example,
Alice has to lock $x_a$ coins for a period of $t_a$ units. If Bob grieves, Alice’s collateral locked will be $O(x_a t_a)$. She calculates the collateral cost of locking $x_a$ coins for $t_a$, let it be defined as $c(x_a t_a)$. Similarly, Bob calculates the collateral cost of locking $y_b$ coins for $t_b$, let it be $c(y_b t_b)$. The steps for locking the coins proceeds as follows:

(i) Bob first locks $c(x_a t_a)$ coins in Chain-b for $D + \Delta$ units of time where $\tau_a + 2\tau_b < D + \Delta < t_b$. We will explain later why an additional $\Delta$ unit is chosen.

(ii) After Alice gets a confirmation of the grieving-premium locked by Bob, she locks $x_a$ for $t_a$ units and grieving-premium $c(x_a t_a) + c(y_b t_b)$ for $D > \tau_a + \tau_b$ units of time, both in Chain-a.

(iii) After Bob gets a confirmation of the grieving-premium locked by Alice, he locks $y_b$ for $t_b$ units of time in Chain-b.

If Bob doesn’t want to proceed after step [ii] and cancels the swap, then Alice can unlock $x_a$ coins from Chain-a. If Bob grieves, then Alice gets the compensation $c(x_a t_a)$ after $D + \Delta$ elapses instead of being griefed for $t_a$. The principle we follow here is “Coins you have now is better than coins you have later”. Alice can use the compensation from $t_a - (D + \Delta)$. Had we set the timelock for locking $c(x_a t_a)$ to $t_a$, Bob can still grief by canceling the contract at time $t_b - \epsilon$. We will discuss later how should we choose $D$ to ensure a faster compensation.

If Alice initiates the swap, she claims $y_b$ coins and withdraws the compensation from Chain-a. Bob gets to claim $x_a$ coins and he withdraws the compensation from Chain-b. If Alice cancels the swap, then Bob unlocks $y_b$ coins from Chain-b. If Alice delays beyond $D$ unit of time, then Bob gets compensation for the loss. Since $D + \Delta < \tau_b$, Alice can delay beyond this point as well. In that case, she gets a compensation of $c(x_a t_a)$ and her net gain is $g_a = c(x_a t_a) - c(y_b t_b)$ and Bob’s net loss is $-g_a$. Thus to prevent Bob from incurring a loss, Alice is forced to pay a compensation of $c(x_a t_a) + c(y_b t_b)$. Even if she does not respond, Bob is entitled to a compensation of $c(y_b t_b)$ after refunding $c(x_a t_a)$ coins to Alice.

A. Formal Description of Quick Swap

1) System Model and Assumption: The system model and assumptions are same as HTLC-based atomic swap. Since $t_a \approx 2t_b$, for ease of analysis, we consider $c(x_a t_a) \geq 2c(y_b t_b)$. For a fixed rate of grieving-premium, we consider $Q = c(x_a t_a)$ and $\frac{Q}{2} = c(y_b t_b)$. Alice locks a griefing-premium of $1.5Q$ and Bob locks griefing-premium $Q$ coins. We denote Alice as A and Bob as B while describing the protocol.

2) Detailed Description: The protocol has the following phases: (A) Preprocessing Phase, (B) Lock Phase and (C) Claim Phase. An instance of successful execution of Quick Swap is shown in Fig. 4.

(A) Preprocessing Phase: The steps are defined as follow:

Sampling Cancellation Hash and Payment Hash:

(i) A’s pair of secret and public key is $(sk_a, pk_a)$ and B’s pair of secret and public key is $(sk_b, pk_b)$. A uses $sk_a$ and B uses $sk_b$ for signing transactions. $pk_a$ and $pk_b$ are used for verifying each such signed transaction.

(ii) A samples random values $s_1$ and $s_3$, creates payment hash $H_1 = \mathcal{H}(s_1)$ and cancellation hash $H_3 = \mathcal{H}(s_3)$. She shares $H_1$ and $H_3$ with B.

(ii) B samples a random value $s_2$ and creates cancellation hash $H_2 = \mathcal{H}(s_2)$ and shares it with A.

