SoK: Payment Channel Networks

Kartick Kolachala, Mohammed Ababneh, Roopa Vishwanathan
New Mexico State University, USA
{kart1712, mababneh, roopav}@nmsu.edu

Abstract—**Payment Channel Networks (PCNs) have been proposed as an alternative solution to the scalability, throughput, and cost overhead problems associated with blockchain transactions. By facilitating offchain execution of transactions, PCNs significantly reduce the burden on the blockchain, leading to faster transaction processing, reduced transaction fees, and enhanced privacy. Despite these advantages, the current state-of-the-art in PCNs presents a variety of challenges that require further exploration. In this paper, we survey several fundamental aspects of PCNs, such as pathfinding and routing, virtual channels, state channels, payment channel hubs, and rebalancing protocols. We aim to provide the reader with a detailed understanding of the various aspects of PCN research, highlighting important advancements. Additionally, we highlight the various unresolved challenges in this area. Specifically, this paper seeks to answer the following crucial question: *What are the various interesting and non-trivial challenges in fundamental infrastructure design leading to efficient transaction processing in PCN research that require immediate attention from the academic and research community?* By addressing this question, we aim to identify the most pressing problems and future research directions, and we hope to inspire researchers and practitioners to tackle these challenges to make PCNs more secure and versatile.**

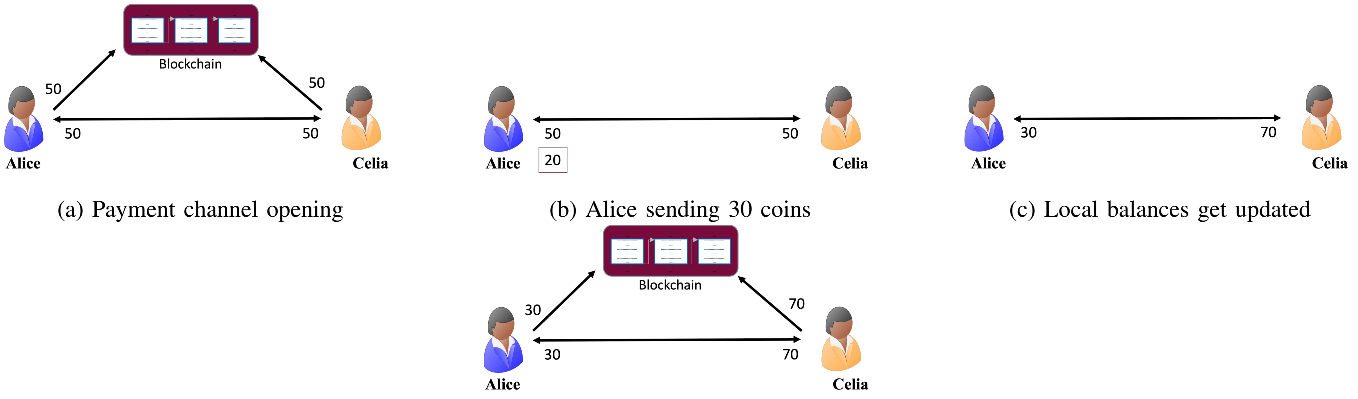
Index Terms—**Payment channel network, Blockchain, Layer-2**

I. Introduction

Cryptocurrencies and cryptocurrency based transactions have become increasing popular. Currently, the total market value of all cryptocurrencies in use has surpassed 2.5 Trillion USD. The cyrptocurrency market is increasing at a rate of ≈ 8.00% every year [1]. This rise in popularity can be attributed to the following reasons: 1) cryptocurrency transactions can be carried out without the presence of a trusted entity. Fiat currency transactions on the other hand, require the presence of a trusted financial organization such as a bank. 2) cryptocurrency transactions do not subject the user to any limits on the number and type of transactions. Fiat currency transactions are limited in their amount and number, and depend on several factors such as the currency, geographical location, etc. [2].
Each transaction posted to the Bitcoin blockchain takes around 7 seconds to be validated [3], [4]. The procedure of validation involves verifying that the transaction posted to the blockchain contains all the required fields and if the signature of the user creating the transaction tuple is valid. Once the validation procedure is successfully completed, the transaction is included in a block that would be mined on the blockchain at some time in the future. The process of mining the block successfully takes ≈ 2 hours [5] (as of June 13^{th} 2024). This delay in the transactions and blocks getting confirmed is termed as the blockchain scalability problem [6]. In contrast, Visa, a company which globally processes transactions using fiat currencies, processes around 24,000 transactions per second [7]. Due to the delay in transaction processing caused by the blockchain scalability problem, blockchain-based transactions cannot process payments instantaneously.
As an alternative to processing transactions by posting to the blockchain, payment channels have been proposed. Two parties with the intent of processing payments between them open a payment channel by creating a transaction tuple called the funding transaction. This funding transaction contains the initial deposits from both the parties. These initial deposits are also called as the initial balances of the parties in the payment channel. The sum aggregate of these initial balances is called as the channel capacity. The funding transaction contains the signatures of both the parties involved in the payment channel making it a 2-2 multi signature transaction. This means that the funds in the funding transaction cannot be spent without the signatures of both the parties. This funding transaction is validated and included in a block. Once this block has been successfully mined and confirmed on the blockchain, the payment channel is opened between the two parties. The two parties can now be involved in an unlimited number of transactions with each other as long as the amount of a single transaction does not exceed their local balances.

An example of a payment channel is given in Figure 1a. Two users (also called nodes) Alice and Celia deposit 50 coins each into a 2-2 multi signature transaction. This transaction is posted to the blockchain, upon which it is validated and included in a block. The block is mined and confirmed, at which time a payment channel is said to open between Alice and Celia. The sum aggregate of the individual balances of Alice and Celia, which is the channel capacity, is 100 coins. Alice making a payment of 20 coins to Celia is shown in the Figure 1b and the updated balances of Alice and Celia are shown in Figure 1c. After the payment has been made, the channel capacity between Alice and Celia still remains constant at 100 coins. In this manner, Alice and Celia can be involved in an unlimited number of payments between each other. Each payment made in the payment channel in an off-chain manner contains the signatures of both Alice and Celia. When either Alice or Celia decide to close the payment channel, they post their final balances to the blockchain and the payment channel is closed as shown in Figure 1d.

For each off-chain transaction in the payment channel, both the parties involved in the payment channel create a commitment. This commitment is essentially an agreement for the new balances signed by both the parties. Exchanging of



(a) Payment channel opening

(b) Alice sending 30 coins

(c) Local balances get updated

(d) Posting final balances to blockchain

Fig. 1: Payment channel operations

commitments signifies that both the parties have agreed to the change in their respective balances. Each pair of commitments (for each transaction), contains a unique sequence number called the revocation sequence maturity number. For each new transaction made in the payment channel, the sequence number of the prior transaction is invalidated by revocation keys of both the parties. These revocation keys are created by both the parties before opening of the payment channel. If a malicious party in the payment channel broadcasts an older balance to the blockchain, the honest party in the channel has a certain time period during which it can contest this behavior on the blockchain. Before this time period expires, the honest party in the payment channel will broadcast the revocation of this old state signed by both the parties. The broadcasting of this revocation to the blockchain prevents the malicious party from stealing funds of the honest party.

