Fast and Atomic Cross-blockchain Asset Exchange for Metaverse Interoperability

Shan Jiang, Jiannong Cao, Hanqing Wu
Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China
cs-shan.jiang@polyu.edu.hk, jiannong.cao@polyu.edu.hk, hanqing.91.wu@connect.polyu.hk

***Abstract*—This work concerns metaverse interoperability. It is a hot topic because more than 240 metaverses were launched in the past years but are merely interoperable. Existing work only focused on particular issues, e.g., asset exchange and object visualization in different metaverses, but neglected the overall picture. Moreover, the existing asset exchange protocols are time-consuming and can hardly afford high-frequency metaverse transactions. To address the issues, we first introduce a layered metaverse interoperability framework that comprehensively considers interoperable cyber worlds, compatible interaction mechanisms, and consistent physical infrastructures. Furthermore, we propose a novel cross-blockchain asset exchange protocol based on key exchange and smart contracts. The proposed protocol is secure and reduces the time overhead from linear to constant.**

***Index Terms*—Metaverse, metaverse interoperability, asset exchange, blockchain interoperability, decentralized exchange, cross-chain swaps.**

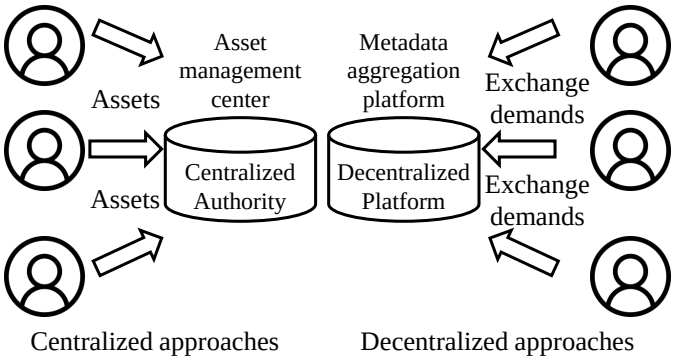

Fig. 1. Traditional cross-blockchain asset exchange approaches (left: centralized approaches; right: decentralized approaches). The centralized approaches manage the assets, while the decentralized approaches aggregate the asset exchange metadata.

I. Introduction

In recent years, metaverse technology has become increasingly popular [1] with a series of impactful projects, such as Decentraland and The Sandbox. According to CoinMarketCap[1], the total capitalization of the global metaverse market has been beyond US$16 billion since Feb 2023. A metaverse is a single, universal, and immersive virtual world in which people behave and interact, including attending remote meetings, playing immersive games, and trading non-fungible tokens [2]. Besides finance, the metaverse technology has been successfully applied in education [3], healthcare [4], transportation [5], etc.

Despite the various applications, the metaverse technology is still in its infancy due to the lack of interoperability. One of the ultimate goals of the metaverse is to be single and universal. However, there are more than 240 metaverses (till Feb 2023) in the market, and they are merely interoperable. For example, people can hardly browse and travel across the metaverses or spend assets of one metaverse in another. The isolation of metaverses degrades and limits the value of each individual. There is an urgent need to design protocols making metaverses interoperable.

In the literature, there is only limited work [6], [7] about the approaches of metaverse interoperability and even the definition. They focused on some particular perspectives, e.g., user experience and data management, but neglected the overall picture. This work fills the gap. We comprehensively consider the three layers of a metaverse, i.e., cyber world, interaction mechanism, and physical infrastructure, and discuss the challenging issues towards metaverse interoperability in each layer. In particular, we highlight the importance of interoperable physical infrastructure and pay special attention to the cross-blockchain asset exchange issue.

Existing cross-blockchain asset exchange can be divided into centralized and decentralized approaches, as shown in Fig. 1. In centralized approaches, assets from users are accumulated and fully managed by a central authority assumed to be trustworthy. Then, the authority provides a platform for users to exchange assets in an instant and low-cost manner. Such an approach makes asset exchange happen outside the blockchain and avoid high transaction fee and confirmation time. However, it is exceptionally unsafe because the central authority can manipulate users' assets arbitrarily [8]. It is reported that more than 30 hacks happened to centralized authorities in the past decade [9].

On the contrary, decentralized approaches remove the reliance on central authorities. The decentralized platform will only aggregate the asset exchange information, or metadata, rather than manage the assets. Then, a user will be able to perform asset exchanges with another based on the metadata on the platform. The decentralized approach demands a secure cross-chain swap protocol, enabling multiple parties to exchange assets across multiple blockchains. The protocol should guarantee that all or none of the desired asset exchanges happen. In the literature, hash time lock contracts are used for the purpose [10], [11], [12]. The decentralized approaches are secure because the asset from one user will not be

[1]https://coinmarketcap.com/view/metaverse/

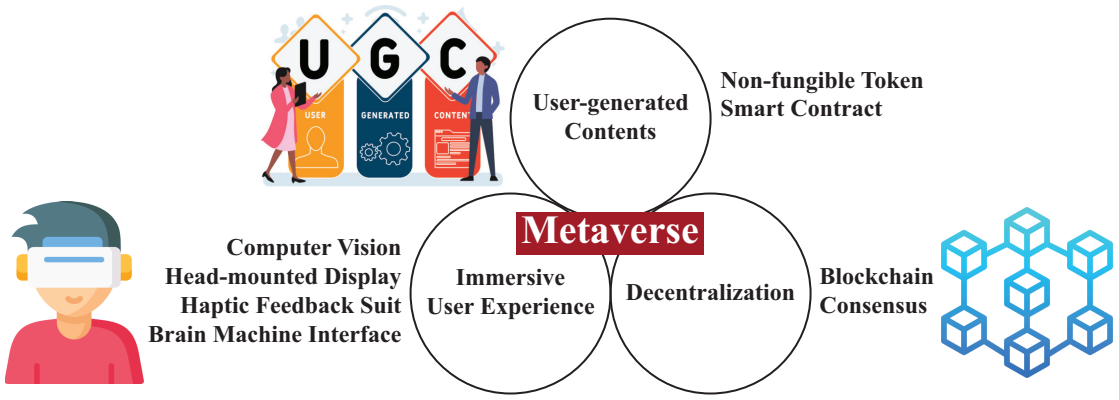

Fig. 2. The metaverse relies on three essentials: immersive user experience, user-generated content, and decentralization. Their enabling technologies range from sensing, computer vision, and brain-machine interfaces to virtual reality.

manipulated unless the asset exchange happens. However, the time overhead of such approaches is high because the cross-chain swaps take a long time to deploy smart contracts and transmit the secret [13].