(B) Locking Phase:

(i) At time $t_1$, B creates and signs transaction grieving_premium_lock_B using funding address addr_funding_penalty_B that locks Q coins address addr_lock_penal, and publishes griefing_premium_lock_B in Chain-b. B encodes the condition whereby Q coins can be claimed either by revealing the preimage of $H_1$ or $H_2$, or it can be spend by A after $D + \Delta$ units of time.

(ii) A checks whether griefing_premium_B is confirmed within $\tau_b$ units. Once that is confirmed, at time $t_2 = t_1 + \tau_b$, B creates and signs a transaction griefing_premium_lock_A using funding address addr_funding_penalty_A that locks 1.5Q coins into address addr_lock_penal_A. The condition for spending the coins are as follows:

(a) Either provide the preimage of $H_1$ or $H_2$;

(b) If no preimage is provided, B can spend the coins locked after D units of time.

Simultaneously, A creates and signs another transaction principal_lock_A using funding address addr_funding_principal_A that locks $x_a$ coins into address addr_lock_principal_A. The coins can be spend either by revealing the preimage of $H_1$, or, A can refund the coins either after $t_a$ unit elapses
or by revealing preimage of $H_2$, whichever occurs first. She publishes both $\text{principal\_lock\_A}$ and $\text{griefing\_premium\_lock\_A}$ in Chain-a.

(iii) B checks whether the transactions broadcasted by A gets confirmed in another $\tau_a$ units. At time $t_3 = t_2 + \tau_a$. He creates and signs another transaction $\text{principal\_lock\_B}$ using funding address $\text{addr\_funding\_principal\_B}$ that locks $y_b$ coins into address $\text{addr\_lock\_principal\_B}$. The coins can be spent either by revealing the preimage of $H_1$, or, B can refund the coins either after $t_b$ unit elapses or by revealing preimage of $H_3$, whichever occurs first. He then proceeds to publish $\text{principal\_lock\_B}$ in Chain-b.

(C) Claim Phase: Once A observes that B has locked $y_b$ coins in Chain-b, he initiates the claim phase at time $t_4 = t_3 + \tau_b$, where $\tau_a\tau_b$ is the time taken for $\text{principal\_lock\_B}$ to be confirmed.

**Redeem:** If A wishes to redeem the coins,
a. At time $t_4 = t_3 + \tau_b < D$:
   (i) A releases the preimage $s_1$ for payment hash $H_1$ and claims the output of transaction $\text{principal\_lock\_B}$ with her signature in Chain-b. This allows her to claim $y_b$ coins.
   (ii) A uses $s_1$ to refund the griefing-premium of $1.5Q$ locked in Chain-a.

b. At time $t_5 = t_4 + t_a$:
   (i) B uses the preimage $s_1$ and unlocks the output of the transaction $\text{principal\_lock\_A}$ with his signature in Chain-a, claiming $x_a$ coins.
   (ii) B uses $s_1$ to refund $Q$ locked in Chain-b.

**Refund:** If A wishes to cancel the swap,
a. At time $t_4 = t_3 + \tau_b < D$:
   (i) A releases the preimage $s_3$ for cancellation hash $H_2$ with her signature in Chain-a, unlocking $1.5Q$ coins.
   (ii) A uses $s_3$ to refund $y_b$ coins from Chain-b at time $t_4 + t_a$. This results in cancelation of swap before $t_b$ elapses.

b. At time $t_5 = t_4 + \tau_b + t_a$:
   (i) B releases the preimage $s_2$, unlocks $Q$ coins with her signature from Chain-b.
   (ii) A uses $s_2$ for refunding $x_a$ coins from Chain-a at time $t_4 + \tau_b + 2t_a$.

### B. Proof of Correctness, Safety and Liveness

It is necessary to argue the state of a proposed protocol in presence of both compliant and malicious parties. Parties either may choose to follow the protocol or they may deviate. We prove that Quick Swap satisfies both safety and liveness. By safety, we mean that “compliant parties should end up “no worse off,” even when other parties deviate arbitrarily from the protocol” [Herlihy et al. (2022)]. Simultaneously, the liveness property states that none of the parties end up keeping their coins locked forever. Before arguing for liveness and safety, we prove the correctness of the protocol in presence of compliant parties - either all agreed coin exchange take place, and no exchange take place.

**Property 1:** (Correctness) If all parties are compliant, then the swap either succeeds with coins being exchanged, or the swap gets canceled with the coins being refunded.