The idea of a payment channel that exists between two parties can be extended to a number of nodes, creating a network of payment channels, called a *payment channel network* or PCN. PCNs enable users that are not connected by a direct payment channel to make payments between each other in an off-chain manner. An example PCN is shown in Figure 2. In the figure, consider Alice who intends to make a payment to Hector, with whom she does not share a payment channel. The naïve way to process this transaction would be for Alice to open a payment channel with Hector, which involves Alice making an expensive blockchain write for the channel opening. Each payment channel opening costs 2.4 USD for blockchain writes [8]–[10]. If Alice intends to send an amount of 1 coin to Hector, it may not be economical for her to open a direct payment channel. Alice can make use of the PCN and make a payment to Hector by forwarding the payment along the path Alice → Celia → Michael → Rajiv → Charlie → Garcia → Hector. This process of using intermediate nodes in a PCN to forward to the payment to the intended destination is called as routing in payment channel networks.

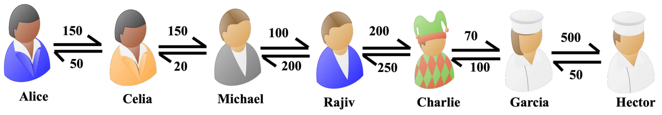


Fig. 2: Payment channel network

Motivation and timeliness of PCN research: A significant advantage of PCNs is their capability to facilitate micro-payment transactions, with minimum amounts as low as 10^{-7} BTC [11]. In contrast, the average transaction cost for a single on-chain transaction on the BTC blockchain is approximately 4.612 USD as of June 2024 [12] [1], regardless of the transaction amount. This cost can be avoided by using off-chain PCNs, which incur no additional fees. Additionally, transactions on the BTC blockchain take around 2 hours to be confirmed as of June 13, 2024 [5], whereas PCNs can process transactions instantly. An example of a real-world PCN is the Lightning Network (LN) on the Bitcoin blockchain [13], which has a 24-hour trading volume of $63,200 and a market capitalization of $7 million [14] as of June 2024. These figures reflect LN's size and growth.

In this SoK, we do not survey the various types of attacks in PCNs [15]–[21], [21]–[41]. The attacks in PCNs usually have overlaps in their strategy and execution, and most of them currently do not have efficient and fully developed mitigation mechanisms proposed. Our conjecture is that, their countermeasures might also have design overlaps as and when they are proposed. Hence, we believe attacks in PCNs and their countermeasures require their own taxonomy. For this SoK, we have covered papers in various areas of PCNs during the time period of 2019-2024 across Tier-1, Tier-2, Tier-3 security conferences in CS, since the norm for security/privacy research

[1] Opening a payment channel requires two inputs (for funds from both the parties) and one output (the funds are locked in a single 2-2 multi signature address). Whereas making a payment by posting it to the blockchain requires one input (from the party that intends to make the payment) and two outputs (one to the receiver and one to the validator/miner for his fees/reward). This difference makes the transaction cost associated with opening a payment channel cheaper than making a payment using a blockchain write.

and computer science research in general is peer-reviewed conferences.

Contributions:

1) We qualitatively compare the recent work in several aspects of building PCNs, viz. pathfinding and routing, virtual channels, state channels, payment channel hubs, and rebalancing using several relevant properties (metrics) along with providing a reasoning why these metrics have been chosen for comparison.

2) We point out the open problems in all the areas that we survey and we also discuss why solving those problems is a hard research challenge.

Outline: In Section II, we start with describing the concept of pathfinding and routing in PCNs and qualitatively compare work published in that area. In Section V, we describe virtual channels which have been proposed to address issues with multi-hop routing in PCNs, and compare work in this area. In Section IV, we describe and compare state channel protocols, which are a more generalized adaptation of virtual channels and can facilitate the execution of arbitrary applications between nodes in the PCN (not just payments). In Section V, we cover payment channel hubs, which are similar to virtual channels and state channels, but facilitate a different use case for payments in PCNs. All of the aforementioned PCN mechanisms consume one common resource: the local balance of a node in a PCN. In Section VI, we discuss rebalancing, which addresses the important function of replenishing channel funds in the PCN. In Section VII, we present the reader with the current research gaps in all of the aforementioned areas and also describe why bridging those gaps is hard. In Section VIII, we conclude the paper.

Prior work: Prior works by Khojasteh *et al.* [42] and Erdin *et al.* [43] survey the work done only in the area of pathfinding and routing protocols and their privacy aspects in PCNs. Whereas, in this paper we cover the entire spectrum of PCN research: rebalancing, virtual channels, state channels, pathfinding and routing, and tumblers. Apart from this [42], [43] do not provide any information about the open problems in PCNs, which we do in our work. The SoK by Gudgeon *et al.* [44], surveys several layer-2 protocols, whereas, we focus exclusively on PCNs. Furthermore [44] was published in 2020 and does not cover most of the recent work published in PCNs.

II. Pathfinding and Routing

Motivation: One of the areas in PCNs that has garnered significant attention from the academic community is pathfinding and routing. Pathfinding is defined as the process of finding a path comprising several nodes from a sender to a receiver in a PCN along which a payment can potentially be forwarded, and routing is the process of actually forwarding the payment along the found path. Intuitively, it may seem that well-known pathfinding and routing protocols from the wired and wireless networks domain can be easily applied to PCNs. Unfortunately, there are several problems with this: 1) Traditional networks focus on the transfer of data, PCNs on the other hand, transfer money in a decentralized manner. 2) Data transfer in traditional peer-peer networks does not alter the bandwidth, whereas money transfer in PCNs alters the monetary state of the nodes involved. 3) Cost in traditional networks is measured in terms of latency, whereas in PCNs, it involves routing fees, leading to greedy behavior among users and makes PCNs vulnerable to various attacks [44].

The properties on the columns in Table I represent the fundamental principles of fiat currency transactions and on-chain payment mechanisms, which we want reflected in off-chain payments. These properties are generally agreed upon in the literature by several works such as [45], [46], [48], [62]–[64] as common evaluation metrics for pathfinding and routing protocols in PCNs. Fulfilling these properties while providing efficient pathfinding and routing is a non-trivial challenge, and necessitates the design of new pathfinding and routing protocols. Several elegant pathfinding and routing protocols have been proposed in the literature. In Table I, we present a qualitative comparison of these routing protocols with respect to the properties they achieve. In this paper across all sections, if any prior work has identified a property as ideal or has identified a gap in research, we give an appropriate citation(s). If there is no citation provided, it indicates that the corresponding property/research gap has been identified by us.

Ideal properties: 1) **Concurrency**: Concurrency is achieved when a pathfinding and a routing protocol enables the nodes to forward more than one payment simultaneously [65]. *Importance*: At a given instant of time, many users will be using the PCN to make offchain payments. Hence it is important for a routing protocol to support concurrency. 2) **Privacy**: Privacy is maintained when a node's real identity is known only to its immediate neighbors and not to the entire network. *Importance*: Information of a node such as its identity, local balance, connections with other nodes in the network and its transaction history are private and should not be known to anyone else. 3) **Topology privacy**: Topology privacy is preserved when no single node has knowledge of the entire network topology. *Importance*: If topology privacy is not preserved, it violates the privacy of every node in the network. Making network topology public can potentially lead an adversary to reconstruct transaction paths, which in turn can lead to an adversary selectively targeting a certain set of nodes. 4) **Avoids source routing**: Source routing is avoided when the sender does not determine the path to the receiver. *Importance*: If a sender determines the complete path to the receiver, it means that he has access to the entire network topology. PCNs are highly dense and dynamic in nature. It is practically infeasible for a node to maintain an updated network topology all the time. 5) **Decentralization**: Decentralization is achieved when there are no centralized, trusted entities responsible for constructing paths for senders. *Importance*: Cryptocurrency payments made using the blockchain are by nature decentralized, hence routing protocols which facilitate offchain cryptocurrency payments should also be decentralized. 6) **Atomicity**: Atomicity is ensured when the payment is routed all the way from the sender to the receiver, or the payment is not routed at all. *Importance*: Atomicity

