Babich V, Birge J, Hilary G (eds) Innovative Technology at the interface of Finance and Operations. Springer Series in Supply Chain Management, forthcoming, Springer Nature

Blockchain Intra- and Interoperability

A. Lipton[1], **T. Hardjono**[2]

[1]MIT Connection Science and Engineering,
Massachusetts Institute of Technology, Cambridge, MA 02139 USA e-mail: alexlipt@mit.edu

[2]MIT Connection Science and Engineering,
Massachusetts Institute of Technology, Cambridge, MA 02139 USA e-mail: hardjono@mit.edu

The date of receipt and acceptance will be inserted by the editor

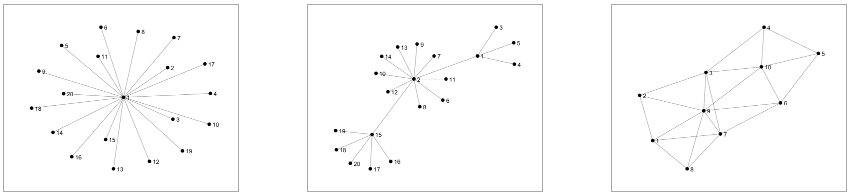
Abstract We introduce blockchains and distributed ledgers and study their intra- and interoperability. Blockchain intraoperability allows one to swap different assets defined on the same blockchain supporting smart contracts. Blockchain interoperability enables one to exchange or move assets residing on different blockchains. Finding practical mechanisms for intra- and interoperability is of paramount importance for the ultimate success of blockchain technology. We recommend using automated market makers for intraoperability and gateways and atomic swaps for interoperability.

Key words Blockchains – Digital Assets – Intraoperability – Interoperability

1 Introduction

The publication of the seminal white paper by Satoshi Nakamoto in October of 2008 opened a new era of technological innovation - the era of distributed ledgers; see [25]. Nakamoto's paper introduced the public blockchain concept and explained how one could build such a blockchain in practice. It started a revolution, which shows no signs of abating.

At the basic level, blockchain is a shared, distributed ledger capable of supporting ownership of assets, tracking transactions between assets' owners, and changing ownership accordingly; see, e.g., [13]. In theory, blockchains can support tangible assets, such as money, shares, real estate, and intangible assets, such as intellectual property. Eventually, blockchain networks can support ownership of anything of value. Recording ownership of a blockchain reduces risks by increasing business interoperability, cuts costs for all involved, and eliminates intermediaries. By using blockchains, one could radically reorganize business *modus operandi.*



(a) centralized

(b) decentralized

(c) distributed

Fig. 1: Three possible ways of organizing networks.

One can think about three complementary ways of organizing information: (a) centralized, (b) decentralized, (c) distributed; see Figure 1. Currently, the centralized hub and spoke model is standard for most industries, for example, banking in a single country. In this example, the hub is the central bank, and the spokes are individual commercial banks. A decentralized system with several hubs and spokes is also quite common. A typical example is cross-border banking. The hubs are central banks, and the spokes are individual commercial banks in their respective countries. Finally, a distributed system relies on direct or peer-to-peer (P2P) business organization (currently a rare instance) and potentially can be viewed as the most robust of the three. The great hope for the distributed ledger technology (DLT) is that it can serve as the backbone of distributed business models.

Using the recent developments in DLT, we can reorganize business activities in general. In Figure 2, we compare two possibilities: (a) the current system, with each participant holding her own ledger, reconciled against other ledgers periodically, (b) a future system relies on all parties, maintaining a shared ledger as a group. The current system enjoys several advantages, including each participant's ability to rely on her trusted systems to control the information released to other participants. However, such an approach results in redundancies, sporadical errors, and potentially fraudulent activities by its very nature. In the future system, each participant can write transactions on the ledger and access the relevant information she is entitled to know. Such a system reduces the business process's overall frictions and increases its robustness because of built-in (rather than accidental) redundancies. Clearly, streamlining the business process using DLT is not free because it requires maintaining consensus on the shared ledger and properly obfuscating private data.

Figure 3 convincingly illustrates the fact that Google trends showing interest in DLT, undeniably kindled by the meteoric price appreciation of Bitcoin and other cryptocurrencies, did not go away (although naturally diminished) after Bitcoin's bubble burst. It suggests that DLT has many more usages in addition to its applications to cryptocurrencies.

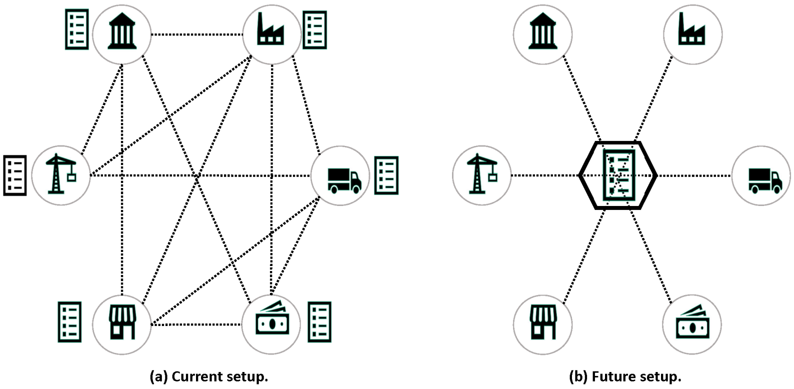


Fig. 2: (a) Current arrangement - Each participant maintains her ledger; participants periodically reconcile ledgers against each other. (b) Future arrangement - Each participant contributes to the shared distributed ledger; a suitable consensus algorithm maintains the ledger's integrity. The information has to be suitably encrypted to satisfy business requirements and data privacy laws.

There is currently tremendous interest in using blockchain and smart contracts technology to re-implement many of the existing functions within the financial sector (e.g. automated market maker) in a decentralized fashion. Numerous "decentralized finance" (DeFi) projects or offerings have been reported in the media in the past year. A core proposition in decentralized finance is that anyone can be a market-maker, and that the replicated copies of smart contracts ensures the "decentralization" of functions.

However, we believe that decentralization should occur across distinct blockchain systems in a manner that provides the application developer with a single view regardless of how many distinct blockchain systems are involved. That is, true decentralization means that functions are spread across distinct blockchain systems, where nodes may be implemented using different software stacks and where each blockchain system may employ different consensus mechanisms and different ledger data structures [15]. Indeed, this is how the TCP/IP Internet is architected and this is largely

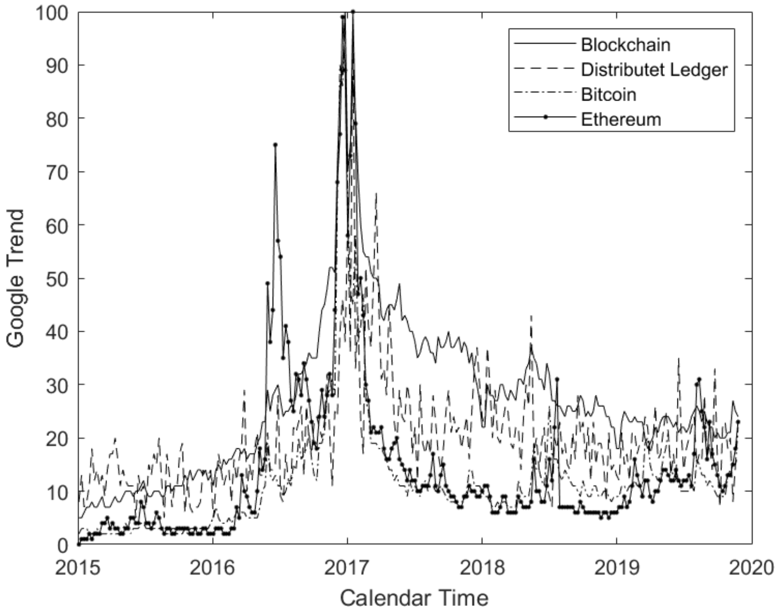


Fig. 3: Google trends reflecting the worldwide interest in blockchains, distributed ledgers, Bitcoin, and Ethereum. Source: https://trends.google.com/

why the Internet has been able to grow in size and in traffic capacity to serve end-users at a global scale. The Internet is not a single contiguous IP network. It is in fact a collection of interconnected *Autonomous Systems* (AS) [6], where each AS has a well-defined physical boundary and each AS is operated by an ISP. In contrast, in the case of the recent DeFi offerings, most (if not all) of the DeFi efforts have been conducted on one blockchain platform, namely Ethereum.

Although much of DeFi activities are today on the Ethereum platform, the history of the Internet indicates that it is unlikely that there will be a single “world computer”. Private (permissioned) blockchains and DLT systems today are used in closed communities (e.g. see [32]), and maintaining separate blockchains may be part of the business survivability of many of these communities. Other platforms are being developed (e.g. Dfinity [24, 19]), and geopolitical factors may prevent the market dominance by a single blockchain platform [21].

If blockchain technology is to become a foundation for the future decentralized finance then the issue of interoperability must be addressed. Standard technical protocols for the various aspects of interoperability must be created, tested and deployed – just as numerous standard protocols have been developed over the past three decades for the TCP/IP Internet.