**Proof:** A and B exchange the hashes $H_1$, $H_2$ and $H_3$ before the start of the protocol. At time $t_1$, B locks $Q$ coins in Chain-b that can be unlocked contingent to either providing preimage of $H_1$ or $H_2$. The coins are locked for $D + \Delta$ units, after which A can claim $Q$ coins and begins the next phase of locking at $t_2 = t_1 + \tau_a$. She locks $x_a$ coins in Chain-a that can be claimed by B contingent to providing the preimage of $H_1$, else A refunds after time $t_a$ or using preimage of $H_2$. She also locks $1.5Q$ coins at $t_2$ in Chain-a. The amount can be unlocked contingent to either providing preimage of $H_1$ or $H_3$. The coins are locked for $D$ units, after which B can claim $1.5Q$ coins. The last locking phase starts at $t_3 = t_2 + \tau_b$, when B locks $y_b$ coins in Chain-b. The latter can be claimed by A contingent to providing the preimage of $H_1$, else B refunds after time $t_b$ or using preimage of $H_3$. The correctness of Claim Phase follows from the description of Redeem and Refund defined in Section III-A2.

**Property 2:** (Safety Property) The safety property states that compliant parties should be as better off as they had been before the protocol execution, even when other parties deviate arbitrarily from the protocol.

**Proof:** After B has locked $Q$ coins Chain-b, if A does not lock any coins Chain-a, B unlocks it after a certain timeperiod $\delta < D + \Delta$. If B locks $Q$ coins, A locks $x_a + 1.5Q$ coins, but B aborts without locking $y_b$ coins, then A unlocks $1.5Q$ coins revealing preimage $s_3$. She gets a compensation $Q$ coins after $D + \Delta$ that covers up for the lost opportunity cost of keeping $x_a$ coins locked for $t_a$ units. If all the parties have locked their coins but A delays beyond $D$ or griefs, B gets compensation of $1.5Q$. He may lose $Q$ coins (if A cancels before elapse of $D + \Delta$ but after timeout $D$, then B is entitled to the full compensation $1.5Q$) but we ensure that $1.5Q - Q = 0.5Q$ coins are enough to compensate for locked collateral $O(y_b)$.  

**Property 3:** (Liveness) Coins No asset belonging to a compliant party do not remain locked forever.

**Proof:** If A doesn’t take any action by $t_2 = t_1 + \tau_a$, B unlocks $Q$ coins after $\tau_b > \delta > 0$ units. If B does not lock coins at $t_3 = t_2 + \tau_a$, then by time $t_3 + \tau_b$, A refunds the griefing-premium $1.5Q$ by revealing preimage of $H_3$ in its own interest. B observes that A has canceled the swap by withdrawing the griefing-premium. If he is rational, then he will releases preimage $s_3$ for $H_3$, unlock $Q$ coins and allow A to unlock $x_a$ coins before $D$ elapses. If B is malicious, then he will end up losing $Q$ coins after $D + \Delta$ units, but A will be able to unlock $x_a$ coins after $t_a$ elapses. If at time $t_4 = t_3 + \tau_b$, A aborts then she loses compensation of $1.5Q$ to B. The latter can unlock $y_b$ coins after $t_b$ has elapsed but loses his compensation of $Q$ coins.

### C. Game-Theoretic Analysis

We model the interaction between the two entities A and B as a sequential game $\Gamma_{\text{quick swap}}$. B initiates the lock phase by...
locking griefing-premium $Q$ for duration $D + r_a + r_b + A$ units in Chain-b at time $t_1$, followed by A choosing either to lock $x_a$ coins for duration $t_2$ units or abort at $t_2 = t_1 + r_b$. If A has not responded then B either cancels the swap at time $t_2 + r_b$ by unlocking $Q$ or he stops. If A wishes to continue, she locks the principal amount and the griefing-premium $1.5Q$ for $D$ units in Chain-a at time $t_2$. At time $t_3 = t_2 + r_a$, if B observes that A has locked the principal amount as well as griefing-premium, then B either locks $y_b$ coins for $t_2$ units in Chain-b or stops. If he does not lock coins, then at time $t_3 + r_a$, A cancels the swap by unlocking $1.5Q$ coins from Chain-a or stops. If B has not responded, he loses the griefing-premium at time $t_3 + r_b$. Else if he chooses to cancel, then A will be able to withdraw the principal amount as well. If B has chosen to continue, A decides at time $t_4 = t_3 + r_b$, whether to continue or cancel the swap.