TABLE I: Comparison of Pathfinding and Routing Protocols in PCNs

Protocols	Concurrency	Privacy	Topology privacy	Avoids source routing	Decentralized	Atomicity	Disjoint graphs	Fees	Year
SilentWhispers [45]	✗	✓	✓	✓	✓	✗	✗	✗	2017
SpeedyMurmurs [46]	✓	✓	✓	✓	✓	✗	✗	✗	2018
Coinexpress [47]	✓	✗	✓	✓	✓	✓	✗	✗	2018
Blanc [48]	✓	✓	✓	✓	✓	✓	✗	✗	2019
Robustpay [49]	✗	✗	✗	✗	✗	✓	✗	✓ (flat)	2019
Flash [50]	✗	✗	✗	✗	✗	✓	✗	✗	2019
Cheapay [51]	✗	✓	✓	✓	✓	✗	✗	✓ (flat)	2019
Eckey *et al* [52]	✗	✗	✓	✓	✓	✓	✗	✗	2020
FSTR [53]	✗	✗	✗	✗	✗	✗	✗	✗	2020
Spider [54]	✗	✗	✗	✗	✗	✗	✗	✗	2020
Vein [55]	✗	✗	✗	✗	✗	✗	✗	✓ (dynamic)	2021
Kadry *et al.* [56]	✗	✗	✗	✗	✗	✗	✗	✗	2021
Webflow [57]	✗	✓	✓	✓	✓	✗	✗	✗	2021
Robustpay+ [58]	✗	✗	✗	✗	✗	✓	✗	✓ (flat)	2021
MPCN-RP [59]	✗	✗	✗	✗	✗	✓	✗	✓ (flat)	2022
Auto tune [60]	✗	✗	✗	✗	✗	✗	✗	✓ (flat)	2023
Yang *et al.* [61]	✓	✓	✓	✓	✗	✗	✗	✗	2023
RACED [62]	✓	✓	✓	✓	✓	✓	✓	✗	2024
Auroch [63]	✓	✓	✓	✓	✓	✓	✗	✓ (dynamic)	2024
SPRITE [64]	✓	✓	✓	✓	✓	✓	✗	✗	2024

is important since it ensures that honest people do not lose their funds because of malicious behavior by other parties in the system. 7) **Disjoint graphs**: A pathfinding and routing protocol is considered applicable to disjoint graphs if it functions even when the network graph consists of islands. *Importance*: PCNs often comprise of islands which only have a couple of nodes. These islands are often disconnected from other dense parts of the PCN. A routing protocol should be able to facilitate transactions between any pair of nodes irrespective of their location. 8) **Fees**: Routing fees is the amount a node charges for forwarding the payment to the next node along a path from the sender to receiver. This fees can be charged in two ways. Flat/fixed fees means that the fees charged for routing payments remains the same irrespective of the transaction amount being routed. If the fees charged by a node varies according to the transaction amount, it is referred to as dynamic fees, typically a percentage of the amount. *Importance*: Every node along a payment path aids the sender in transaction processing by forwarding the payment to the next node along the path. The nodes need to be paid in the form of routing fees for their service.

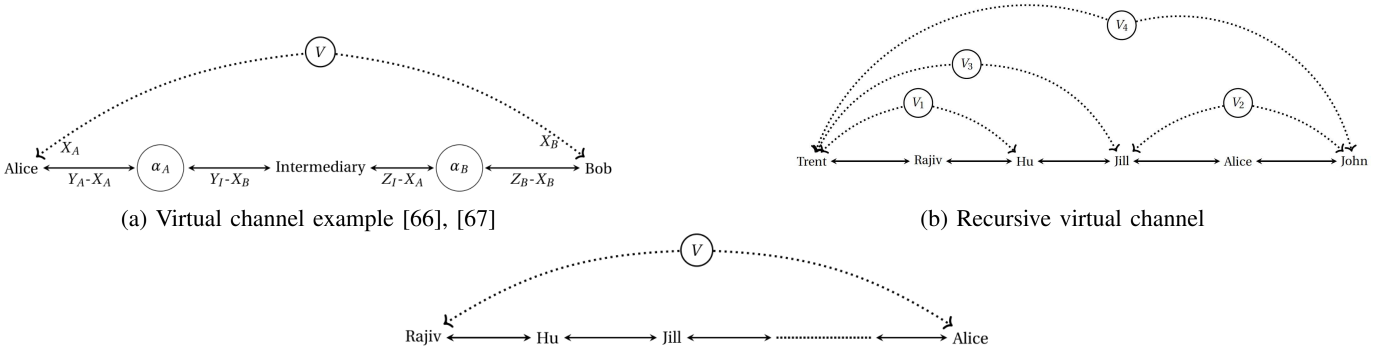
As illustrated in Table I, routing protocols for PCNs have evolved significantly over the years. The two most significant advancements are taking routing fees into consideration and providing support for privacy. For instance, LN provides users with two sets of keys, a long-term keypair and an alias (a temporary identity) helping to conceal their identities and ensure privacy. Despite these developments, two overarching research problems remain that require attention. We discuss them in detail in Section VII.

III. Virtual channels

Motivation: Transactions in PCNs are routed from the sender to the receiver using a path of intermediate nodes. Current pathfinding and routing mechanisms require the nodes along the payment path to be available for a transaction to be processed. However, nodes can sometimes choose to go offline or there can be network/service disconnections causing transaction failures. Furthermore, each node along a payment path charges its own fees for forwarding the payment, which is paid by the sender and increases with the path length, hence the time taken to route a payment grows linearly in the path length. Virtual channels, which are built on top of existing payment channels, solve these problems. Initial constructions of virtual channels facilitated payments between a pair of nodes using a single intermediate node [66]. The intermediate node needs to have individual payment channels open with the other two nodes. The intermediary and the pair of nodes lock coins with each other in their respective payment channels and a virtual channel is established. Upon establishment of the virtual channel, the pair of nodes can be involved in a unlimited number of payments. These payments can be processed without the intermediate node being online. It might be better to use virtual channels from a routing fees perspective, since unlike routing protocols, nodes in virtual channels do not charge a routing fee for every transaction.

Alice, Bob, and an intermediary establish a virtual channel as shown in Figure 3a. Alice locks Y_A coins and the intermediary locks Y_I coins in the payment channel α_A. Similarly, Bob locks Z_B coins and the intermediary locks Z_I coins in their channel α_B. The virtual channel V is created once Alice locks X_A coins from her balance in α_A and Bob locks X_B coins from his balance in α_B. Now Alice and Bob can process payments without the intermediary's online presence.

The idea of a virtual channel between a pair of nodes involving a single intermediary has been extended to establishing a virtual channel recursively over several hops involving several intermediaries, leading to the construction of a recursive virtual channel [70], [72], [73]. An example of a recursive



(a) Virtual channel example [66], [67]

(b) Recursive virtual channel

(c) Multi-hop virtual channel (the dots between Jill and Alice depict several intermediate nodes)

Fig. 3: Different types of virtual channels

TABLE II: Comparison of Virtual Channel Protocols

Protocol	BC	Validity	Fee	Privacy	Off-chain dispute resolution	Recursive	Multihop	Year
Generalized state channels [68]	TC	Limited	✗	✗	✗	✓	✗	2018
Eckey *et al.* [69]	TC	Limited	✗	✗	✗	✓	✗	2019
Perun [66]	TC	Limited	✗	✓	✗	✗	✗	2019
Jourenko *et al.* [70]	UTXO	Limited	✗	✗	✗	✗	✗	2020
Aumayr *et al.* [71]	UTXO	Limited	✓ (fixed)	✗	✗	✗	✗	2021
Elmo [72]	UTXO	Unlimited	✗	✗	✗	✓	✗	2021
Donner [73]	UTXO	Unlimited	✓ (time based)	✓	✗	✓	✗	2023
Jia *et al.* [74]	UTXO, TC	Limited	✓ (fixed)	✗	✗	✗	✗	2023

channel is shown in Figure 3b. A recursive virtual channel enables transaction processing between any pair of nodes, by recursively establishing virtual channels between several intermediaries, as opposed to [66], which facilitates virtual channels only between a pair of nodes connected directly to an intermediary.