The purpose of this paper is to introduce the concepts of blockchain intra- and interoperability and describe their potential practical applications. It is organized as follows. Section 2 describes several essential examples, including Bitcoin, and Ethereum. Section 3 introduces the concept of blockchain intraoperability and shows how to achieve intraoperability in an Ethereum-like blockchain by using automated market makers. Section 4 defines the concept of blockchain interoperability and develops several viable approaches to designing the corresponding mechanisms, including gateway asset transfer protocols. Conclusions are drawn in Section 5.

A recent paper by Hardjono *et al.* and a book by Lipton and Treccani cover this chapter's material in much greater detail and contains additional references; see [15,23].

2 Blockchains in a nutshell

2.1 Bitcoin

Nakamoto described her intentions as follows: "I've been working on a new electronic cash system that's fully peer-to-peer, with no trusted third party — The main properties: Double-spending is prevented with a peer-to-peer network. No mint or other trusted parties. Participants can be anonymous. New coins are made from Hashcash style proof-of-work. The proof-of-work for new coin generation also powers the network to prevent double-spending." The result of these efforts is the celebrated Bitcoin protocol for moving the corresponding token known as BTC from one of the protocol participants to the next. Nakamoto's designed blockchain relies on public-key or asymmetric cryptography, specifically elliptic-curve cryptography (ECC). It uses pairs of keys: a public key, known to all, and a secret key, known only to the owner. Public keys are the protocol participants' addresses, while secret keys are tools for unlocking BTCs held at the corresponding addresses.

All BTC transactions are cryptographically secured via the elliptic curve digital signature algorithm (ECDSA) and do not require further efforts to ensure that they are valid. However, in the absence of strict controls, nothing prevents the owner of a particular address from spending her money BTCs twice - the celebrated double-spend problem. To ensure that the Bitcoin protocol is self-consistent and prevents double-spending, it is necessary to have a distinctive class of participants in the Bitcoin protocol, called miners. Miners listen to the network for upcoming transactions, assemble these transactions into blocks, and participate in a competition to have their block added to the system. Other miners will accept a new block only if it contains valid transactions and is correctly stamped by the successful miner. To stamp a block, miners need to solve a computational hash puzzle, requiring spending electricity and other resources on a prodigious scale.[1] Thus, the name - a proof-of-work (PoW) consensus algorithm.

2.2 Ethereum

Ethereum addresses the limitations of Bitcoin scripts by introducing smart contracts (SCs), which are stateful and Turing-complete scripts, at least in theory. Ethereum considerably augments the capabilities of DLTs and their possible applications. While the Bitcoin protocol focuses on a single use case of consistently mining and moving BTC, which it views as electronic cash, the Ethereum protocol offers a decentralized, trusted computing platform for executing arbitrary code. Its native cryptocurrency, ether (ETH), while undeniably very important, merely acts as the token financing the execution of SCs by thousands of machines. Since SCs regulate all kind of separate use cases, the best way of thinking about Ethereum is as consensus as a service (CaaS) provider for any application that requires a high level of trust and auditability.[2]

Ethereum was never shy about its grandiose ambitions. The Ethereum white paper states its purpose as follows: “Satoshi Nakamoto’s development of Bitcoin in 2009 has often been hailed as a radical development in money and currency, being the first example of a digital asset which simultaneously has no backing or intrinsic value and no centralized issuer or controller. However, another - arguably more important - part of the Bitcoin experiment is the underlying blockchain technology as a tool of distributed consensus, and attention is rapidly starting to shift to this other aspect of Bitcoin.”

The Bitcoin and Ethereum protocols have many superficial similarities since both are public distributed ledgers relying on ECDSA and PoW.[3] However, there are profound differences, too. Bitcoin is a protocol squarely aimed at supporting BTC as an alternative currency. Ethereum is a protocol designed for running a distributed computing platform capable of storing and executing arbitrary code on the Ethereum Virtual Machine (EVM). In addition to externally owned accounts (EOA), which operate just like Bitcoin accounts, Ethereum supports SCs, maintaining a ledger for external tokens and defining applications unrelated to financial instruments. For example, one can trivially implement an equivalent to BTC as a token contract on the Ethereum platform.

[1] Last year (2019), the Bitcoin protocol confirmed 100 million transactions. For doing so, it used more electricity than Denmark.

[2] Consensus as a service is an end-to-end process ensuring the overall consistency of operations run on distributed ledgers.

[3] Ethereum 2.0 is going to switch to a proof of stake (PoS) consensus algorithm.

2.3 Other blockchains

The Bitcoin and Ethereum protocols launched the blockchain revolution. By now, there are thousands of blockchain utilizing a wide variety of consensus mechanisms, from the classical PoW, to the proof-of-stake (PoS), to the practical Byzantine- fault-tolerant consensus. Some of these blockchains are public and open to all and sundry to join; others are private and can be joint only by preapproved entities.

What is important to us is that many of the blockchains can support smart contracts so that they are CaaS providers carrying all kinds of digital assets representing both tangible and intangible real-world assets. Accordingly, two exciting and practically relevant problems arise: (a) how to exchange assets defined on the same blockchain; (b) how to move assets from one blockchain to the next. We explicitly formulate and discuss these problems in detail in Sections 3, 4.

3 Challenges in Blockchain Intraoperability

We define blockchain intraoperability as exchanging different assets defined on the same blockchain supporting smart contracts. Swapping of stable coins, described below, is particularly important. Exchanging Ethereum-based stable coins, such as True USD (TUSD) and USD Coin (USDC), is a typical example of intraoperability in action. Exchanging Tether (USDT), a Bitcoin-based stable coin, for USD Coin (USDC), an Ethereum-based stable coin, is an example of blockchain interoperability. Interoperability forms the backbone of the burgeoning field of decentralized finance (DeFi). We implement blockchain intraoperability by using Automatic Market Makers (AMMs).

For illustrative purposes, we show prices of USDT and TUSD in Figure 4.

3.1 Stablecoins

Despite occasional claims to the contrary, conventional cryptocurrencies, such as BTC and ETH, are ill-suited to commerce's needs. There are several reasons why it is the case, the most apparent being that cryptocurrencies have colossal volatility. Hence, considerable efforts are directed towards building the so-called stablecoins, which live on blockchains, but have low volatility. The ultimate culmination of these efforts would be introducing central bank digital currencies (CBDCs), which are digital representations of the corresponding fiat.

The simplest way of building a stablecoin is to use the Ethereum protocol as a CaaS provider, designing such a coin as a token pegged to an asset or a basket of assets, viewed as stable in a conventional economic sense.

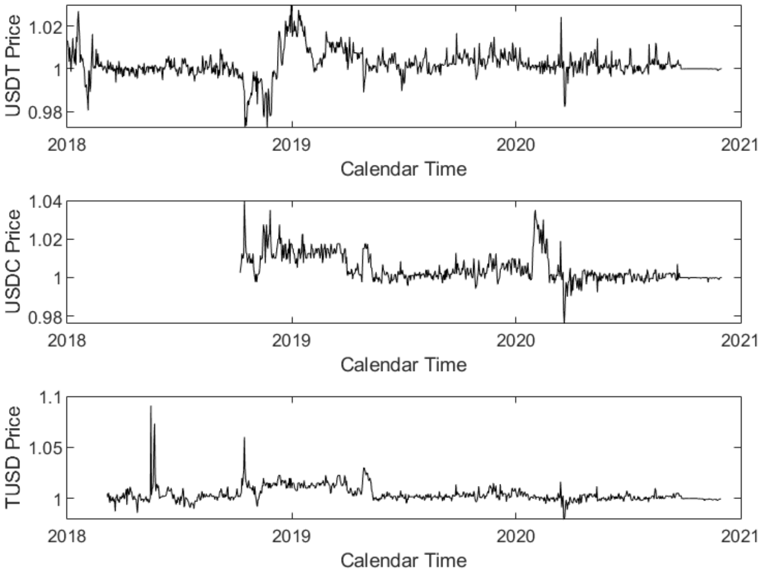


Fig. 4: Prices of USDT, USDC, and TUSD since inception. Their market capitalizations are 19, 3, and 0.3 billion USD, respectively. Source: https://coinmarketcap.com/

Because of its technical capabilities, the Ethereum protocol is often viewed as a natural tool for building stablecoins. In contrast, the Bitcoin protocol is considered ill-suited for anything other than recording BTC transactions. In reality, the biggest by far of all stablecoins is Tether, a Bitcoin Omni Layer token. Several other blockchains, such as Stellar or Algorand, can be used to support stablecoins. In many respects, they are better than Ethereum, simply because using Ethereum might be prohibitively expensive.

The development of robust and trustworthy stablecoins is of paramount importance for the future real-life applications of DLT. Here we mention that there are four viable types of stablecoins:

- fully collateralized by individual fiat currencies;
- fully collateralized by baskets of fiat currencies;
- overcollateralized with native token;
- asset-backed.

Other possibilities considered in the literature are partially collateralized and algorithmically stabilized stable coins. Such coins are not viable and are not worth discussing.

Although the prices of stable coins, even fully fiat-backed, deviate from their equilibrium values due to the varying supply-demand and other considerations, these deviations are kept relatively small by arbitrageurs.