1) Game Model & Preference Structure: The extensive-form game $\Gamma_{quick\ swap}$ is similar to $\Gamma_{swap}$, except that A and B’s strategy space has the option to cancel, apart from continue and stop. The analysis is done by applying backward induction on $\Gamma_{quick\ swap}$.

If A delays instead of making a move at $t_3 + r_b$, then the opportunity cost of coins locked as griefing-premium will rise. Canceling the swap at $t_3 + r_b$ will lead to A’s utility as $x_a$.

If A chooses to delay and cancel at time $t_3 + r_b + t$ for $t > 0$, then the utility drops further, i.e., $x_a(e^{r_a(t+2r_b+r_a)} + 1.5Q)e^{r_a(t+2r_b+r_a)}$. In the previous HTLC-based atomic swap, if A finds that the utility on continuing is less than the utility of the swap till the lock time of HTLC expires, she speculated till the situation turns in her favor. However, the situation is different now as A is allowed to abort the swap much earlier without waiting for the lock time to elapse. A rational A will choose not to delay anticipating that the situation may turn worse later. At time $t_4$, if A continues and B follows:

$$u_{A,int}(cont, t_4) = (1 + s_{A}) \frac{\mathcal{E}(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - f_a - f_b$$

$$u_{B,int}(cont, t_4) = (1 + s_{B}) \frac{\mathcal{E}(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - f_a - f_b$$

At time $t_4$, if A cancels then B cancels the deal as well. The payoffs are $u_{A,int}(cancel, t_4) = \frac{x_a}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - 2f_a$ and $u_{B,int}(cancel, t_4) = \frac{\mathcal{E}(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - 2f_b$. A will continue at $t_4$ over canceling the swap, if the following condition holds:

$$(1 + s_{A}) \frac{\mathcal{E}(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - f_a - f_b > \frac{x_a}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - 2f_a$$

or, $x(y, t_4) > \frac{x_a}{(1 + s_{A}) e^{r_a(t_a+2r_b+r_a)}} + f_a + f_b$.

1. We derive $x(y, t_4)$ in terms of $x$ for the above inequality holds. If $x(y, t_4) \geq x(y, t_4)$ then A claims the coins.

At time $t_3$, B decides to continue, with probability $1 - \theta_1$, A is malicious and will delay till $D - \epsilon (D - \epsilon \rightarrow D)$. The utility is expressed as follows:

$$u_{B,int}(cont, t_3) = \theta_1 \left( \int_0^\infty \frac{p(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp + \int_0^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp \right) + (1 - \theta_1) \left( \int_0^\infty \frac{p(x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp + \int_0^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp \right)$$

At time $t_3$, B chooses stop, then the utility for A and B is defined as:

$$u_{A,int}(stop, t_3) = \frac{x_a}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - 2f_a + \frac{Q}{e^{r_a(t_a+2r_b+r_a)}} - f_b$$

If at time $t_3$, B chooses cancel then he unlocks the premium $Q$ locked in Chain-b by releasing preimage of $H_2$. A observes that swap is canceled, so she unlocks $x_a$ coins and griefing-premium $1.5Q$ from Chain-a. The utility is defined as $u_{B,int}(cancel, t_3) = x(y, t_3) + \frac{Q}{e^{r_a(t_a+2r_b+r_a)}} - f_b$ and $u_{A,int}(cancel, t_3) = \frac{x_a}{e^{r_a(t_a+2r_b+r_a)}} + \frac{1.5Q}{e^{r_a(t_a+2r_b+r_a)}} - 2f_a$. We observe that it is better to cancel the swap than wait for the contract to expire as B will lose his griefing-premium in the process. Thus cancel strictly dominates stop in our protocol. B will continue at $t_4$ over canceling the swap, if the following condition holds:

$$u_{B,int}(cont, t_3) > u_{B,int}(cancel, t_3)$$

At time $t_2$, A decides to continue, then utility is:

$$u_{A,int}(cont, t_2) = \theta_2 \left( \int_{t_2}^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp + \int_{t_2}^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp \right) + (1 - \theta_2) \left( \int_{t_2}^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp + \int_{t_2}^\infty \frac{P(p, x, y, t_a, t_a)}{e^{r_a(t_a+2r_b+r_a)}} \, dp \right)$$