This work proposes a fast and atomic cross-blockchain asset exchange protocol based on key exchange and smart contracts for metaverse interoperability. First, each party generates multiple key pairs according to the desired asset exchanges, and a smart contract generates a random number. Second, a one-time address for each desired asset exchange is generated based on the keys and random number. The assets are also deposited into one-time addresses. Each party will then scan the transactions to ensure the desired assets are deposited into the one-time address corresponding to itself. Third, if every party agrees, the smart contract will reveal the secret so that every party can compute the private keys of corresponding one-time addresses and get the assets. Such a protocol is secure due to the high security of key exchange and smart contracts. Furthermore, the time overhead is significantly reduced, especially when there are many asset exchanges, because it makes the exchanges happen in parallel.

The main contributions of this work are as follows:

- We present the first metaverse interoperability framework and discuss interoperable cyber worlds, compatible interaction mechanisms, and consistent physical infrastructures towards metaverse interoperability.
- We comprehensively reviewed the state-of-the-art research and projects concerning blockchain and metaverse interoperability.
- We propose a cross-blockchain asset exchange protocol based on key exchange and smart contracts for metaverse interoperability. The time overhead is significantly reduced compared to traditional protocols.

The rest of this paper is organized as follows. Sec. II introduces the preliminary knowledge of metaverse and our categorization of metaverse interoperability in three layers. Sec. III presents the related research and projects and articulates the motivations of this work. Sec. IV and Sec. V show the proposed cross-metaverse asset exchange protocol in detail and performance evaluation results, respectively. Finally, Sec. VI concludes this work.

II. Preliminaries

This section introduces this work's preliminary knowledge concerning the metaverse concept and its interoperability. We will explain the essentials of the metaverse and the general framework for realizing metaverse interoperability.

A. Metaverse

The term *metaverse* appeared for the first time in the science fiction *Snow Crash* published in 1992, referring to a synthetic urban environment in which the users can develop and trade the virtual real estates [14]. *Metaverse* is a word combining "meta" (meaning *beyond*) and "verse" (meaning *universe*) [15]. After nearly three decades of development, the concept metaverse caught people's attention again in 2018 due to the great success of the movie *Ready Player One*, in which the *OASIS*, a virtual reality entertainment universe with a high degree of freedom, was presented [16]. In the movie, the users can wear head-mounted displays and haptic feedback suits to enter *OASIS*, act as highly-customized avatars, interact with non-players characters and other users, play various immersive games, etc. It raises a question to the scientists and developers whether the technologies, such as sensing, brain-machine interfaces, and virtual/augmented/mixed reality, are ready for developing a metaverse [17].

Recently, several metaverse projects were initialized and launched, including Decentraland, The Sandbox, and Axie Infinity. The research community and industry are also investigating the feasibility of metaverse in shopping [18], education [19], hospitalization [20], etc. Compared to *OASIS*, the metaverses in these projects are decentralized instead of owned and managed by a single stakeholder [21]. Most of these projects are developed on top of existing public blockchains, mostly Ethereum [22]. The market capitalization of a single project can be billions of US dollars, e.g., Decentraland reached 6.6 billion US dollars in Dec 2021 [23]. In these projects, the users can create the cyber islands or buildings and trade them using the built-in cryptocurrencies. The cyber assets are also called non-fungible tokens (NFTs) [24].

From the existing metaverse projects, we find three essentials of a metaverse, namely immersive user experience, user-generated content, and decentralization, as shown in Fig. 2.

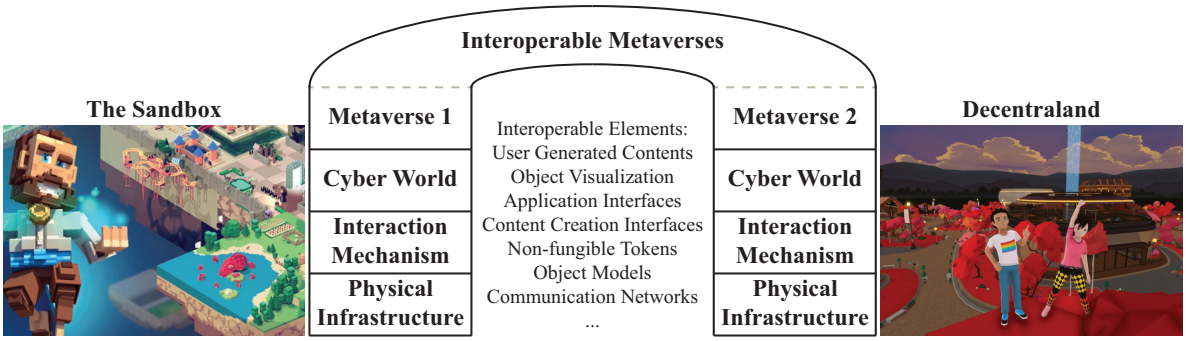

Fig. 3. The interoperability of metaverses is in three layers: cyber world, physical world, and interaction mechanism. Proper mechanisms in single and cross layers should be designed to enable metaverse interoperability.

First, a metaverse should render a nearly realistic and real-time virtual environment simulating the primary senses of touch, sight, hearing, smell, and taste. The enabling technologies include but are not limited to computer vision, head-mounted displays, haptic feedback suits, and brain-machine interfaces. Second, a metaverse should provide the users with easy-to-use interfaces to create highly-customized content. Copyright and terms of usage should be accompanied by user-generated content to protect users' rights and interests. Finally, a metaverse should never be controlled by any centralized stakeholder, such as an enterprise or an organization. Instead, the users can control the metaverse in a decentralized manner. Blockchain and smart contracts [25], [26], [27] are promising technologies that enable user-generated content and decentralization.