A multihop virtual channel is constructed by establishing a single virtual channel between a pair of users over a path comprising of several intermediate nodes. An example of multihop virtual channel is shown in Figure 3c. A multihop virtual channel is an improvement over recursive virtual channels. In a multihop virtual channel, a single virtual channel can be established between any pair of nodes which are separated by several intermediaries.

Virtual channels should not add an unnecessary burden to users, and should mirror the operations of payment channels as closely as possible, with the added benefit of no on-chain transactions at all, while maintaining comparable security/privacy properties. We now give the properties desired from an ideal virtual channel, and compare the works in this area on the extent to which they achieve these properties.

Blockchain terminology: In the rest of the paper BC denotes a blockchain. TC is a blockchain that supports a Turing complete programming language, such as Ethereum, and UTXO is a blockchain that supports the UTXO-based scripting mechanism such as Bitcoin. In Table II, we present a qualitative comparison of virtual channel protocols.

Ideal properties: 1) **Validity**: This determines the validity of the virtual channel. A limited validity means that the virtual channel is valid for a predetermined time period (which is decided by nodes involved in the virtual channel). Unlimited validity means that the virtual channel can stay open until the nodes involved decide to initiate closing [69]. *Importance*: Unlimited validity provides better and efficient transaction processing between users who intend to have frequent payments made between each other. 2) **Fee**: This metric determines if the virtual channel takes into account the fees charged by the intermediate node(s) involved in the channel's establishment. The fixed fee model implies that a predetermined, fixed fee is paid to the intermediate node(s) which is agreed upon by all the nodes in the virtual channel before the channel establishment. The time-based fee model implies that the fee paid to the intermediary depends on the time for which the virtual channel stays open [69]. *Importance*: Having an well-defined fee structure will motivate the intermediary/intermediaries to participate in virtual channel creation. 3) **Privacy**: In any virtual channel construction, the real identity of a node should only be known to its immediate neighbor(s) [66]. *Importance*: Privacy is important since it helps in preserving topology privacy. 4) **Offchain dispute resolution**: This metric determines if the transaction disputes in a virtual channel require a blockchain write. *Importance*: Ideally, we would want a virtual channel construction to have off-chain dispute resolution since blockchain writes are expensive and time-consuming. 5) **Support for multihop virtual channels with several intermediaries**: A virtual channel is said to be multihop if it can facilitate payments between a sender and a receiver across a path comprising of several intermediate nodes by constructing a single virtual channel from the sender to the receiver, without establishing virtual channels between any pair of intermediate nodes along the path. *Importance*: This property is ideal since it facilitates payments between any pair of nodes in the network by establishing only a single virtual channel, as opposed to a recursive virtual channel in which a sender/receiver, and the intermediate nodes lock coins in multiple virtual channels at the same time.

The most significant developments in virtual channels over the years are that newer protocols have incorporated a fees to be paid for the intermediary(ies) that lock coins in virtual channels and virtual channels now offer support for both TC based and UTXO based blockchains. Despite these developments, efficient virtual channel protocol design has three overarching research problems, which are discussed in Section VII.

IV. State channels

A state channel is an off-chain protocol that can facilitate execution of an arbitrary decentralized application, also called as DApp (such as a two player game) between two users in a decentralized and distributed network. An example of a state channel is given in Figure 4a. Let us consider two users Alice and Celia who intend to play a game of chess. They initially interact with the chess game application deployed on the blockchain by a third party service provider. Both Alice and Celia deposit 30 coins each from their possession into a 2-2 multi signature transaction (step 1). Upon locking the funds, the chess game is instantiated between Alice and Celia by the DApp (step 2). Alice and Celia engage in a series of moves for the chess game (step 3, step 4) and let us assume Alice eventually wins (step 5) as shown in Figure 4b. Alice gets paid 15 coins as the prize money in Figure 4c. In this game, Alice and Celia countersign each other's moves until there is a winner or a draw. In the event of malicious activity by one party (such as undoing a prior move), the honest party will broadcast all the moves to the blockchain for dispute resolution.

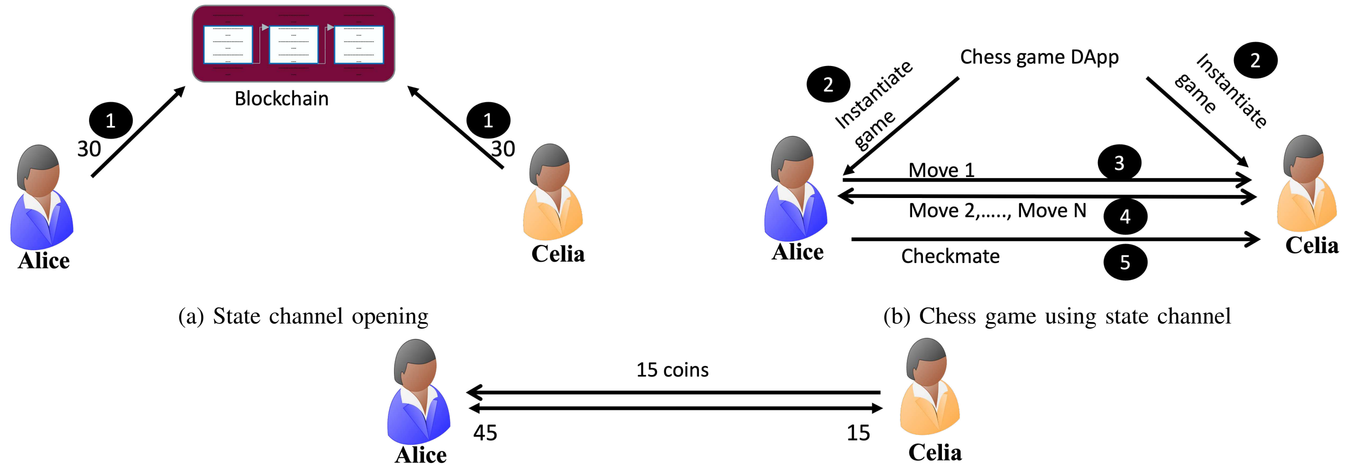
Motivation: State channels allow the already existing payment channels to facilitate execution of arbitrary applications such as games, e-commerce, etc. This leads to the PCN becoming more versatile. State channels should be deployable on any blockchain regardless of the underlying programming/scripting requirements and should be able to facilitate the execution of any decentralized application between the parties involved, while maintaining comparable security properties (to payment channels) and should ideally be able to resolve disputes without accessing the blockchain. We now give the properties desired from an ideal state channel, and compare the works in this area on the extent to which they achieve these properties. We give a qualitative comparison of various state channel protocols in Table III. We describe the metrics used for comparison below.