3.2 Automated market makers

Let us design an SC, capable of making markets between two tokens TN_1, TN_2, whose relative price, i.e., the price of the second token expressed in terms of the first one, is denoted by P. We call such a contract an automated market maker (AMM). AMMs gained popularity over the last couple of years. Initially, anyone can become a market maker by delivering TN_1 and TN_2 simultaneously in the right proportion to the collateral pool. Subsequently, anyone can remove one of the tokens from the pool by simultaneously providing the other token to the pool according to the rule defining the underlying SC. The best use case for AMMs is swapping stablecoins. However, exchanging other tokens against each other, for instance, a stablecoin against ETH, is also possible.

The actual exchange rate relies on rules, which have to be agreed upon in advance. We consider three possible choices: the constant sum, constant product, and mixture rules. Several sources cover AMMs; see, e.g., [1,7,23, 28,34].

We assume that the initial prices of TN_1, TN_2 tokens are equal to each other and consider an automated market maker defined by the constant sum rule:

$$X + Y = \Sigma_0, \quad X_0 = Y_0 = N, \quad \Sigma_0 = 2N. \tag{1}$$

Here X, Y are the numbers of TN_1, TN_2 in the pool. Eq. (1) yields:

$$Y = \Sigma_0 - X, \quad \left|\frac{dY}{dX}\right| = 1. \tag{2}$$

According to Eq. (2), the pool becomes exhausted when $X = \Sigma_0$, since increasing X from N to $2N$ is a rational thing for an arbitrageur to do when TN_2 becomes more expensive than TN_1. The marginal price of TN_2 expressed in terms of TN_1, is given by the second equation (2). This price is constant and equal to one. A constant sum AMM makes sense when TN_1, TN_2 are stablecoins, with prices weakly fluctuating around their equilibrium values. If transaction fees are zero, it is rational to exhaust the pool even if the deviation from equilibrium is minuscule. However, in the more realistic situation when transaction fees are nonzero, the corresponding deviation has to be above a threshold to make arbitrage profitable.

The constant product rule defines more interesting (and practically important) AMMs:

$$XY = \Pi_0, \quad X_0 = Y_0 = N, \quad \Pi_0 = N^2. \tag{3}$$

It is clear that

$$Y = \frac{\mathit{\Pi}_0}{X}, \quad \left|\frac{dY}{dX}\right| = \frac{\mathit{\Pi}_0}{X^2}. \tag{4}$$

Thus, an arbitrageur cannot exhaust such a pool, so that it shall exist indefinitely. It is clear that the price of TN_2 expressed in terms of TN_1 is no longer constant and increases (decreases) when X decreases (increases).

If necessary, we can generalize the constant sum, and constant product rules. The constant sum and constant product rules, given by Eqs (1), (3), can be written as follows:

$$\begin{aligned} &\left(\tfrac{\mathit{\Sigma}}{\mathit{\Sigma}_0} - 1\right) = 0, \quad X_0 = Y_0 = N, \quad \mathit{\Sigma}_0 = 2N, \\ &\left(\tfrac{\mathit{\Pi}_0}{\mathit{\Pi}} - 1\right) = 0, \quad X_0 = Y_0 = N, \quad \mathit{\Pi}_0 = N^2. \end{aligned} \tag{5}$$

where $\mathit{\Sigma} = X + Y$, $\mathit{\Pi} = XY$ are the current sum and product, respectively. These rules can be combines as follows:

$$\begin{aligned} &\left(\tfrac{\mathit{\Pi}_0}{\mathit{\Pi}} - 1\right) \cancel{+} \alpha \left(\tfrac{\mathit{\Sigma}}{\mathit{\Sigma}_0} - 1\right) = 0, \\ &X_0 = Y_0 = N, \quad \mathit{\Sigma}_0 = 2N, \quad \mathit{\Pi}_0 = N^2. \end{aligned} \tag{6}$$

where $\alpha > 0$ is an adaptive parameter that characterizes the transition from the constant product to the constant sum rule. The product $\mathit{\Pi}$ appears in the denominator to avoid the possibility of exhausting the entire pool and ensuring that:

$$Y(X) \underset{X \to 0}{\to} \infty, \qquad X(Y) \underset{Y \to 0}{\to} \infty. \tag{7}$$

Of course, by opening herself to arbitrageurs' actions, an AMM will suffer a loss caused by a drop in the collateral value below its buy-and-hold level. In the language of mathematical finance, an AMM, who can be viewed as an option seller, suffers from negative convexity. To compensate for this loss, AMMs have to charge transaction fees, see the next section. The AMM's loss is called impermanent because, under the mean-revertion assumption, it tends to disappear. However, the mean-reversion assumption might or might not hold in real life. Introducing x, y, such that $X = Nx$, $Y = Ny$, we rewrite Eqs (1) – (2) as follows:

$$x + y = 2, \quad x_0 = y_0 = 1, \tag{8}$$

$$y(x) = 2 - x, \quad \left|\frac{dy}{dx}\right| = 1. \tag{9}$$

In terms of x, y, the constant product rule given by Eqs (3) – (4) can be written as follows:

$$xy = 1, \quad x_0 = y_0 = 1, \tag{10}$$

$$y(x) = \frac{1}{x}, \quad \left|\frac{dy}{dx}\right| = \frac{1}{x^2}. \tag{11}$$

Finally, Eqs (6) written in terms of x, y become:

$$\left(\frac{1}{xy} - 1\right) \cancel{+} \alpha \left(\frac{x+y}{2} - 1\right) = 0, x_0 = y_0 = 1. \tag{12}$$

A simple algebra yields:

$$\begin{aligned} y_\alpha &= \tfrac{1}{2\alpha}\left(-\left(2\left(1-\alpha\right)+\alpha x\right)+\sqrt{\left(2\left(1-\alpha\right)+\alpha x\right)^2+\tfrac{8\alpha}{x}}\right), \\ \tfrac{dy_\alpha}{dx} &= \tfrac{1}{2}\left(-1+\tfrac{2(1-\alpha)+\alpha x-4/x^2}{\sqrt{(2(1-\alpha)+\alpha x)^2+8\alpha/x}}\right). \end{aligned} \tag{13}$$

For brevity, below we suppress N, the initial number of tokens delivered to the pool.

Assume that P moves away from its equilibrium value $P_0 = 1$. Let $P > 1$. For the constant sum contract, an arbitrageur can choose a number x, $1 < x \leq 2$, and deliver $(x-1)$ of TN_1 tokens to the pool in exchange for getting $(x-1)$ of TN_2 tokens. Her profit or loss is given by

$$\mathit{\Omega}\left(x\right) = \left(P-1\right)\left(x-1\right). \tag{14}$$

Since $\mathit{\Omega}$ is a linear function of x, it is rational to exhaust the entire pool, by choosing the following optimal values $(x^*, y^*, \mathit{\Omega}^*)$:

$$x^* = 2, \quad y^* = 0, \quad \mathit{\Omega}^* = \left(P-1\right). \tag{15}$$

Similarly, when $P < 1$:

$$x^* = 0, \quad y^* = 2, \quad \mathit{\Omega}^* = \left(1-P\right). \tag{16}$$

The arbitraged portfolio's value is $\pi^*\left(P\right)$, where

$$\pi^*\left(P\right) = \begin{cases} 2, & P \geq 1, \\ 2P, & P < 1. \end{cases}, \tag{17}$$

while the buy and hold portfolio's value is $(P+1)$. The difference ω has the form

$$\omega = \left(P+1\right) - \pi^*\left(P\right). \tag{18}$$

In DeFi, ω is called the impermanent loss. This terminology is misleading because the loss can quickly become permanent when P drifts away from its "equilibrium" value of one. The percentage loss of the realized portfolio compared to the buy and hold portfolio has the form:

$$\lambda = 1 - \frac{|P-1|}{P+1}. \tag{19}$$

We can repeat the above arguments for the constant product contract. Assuming that P deviates from one, an arbitrageur can choose a number $x > 1$ and deliver $(x-1)$ TN_1 tokens to the pool and take $(1-y)$ TN_2

tokens from the pool, where $y = 1/x$. In this case, her profit or loss can be written as follows:

$$\Omega(x) = \left(P\left(1 - \frac{1}{x}\right) - (x - 1)\right). \tag{20}$$

The optimality condition has the form

$$\Omega'(x) = \left(\frac{P}{x^2} - 1\right) = 0, \tag{21}$$

so that the corresponding optimal values (x^*, y^*, Ω^*) are

$$x^* = \sqrt{P}, \quad y^* = \frac{1}{\sqrt{P}}, \quad \Omega^* = \left(\sqrt{P} - 1\right)^2. \tag{22}$$

Thus, a constant product collateral pool can never be exhausted. At every stage, the optimal amounts of TN_1 and TN_2 held in the portfolio are equal to $\sqrt{P}$, each. Since the value of both tokens in the portfolio has to be equal, the implied optimal value of TN_2 expressed in terms of TN_1 is $P^* = x^*/y^* = P$. The arbitraged portfolio's value is $\pi^* = 2\sqrt{P}$, while the buy-and-hold portfolio's value is $(P + 1)$. The difference ω is given by

$$\omega = (P + 1) - 2\sqrt{P}. \tag{23}$$

The corresponding percentage loss is

$$\lambda = 1 - \frac{2\sqrt{P}}{(P + 1)} = \frac{\left(\sqrt{P} - 1\right)^2}{(P + 1)}. \tag{24}$$