B. Metaverse Interoperability

According to the statistics from CoinMarketCap, there were up to 240 metaverse platforms till Feb 2023. Despite the large and increasing number of metaverses, they are fully isolated. First, the contents generated in one metaverse cannot be utilized in another. Second, the users can hardly explore crossing metaverses seamlessly using a single set of equipment. Furthermore, different metaverses employ heterogeneous infrastructure, including separated communication networks and distinct consensus mechanisms. In the future, more and more metaverse platforms will be established, and the interoperability of the vast number of metaverse platforms will become a critical issue.

Fig. 3 depicts the interoperability between two metaverses, e.g., Decentraland and The Sandbox. The support for metaverse interoperability should be provided threefold in the cyber world, interaction mechanisms, and physical infrastructure. The detailed explanations are as follows:

- The cyber worlds should be interoperable. The rendering and visualization of the cyber objects should be consonant, and the user-generated contents can be easily migrated from one metaverse to another.
- The interaction mechanisms should be compatible. In particular, the head-mounted displays and haptic feedback suits should be general for all metaverses rather than particular ones. The application interfaces and content creation interfaces should be compatible.
- The physical infrastructures should be consistent. For example, the non-fungible tokens are exchangeable, the communication networks can be seamlessly switched, and the object models are transformable.

There are extensive benefits to achieving metaverse interoperability. From the users' perspectives, they can conveniently interact across metaverses using a single set of equipment and payment channel, including exploring the virtual worlds, creating the contents, and exchanging the assets. From the metaverses' perspectives, they will form an integrated ecosystem by sharing the users and user-generated content. Such an ecosystem will be more robust to adversarial attacks than individuals and is promising to attract more users.

Among the three layers, the consistency of physical infrastructures is particularly important because they are the fundamental mechanisms of metaverses. This work focuses on cross-blockchain asset exchange, a challenging issue of consistent physical infrastructures, and proposes a secure and time-efficient protocol.

III. Related Work

This section summarizes the related work in academia and industry concerning cross-blockchain asset exchange and blockchain interoperability [28]. Based on the summary, we articulate the novelty and motivations of this work.

A. Cross-blockchain Asset Exchange

Cross-blockchain asset exchange refers to transferring or exchanging tokens among multiple blockchains. If the assets of two blockchains or metaverses are not exchangeable, then users' tokens or properties in one metaverse cannot be used or traded in another one. Such an issue will severely degrade users' quality of experience. Considering a scenario with two blockchains, a user exchanges 1 BTC (token in Bitcoin) for 16.27 ETH (token in Ethereum). There are two approaches to achieving so. On the one hand, the user A can transfer 1 BTC in Bitcoin to another user B who transfers 16.27 ETH back to A in Ethereum. On the other hand, the user A can destroy 1 BTC in Bitcoin, and 16.27 ETH are minted and deposited to A's account in Ethereum simultaneously. Cross-blockchain asset exchange can be divided into transfer-based and mint-based depending on whether destroyed tokens exist.

Transfer-based asset exchange has been extensively studied in the literature, also known as cross-chain swap [11]. The research community has been designing cross-chain swap protocols guaranteeing atomicity and improving the efficiency [12]. More specifically, atomicity means that the multiple transactions involved in cross-chain swaps must be either all confirmed either or none, and efficiency refers to the time and cost of completing a cross-chain swap [29]. Herlihy proposed the first atomic cross-chain swap protocol based on hash time lock contracts (HTLC) [10]. In particular, a secret hash value and multiple hash values-based smart contracts on the target blockchains are used and spread among asset dealers to enable the multiple asset-transfer transactions [30]. The multiple transactions in a cross-chain swap were modeled as a directed graph. It is proved that the swap is atomic if and only if the modeled graph is strongly connected. Imoto significantly improved the space and local time complexity by avoiding the excessive storage of the swap topology [13]. Liu et al. proposed AucSwap that leverages the Vickrey auction scheme to achieve efficient cross-chain asset transfer [31].

HTLC-based cross-chain asset exchange protocols intrinsically suffer from low time efficiency and high financial cost. Trusted hardware is also a promising approach to enabling efficient transfer-based asset exchange. Tesseract is the first trusted hardware-based cross-chain asset exchange protocol. In Tesseract, an asset exchange transaction is regarded as multiple asset transfer transactions on multiple blockchains. The transactions on independent blockchains are committed all or none owing to the trusted hardware [32].

More recently, zero-knowledge proof has been used to bridge two blockchains and provide asset exchange services [33]. Specifically, a relay blockchain C is used to store the block headers of blockchain A, relay the information on A to blockchain B, and provide transaction proof to B using zero-knowledge proof. Cross-chain asset exchange can be achieved by setting the relayed information properly. Although such an approach is decentralized, it incurs high computation overhead and transaction confirmation delay.

The mint-based solutions originate from proof-of-burn, an energy-efficient blockchain consensus protocol [34]. Karantias et al. studied the proof-of-burn consensus protocol from a cryptographic perspective [35]. In the proof-of-burn, the users send cryptocurrencies to an address that is unspendable by anyone so that a new corresponding account can be set up in a target blockchain. The cryptographic functions are used for the generation and verification of unspendable accounts. The proof-of-burn protocol achieves cross-chain asset exchange by enabling the destruction of cryptocurrencies in one blockchain and the minting of new ones in another blockchain [36].

B. Industrial Blockchain Interoperability Solutions

In industry, many cross-chain solutions exist, including Cosmos, Polkadot, Wanchain, and Plasma [37]. These cross-chain solutions primarily aim to create a blockchain ecosystem so that the tokens on different blockchains can be securely exchanged. In the following, we introduce the two representative solutions: Cosmos and Polkadot.