Ideal properties: 1) **General purpose state channels**: This is the ability of the state channel to facilitate the execution of any application supported by the underlying blockchain in an off-chain manner *Importance*: This is an ideal feature since it facilitates the deployment of any application without accessing the blockchain, making state channels more versatile. 2) **Graceful exit**: A state channel protocol is said to employ a graceful exit if it has clear and well-defined mechanisms for nodes joining or leaving the state channel without accessing the blockchain. *Importance*: This is an important property since most of the state channel protocols require the user to interact with the blockchain when they join/leave the state channel(s) they are a part of. 3) **Off-chain dispute resolution**: This property indicates if a dispute that occurs between the parties involved in the state channel can be resolved without accessing the blockchain. *Importance*: Ideally, we would want a state channel construction to have off-chain dispute resolution since blockchain writes are expensive and time-consuming.

V. Tumblers

Motivation: A payment channel hub (tumbler) is a multi-party off-chain system where users can establish payment channels with a central hub, which acts as an intermediary. It allows multiple users to send payments to each other without the need for direct payment channels between each user pair. The hub coordinates payments between different participants. The intermediary which facilitates payments is called a tumbler. Though a payment channel hub uses the same underlying infrastructure as that of PCNs and virtual channels, each of these constructions have their own use cases. PCNs are usually used when two nodes Alice and Bob transact on an infrequent basis. Virtual channels are used if Alice and Bob transact frequently, e.g., if Bob provides Alice with a service every month. Payment channel hubs are used when Alice needs to pay several receivers on a frequent basis and she does not want the tumbler to know the receivers.

Payment channel hubs can be classified into two types: onchain and off-chain. Early hubs, also called hubs or mixers, were on-chain [91]–[102], but they all suffered from scalability issues due to having to post each transaction on the blockchain. The scalability issues of on-chain payment channel hubs have led to the development of offchain payment channel hubs for specific blockchains, e.g., Bolt [81] for Zcash, Nocust [83] and MixCT [88] for Ethereum. The most general-purpose payment channel hubs are Tumblebit [82], A^2L [85], and Blindhub [90]. A payment channel hub should be able to



(a) State channel opening

(b) Chess game using state channel

(c) State channel close

Fig. 4: State channel example with a chess game decentralized application (DApp)

TABLE III: Comparison of State Channel Protocols

Protocol	BC	General/purpose state channels	Security	Graceful exit	Off-chain dispute resolution	Year
Forcemove [75]	TC	✗	✓	✗	✗	2018
Pisa [76]	TC	✗	✓	✗	✗	2019
Sprites [77]	UTXO	✓	✓	✗	✗	2019
Hydra [78]	UTXO	✓	✓	✗	✗	2021
Aumayr *et al.* [79]	UTXO, TC	✗	✓	✗	✗	2021
Origami [80]	TC	✗	✓	✓	✗	2023

TABLE IV: Comparison of Payment Channel Hubs

Protocol	BC	Relationship anonymity	Privacy against aborts	Independent of epochs	Dynamic corruption	Atomicity	Value privacy	Variable amount	Year
BOLT [81]	TC	✓	✗	✓	✗	✓	✓	✓	2017
Tumblebit [82]	UTXO	✓	✗	✗	✗	✓	✗	✗	2017
Nocust [83]	TC	✓	✗	✗	✗	✓	✗	✓	2018
Teechain [84]	TC	✓	✗	✗	✗	✓	✓	✓	2019
A^2L [85]	UTXO, TC	✓	✗	✗	✗	✓	✓	✗	2021
Accio [86]	TC	✓	✓	✓	✗	✓	✓	✓	2021
Boros [87]	TC	✗	✗	✓	✗	✓	✗	✓	2021
MIXCT [88]	TC	✓	✗	✗	✗	✓	✓	✓	2022
Turbo [89]	TC	✓	✗	✗	✗	✗	✗	✗	2022
Blindhub [90]	UTXO, TC	✓	✗	✗	✗	✓	✓	✓	2023

facilitate payments between a sender and receiver, who do not have a payment channel open between them such that the hub cannot link a given transaction amount to a particular sender/receiver pair. Furthermore, the payment channel hub should also guarantee the fundamental property of atomicity (ensuring that the payment is sent to the receiver or it does not go through at all). We now give the properties desired from an ideal payment channel hub, and compare the works in this area on the extent to which they achieve these properties. In Table IV, we qualitatively compare several off-chain tumblers.

Ideal properties: 1) **Relationship anonymity**: It ensures that the relationship between a sender and a receiver for a given transaction should not be known to the tumbler [81]. *Importance*: This is an important property since one of the main uses and design goals of a tumbler is to facilitate anonymous transactions. Hence, any tumbler construction should satisfy this property. 2) **Privacy against aborts**: Tumbler should not be able to deduce the identities of a sender/receiver in case of a transaction abort, regardless of which party is responsible for the abort [90]. *Importance*: This is an important property to have since aborts can happen due to several reasons such as network disconnections, power outages, etc. Malicious nodes can also deliberately abort a protocol. In the case of tumbler protocols, the nodes whose payments did not succeed can be linked to each other once the protocol execution completes. 3) **Independent of time epochs**: The tumbler processes transactions in discrete fragments of time called epochs, i.e., transactions only take place during a time epoch [81]. This is not ideal. *Importance*: A payment channel hub that supports transaction processing based on time epochs is not efficient because nodes cannot process payments between time epochs, which can be of arbitrary length. The duration between time epochs depends on several factors such as the unit of the time epoch (absolute time or relative time measured in terms of block height), number of users that will most likely involved in the next time epoch, etc. 4) **Protection against dynamic corruption of nodes**: Current tumbler protocols corrupt nodes at the beginning of protocol execution and assume that the set of corrupted nodes remains constant until the protocol execution terminates. Ideally, a payment channel hub should be able to handle the deviation of any party from the protocol at any point during its execution. *Importance*: This is an important property since its infeasible from a practical standpoint to assume that the set of adversaries in the payment channel hub protocol remains constant during execution. 5) **Atomicity**: Atomicity is ensured if either the payment is routed all the way from the sender to the receiver or the payment is routed at all. *Importance*: This is important since it prevents honest parties from losing their funds because of malicious parties. 6) **Value privacy**: Value privacy is ensured by a payment channel hub, when, given a transaction amount, the tumbler cannot link it to a sender/receiver pair [81]. *Importance*: Value privacy is important since only the sender and receiver should know the amount being transacted. In the event of every sender/receiver pair transacting a unique amount, lack of value privacy will enable the tumbler to link a sender/receiver pair to a particular transaction, violating relationship anonymity. 7) **Variable amount**: The tumbler should be able to process transactions of any amount [81]. *Importance*: A tumbler having the ability to process amounts of variable value is important since its practically infeasible to assume that all sender/receiver pairs will always transact a fixed amount.

Over the years, various payment channel hub constructions have been developed to address specific challenges based on their unique design goals. However, we have identified three overarching research problems in payment channel hubs that require attention. We discuss them in Section VII.