At time $t_2$, A decides to abort then B initiates cancellation by releasing preimage of $H_2$. Note that A will not take any action if she intends to cancel as she has not locked any coins. Thus both cancel and stop means the same payoff for A. The payoff is defined as $u_{A,int}(stop, t_2) = u_{A,int}(cancel, t_2) = x_a + 1.5Q$ and $u_{B,int}(cancel, t_2) = x(y, t_2) + \frac{Q}{e^{r_a(t_a+2r_b+r_a)}} - f_b$. A will continue at $t_2$ over stopping the swap, if the following condition holds:

$$u_{A,int}(cont, t_2) > u_{A,int}(cancel, t_2)$$

Proposition 2: Quick Swap is more participant-friendly compared to HTLC-based atomic swap.

Proof: In Quick Swap, for given values of $\theta_1$ and $\theta_2$, success rate or SR is function of $x_a$ (or A’s tokens) since there is no
delay involved. It is expressed as:

$$SR(x_a) = \int x_a^2 A(x_a) B(x_a) \, dp$$

where $A(x_a) = \theta_2 P(p, x(y_b, t_2), \tau_a)$ and $B(x_a) = C(x(y, t_4)[x_a], p, \tau_b)$

A is able to estimate the success rate now as it is dependent solely on $x_a$. There is no uncertainty involved, unlike in HTLC-based atomic swap where a higher delay leads to violation of participation constraint. Additionally, the range of $x_a$ for which Quick Swap has a non-zero success rate is larger. The provision of cancellation, even before the elapse of violation of participation constraint. Additionally, the range of $x_a$ for which Quick Swap has a non-zero success rate is larger. The provision of cancellation, even before the elapse of the contract’s locktime, makes the protocol robust and more resilient to fluctuation in asset price.

D. Discussions

(a) Our protocol requires an extra round compared to HTLC-based atomic swap. The number of transactions created for Quick Swap is double the number of transactions needed for the latter.

(b) The parameter $D$ can be chosen by A and B, after they have negotiated with each other. A rational honest party will choose a shorter delay for keeping the griefing-premium locked.

IV. Construction of a Fair Multiparty Atomic Swap

Suppose Alice wants to exchange coins with Bob where the former has some Bitcoins and later has Ethers, but Bob wants to exchange his Ethers for Litecoins. In such a scenario, they take the help of some intermediaries for assisting in the exchange of coins. If we consider just a three-party situation, then there may exist a participant Carol who is willing to exchange Litecoins for Bitcoins. So Alice sends $x_a$ BTC to Carol and Carol sends $z_c$ LTC to Bob, and finally Bob sends $y_b$ ETH to Alice. In a real situation, Carol will charge a fee from Alice for facilitating the swap but we ignore the fee in this paper.

Problem of Grieving: In an HTLC-based setting, Alice samples a secret $s$, shares $H = \mathcal{H}(s)$ with Carol and Bob. Alice forwards HTLC to Carol locking $x_a$ BTC in Chain-a for $T_1$ unit of time contingent to providing preimage of $H$. The confirmation time of a transaction in Chain-a is in $\tau_a$. Carol forwards an HTLC to Bob using the same condition for a time period of $T_2$ units where $T_2 < T_1$, locking $z_c$ LTC in Chain-b. The confirmation time of a transaction in Chain-b is in $\tau_b$. Finally, Bob forwards the HTLC, locking $y_b$ BTC in Chain-c for time period of $T_3$ units where $T_3 < T_2$. The confirmation time of a transaction in Chain-c is in $\tau_c$. The problem of grieving persists as Carol may not choose to lock coins based on the fluctuation rate of Bitcoin and Litecoin, and even if Carol locks her coins, Bob may abort. If all the parties have locked coins, Alice may abort, and makes Bob and Carol suffer. We discuss a fix to this problem by extending Quick Swap from a two-party setting to a three-party setting.

High Level Overview of the protocol: We discuss the steps followed upon extending Quick Swap to three party setting. The steps have been illustrated in Fig. 5.