For the mixed rule AMM, the arbitrageur's profit for $P > 1$ has the form:

$$\Omega(x) = (P(1 - y_\alpha(x)) - (x - 1)), \tag{25}$$

with the optimum achieved at $x_\alpha^*, y_\alpha^*, \Omega_\alpha^*$ of the form:

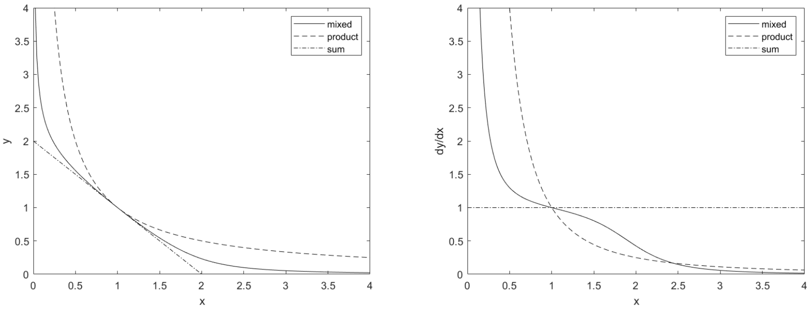
$$y'_\alpha(x_\alpha^*) = -\frac{1}{P}, \quad y_\alpha^* = y_\alpha(x_\alpha^*), \quad \Omega_\alpha^* = (P(1 - y_\alpha^*) - (x_\alpha^* - 1)). \tag{26}$$

We find the optimal x_α^* via the Newton-Raphson method starting with a suitable $x_\alpha^{(0)}$:

$$x_\alpha^{(n+1)} = x_\alpha^{(n)} - \frac{y'_\alpha\left(x_\alpha^{(n)}\right) + \frac{1}{P}}{y''_\alpha\left(x_\alpha^{(n)}\right)}. \tag{27}$$

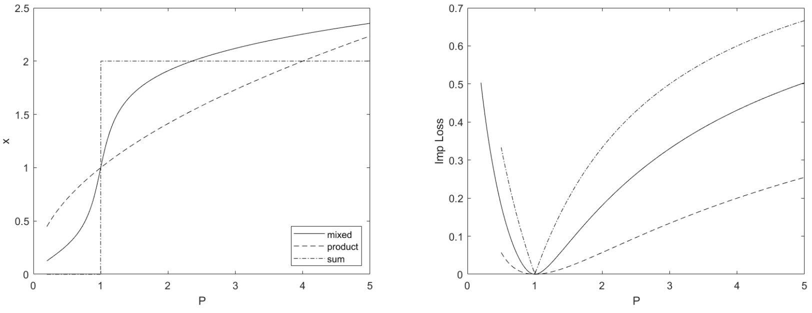
Since the Newton-Raphson method has quadratic convergence, ten iterations provide machine accuracy, so that we can set $x_\alpha^* = x_\alpha^{(10)}$. The value of the arbitraged portfolio is

$$\pi^* = x_\alpha^* + P y_\alpha(x_\alpha^*). \tag{28}$$



(a) (b)

Fig. 5: (a) y as a function of x for three different types for constant sum, constant product, mixed rule AMMs. (b) The relative price of TN_2 expressed in terms of TN_1 equal to $|dy/dx|$ for constant sum, constant product, and mixed rule AMMs. For the mixed rule, $\alpha = 10$.



(a) (b)

Fig. 6: (a) x as a function of P for three different types of AMM: constant sum, constant product, mixed rule; (b) Impermanent loss as a function of P: constant sum, constant product, mixed rule. For the mixed rule, $\alpha = 10$.

The constant sum, constant product, and mixed rule curves, as well as relative prices of TN_2, expressed in terms of TN_1, and impermanent losses are presented in Figures 5, 6. These figures demonstrate that a market maker experiences a loss whenever the tokens' relative price deviates from its equilibrium value. The impermanent loss is relatively small for the constant product rule, moderate for the mixed rule, and very high for the constant sum rule. Even when the price P deviates by a factor of five from its equilibrium value, the impermanent loss for the constant product rule is tolerable, especially compared to the mixed rule.

3.3 P&L modelling for AMMs

Of course, the objective of any market maker is profit. This profit comes from transaction fees charged by the pool, which have to exceed the impermanent loss caused by a drop in the collateral value below its buy-and-hold level. In this section, we model the profit and loss (P&L) distribution of an AMM. We consider an AMM charging proportional transaction fees every time an arbitrageur or a generic market participant removes tokens of one kind and adds tokens of the other kind. These fees have to cover the impermanent loss and then some. An arbitrageur has to add more tokens to the pool than is required by its constituent rule to cover transaction fees. Consider an AMM defined by the constant product rule. Let ε be a percentage fee. Denote by T_0 and T_1 two time slices. At time T_0 the price is P_0, the pool's composition is (x_0, y_0), and the product value is $\pi_0 = x_0 y_0$. At time T_1 the price is P_1 and the arbitraged pool's composition (x_1, y_1) has to be determined. First, we assume that $P_1 > P_0$, so that, with zero transaction costs, an arbitrageurs would withdraw a certain amount of TN_2 and add the corresponding amount of TN_1. With non-zero transaction costs her decision is more complicated. She can choose a number $x_1 > x_0$ and deliver $(1+\varepsilon)(x_1 - x_0)$ of TN_1 tokens to the pool in exchange for getting $(y_0 - y_1)$ of TN_2 tokens from the pool, where $y_1 = \pi_0 / x_1$. The profit or loss is as follows:

$$\Omega(x) = \left(P_1 \left(y_0 - \frac{\pi_0}{x} \right) - (1+\varepsilon)(x - x_0) \right). \tag{29}$$

The profit is maximized when

$$\Omega'(x) = \left(\frac{P_1 \pi_0}{x^2} - (1+\varepsilon) \right) = 0, \tag{30}$$

so that

$$\begin{gathered} x_1^* = \max \left(\sqrt{\tfrac{P_1 \pi_0}{(1+\varepsilon)}}, x_0 \right), \quad y_1^* = \tfrac{\pi_0}{x_1^*}, \\ \pi_1^* = \tfrac{((1+\varepsilon)(x_1^* - x_0) + x_0)\pi_0}{x_1^*}. \end{gathered} \tag{31}$$

We have to take the maximum in Eq. (31) to ensure that $x_1^* \geq x_0$, so that TN_1 tokens are added to the pool, rather than withdrawn from it. Thus, for $P_1 > P_0$, the adjustment occurs only when

$$P_1 > \frac{(1+\varepsilon) x_0}{y_0}. \tag{32}$$

Similarly, for $P_1 < P_0$,

$$\begin{gathered} y_1^* = \max \left(\sqrt{\tfrac{\pi_0}{(1+\varepsilon) P_1}}, y_0 \right), \quad x_1^* = \tfrac{\pi_0}{y_1^*}, \\ \pi_1^* = \tfrac{((1+\varepsilon)(y_1^* - y_0) + y_0)\pi_0}{y_1^*}, \end{gathered} \tag{33}$$

so that adjustment happens when

$$P_1 < \frac{x_0}{(1+\varepsilon)\,y_0}. \tag{34}$$

Finally, for

$$\frac{x_0}{(1+\varepsilon)\,y_0} \leq P_1 \leq \frac{(1+\varepsilon)\,x_0}{y_0}, \tag{35}$$

it is suboptimal to adjust the composition of the pool, so that $(x_1^*, y_1^*) = (x_0, y_0)$, $\pi_1^* = \pi_0$.

Eqs (32), (34), (35) show that in the presence of non-zero transaction costs the actual composition of the pool is not only time-dependent, as expected, but, more surprisingly, path-dependent.

Let us study the profitability of a constant product rule AMM. We shall assume that the relative price $P(t)$ is mean-reverting and is driven by an Ornstein-Uhlenbeck process:

$$\begin{aligned} P(t) &= \exp(p(t)), \\ dp(t) &= -\kappa p(t)\,dt + \sigma dW(t), \quad p(0) = 0. \end{aligned} \tag{36}$$

Here $W(t)$ is the Wiener process driving random variations of the log-price, κ is mean-reversion rate, and σ is volatility; κ and σ are measured in the units of $[1/T]$ and $\left[1/\sqrt{T}\right]$, respectively. It is helpful to switch to non-dimensional units. To this end, we introduce $\bar{t} = \kappa t$, $\bar{\sigma} = \sigma/\sqrt{\kappa}$, and rewrite Eq. (36) as follows:

$$\begin{aligned} P(\bar{t}) &= \exp(p(\bar{t})), \\ dp(\bar{t}) &= -p(\bar{t})\,d\bar{t} + \bar{\sigma} dW(\bar{t}), \quad p(0) = 0. \end{aligned} \tag{37}$$

Below we omit overbars for brevity. When σ is small, the log-price is almost deterministic and mean-reverting, when σ is large, it is strongly stochastic. Since small price changes do not result in adjustments of portfolio composition, see Eq. (35), we discretize Eq. (37) with a time step $\varDelta t$, for instance, one day, and rewrite it as follows:

$$\begin{aligned} P_{l+1} &= \exp(p_{l+1}), \\ p_{l+1} &= (1-\varDelta t)\,p_l + \sigma\sqrt{\varDelta t}\eta_l, \quad p_0 = 0. \end{aligned} \tag{38}$$

Here η_l is the standard normal random variable.