TABLE V: Comparison of Rebalancing Protocols; C is centralized and D is distributed

Protocol	BC	Trusted entity	Graph compatability	Privacy	Year
Revive [103]	TC	✓ (C)	Cycles only	✗	2017
Subramanian *et al.* [104]	TC and UTXO	✗	Agnostic	✓	2019
Rebal [105]	UTXO	✗	Cycles only	✓	2021
Hide & Seek [106]	UTXO	✓ (D)	Cycles only	✓	2022
Cycle [107]	TC	✗	Cycles only	✓	2022
Shaduf [108]	TC	✗	Agnostic	✓	2022
Musketeer [109]	UTXO	✗	Cycles	✗	2023
Chen [110] *et al.*	UTXO	✗	Cycles	✗	2024

VI. Rebalancing

Motivation: Rebalancing in PCNs involves redistributing the balance within an existing payment channel or across multiple channels to maintain or improve the network's liquidity and usability. This is done to ensure that both parties in the channel have sufficient funds on their respective sides to send and receive payments. Rebalancing is crucial to payment channel networks. If the link weights of nodes become zero/depleted as a result of being involved in transactions, it will prevent them from taking part in further payments until those link weights are replenished. Such nodes are called as dormant nodes in the network. A PCN which has a large number of such dormant nodes experiences an overall reduction in transaction throughput and an overall reduction in liquidity. Hence, the link weights of the nodes in the PCN need to be rebalanced to prevent failure of transactions due to lack of liquidity. A rebalancing protocol should be be able to replenish the link weight of dormant nodes in the PCN irrespective of the PCNs topology, while not having to employ any trusted entity and should not compromise privacy of nodes. We now give the properties desired from an ideal rebalancing protocol, and compare the works in this area on the extent to which they achieve these properties. We give a qualitative comparison of rebalancing protocols in Table V. We describe the metrics as follows:

Ideal properties: 1) **Trusted entity**: This metric determines the nature of trust required among parties to deploy the rebalancing protocol. Centralized trust (C) indicates the presence of a single trusted entity and distributed trust (D) indicates the distribution of trust across various entities. *Importance*: It is ideal to have distributed trust for a rebalancing protocol since, PCN payments are by nature decentralized. 2) **Graph compatibility**: This indicates the nature of the topology of the network on which a rebalancing protocol can be deployed. Cycles indicate that the rebalancing protocol can only deployed on cyclic graphs and agnostic indicates that the rebalancing protocol can be deployed irrespective of the network topology. *Importance*: Ideally a rebalancing protocol should be deployable irrespective of the topology of the graph since its practically infeasible to assume that every node in a PCN will be a part of a cycle. 3) **Privacy**: Privacy is said to be achieved when sensitive information such as the local balance and the identities of nodes are not known to anyone except for their immediate neighbors. *Importance*: The importance of privacy for layer-2 protocols has already been described in Section II.

Rebalancing is a relatively mature area and significant challenges have already been addressed.

VII. Research Gaps & Open Problems

In this section, we highlight the gaps in research published up until now in the areas of pathfinding and routing, virtual channel construction, state channels and payment channel hubs. The gaps are described as research questions, denoted by **RQ**.

RQ1: Why is super node liquidity validation in PCNs hard? A super node, variously called as a trampoline node, routing node, routing helper, landmark node, router, etc. [45], [48], [54], [62], [64] is a highly connected node with numerous high liquidity channels, that helps in pathfinding and routing payments. One of the main problems with the super nodes is that a sender has no way to know whether the super node possesses enough liquidity on its channels to route a payment. The local balance of a super node in a given channel (or of any node in a PCN) is a private value and should not be known to any node except for its immediate neighbor that it shares the channel with. Currently, if a super node does not have enough liquidity to route the payment of a sender, the transaction fails and it has to be retried by the sender. In LN, one of the most widely used PCN, this is a significant problem. Sometimes the sender might have to keep retrying for ≈ 1 hour to have a successful payment [111]. The main goal of PCNs is to facilitate instantaneous payments and these transaction retries render such payments almost impossible. It will greatly benefit the sender if it has a mechanism to validate whether a super node has enough liquidity (balance) to route its payment without violating any privacy concerns.

RQ2: Why is channel verification in a PCN hard? To be a part of any PCN, nodes will open payment channels with other nodes in the network. Two nodes open a payment channel between them by posting a transaction to the blockchain. This transaction can be posted on the blockchain or as a function call to an existing smart contract. In the most popular PCN, LN, the procedure of verifying whether a payment channel really exists on the blockchain is very inefficient. A node who wants to verify a channel needs to request the block in which the channel opening transaction has been included, verifying whether the transaction has been successfully executed by the validator/miner and finally verifying if the channel opening transaction corresponds to a 2-2 multi signature address on the blockchain. The verifier performing these steps is inefficient since all these steps will have already been performed by the miner. Finding a way to do this without blockchain access and in a blockchain agnostic manner in a hard research challenge.

RQ3: Why is designing pathfinding protocols for PCNs, that comprise of several distinct well-connected components a hard problem? Though solutions such as [62] exist that solve this problem to a certain extent by using routing helpers/trampoline nodes, the aspect that makes it hard is to quantify the denseness/sparseness of a well-connected component. There may be well-connected components in the PCN that comprise of only a few nodes (i.e, islands). In such a case, the nodes in the islands would ideally need to establish payment channels with either a trampoline in their component or a non-trampoline node belonging to other well-connected components. If there is no trampoline available for a well-connected component, the nodes in that component might have to establish payment channels with ideally more than one node from a well-connected component that has a trampoline. This directly contradicts the advantage of payment channels which is to facilitate payments without accessing the blockchain.

RQ4: Why is designing a routing protocol that supports concurrent payments and is resilient to channel gaming a hard problem? Processing concurrent transactions requires the design of a mechanism that allows a node to lock a portion of its liquidity in a channel with an immediate neighbor for one transaction while simultaneously using the remaining liquidity to process another. Though there are protocols that support concurrency [45], [47], [48], [62]–[64], they are not resilient to the presence of potentially malicious nodes in the PCN, which may initiate transactions with the sole intent of locking liquidity, leading to congestion and disruption in the network.

RQ5: Why is having a well-defined fee structure for virtual channels hard? The intermediary(s) involved in the virtual channel construction additionally lock coins in virtual channels apart from the ones locked in the underlying payment channel. Currently, nodes get paid routing fees for every transaction they process. In the case of virtual channels, having a well-defined fee structure is difficult due to the following reasons: 1) The fee structure should take into account the amount of funds and the time for which these funds of the intermediary(ies) are locked in a virtual channel. 2) It also needs to take into account the routing fee an intermediary could have earned by not locking up the coins in the virtual channel.

RQ6: Why is off-chain dispute resolution in virtual channels hard? There is no offchain consensus mechanism for dispute resolution in a PCN, as opposed to the 51% honest majority assumption that exists among validators on the blockchain. This honest majority helps resolve disputes in the transactions posted to the blockchain. Designing such a dispute mechanism for layer-2 transactions is hard since transactions are private (not posted to the blockchain), and nodes do not broadcast their activities to the entire network.

RQ7: Why is providing support for a multihop virtual channel a hard problem? This is hard since a multi-hop virtual channel construction should ensure that neither sender/receiver nor the intermediate nodes should lock coins in multiple channels at the same time.

RQ8: Why is ensuring privacy in a virtual channel protocol

a hard problem? In a recursive virtual channel, new virtual channels are constructed on top of existing virtual channels to facilitate payments. This staggered nature makes it mandatory to reveal the identity of at least one endpoint node (sender or receiver). This is because, at least one node among the sender/receiver is involved in all virtual channels. The solution to this problem is to design a multihop virtual channel.

RQ9: Why is designing a payment channel hub (PCH) that is resistant to privacy against aborts and dynamic corruption a hard challenge? PCHs usually use transaction mixing for enhancing privacy, which is a process in which multiple payments from different users are mixed together in such a way that it is infeasible for the hub to link the sender and recipient of a specific transaction. This process helps obscure the flow of funds, providing unlinkability. Designing a payment channel hub that is resistant to privacy against aborts is hard because, if a PCH selectively aborts a payment from a sender/receiver, the counter party whose payment also failed can be linked. If sender/receiver gets corrupted during the PCH's execution, the corresponding transaction has to be aborted to ensure atomicity, which is why the existing tumbler constructions assume a static adversary, in which certain nodes are designated as corrupted before the PCH begins execution. The trade off here is preserving transaction unlikability during a corrupted party's transaction abort.