(i) Alice samples hashes $H_1$ and $H_2$ using a randomly sampled secret $s_1$ and $s_2$ respectively, and shares it with Bob and Carol. Bob samples hash $H_3$ using secret $s_3$ and Carol samples hash $H_4$ using secret $s_4$.

(ii) Bob locks griefing-premium $c(x_a, T)$ in Chain-c for a time-period of $D + 2\Delta < T_3$ using hashlock $H_3 \lor H_1$, at time $t_1$.

(iii) Alice locks principal amount $x_a$ for time period $T_1$ using hashlock $H_1$ at time $t_2 = t_1 + \tau_a$, with the provision of refunding earlier if preimage of $H_3$ is revealed. She samples a hash $H_2$ using secret $s_2$, and locks griefing-premium $c(z_c, T_2)$ for a time period $D + \Delta$, hashlock $H_1 \lor H_2$ in Chain-a at time $t_2$.

(iv) Carol locks principal amount $z_c$ for time period $T_2$, hashlock $H_1$, at time $t_3 = t_2 + \tau_b$, at time $t_2$. She can refund before $T_2$ elapses if preimage of $H_2$ is revealed. Carol samples hash $H_4$ using secret $s_4$, and locks griefing-premium $c(y_b, T_3)$ for timeperiod $D$, hashlock $H_1 \lor H_4$ in Chain-b.

(v) Finally, Bob locks $z_c$ coins in Chain-c for timeperiod $T_3$, hashlock $H_4$, at time $t_2 = t_1 + \tau_c$. He has a provision to refund earlier if the preimage of $H_4$ is revealed.

To initiate the swap, Alice reveals secret $s_3$ and everyone is able to unlock their griefing-premium and the swapped coins. If any of the parties want to cancel, he or she will choose to reveal either secret $s_2$, $s_3$ or $s_4$.

A. Generic n-party fair cyclic atomic swap

1) System Model & Assumption: Party $P_0$ wants to exchange $a_0$ coins for $a_n$ coins of party $P_n$, taking help of $n - 1$ intermediaries $P_1, P_2, \ldots, P_{n-1}$. A party $P_i$ has account in Chain-i and Chain-((i - 1) mod n + 1) . Blockchain Chain-i has transaction confirmation time $\tau_i$ where $i \in [0, n]$.

2) Detailed Construction: We describe the steps:

- $P_0$ samples payment hash $H$ and shares with neighbors $P_n$ and $P_1$.

- $P_n$ initiates the Locking Phase, samples cancellation hash $H_n$. He locks griefing-premium $c(a_0, T_0)$ for locktime $D + n \Delta < T_n$ in Chain-n, using hashlock $H \lor H_n$, at
time \( t_1 \).

- Rest of the parties \( P_i, i \in [0, n-1] \) does the following: \( P_i \) generates a cancellation hash \( H_i \) and locks \( a_i \) coins for locktime \( T_i \), using hashlock \( \bar{H} \) at time \( t_{i+2} = t_{i+1} + T_i \) mod \( n+1 \) in Chain-1. The coins can be refunded before \( T_i \) if preimage of \( H_i \) is revealed. He also locks griefing-premium \( c(a_i+1)T_i+1) \) at time \( T_i+2 \), for locktime \( D+(n-1-i)\Delta \), using hashlock \( H \) in Chain-2. The locktimes assigned follow a strictly decreasing order: \( T_0 > T_1 > \ldots > T_n \).

- Finally, \( P_n \) locks \( a_n \) coins for locktime \( T_n \), using hashlock \( H \), at time \( t_{n+2} = t_{n+1} + T_n \) in Chain-2. He has an option to refund the coins if \( P_{n-1} \) cancels the swap by revealing the preimage of \( H_{n-1} \). The locktimes assigned follow a strictly decreasing order: \( T_0 > T_1 > \ldots > T_n \).

V. RELATED WORKS

The HTLC-based atomic swap was first proposed in [TN1(2013)]. However, the design lacks fairness and is susceptible to griefing attacks. Later Hao et al. [Han et al.(2019)] suggested the use of premium to counter griefing attacks. However, the protocol assumed that in a two-party setting where Alice wants to exchange currency with Bob, only Alice can be at fault. So she must lock premium and Bob is not required to do so. In an American-style option-based swap, Bob gets the premium even though Alice initiates the swap on time. In currency exchange-based atomic swap, Bob gets the premium if Alice doesn’t respond within the time period of the contract. The protocol is not fair as Bob can grief as well. The construction cannot be realized in Bitcoin scripts as it requires the inclusion of an additional opcode.