A Cosmos network [38] consists of a hub and multiple zones that are blockchains built by the Tendermint consensus protocol [39]. The zones are independent blockchains connected by the hub. Users can send assets from one zone to another through the hub. A network-level protocol, i.e., the inter-blockchain communication protocol [40], was developed to facilitate the message passing among the hub and zones. The Cosmos is a permissionless blockchain network open for developers to develop and integrate new zones.

The Polkadot network consists of a relay chain, multiple parachains, and bridges [41]. The relay chain is responsible for the underlying Polkadot nodes to make consensus and confirm transactions. In contrast, the parachains do not confirm transactions but only receive the ones from the relay chain and process them. A bridge is between a parachain and the relay chain to connect them. It enables communication among the relay chain, parachains, and even other public blockchains not in the Polkadot network. The Polkadot network employs the consensus protocol of the Substrate [42].

The Cosmos and Polkadot solutions share specific characteristics. Both of them have a main chain run by a specialized consensus protocol (the hub run by the Tendermint protocol in the Cosmos and the relay chain run by the Substrate protocol in the Polkadot) and multiple interconnected side chains (the zones in the Cosmos and the parachains in the Polkadot).

C. Programming Multiple Blockchains

Blockchain interoperability concerns cross-blockchain asset exchange and cross-blockchain programming and queries. Specifically, programming over multiple blockchains is essential due to the popularity of developing decentralization applications through smart contracts. As the number of blockchains and volume of blockchain data keep exploding, cross-blockchain queries are essential to support applications involving multiple blockchains.

HyperService is the first platform that supports programming across multiple heterogeneous blockchains [43]. The key enabling components are a cross-blockchain programming abstraction and a cryptography protocol facilitating secure program execution. The major limitations of Hyperservice are twofold. On the one hand, Hyperservice incurs an unavoidable delay in creating handshaking sessions for decentralization applications on heterogeneous blockchains. On the other hand, the programming support is inadequate, e.g., conditional branching, looping, and calling and returning from procedures are not available.

Vassago is the first protocol enabling provenance queries over multiple blockchains [44]. Without Vassago, cross-blockchain provenance queries are unreliable and inefficient because of the lack of global knowledge and sequential queries over multiple blockchains. In Vassago, a shared blockchain is employed to validate the results of cross-blockchain provenance queries. Moreover, the queries over multiple blockchains are parallelized to boost query efficiency. The limitation of Vassago lies in its undue reliance on the shared blockchain, which raises security and efficiency concerns.

To summarize, most existing industrial and academic studies concern cross-blockchain asset exchanges while neglecting the support of upper-level applications involving multiple blockchains. Only several studies discuss the programming and queries over multiple blockchains. However, the efficiency of the approaches is still limited. To our knowledge, there is no work considering the interoperability of multiple metaverses.

IV. Fast and Atomic Cross-blockchain Asset Exchange

We propose to leverage smart contract and hash-locking technologies to establish a fast and atomic cross-blockchain asset exchange protocol for metaverse interoperability.

A. Terminologies

We provide the terminology of atomic swap as follows:

- G: a base point; $G = (x, -4/5)$;
- E: an elliptic curve equation; $-x^2 + y^2 = 1 + dx^2y^2$;
- Private ec-key: a standard elliptic curve private key: a number $a \in [1, l-1]$;
- Public ec-key: a standard elliptic curve public key: a point $A = aG$;
- One-time keypair: a pair of private and public ec-keys;
- Private user key: a pair (a, b) of two different private ec-keys;
- Tracking key is: a pair (a, B) of private and public ec-key (where $B = bG$ and $a \neq b$);
- Public user key: a pair (A, B) of two public ec-keys derived from (a, b)
- Standard address: a representation of a public user key given into a human-friendly string with error correction;
- Truncated address: a representation of the second half (point B) of a public user key given into a human-friendly string with error correction.
- Digital Signatures: $(pk, sk) = generateKeys(Any_Keysize)$ where pk is public verification key (public address) and sk is the secret signing key respectively.

B. Protocol Overview

Atomic cross-chain swap is a distributed coordination task where multiple parties exchange assets across multiple blockchains, e.g., trading bitcoin for ether. It should guarantee that:

- if all parties conform to the protocol, then all swaps take place;
- if some coalition deviates from the protocol, then no conforming party ends up worse off;
- no coalition has the incentive to deviate from the protocol.

However, the existing method (Hashed-Timelock Agreements) is not time-efficient. It requires at least 2Δ time slots, in Fig. 4, where n is the number of users, and Δ is a period for deploying the smart contract.

Specifically, given a set of TXs proposed by nodes from different Blockchains, and each TX indicating an event/operation on one blockchain; We assume that the nodes can communicate with each other safely and that each blockchain supports smart contract; The objective is to guarantee the atomicity: all transactions confirmed by affiliated blockchains or not, and time efficiency: reduce the latency of swap.

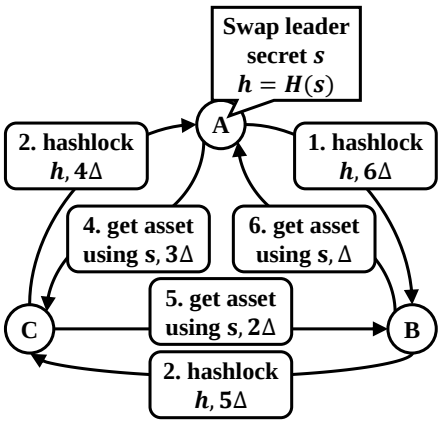

Fig. 4. Procedures of the hash time lock contracts-based approach. The swap leader A generates a secret and transmits it to B, C, and itself. Then, node A reveals the secret to getting the expected assets. The other nodes acquire the secret from A and can also get the expected assets. The approach takes $\mathcal{O}(n)$ time, where n is the number of participating parties.