RQ10: Why is having an offchain dispute resolution for a state channel hard? Current state channel protocols use an onchain transaction or a function call to the onchain smart contract in the case of dispute resolution. An ideal state channel protocol should be able to facilitate dispute resolutions in an offchain manner. This is a hard research challenge since disputes on the blockchain are usually resolved by the underlying consensus mechanisms. In an offchain scenario, there is no such consensus that guarantees transaction validity. Also, in an offchain protocol such as the state channel, nodes which are not part of the state channel protocol do not know any details of the protocol's execution due to privacy concerns. Onchain transactions on the hand are publicly accessible.

VIII. Conclusion

In this paper, we qualitatively compared the recent work in various foundational areas of PCN research: pathfinding and routing, virtual channels, state channels, payment channel hubs, and rebalancing protocols. We also discussed the gaps in research in these areas along with reasons why fulfilling those gaps is non-trivial. We hope that this paper motivates researchers to build robust protocols that address these gaps that would go a long way towards building out and developing a decentralized financial ecosystem.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Award No. 2148358, 1914635, 2417062, and the Department of Energy under Award No. DE-SC0023392. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation and the Department of Energy.

References

[1] "Top cryptocurrency statistics and trends in 2024," https://bit.ly/3WnCkYE.

[2] Nerdwallet, "Xoom money transfer review," https://bit.ly/3QnPOAu.

[3] ycharts, "Bitcoin average confirmation time," https://bit.ly/3AhRlnf, 2023.

[4] miamiherald, "Btc transction througput," https://www.miamiherald.com/software-business/article274817896.html, 2023.

[5] "Btc confirmation time," https://bit.ly/3AhRlnf.

[6] C. Link, "Blockchain scalability: Execution, storage, and consensus," https://chain.link/education-hub/blockchain-scalability, 2023.

[7] "Visa," https://bit.ly/4dszZ5v.

[8] Medium, "How to calculate bitcoin transaction fees: What businesses need to know about bitcoin," https://bit.ly/3ygDZqN.

[9] 99BTC, "The complete guide to bitcoin fees from the complete guide to bitcoin fees & transactions in 2024," https://99bitcoins.com/bitcoin/fees/, 2023.

[10] coingecko, "Btc transction cost," https://bit.ly/3WtSzVh, 2023.

[11] MIT, "layer 2 the lightning network," https://dci.mit.edu/lightning-network.

[12] BTC, "Bitcoin average transaction fee," https://bit.ly/4djgHQ6.

[13] LND, "Lnd," https://docs.lightning.engineering, 2023.

[14] coinmarket, "Ln trading volume," https://bit.ly/3Yo6EEU.

[15] Z. Avarikioti, P. Kdzior, T. Lizurej, and T. Michalak, "Bribe & fork: Cheap bribing attacks via forking threat," *arXiv preprint arXiv:2402.01363*, 2024.

[16] B. Weintraub, S. P. Kumble, C. Nita-Rotaru, and S. Roos, "Payout races and congested channels: A formal analysis of security in the lightning network," *arXiv preprint arXiv:2405.02147*, 2024.

[17] T. Von Arx, M. Tran, and L. Vanbever, "Revelio: A network-level privacy attack in the lightning network," in *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*. IEEE, pp. 942–957.

[18] A. A. Khalil, M. A. Rahman, and H. A. Kholidy, "Fakey: Fake hashed key attack on payment channel networks," in *2023 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2023, pp. 1–9.

[19] S. Mazumdar, P. Banerjee, A. Sinha, S. Ruj, and B. K. Roy, "Strategic analysis of griefing attack in lightning network," *IEEE Transactions on Network and Service Management*, vol. 20, no. 2, pp. 1790–1803, 2022.

[20] C. Sguanci and A. Sidiropoulos, "Mass exit attacks on the lightning network," in *2023 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2023, pp. 1–3.

[21] C. Shikhelman and S. Tikhomirov, "Unjamming lightning: A systematic approach," *Cryptology ePrint Archive*, 2022.

[22] A. Biryukov, G. Naumenko, and S. Tikhomirov, "Analysis and probing of parallel channels in the lightning network," in *International Conference on Financial Cryptography and Data Security*. Springer, 2022, pp. 337–357.

[23] A. Riard and G. Naumenko, "Time-Dilation Attacks on the Lightning Network," *Cryptoeconomic Systems*, vol. 1, no. 2, oct 22 2021.

[24] A. Mizrahi and A. Zohar, "Congestion attacks in payment channel networks," in *International conference on financial cryptography and data security*. Springer, 2021, pp. 170–188.

[25] B. Weintraub, C. Nita-Rotaru, and S. Roos, "Structural attacks on local routing in payment channel networks," in *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, pp. 367–379.

[26] C. Pérez-Sola, A. Ranchal-Pedrosa, J. Herrera-Joancomartí, G. Navarro-Arribas, and J. Garcia-Alfaro, "Lockdown: Balance availability attack against lightning network channels," in *Financial Cryptography and Data Security: 24th International Conference*. Springer, 2020, pp. 245–263.

[27] J. Harris and A. Zohar, "Flood & loot: A systemic attack on the lightning network," in *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, 2020, pp. 202–213.

[28] S. Tochner, A. Zohar, and S. Schmid, "Route hijacking and dos in off-chain networks," in *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, 2020, pp. 228–240.

[29] S. Mazumdar, P. Banerjee, and S. Ruj, "Griefing-penalty: Countermeasure for griefing attack in lightning network," *arXiv preprint arXiv:2005.09327*, 2020.

[30] E. Rohrer and F. Tschorsch, "Counting down thunder: Timing attacks on privacy in payment channel networks," in *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, 2020, pp. 214–227.

[31] Z. Lu, R. Han, and J. Yu, "General Congestion Attack on HTLC-Based Payment Channel Networks," in *3rd International Conference on Blockchain Economics, Security and Protocols (Tokenomics 2021)*, ser. Open Access Series in Informatics (OASIcs), V. Gramoli, H. Halaburda, and R. Pass, Eds., vol. 97. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, pp. 2:1–2:15.

[32] G. Van Dam, R. A. Kadir, P. N. Nohuddin, and H. B. Zaman, "Improvements of the balance discovery attack on lightning network payment channels," in *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer, 2020, pp. 313–323.

[33] E. Rohrer, J. Malliaris, and F. Tschorsch, "Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks," in *2019 IEEE european symposium on security and privacy workshops (EuroS&PW)*. IEEE, pp. 347–356.

[34] M. Romiti, F. Victor, P. Moreno-Sanchez, P. S. Nordholt, B. Haslhofer, and M. Maffei, "Cross-layer deanonymization methods in the lightning protocol," in *International Conference on Financial Cryptography and Data Security*. Springer, 2021, pp. 187–204.

[35] S. Tikhomirov, P. Moreno-Sanchez, and M. Maffei, "A quantitative analysis of security, anonymity and scalability for the lightning network," in *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pp. 387–396.

[36] T. von Arx, M. Tran, and L. Vanbever, "Revelio: A network-level privacy attack in the lightning network," in *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pp. 942–957.

[37] G. Kappos, H. Yousaf, A. Piotrowska, S. Kanjalkar, S. Delgado-Segura, A. Miller, and S. Meiklejohn, "An empirical analysis of privacy in the lightning network," in *Financial Cryptography and Data Security: 25th International Conference, FC 2021*. Springer, pp. 167–186.

[38] P. Kumar Sharma, D. Gosain, and C. Diaz, "On the anonymity of peer-to-peer network anonymity schemes used by cryptocurrencies," in *The Network and Distributed System Security Symposium*. Internet Society, 2023.