Similar work has been done that talks about locking premium by both the parties involved in exchanging currency [Xue and Herlihy(2021)]. However, the protocol is not compatible with Bitcoin scripts and suffers from the problem of mismatched premiums, and lacking fairness. Further, the authors have bootstrapped the premium, whereby small valued premiums get locked first, and with each iteration, the premium amount increases. This leads to multiple round communication, creation of multiple contracts for each iteration, and longer lock time than [TN1(2013)] and griefing on the locked-up premium is possible [Nadahalli et al.(2022)]. Nadahalli et al. [Nadahalli et al.(2022)] have proposed a protocol that is grief-free and compatible with Bitcoin scripts. The protocol is efficient regarding the number of transactions and the worst-case timelock for which funds remain locked. However, the problem of mismatched premium exists. The model lacks flexibility due to the coupling of premium with the principal amount, and thus cannot be extended to multi-party atomic swap setting involving more than two blockchains [Herlihy(2018)]. Our proposed protocol overcomes several such shortcomings. However, there is a constant factor increase in overhead of transaction and communication round compared to [Han et al.(2019)] and [Nadahalli et al.(2022)]. We summarize the discussion by performing a comparative analysis of Quick Swap with other state-of-the-art protocols in Table [II]

| Protocol | Fairness of Premium Locked | Counters Speculation | Single Asset Light-Tier | Multi-Party Multi-Blockchain |
|----------|----------------------------|----------------------|-------------------------|-----------------------------|
| Quick Swap | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P1 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P2 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P3 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P4 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P5 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |
| P6 | Supported by Bitcoin script | Supported | Supported by Bitcoin script | Supported by Bitcoin script |

TABLE II: Comparative Analysis of Quick Swap with existing Atomic Swap protocols in terms of P1: Countering griefing attack, P2: Cancellation Allowed, P3: Counters speculation, P4 Fairness of premium locked, P5: Supported by Bitcoin script and P6: Extension to multi-party cyclic swap

VI. CONCLUSION

In this paper, we perform a game-theoretic analysis of HTLC-based atomic swap. We observe that the protocol lacks fairness and it is not at all participant-friendly. We propose Quick Swap that is robust and allows faster settlement of the transaction. We discuss the step for extending Quick Swap to a multiparty setting involving more than two blockchains. As a part of our future work, we would like to analyze Quick Swap in presence of rational miners in underlying Blockchains.

ACKNOWLEDGEMENT

We thank Dr. Abhinandan Sinha, Assistant Professor at Ahmedabad University, for his initial valuable comments on this work.

REFERENCES

[TN1(2013)] 2013. TierNolan. Technical Report. https://github.com/TierNolan.

[Nadahalli et al.(2022)] Nadahalli, M., Herlihy, M., and Shrira, L. 2022. Deterministic cross-blockchain token transfers. IEEE Access 10 (2022), 59837–59852.

[Hao et al.(2019)] Hao, R., Lin, H., and Yu, J. 2019. On Quick Swap. IEEE Access 7 (2019), 110330–110342.

[Nadahalli et al.(2022)] Nadahalli, M., Herlihy, M., and Shrira, L. 2022. Deterministic cross-blockchain token transfers. IEEE Access 10 (2022), 59837–59852.

[Han et al.(2019)] Han, R., Lin, H., and Yu, J. 2019. On Quick Swap. IEEE Access 7 (2019), 110330–110342.

[Xue and Herlihy(2021)] Xue, C. and Herlihy, M. 2021. Using correlated stochastic differential equations to forecast cryptocurrency rates and social media activities. Applied Network Science 6, 1 (2021), 1–30.

[Herlihy(2018)] Maurice Herlihy. 2018. Atomic Cross-Chain Swaps. In Proceedings of the 1st ACM Conference on Advances in Financial Technologies. 62–75.

[Herlihy et al.(2018)] Maurice Herlihy, Barbara Liskov, and Maria Potop-Butucaru. 2021. R-SWAP: Relay based atomic cross-chain swap protocol. Ph.D. Dissertation. Sorbonne Université.