Our proposed approach aims to finish the swap within a constant $c\Delta$ time. However, we need to solve the following three challenging issues:

1) How to temporarily hold the deposit? We need a mechanism to manage the private key to the deposit, which is intended for exchange.
2) How to guarantee that the deposit money can be returned if a bad thing happens? When a collision happens, the conforming party should not lose the deposit.
3) How to protect the assets once reveal the deposit key? E.g., Alice may retrieve the deposit, which is sent by Alice ahead of Bob, if Alice knows the key.

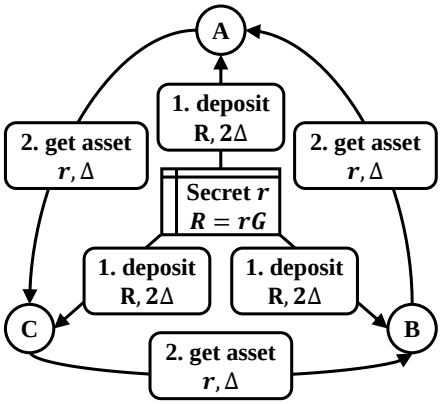

Fig. 5. Procedures of the proposed approach. A predefined smart contract initializes the asset exchange with a secret. Then, the participating parties deposit the assets into the smart contract. Finally, the secret is revealed, and all the parties can get the expected assets. The approach takes $\mathcal{O}(1)$ time because the participating parties take actions in parallel.

We propose using a two-phase approach to reduce the latency, as depicted in Fig. 5. The existing solution uses a

ring-based method where all the operations are carried out individually. In our approach, the first phase is to deposit. Users will invoke a smart contract to temporarily lock the assets using an encrypted one-time address simultaneously. It will only cost Δ time. The second phase is to Retrieve. Users will scan every passing transaction with the private key to locate transactions sent to them. Then users will reveal their secret random number to decrypt the one-time address and claim the assets simultaneously. It will only cost Δ time as well. Since all the operations in each phase happen simultaneously, we can reduce the $n\Delta$ time to $c\Delta$ time, which is a constant value. We will explain the solution using an example as follows.

C. Protocol Step by Step

This work considers two users, Alice and Bob, as an example. Alice wants to exchange x BTC with Bob and receive y ETH from Bob. The exchange rate x/y is predefined. Alice has two key pairs: (a_{alice}, A_{alice}) and $(b_{alice}, B_{alice}$ where $A_{alice} = a_{alice} \cdot G$ and $B_{alice} = b_{alice} \cdot G$. Similarly, Bob has two key pairs: (a_{bob}, A_{bob}) and $(b_{bob}, B_{bob}$ where $A_{bob} = a_{bob} \cdot G$ and $B_{bob} = b_{bob} \cdot G$.

Step 1: random number generation. Alice and Bob agree to exchange the assets using a smart contract. First, Alice will generate a random number key pair $(r_{alice}, R_{alice}$ where $R_{alice} = r_{alice} \cdot G$. Alice publishes R_{alice} to the network. At the same time, Bob will generate a random number key pair $(r_{bob}, R_{bob}$ where $R_{bob} = r_{bob} \cdot G$, and publish R_{bob} to the network as well. Then, the smart contract will also generate a random number key pair (r_{sc}, R_{sc}) where $R_{sc} = r_{sc} \cdot G$, and publishes R_{sc} to the network. Until this step, Alice and Bob have three key pairs, two for the account and one for the one-time random number. The smart contract has one key pair that is a one-time random number.

We use two key pairs for each account because the receiver needs to scan every passing transaction using their own key pair in future steps. Such an action costs extra time if done by the user. Using two key pairs allows the user to pass the tracking key pair, a *truncated address*, to a third party for locating the transactions. The approach is helpful when the receiver lacks network bandwidth and computation capabilities.

Step 2: asset deposition. Alice has A_{bob}, B_{bob}, R_{sc} because they are publicly available. Alice will deposit x BTC to the one-time address $Pay.toBob$ computed as:

$$Pay.toBob = H(r_{alice} \cdot A_{bob}) \cdot G + a_{alice} \cdot R_{sc} \cdot B_{bob} \quad (1)$$

Then, Alice broadcasts the transaction to the blockchain network.

Step 3: transaction scanning and location. Bob checks every passing transaction using their private key a_{bob} and b_{bob} by computing:

$$P' = H(R_{alice} \cdot a_{bob}) \cdot G + A_{alice} \cdot R_{sc} \cdot b_{bob} \quad (2)$$

Note that R_{alice}, A_{alice}, and R_{sc} are publicly available. Bob can check whether $Pay.toBob = P'$ for each transaction to determine whether its receipt is Bob. This is because substituting Eq. 3 and Eq. 4 into Eq. 1 gets Eq. 2.

$$r_{alice} \cdot A_{bob} = r_{alice} \cdot G \cdot a_{bob} = R_{alice} \cdot a_{bob} \quad (3)$$

$$a_{alice} \cdot R_{sc} \cdot B_{bob} = a_{alice} \cdot R_{sc} \cdot b_{bob} \cdot G = A_{alice} \cdot R_{sc} \cdot b_{bob} \quad (4)$$

In this step, only Alice and Bob know the meaning of the one-time address $Pay.toBob$. Any other user from the network can see the transactions from Alice but cannot link them back to Bob's account. After this step, Alice will deposit the assets to the one-time address, and Bob notices such a transaction. However, Bob does not know the private key to spend the assets.

Step 4: asset claiming. The smart contract publishes the secret key r_{sc} based on the contract script. Bob cab recover the corresponding one-time private key p_x as follows:

$$p_x = H(R_{alice} \cdot a_{bob}) + A_{alice} \cdot r_{sc} \cdot b_{bob} \quad (5)$$

The private key p_x can claim the assets in $Pay.toBob$ because $Pay.toBob = p_x \cdot G$. The calculation is as follows:

$$Pay.toBob = (H(R_{alice} \cdot a_{bob}) + A_{alice} \cdot r_{sc} \cdot b_{bob}) \cdot G \quad (6)$$

It means Bob can use p_x to sign a transaction that spends the assets in $Pay.toBob$. Similarly, Bob will deposit y ETH into a one-time address $Pay.toAlice$, and Alice can use a private key to withdraw the asset.