We assume that pool's liquidity provider has a time horizon $T = L\varDelta t$ and model the evolution of the system as a whole for L steps. The corresponding dynamics can be described as follows:

$$\begin{aligned} p_{l+1} &= (1-\varDelta t)\,p_l + \sigma\sqrt{\varDelta t}\eta_l, & p_0 &= 0, \\ x_{l+1} &= f(\pi_l, p_{l+1}, p_l, x_l, y_l), & x_0 &= 1, \\ y_{l+1} &= g(\pi_l, p_{l+1}, p_l, x_l, y_l), & y_0 &= 1, \\ \pi_{l+1} &= h(\pi_l, p_{l+1}, p_l, x_l, y_l), & \pi_0 &= 1, \\ P_{l+1} &= \exp(p_{l+1}), & P_0 &= 1 \end{aligned} \tag{39}$$

Here

$$x_{l+1} = \begin{cases} \sqrt{\frac{P_{l+1}\pi_l}{(1+\varepsilon)}}, & \frac{(1+\varepsilon)x_l}{y_l} < P_{l+1}, \\ x_l, & \frac{x_l}{(1+\varepsilon)y_l} \leq P_{l+1} \leq \frac{(1+\varepsilon)x_l}{y_l}, \\ \frac{\pi_l}{y_{l+1}}, & P_{l+1} < \frac{x_l}{(1+\varepsilon)y_l}. \end{cases}$$

$$y_{l+1} = \begin{cases} \frac{\pi_l}{x_{l+1}}, & \frac{(1+\varepsilon)x_l}{y_l} < P_{l+1}, \\ y_l, & \frac{x_l}{(1+\varepsilon)y_l} \leq P_{l+1} \leq \frac{(1+\varepsilon)x_l}{y_l}, \\ \sqrt{\frac{\pi_l}{(1+\varepsilon)P_{l+1}}}, & P_{l+1} < \frac{x_l}{(1+\varepsilon)y_l}. \end{cases} \tag{40}$$

$$\pi_{l+1} = \begin{cases} \frac{((1+\varepsilon)(x_{l+1}-x_l)+x_l)\pi_l}{x_{l+1}}, & \frac{(1+\varepsilon)x_l}{y_l} < P_{l+1}, \\ \pi_l, & \frac{x_l}{(1+\varepsilon)y_l} \leq P_{l+1} \leq \frac{(1+\varepsilon)x_l}{y_l}, \\ \frac{((1+\varepsilon)(y_{l+1}-y_l)+y_l)\pi_l}{y_{l+1}}, & P_{l+1} < \frac{x_l}{(1+\varepsilon)y_l}. \end{cases}$$

The corresponding $P\&L$ versus the buy-and-hold strategy is given by the following formula:

$$P\&L = (x_L + P_L y_L) - (1 + P_L). \tag{41}$$

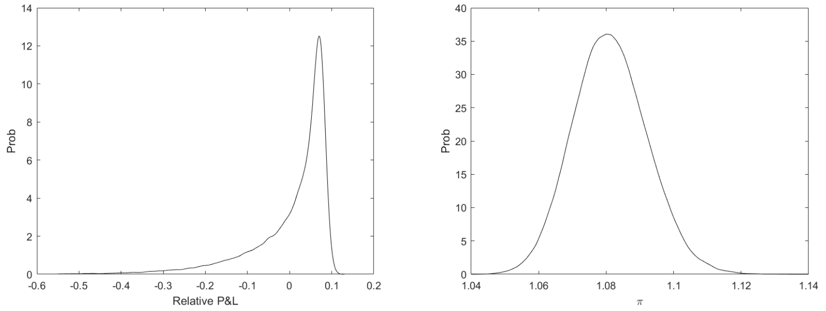
The market making activity makes sense only when $P\&L > 0$. Perhaps, a more informative quantity is the relative $\overline{P\&L}$, which shows the percentage return on market making, compared with the buy-and-hold strategy:

$$\overline{P\&L} = \frac{P\&L}{(1+P_L)} = \frac{(x_L + P_L y_L)}{(1+P_L)} - 1. \tag{42}$$

Of course, since the log-price is stochastic, we can only analyze $\overline{P\&L}$ in the probabilistic sense by running Monte Carlo simulations. To this end, we consider M MC paths, which are characterized by a random matrix (η_{ml}), calculate a set of M P&L values, $\left\{\overline{P\&L}_m\right\}$, and study its statistical properties. Figure 7 presents the corresponding results.

It is clear that for the choice of parameters used in Figure 7 automated liquidity provision is profitable on average. This profitability comes from the fact that the AMM accumulates more tokens at the end of the process than she had in the beginning. This figure illustrates the complicated dependence of $\overline{P\&L}$ on L, σ, ε. The reader, wishing to become an AMM, needs to explore this dependence in detail. One can analyze the P&L generated by the mixed rule AMM by using a similar technique. We leave the corresponding analysis to the reader.

We emphasize that it is exceedingly hard to protect AMMs against meta-threats, such as the underlying SC not being robust enough or the underlying token losing value permanently because of regulatory pressures, poor design, or outright theft of the collateral. Even if none of the above happens, the portfolio consisting of tokens TN_1, TN_2 can lose value vis-a-vis fiat because both tokens depreciate against fiat currency simultaneously. Engaging



(a) (b)

Fig. 7: (a) The pdf for P&Loss; (b) The pdf for $\pi_L = x_L y_L$. The corresponding parameters are as follows: $L = 200$, $\varDelta t = 0.0025$, $\sigma = 1$, $\epsilon = 0.05$.

in liquidity provision is advantageous to someone who is already invested in tokens and wants to stay invested in them in the long run. Another attractive opportunity is to provide liquidity to the stablecoin universe and view AMM gains (if any) as a way to earn interest on one's investments.

4 Challenges in Blockchain Interoperability

The Internet architecture lends itself to scalability because it (i) permits each AS to employ its own interior routing protocols, while standardizing the interfaces between autonomous systems to permit data-packet flow across these systems; and (ii) places the higher layer semantics (content or meaning) of a connection to the edges of the network, where the sender and receiver are located. The intervening autonomous system between the sender and receiver are oblivious to the content of the message being sent. Messages are in fact broken down in to IP packets (datagrams [5]), and each IP packet may traverse differing routes across different sets of autonomous systems end-to-end.

In order for blockchain systems to have a high degree of interoperability, each blockchain system must be free to implement its interior ledger maintenance protocols (e.g. consensus protocol, ledger data structures, etc.), with standardized APIs defined for cross-blockchain transfers of the digital value-representation tokens (assets). Secondly, for digital assets that are blockchain-based the notion of "value" (economic value) must be separated from the blockchain infrastructures that manage the value-representation tokens. That is, if we view on-chain tokens as a counterpart of the IP datagrams on the Internet – where the economic value is discernible only to the sender and receiver at the edges – then a proper interoperable blockchain architecture must permit the tokens to traverse (hop across) multiple blockchain

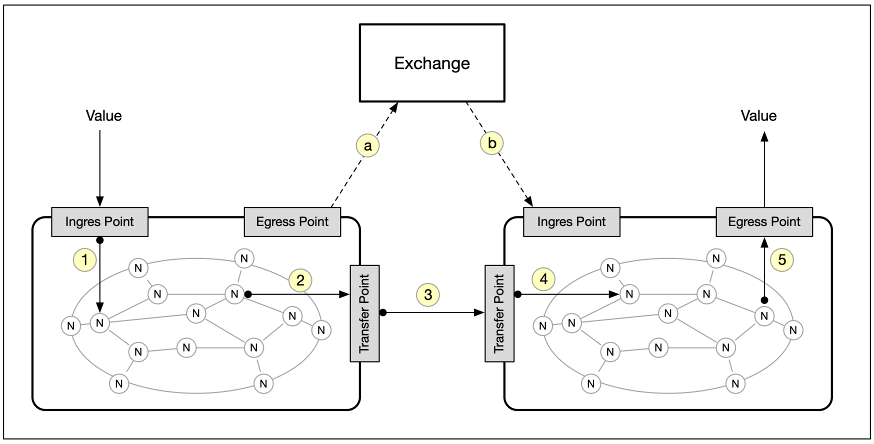


Fig. 8: Overview of interoperability architecture for blockchain networks

networks in an agnostic manner without loss to the economic value represented by the tokens [15].

A corollary of the last point above is the need to separate the economic value represented by the tokens from the operating costs of the blockchain infrastructure supporting the tokens. More specifically, the degree of user-demand for infrastructure services to transact with a token (i.e. move token/asset ownership from one public-key to another) should not be tied to the economic value of the token, as this renders the infrastructure to becoming too costly to employ. A case in point is Bitcoin [11,10] and Ethereum [29], both of which have become too costly to be used for small-amount transactions. Alternative incentives mechanisms for infrastructure (node) operators need to be developed based on this principle of the separation of token economic value from infrastructure operating costs. For example, a traditional subscriber-fee model akin to that used by Internet ISPs has been proposed in [16].

It is worth reflecting that if starting in the 1990s the ISPs charged consumers for Internet access based on the sheer number of IP datagrams routed through their subnets, the Internet would not be the success that it is today.