V. Performance Evaluation

A. Discussion

In Eq. 1, we separate $H(r_{alice} \cdot A_{bob}) \cdot G$ or $H(R_{alice} \cdot a_{bob}) \cdot G$ before the $+$ mark. The purpose is to limit access to the one-time address to Alice and Bob only since r_{alice} and a_{bob} are both private.

In the later transformation part, we creatively utilize the *three (multi)-party Diffie-Helman key exchanges*. To our knowledge, such a technique has not yet been used in blockchain applications. We use it to lock both users' assets temporarily. So far, only Monero and its forked projects use *two-party Diffie-Helman key exchanges* technique to improve transaction anonymity and remove traceability. We list the full equation transformation with its operation meanings below:

$$\begin{aligned}
Pay.toBob &= H(r_{alice} \cdot A_{bob}) \cdot G + a_{alice} \cdot R_{sc} \cdot B_{bob} \\
&= H(r_{alice} \cdot A_{bob}) \cdot G + A_{alice} \cdot R_{sc} \cdot b_{bob} \\
&= (H(r_{alice} \cdot A_{bob}) + a_{alice} \cdot r_{sc} \cdot B_{bob}) \cdot G \\
&= (H(r_{alice} \cdot a_{bob} \cdot G) + a_{alice} \cdot r_{sc} \cdot b_{bob} \cdot G) \cdot G \\
&= (H(r_{alice} \cdot G \cdot a_{bob}) + a_{alice} \cdot G \cdot r_{sc} \cdot b_{bob}) \cdot G \\
&= (H(R_{alice} \cdot a_{bob}) + A_{alice} \cdot r_{sc} \cdot b_{bob}) \cdot G
\end{aligned}$$

The benefits of the proposed solution are as follows:

- The assets are controlled by Alice and Bob all the time,
- The transaction is only visible to Alice and Bob and anonymous to others,
- During the swap, the assets will be temporarily locked by the smart contract with a one-time random secret, and
- If only Alice deposits the assets, r_{sc} can be revealed only to Alice automatically to return the assets.

B. Experimental Results

From the existing hash time lock contracts (HTLC)-based method for atomic swap [10], the time efficiency can be improved since it requires at least $2n \cdot \Delta$ time slots where n is the number of participants in this swap, and Δ is a period for deploying the smart contract. Given an example of three parties exchange as depicted in Fig. 6 and Fig. 7, it requires at least 6Δ time slots to complete the swap process where six is calculated by three members to multiply two main phases, specifically deploying contracts and triggering arcs.

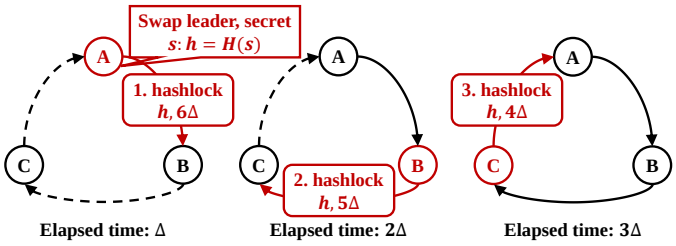

Fig. 6. Phase one of the HTLC approach is contract deployment. It runs step by step as follows: 1) the leader A generates a secret s and deploys an HTLC to B using $H(s)$, 2) B observes A's HTLC and deploys an HTLC to C using $H(s)$, and 3) C observes B' HTLC and deploys an HTLC to A using $H(s)$. Each step takes Δ time, and phase one takes 3Δ time.

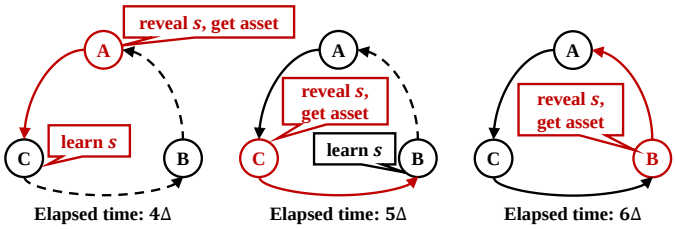

Fig. 7. Phase two of the HTLC approach is triggering the contract. It runs step by step as follows: 1) A uses secret s to get the assets in C's HTLC, 2) C knows the secret s and uses it to get the assets in B's HTLC, and 3) B knows the secret s and uses it to get the assets in A's HTLC. Each step takes Δ time, and phase two takes 3Δ time.

We propose the new two phases-based approach, which can reduce the swap latency from existing $\mathcal{O}(n)$ to $\mathcal{O}(1)$. Instead of using the existing ring-based approach, we parallelize users' operations into two major phases, *deposit* and *retrieve*.

The *deposit* includes steps 1 and 2. Each user will prepare and deposit the assets to the target account using the new computed one-time address, e.g., $Pay.toBob$ in equation (1). Since the operation can be done by all users (participants in this swap) simultaneously, the time complexity is 1Δ time, which is a constant value. *Retrieve* includes step 3 and step 4, the receiver will scan all passing transactions using its private key pairs and claim assets once the smart contract reveals the r_{sc}. This operation can also be done by all users simultaneously where the time complexity is 1Δ time, still a constant value. The total time complexity is 1Δ time from *deposit* plus 1Δ time from *Retrieve*, which equals to 2Δ time, still a constant value as $\mathcal{O}(1)$. The total swap time complexity comparison is depicted in Fig. 8.

In summary, time efficiency is improved by operation parallelization. The private-public key generation and management

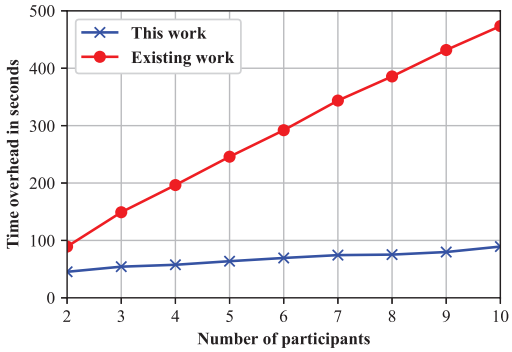

Fig. 8. Comparison of time overhead between the traditional and proposed approaches when $\Delta = 20s$. The time overhead of the HTLC approach increases linearly to the number of participating parties. The HTLC approach takes up to 500 seconds when there are ten parties. Despite the number of participating parties, the proposed approach takes nearly unchanged time. It takes around 60 seconds to complete every cross-blockchain asset exchange.

in mainstream blockchains are still limited to single key pair, $P = x \cdot G$ with $P = x \cdot G + B = (x + b) \cdot G$ where $(x + b)$ is the new private spent key. This work is a totally new solution because the account is controlled by two key pairs, not only one. The sender can compute the receiver address for one-off use and deposit the assets. Although the sender knows the xG and B, the private key to spend this transaction, is unknown to the sender, the assets are safe to the receiver, who can spend this transaction later.

VI. Conclusion

This work is the first to study metaverse interoperability comprehensively. We introduce a three-layer metaverse interoperability framework discussing the interoperable cyber worlds, compatible interaction mechanisms, and consistent physical infrastructures. Then, we focus on the important asset exchange issue contributing to consistent physical infrastructures. We investigate the existing approaches and find they are time-consuming and can hardly afford the high-frequency transactions in metaverses. To this end, we propose a novel cross-blockchain asset exchange protocol based on three-party Diffie-Helman key exchanges and smart contracts. The proposed protocol achieves second-level asset exchange despite the number of participating parties.

Acknowledgement

This work was supported by the Research Institute for Artificial Intelligence of Things, The Hong Kong Polytechnic University, Hong Kong SAR, China; the Hong Kong Research Grants Council Collaborative Research Fund, under Grant C2004-21GF; and the Hong Kong Research Grants Council Research Impact Fund, under Grant R5034-18.

References

[1] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Communications Surveys & Tutorials*, 2022.

[2] M. Xu, W. C. Ng, W. Y. B. Lim, J. Kang, Z. Xiong, D. Niyato, Q. Yang, X. S. Shen, and C. Miao, "A full dive into realizing the edge-enabled metaverse: Visions, enabling technologies, and challenges," *IEEE Communications Surveys & Tutorials*, 2022.

[3] R. Hare and Y. Tang, "Hierarchical deep reinforcement learning with experience sharing for metaverse in education," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022.

[4] T. Zhang, J. Shen, C.-F. Lai, S. Ji, and Y. Ren, "Multi-server assisted data sharing supporting secure deduplication for metaverse healthcare systems," *Future Generation Computer Systems*, vol. 140, pp. 299–310, 2023.

[5] M. Deveci, A. R. Mishra, I. Gokasar, P. Rani, D. Pamucar, and E. Özcan, "A decision support system for assessing and prioritizing sustainable urban transportation in metaverse," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 2, pp. 475–484, 2022.

[6] T. Li, C. Yang, Q. Yang, S. Zhou, H. Huang, and Z. Zheng, "Meta-opera: A cross-metaverse interoperability protocol," *arXiv preprint arXiv:2302.01600*, 2023.

[7] B. Chen, C. Song, B. Lin, X. Xu, R. Tang, Y. Lin, Y. Yao, J. Timoney, and T. Bi, "A cross-platform metaverse data management system," in *IEEE International Conference on Metrology for Extended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*. IEEE, 2022, pp. 145–150.

[8] A. Barbon and A. Ranaldo, "On the quality of cryptocurrency markets: Centralized versus decentralized exchanges," *arXiv preprint arXiv:2112.07386*, 2021.

[9] P. Xia, H. Wang, B. Gao, W. Su, Z. Yu, X. Luo, C. Zhang, X. Xiao, and G. Xu, "Trade or trick? detecting and characterizing scam tokens on uniswap decentralized exchange," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 5, no. 3, pp. 1–26, 2021.

[10] M. Herlihy, "Atomic cross-chain swaps," in *ACM Symposium on Principles of Distributed Computing (PODC)*, 2018, pp. 245–254.

[11] M. Belotti, S. Moretti, M. Potop-Butucaru, and S. Secci, "Game theoretical analysis of cross-chain swaps," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2020, pp. 485–495.

[12] L. Lys, A. Micoulet, and M. Potop-Butucaru, "R-swap: Relay based atomic cross-chain swap protocol," in *International Symposium on Algorithmic Aspects of Cloud Computing (ALGOCLOUD)*. Springer, 2021, pp. 18–37.

[13] S. Imoto, Y. Sudo, H. Kakugawa, and T. Masuzawa, "Atomic cross-chain swaps with improved space and local time complexity," in *International Symposium on Stabilizing, Safety, and Security of Distributed Systems (SSS)*. Springer, 2019, pp. 194–208.

[14] F.-Y. Wang, R. Qin, X. Wang, and B. Hu, "Metasocieties in metaverse: Metaeconomics and metamanagement for metaenterprises and metacities," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 1, pp. 2–7, 2022.

[15] J. D. N. Dionisio, W. G. B. III, and R. Gilbert, "3d virtual worlds and the metaverse: Current status and future possibilities," *ACM Computing Surveys*, vol. 45, no. 3, pp. 1–38, 2013.

[16] H. Duan, J. Li, S. Fan, Z. Lin, X. Wu, and W. Cai, "Metaverse for social good: A university campus prototype," in *ACM International Conference on Multimedia (MM)*. ACM, 2021, pp. 153–161.

[17] M. A. I. Mozumder, M. M. Sheeraz, A. Athar, S. Aich, and H.-C. Kim, "Overview: Technology roadmap of the future trend of metaverse based on iot, blockchain, ai technique, and medical domain metaverse activity," in *International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2022, pp. 256–261.