4.1 Interoperability Across Blockchains and DLT Systems

In considering the use of blockchain technology to represent economic value – through the on-chain value-representation tokens or ledger entries – it is useful to address the problem of interoperability at two levels [15]:

- *Interoperability at the value level*: In order to perform transactions using virtual assets there must be agreement on the notion of value and the standardization of mechanism to represent economic value as tokens within blockchain systems.

A core part of interoperability at the value level is the standardization of *asset profiles* (asset prospectus) definition. This permits transacting parties to refer to the same definition of the virtual asset to be exchanged. We discuss the notion of asset profiles in Section 4.3.

- *Interoperability at the mechanical protocol level*: Standard protocols must be developed to perform value conversions (ingress and egress points in Figure 8) and to perform token-transfers across blockchain networks (transfer points in Figure 8).

  The standard protocols used to carry out the token-transfers across blockchain networks must be agnostic to the economic value represented by the token. We discuss the technical protocol and its desirable properties in Section 4.4.

4.2 Interfaces for Interoperable Functions

An interoperable blockchain architecture must distinguish between *value-representation conversion points* in the blockchain network from *representation translation points* across blockchain networks (see Figure 8):

- *Value-representation conversion points*: These are the *ingress points* for value to enter the blockchain system, and the *egress points* where value departs from the blockchain system. When a blockchain system receives economic value from an external source (e.g. a non-DLT representation) via an ingress point, it creates a representation of that value in the form of the token data-structure defined in that blockchain system. This is shown as Step 1 in Figure 8. Similarly, when a DLT value-representation is to be removed from a blockchain, it departs the blockchain via an egress point. The ledger of the blockchain is marked to indicate that the token no longer exists (i.e. it has been invalidated).

  It is important to note that in some blockchain systems the tokens are an inherent part of the system and that “assets” (their token representation) never enters or leaves the blockchain (e.g. Bitcoin). This is because the value of the virtual asset (i.e. BTCs) not a derivative of an underlying asset [2].
- *Token transfer points*: These are the points in a blockchain network where tokens can be transferred directly from one blockchain to another without any change to the economic value represented by the token. This is shown as Step 3 in Figure 8. A data structure (format) translation of the token may occur if the two blockchain infrastructures employ differing interior ledger data-structures.

  A successful unidirectional token transfer means that the token data-structure in the origin blockchain (Step 2) is destroyed (or marked as being no longer valid), while a new token data-structure is created (added) in the destination blockchain (Step 4). No change to the economic value presented by the tokens must occur during such transfers.

In cases where circumstances prevent the traversal by a token across blockchain networks (e.g. incompatible ledger data structures, jurisdiction constraints, etc.) then both value-representation conversion and token-format translation must occur with the help of a mediating third party (e.g. crypto-exchanges or similar VASPs). This is illustrated in Step (a) and Step (b) of Figure 8. This two-stage process consist of the following. The value departs the origin blockchain in Step (a) through an *egress point* to the exchange entity. Here, the value-representation token in the origin blockchain system is extinguished or destroyed. Secondly, the exchange entity must inject that value at the ingress point (Step (b)) at the destination blockchain system – resulting in the creation of a new value-representation token according to the data-structure employed by the destination blockchain. In this case, the third-party exchange entity must be a participant in both blockchain systems, and it must have the means to perform the process (e.g. it holds sufficient fiat currencies).

4.3 Asset Profiles: Standardizing Asset Prospectus Documents

We define the *asset profile* as prospectus of a regulated asset that includes information and resources describing the virtual asset. This includes the asset name/code, issuing authority, denomination, jurisdiction, and the URLs and mechanisms to validate the information. An asset profile document or claim must be digitally signed by the issuer of the virtual asset. It is an assertion or statement regarding the true existence of the virtual asset within a given system (blockchain-based or otherwise), within a legal jurisdiction. The asset profile is independent from the specific instantiation of the asset (on a blockchain or otherwise) and independent from its instance-ownership information.

There are a number of information fields that could be expressed within an asset profile statement or document for a given virtual asset. Some examples include, but not limited to:

- *Issuer*: This information field pertains to the legal entity that issues or creates the virtual asset. The Issuer could be a single corporation, a community of entities, a government, etc.
- *Jurisdiction*: This information field refers to the legal jurisdiction where the virtual asset is defined and recognized.
- *Virtual asset code*: This information field contains globally unique alphanumeric value assigned to the asset. This ensures that users and systems can acurately refer to a virtual asset, without any errors or ambiguities.
- *Virtual asset type*: This refers to the underlying asset or collateral upon which the virtual asset is based. Examples include bankable assets (i.e. fiat currency), company shares (e.g. equity), tangible assets (e.g. real estate), etc.

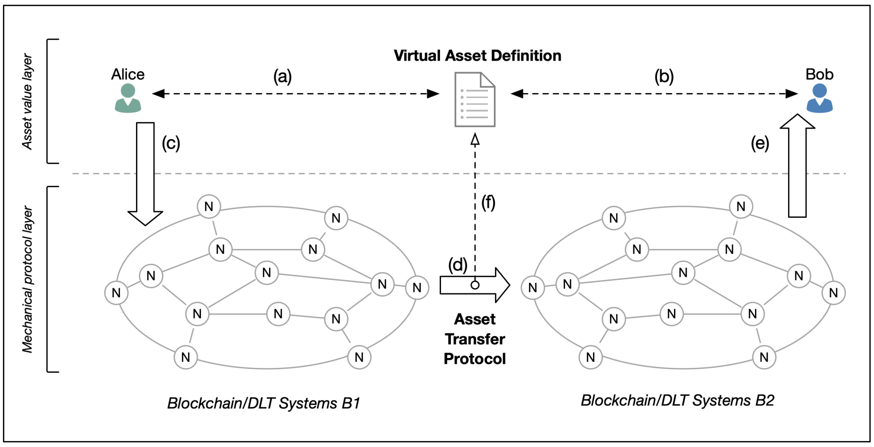


Fig. 9: Asset transfers based on a standard asset definition profile

Note that this information field in the asset profile could be "none", meaning that the value of the virtual asset is not derived from any underlying asset (as is the case with Bitcoin) [2].

- *Total supply*: This information field refers to total supply of the asset, which could be fixed, conditional (e.g. upon some external factors), or flexible (where the supply of the asset is managed flexibly by authorized parties).
- *Issuance date*: This information field refers to the date when the virtual asset become available. This date should be identical to the date of issuance and signing of the asset profile document (e.g. JSON file).
- *Validation endpoint*: This is the URI/URL where any entity can verify the validity status of the asset profile document (e.g. JSON file).
- *Digital signature of Issuer*: This is the digital signature of the Issuer over the entire asset profile document. Typically the digital signature field includes a copy of the public-key of the Issuer, and the signature is achieved using standard algorithms [31,3,22].

Figure 9 provides an overview of the two levels of interoperability and the need for standardization to occur at each level. Alice seeks to transfer ownership of a virtual asset to Bob who is located in a different blockchain. Both sides have agreed upon the definition of the asset to be transferred (Steps (a) and (b)). Alice then invokes the transfer protocol (e.g. smart contract, application, etc.) in Step (c), which results in the asset transfer protocol executing between the two blockchain systems B1 and B1 (Step (d)). The protocol itself must have the means to refer to the definition of the asset being transferred (e.g. reference (f) in Figure 9).

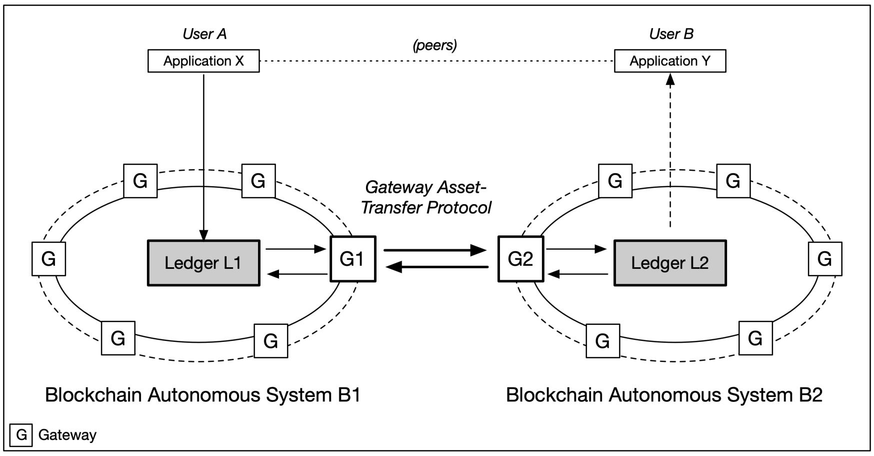

User A
Application X
(peers)
User B
Application Y
Gateway Asset-Transfer Protocol
G
Ledger L1
G1
G2
Ledger L2
Blockchain Autonomous System B1
Blockchain Autonomous System B2
G Gateway

Fig. 10: Overview of the gateway-to-gateway asset transfer protocol

4.4 The Gateway Model: A Framework for Blockchain Interoperability

In this section, we discuss blockchain interoperability following the notion of the *token transfer points* mentioned above, manifesting these transfer points in the form of *blockchain gateways* [15,14]. A gateway stands in" front" of its blockchain systems, and has read/write access to the ledger and other interior resources within its blockchain domain. A gateway in one blockchain interacts with another gateway in a different blockchain in the task of transferring virtual assets between them unidirectional.

Similar to a routing autonomous system being composed of one or more (possibly nested) routing domains, the framework of [15] views a blockchain domain as consisting of *interior nodes* and *gateway nodes*:

- *Interior nodes*: These are nodes and other entities whose main task is maintaining ledger information and conducting transactions within one blockchain domain.
  For certain blockchain configurations (e.g. private or permissioned) the interior nodes are prohibited from engaging external entities without authorization.
- *Gateway nodes*: These are nodes and other entities whose main task is dealing with with cross-blockchain asset transfers involving different blockchain systems.

Figure 10 provides a high level illustration of gateway nodes G within two blockchain domains (interior nodes are not shown). Although Figure 10 shows a small number of nodes G to be designated as inter-domain gateway nodes, ideally all nodes in a given blockchain system should have the capability (i.e. correct software, hardware, trusted computing base) to become gateway nodes. This allows dynamic groups (subsets) of the population of nodes to become *gateway groups* that act collectively on behalf of the blockchain system as a whole [17].

The following assumptions and principles underlie the interoperability framework of [15], and they correspond to the design principles of the Internet architecture:

- *Opaque blockchain-resources principle*: The interior resources of each blockchain system is assumed to be opaque to (hidden from) external entities. Any resources to be made accessible to an external entity must be made explicitly accessible by a gateway node with proper authorization.
  The opaque resources principle permits the interoperability architecture to be applied in cases where one (or both) blockchain systems are permissioned (private). It is the analog of the autonomous systems principle in IP networking [6], where interior routes in local subnets are not visible to other external autonomous systems.
- *Externalization of value principle*: The gateway-to-gateway protocol must be agnostic (oblivious) to the economic or monetary value of the virtual asset being transferred.
  The value-externalization principle permits asset transfer protocols to be designed for efficiency, speed and reliability – independent of the changes in the perceived economic value of the virtual asset. It is the analog of the end-to-end principle in the Internet architecture [27], where contextual information (economic value) is placed at the endpoints of the transaction. In the case of virtual asset transfers, the originator and beneficiary at the respective blockchain systems are assumed to have a common agreement regarding the economic value of the asset.

The goal of a blockchain interoperability framework based on interoperable gateways is to permit two (2) gateway nodes belonging to distinct blockchain systems to conduct a virtual asset transfer between them, in a secure and non-repudiable manner while ensuring the asset does not exist simultaneously on both blockchains (double-spend problem). The notion of a gateway is used because we recognize that blockchain technology is evolving and that in many cases the interior technical constructs in these blockchains maybe incompatible with one another. The architecture therefore assumes that certain types of nodes (gateway nodes) will be equipped with an asset transfer protocol and other relevant resources that permits greater interoperability across these incompatible blockchain systems. In a sense, a gateway "hides" the complexity of its blockchain, and in turn exposes standard APIs to other gateways in order to interoperate.

The resources within a blockchain system (e.g. ledgers, public-keys, consensus protocols, etc.) are assumed to be opaque to external entities in order to permit a resilient and scalable protocol design that is not dependent on the interior constructs of particular blockchain systems. This ensures that the virtual asset transfer protocol between gateways is not conditioned or dependent on these local technical constructs. The role of a gateway therefore is also to mask (hide) the complexity of the interior constructs of the blockchain system that it represents. Overall this approach ensures that a given blockchain system operates as a true autonomous system.

It is important to note that the opaque resources (ledgers) principle has implications on smart contract cross-chain conditionals, such as cross-chain hash-locks [26] and time-locks. Many proposals for cross-chain "atomic swaps" are designed with the underlying assumption that the ledgers on both sides are public or permissionless (e.g. see [8,33,20]). This means that both Alice and Bob are able to read (and invoke) each other's hash-lock smart contract at their respective blockchains. However, we believe that this underlying assumption is unrealistic given the fact that many blockchain systems today are private (permissioned).

The point of the opaque resources principle is to enable the design of cross-blockchain asset transfer protocols under the strictest condition – namely that both blockchains are private and their ledgers and smart-contracts inaccessible to each other. Interaction between them are possible only through their respective gateways. If an asset transfer protocol works for two private blockchains via gateways, then it should also work for cases where one or both of the blockchains are public or permissionless.

4.5 Protocols for Asset Transfers: Desirable Properties

At the mechanical protocol level, there are a number of desirable properties of an asset transfer protocol across blockchain/DLT systems [14]:

- *Atomicity*: An asset transfer must either commit or entirely fail (where failure means there are no changes to asset ownership in the origin blockchain).
- *Consistency*: An asset transfer (commit or fail) must always leave both blockchain systems in a consistent state, where the asset in question is located in one blockchain only. A protocol failure must not result in a "double-existence" of the asset (leading to double-spend in the two blockchain systems respectively).
- *Isolation*: While the asset transfer is occurring, the asset ownership cannot be modified. That is, some kind of temporary disablement of the asset at the origin blockchain must be used (e.g. locking on the ledger, escrow to a gateway, etc). This is to prevent the owner (who requested the cross-chain transfer) from double-spending the asset locally while the transfer protocol is running.
- *Durability*: Once an asset transfer has been committed on both the origin blockchain and destination blockchain, this commitment must remain true regardless of gateway crashes or blockchain unavailability (e.g. blockchain slowdown as in the case of CryptoKitties in Ethereum [4]).

It is crucial to note that these properties must hold true regardless of whether one or both of the blockchain systems are private (permissiond) or public (permissionless).

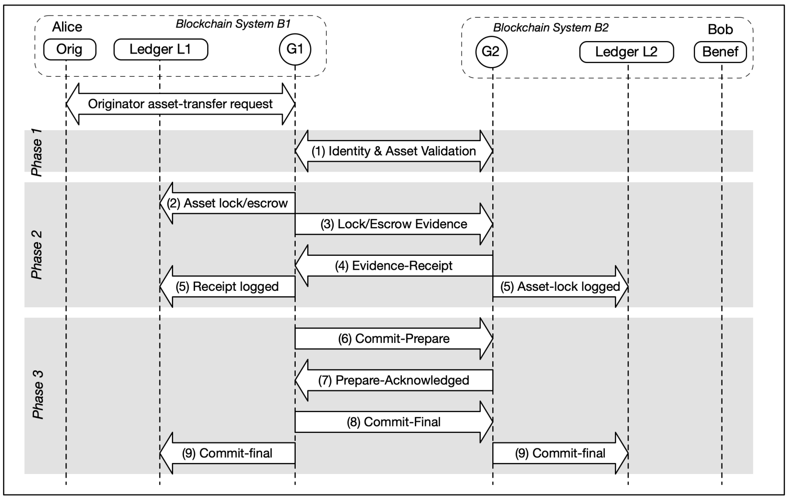


Fig. 11: Overview of the phases of the gateway asset-transfer protocol (after [14])

4.6 Phases of Gateway-to-Gateway Asset Transfers

An asset transfer protocol between two blockchain systems is carried out by two (2) gateway nodes that represent the two respective blockchain systems. A successful transfer results in the asset being extinguished or marked on the origin ledger by the origin-gateway, and for the asset to be introduced by the destination-gateway into the destination ledger. The mechanism to extinguish or introduce an asset from/into a ledger is dependent on the specific blockchain system.

The interaction between the two gateways is summarized in Figure 11, where the origin blockchain is B1 and the destination blockchain is B2. The gateways are denoted as G1 and G2 respectively.

The gateway nodes must implement one (or more) transactional commitment protocols (in Phase 2) that permit the coordination between two gateways, and the final commitment of the asset transfer. The choice of the commitment protocol (type/version) and the corresponding commitment evidence must be negotiated between the gateways during Phase 1. For example, the gateways G1 and G2 may implement the classic 2-Phase Commit (2PC) protocol [12,30] or other variants (e.g. 3PC) as a means to ensure efficient and non-disputable commitments to the asset transfer.

4.6.1 Phase 1: Pre-transfer Asset and Identities Verification In this phase the gateways G1 and G2 initiate a connection to each other in order to perform a number of validation functions. Some of these are as follows:

- Exchange of parameters for secure channel establishment between G1 and G2.

- Delivery of asset-profile information and asset-holder information, including originator and beneficiary identities and public keys (per the Travel Rule [9]), and the gateway owner (VASP) identities and public keys
- Exchange of parameters related to the blockchain systems B1 and B2, the commitment mechanism to be employed between G1 and G2, and the form of the asset-lock evidence to be delivered by G1.

4.6.2 Phase 2: Evidence of asset locking or escrow In this phase, gateway G1 must provide gateway G2 with sufficient evidence that the asset on blockchain B1 is in a locked state (or escrowed) under the control of G1 on ledger L1, and safe from double-spend on the part of its current owner (the originator).

The precise form of the evidence is dependent on the blockchain system in B1, and must be previously agreed upon in Phase 1. The purpose of this evidence is for dispute resolution between G1 and G2 (i.e. entities who own and operate G1 and G2 respectively) in the case that double-spend is later detected.

The gateway G2 must return a signed receipt to G1 of this evidence in order to cover G1 in the case of later denial by G2.

4.6.3 Phase 3: Final commitment of transfer In this phase gateway G1 indicates to G2 its readiness to finally commit to the transfer, and vice versa. Both messages must be signed by G1 and G2 respectively in case of later (post-transfer) disputes.