[18] N. Xi, J. Chen, F. Gama, M. Riar, and J. Hamari, "The challenges of entering the metaverse: An experiment on the effect of extended reality on workload," *Information Systems Frontiers*, pp. 1–22, 2022.

[19] I. A. Akour, R. S. Al-Maroof, R. Alfaisal, and S. A. Salloum, "A conceptual framework for determining metaverse adoption in higher institutions of gulf area: An empirical study using hybrid sem-ann approach," *Computers and Education: Artificial Intelligence*, vol. 3, p. 100052, 2022.

[20] D. Gursoy, S. Malodia, and A. Dhir, "The metaverse in the hospitality and tourism industry: An overview of current trends and future research directions," *Journal of Hospitality Marketing & Management*, pp. 1–8, 2022.

[21] S. M. H. Bamakan, N. Nezhadsistani, O. Bodaghi, and Q. Qu, "Patents and intellectual property assets as non-fungible tokens; key technologies and challenges," *Scientific Reports*, vol. 12, no. 1, pp. 1–13, 2022.

[22] D. Chirtoaca, J. Ellul, and G. Azzopardi, "A framework for creating deployable smart contracts for non-fungible tokens on the ethereum blockchain," in *IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*. IEEE, 2020, pp. 100–105.

[23] C. Goanta, "Selling land in decentraland: the regime of non-fungible tokens on the ethereum blockchain under the digital content directive," in *Disruptive Technology, Legal Innovation, and the Future of Real Estate*. Springer, 2020, pp. 139–154.

[24] L. Kugler, "Non-fungible tokens and the future of art," *Communications of the ACM*, vol. 64, no. 9, pp. 19–20, 2021.

[25] S. Jiang, J. Cao, H. Wu, Y. Yang, M. Ma, and J. He, "Blochie: a blockchain-based platform for healthcare information exchange," in *IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2018, pp. 49–56.

[26] M. Zhang, J. Cao, Y. Sahni, Q. Chen, S. Jiang, and L. Yang, "Blockchain-based collaborative edge intelligence for trustworthy and real-time video surveillance," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1623–1633, 2022.

[27] H. Wu, S. Jiang, and J. Cao, "High-efficiency blockchain-based supply chain traceability," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[28] A. Singh, K. Click, R. M. Parizi, Q. Zhang, A. Dehghantanha, and K.-K. R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, p. 102471, 2020.

[29] J. Xu, D. Ackerer, and A. Dubovitskaya, "A game-theoretic analysis of cross-chain atomic swaps with htlcs," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2021, pp. 584–594.

[30] S. K. Mohanty and S. Tripathy, "n-htlc: Neo hashed time-lock commitment to defend against wormhole attack in payment channel networks," *Computers & Security*, vol. 106, p. 102291, 2021.

[31] W. Liu, H. Wu, T. Meng, R. Wang, Y. Wang, and C.-Z. Xu, "Aucswap: A vickrey auction modeled decentralized cross-blockchain asset transfer protocol," *Journal of Systems Architecture*, vol. 117, p. 102102, 2021.

[32] I. Bentov, Y. Ji, F. Zhang, L. Breidenbach, P. Daian, and A. Juels, "Tesseract: Real-time cryptocurrency exchange using trusted hardware," in *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 2019, pp. 1521–1538.

[33] T. Xie, J. Zhang, Z. Cheng, F. Zhang, Y. Zhang, Y. Jia, D. Boneh, and D. Song, "zkbridge: Trustless cross-chain bridges made practical," in *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 2022, pp. 3003–3017.

[34] I. Homoliak, S. Venugopalan, D. Reijsbergen, Q. Hum, R. Schumi, and P. Szalachowski, "The security reference architecture for blockchains: Toward a standardized model for studying vulnerabilities, threats, and defenses," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 341–390, 2020.

[35] K. Karantias, A. Kiayias, and D. Zindros, "Proof-of-burn," in *International Conference on Financial Cryptography and Data Security (FC)*. Springer, 2020, pp. 523–540.

[36] B. Pillai, K. Biswas, Z. Hóu, and V. Muthukkumarasamy, "Burn-to-claim: An asset transfer protocol for blockchain interoperability," *Computer Networks*, vol. 200, p. 108495, 2021.

[37] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A survey on blockchain interoperability: Past, present, and future trends," *ACM Computing Surveys*, vol. 54, no. 8, pp. 1–41, 2021.

[38] A. Howell, T. Saber, and M. Bendechache, "Measuring node decentralisation in blockchain peer to peer networks," *Blockchain: Research and Applications*, p. 100109, 2022.

[39] Y. Amoussou-Guenou, A. Del Pozzo, M. Potop-Butucaru, and S. Tucci-Piergiovanni, "Correctness of tendermint-core blockchains," in *International Conference on Principles of Distributed Systems (OPODIS)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[40] L. Kan, Y. Wei, A. H. Muhammad, W. Siyuan, L. C. Gao, and H. Kai, "A multiple blockchains architecture on inter-blockchain communication," in *IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. IEEE, 2018, pp. 139–145.

[41] T. Koens and E. Poll, "Assessing interoperability solutions for distributed ledgers," *Pervasive and Mobile Computing*, vol. 59, p. 101079, 2019.

[42] V. Shcherba and R. Hussain, "Blockchain and games: a novel middleware for blockchain-based multiplayer games," in *ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM) Poster and Demo Sessions*. ACM, 2021, pp. 9–11.

[43] Z. Liu, Y. Xiang, J. Shi, P. Gao, H. Wang, X. Xiao, B. Wen, and Y.-C. Hu, "Hyperservice: Interoperability and programmability across heterogeneous blockchains," in *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 2019, pp. 549–566.

[44] R. Han, J. Xiao, X. Dai, S. Zhang, Y. Sun, B. Li, and H. Jin, "Vassago: Efficient and authenticated provenance query on multiple blockchains," in *International Symposium on Reliable Distributed Systems (SRDS)*. ACM, 2021, pp. 132–142.