Gateway G1 marks the ledger L1 that the virtual asset is no longer associated with the public-key of previous owner (originator) and that the asset instance no longer exists on the blockchain system B1. Similarly, gateway G1 marks the ledger L2 in blockchain system B2 to indicate that henceforth the asset is associated with the public-key of the new owner (beneficiary).

Optionally, both G1 and G2 may exchange the local ledger marking information (e.g. block number and transaction number) with each other. This information may aid in future search, audit and accountability purposes from a legal perspective.

4.7 Open Challenges in Interoperability

There are a number of open issues that are related to the asset transfer protocol between gateway nodes. Some of the issues are due to the fact that blockchain technology is relatively new, and that technical constructs designed for interoperability have yet to be addressed. Some of the issues are due to the nascency of the virtual asset industry and lack of conventions, and therefore require industry collaboration to determine these.

- *Global identification of blockchain systems and public-keys*: There is currently no standard nomenclature to identify blockchain systems in a globally unique manner. The analog to this is the AS-numbers associated with IP routing autonomous systems. Furthermore, an address (public-key) may not be unique to one blockchain system. An entity (e.g. user) may in fact employ the same public-key at multiple distinct blockchain systems simultaneously.

– *Standard APIs for Cross Blockchain Transfers* As mentioned previously, standard protocols are needed to perform token-transfers across blockchain networks in a manner that is agnostic to the economic value represented by the token.
  Efforts are underway to begin defining the APIs, messages and payloads for asset transfers across blockchain systems [14,18].
– *Commitment protocols and forms of commitment evidence*: Commitment protocols for asset transfers across blockchain systems should be standardized based on existing well-deployed transaction systems (e.g. based on 2-Phase Commit [12,30]).
  The forms of commitment evidence also need to be standardized for families of blockchain systems that employ similar or compatible ledger data structures (e.g. Ethereum and Quorum).

5 Conclusions

This chapter discussed the emerging field of blockchains and distributed ledgers and introduced two important concepts - blockchain intraoperability and interoperability. We showed that intraoperability and interoperability are achievable by using AMMs and gateway asset transfer protocols, respectively. While there are many specific issues left for future research, the general directions are clear. The successful design of intra- and interoperability mechanisms is a must for making blockchain technology a viable tool for solving some of our times' most challenging technological problems.

We are grateful to Prof. Sandy Pentland and Dr. Marsha Lipton for their invaluable help.

References

1. Angeris, G., Kao, H.T., Chiang, R., Noyes, C., Chitra, T., 2019. An analysis of Uniswap markets. Cryptoeconomic Systems Journal.
2. Ankenbrand, T., Bieri, D., Cortivo, R., Hoehener, J., Hardjono, T.: Proposal for a Comprehensive (Crypto) Asset Taxonomy (2020). URL https://arxiv.org/abs/2007.11877
3. Bartel, M., Boyer, J., Fox, B., LaMacchia, B., Simon, E.: XML Signature Syntax and Processing Version 2.0. W3C Candidate Recommendation, W3C (2015). Available at http://www.w3.org/TR/2015/NOTE-xmldsig-core2-20150723/
4. BBC News: CryptoKitties craze slows down transactions on Ethereum. BBC News (2017). URL https://www.bbc.com/news/technology-42237162

5. Cerf, V.G., Khan, R.E.: A Protocol for Packet Network Intercommunication. IEEE Transactions on Communications **22**, 637–648 (1974)
6. Clark, D.: The Design Philosophy of the DARPA Internet Protocols. ACM Computer Communication Review – Proc SIGCOMM 88 **18**(4), 106–114 (1988)
7. Egorov, M., 2019. StableSwap - efficient mechanism for Stablecoin liquidity. White paper.
8. Ezhilchelvan, P., Aldweesh, A., van Moorsel, A.: Non-blocking two phase commit using blockchain. In: Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock?18), p. 36?41. Association for Computing Machinery, New York, NY, USA (2018). URL https://doi.org/10.1145/3211933.3211940
9. FATF: International Standards on Combating Money Laundering and the Financing of Terrorism and Proliferation. FATF Revision of Recommendation 15, Financial Action Task Force (FATF) (2018). Available at: http://www.fatf-gafi.org/publications/fatfrecommendations/documents/fatf-recommendations.html
10. Godbole, O.: Bitcoin Transaction Fees Rise to 28-Month High as Hashrate Drops Amid Price Rally. Coindesk (2020). URL https://www.coindesk.com/bitcoin-transaction-fees-hashrate
11. Gogo, J.: Bitcoin Transaction Fees Soar 550% in a Month, BCH, Dash Transactions Much Cheaper. Bitcoin.com (2020). URL https://news.bitcoin.com/bitcoin-transaction-fees-soar-550-in-a-month-as-price-surges-bch-dash-cheapest-networks
12. Gray, J.: The Transaction Concept: Virtues and Limitations. In: Very Large Data Bases – Proceedings of the 7th International Conference, pp. 144–154. Cannes, France (1981)
13. Gupta, M., 2017. Blockchain for DUMMIES. John Wiley & Sons.
14. Hardjono, T., Hargreaves, M., Smith, N.: An Interoperability Architecture for Blockchain Gateways. Internet-draft draft-hardjono-blockchain-interop-arch-01, IETF (2020). URL https://www.ietf.org/archive/id/draft-hardjono-blockchain-interop-arch-01.txt
15. Hardjono, T., Lipton, A., Pentland, A.: Towards an Interoperability Architecture Blockchain Autonomous Systems. IEEE Transactions on Engineering Management **67**(4), 1298–1309 (2019). URL doi:10.1109/TEM.2019.2920154
16. Hardjono, T., Lipton, A., Pentland, A.: Towards a Contract Service Provider Model for Virtual Assets and VASPs. In: P2P Financial Systems International Workshop 2020 (2020). URL https://arxiv.org/abs/2009.07413
17. Hardjono, T., Smith, N.: Decentralized Trusted Computing Base for Blockchain Infrastructure Security. Frontiers Journal - Special Issue on Finance, Money & Blockchains **2** (2019). URL https://doi.org/10.3389/fbloc.2019.00024
18. Hargreaves, M., Hardjono, T.: Open Digital Asset Protocol. Internet-draft draft-hargreaves-odap-01, IETF (2020). URL https://www.ietf.org/archive/id/draft-hargreaves-odap-01.txt
19. Heaven, W.D.: A plan to redesign the internet could make apps that no one controls. MIT technology Review (2020). URL https://www.technologyreview.com/2020/07/01/1004725/redesign-internet-apps-no-one-controls-data-privacy-innovation-cloud/

20. Herlihy, M.: Atomic Cross-chain Swaps. In: Proceedings of the ACM Symposium on Principles of Distributed Computing PODC'18, pp. 245–254. Association for Computing Machinery, New York, NY, USA (2018). URL https://dl.acm.org/doi/pdf/10.1145/3212734.3212736
21. Huang, Z.: China-Backed Crypto Guru Wants to Unify World?s Blockchain. Bloomberg (2020). URL https://www.bloomberg.com/news/articles/2020-07-26/china-backed-crypto-guru-wants-to-unify-the-world-s-blockchains
22. Jones, M., Bradley, J., Sakimura, N.: JSON Web Signature (JWS) (2017). URL https://tools.ietf.org/html/rfc7515. IETF Standard RFC7515
23. Lipton, A., Treccani, A., Blockchain and Distributed Ledgers: Mathematics, Technology, and Economics. WSPC, Singapore (2021).
24. Martin, J.: Dfinity?s Internet Computer Could Be a Truly Decentralized Alternative. Cointelegraph (2020). URL https://cointelegraph.com/news/dfinitys-internet-computer-could-be-a-truly-decentralized-alternative
25. Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system.
26. Nolan, T.: Alt chains and atomic transfers (2013). URL https://bitcointalk.org/index.php?topic=193281.msg2224949-msg2224949
27. Saltzer, J., Reed, D., Clark, D.: End-to-End Arguments in System Design. ACM Transactions on Computer Systems **2**(4), 277–288 (1984)
28. Schär, F., 2020. Decentralized Finance: On Blockchain-and Smart Contract-based Financial Markets. Available at SSRN 3571335.
29. Shevchenko, A.: Using a DeFi protocol now costs more than $50 as Ethereum fees skyrocket. Cointelegraph (2020). URL https://cointelegraph.com/news/using-a-defi-protocol-now-costs-more-than-50-as-ethereum-fees-skyrocket
30. Traiger, I.L., Gray, J., Galtieri, C.A., Lindsay, B.G.: Transactions and Consistency in Distributed Database Systems. IBM Research Report **RJ2555** (1979)
31. US Congress: Electronic signatures in global and national commerce act (2000)
32. Whittemore, N.: JPMorgan Launches JPM Coin: Welcome to the Private Currency Era. Coindesk (2020). URL https://www.coindesk.com/jpmorgan-launches-jpm-coin-private-currency
33. Zakhary, V., Agrawal, D., Abbadi, A.E.: Atomic Commitment Across Blockchains (2019). URL https://arxiv.org/pdf/1905.02847.pdf
34. Zhang, Y., Chen, X., Park, D., 2018. Formal specification of constant product (xy= k) market maker model and implementation. 2018.