[39] P. Zabka, K.-T. Foerster, C. Decker, and S. Schmid, "Short paper: A centrality analysis of the lightning network," in *International Conference on Financial Cryptography and Data Security*. Springer, 2022, pp. 374–385.

[40] P. Casas, M. Romiti, P. Holzer, S. B. Mariem, B. Donnet, and B. Haslhofer, "Where is the light(ning) in the taproot dawn? unveiling the bitcoin lightning (ip) network," in *2021 IEEE 10th International Conference on Cloud Networking (CloudNet)*, 2021, pp. 87–90.

[41] L. Heimbach, Y. Vonlanthen, J. Villacis, L. Kiffer, and R. Wattenhofer, "Deanonymizing ethereum validators: The p2p network has a privacy issue," *arXiv preprint arXiv:2409.04366*, 2024.

[42] H. Khojasteh and H. Tabatabaei, "A survey and taxonomy of blockchain-based payment channel networks," in *2021 IEEE High Performance Extreme Computing Conference (HPEC)*. IEEE, pp. 1–8.

[43] E. Erdin, S. Mercan, and K. Akkaya, "An evaluation of cryptocurrency payment channel networks and their privacy implications. arxiv 2021," *arXiv preprint arXiv:2102.02659*.

[44] L. Gudgeon, P. Moreno-Sanchez, S. Roos, P. McCorry, and A. Gervais, "Sok: Layer-two blockchain protocols," in *Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24*. Springer, pp. 201–226.

[45] G. Malavolta, P. Moreno-Sanchez, A. Kate, and M. Maffei, "Silentwhispers: Enforcing security and privacy in decentralized credit networks," in *24th Annual Network and Distributed System Security Symposium, NDSS*, 2017.

[46] S. Roos, P. Moreno-Sanchez, A. Kate, and I. Goldberg, "Settling payments fast and private: Efficient decentralized routing for path-based transactions," in *25th Annual Network and Distributed System Security Symposium, NDSS*, 2018.

[47] R. Yu, G. Xue, V. T. Kilari, D. Yang, and J. Tang, "Coinexpress: A fast payment routing mechanism in blockchain-based payment channel networks," in *27th International Conference on Computer Communication and Networks, ICCCN 2018*. IEEE, pp. 1–9.

[48] G. Panwar, S. Misra, and R. Vishwanathan, "Blanc: Blockchain-based anonymous and decentralized credit networks," in *Proceedings of the Ninth ACM Conference on Data and Application Security and Privacy*, ser. CODASPY '19. Association for Computing Machinery, p. 339–350.

[49] Y. Zhang and D. Yang, "Robustpay: Robust payment routing protocol in blockchain-based payment channel networks," in *2019 IEEE 27th International Conference on Network Protocols (ICNP)*, pp. 1–4.

[50] P. Wang, H. Xu, X. Jin, and T. Wang, "Flash: Efficient dynamic routing for offchain networks," ser. CoNEXT '19. Association for Computing Machinery, 2019, p. 370–381.

[51] Y. Zhang, D. Yang, and G. Xue, "Cheapay: An optimal algorithm for fee minimization in blockchain-based payment channel networks," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1–6.

[52] L. Eckey, S. Faust, K. Hostáková, and S. Roos, "Splitting payments locally while routing interdimensionally," *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 555, 2020.

[53] S. Lin, J. Zhang, and W. Wu, "Fstr: Funds skewness aware transaction routing for payment channel networks," in *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 464–475.

[54] V. Sivaraman, S. B. Venkatakrishnan, K. Ruan, P. Negi, L. Yang, R. Mittal, G. Fanti, and M. Alizadeh, "High throughput cryptocurrency routing in payment channel networks," in *17th USENIX Symposium on Networked Systems Design and Implementation*, 2020, pp. 777–796.

[55] Q. Gong, C. Zhou, L. Qi, J. Li, J. Zhang, and J. Xu, "Vein: High scalability routing algorithm for blockchain-based payment channel networks," in *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2021, pp. 43–50.

[56] H. Kadry and Y. Gadallah, "A machine learning-based routing technique for off-chain transactions in payment channel networks," in *2021 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pp. 66–73.

[57] X. Zhang, S. Shi, and C. Qian, "Webflow: Scalable and decentralized routing for payment channel networks with high resource utilization," *CoRR*, vol. abs/2109.11665, 2021.

[58] Y. Zhang and D. Yang, "Robustpay+: Robust payment routing with approximation guarantee in blockchain-based payment channel networks," *IEEE/ACM Transactions on Networking*, vol. 29, pp. 1676–1686, 2021.

[59] Y. Chen, Y. Ran, J. Zhou, J. Zhang, and X. Gong, "Mpcn-rp: A routing protocol for blockchain-based multi-charge payment channel networks," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1229–1242, 2022.

[60] H.-J. Hong, S.-Y. Chang, and X. Zhou, "Auto-tune: An efficient autonomous multi-path payment routing algorithm for payment channel networks," *Computer Networks*, vol. 225, p. 109659, 2023.

[61] L. Yang, X. Dong, S. Gao, Q. Qu, X. Zhang, W. Tian, and Y. Shen, "Optimal hub placement and deadlock-free routing for payment channel network scalability," in *2023 IEEE 43rd International Conference on Distributed Computing Systems (ICDCS)*.

[62] K. Kolachala, M. Ababneh, and R. Vishwanathan, "Raced: Routing in payment channel networks using distributed hash tables," in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '24. Association for Computing Machinery, p. 1895–1910.

[63] M. Ababneh, K. Kolachala, and R. Vishwanathan, "Auroch: Auction-based multipath routing for payment channel networks," in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '24. Association for Computing Machinery, p. 1861–1877.

[64] G. Panwar, R. Vishwanathan, G. Torres, and S. Misra, "Sprite: Secure and private routing in payment channel networks," in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*. Association for Computing Machinery, p. 1878–1894.

[65] G. Malavolta, P. Moreno-Sanchez, A. Kate, M. Maffei, and S. Ravi, "Concurrency and privacy with payment-channel networks," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 455–471.

[66] S. Dziembowski, L. Eckey, S. Faust, and D. Malinowski, "Perun: Virtual payment hubs over cryptocurrencies," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, pp. 106–123.

[67] K. Kolachala, E. Simsek, M. Ababneh, and R. Vishwanathan, “Sok: Money laundering in cryptocurrencies,” in *Proceedings of the 16th International Conference on Availability, Reliability and Security*, ser. ARES ’21. Association for Computing Machinery, 2021.

[68] S. Dziembowski, S. Faust, and K. Hostáková, “General state channel networks,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 949–966.

[69] S. Dziembowski, L. Eckey, S. Faust, J. Hesse, and K. Hostáková, “Multi-party virtual state channels,” in *Advances in Cryptology–EUROCRYPT 2019*. Springer, pp. 625–656.

[70] M. Jourenko, M. Larangeira, and K. Tanaka, “Lightweight virtual payment channels,” in *International Conference on Cryptology and Network Security*. Springer, 2020, pp. 365–384.

[71] L. Aumayr, M. Maffei, O. Ersoy, A. Erwig, S. Faust, S. Riahi, K. Hostáková, and P. Moreno-Sanchez, “Bitcoin-compatible virtual channels,” in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 901–918.

[72] A. Kiayias and O. S. T. Litos, “Elmo: Recursive virtual payment channels for bitcoin,” *Cryptology ePrint Archive*, 2021.

[73] L. Aumayr, P. Moreno-Sanchez, A. Kate, and M. Maffei, “Breaking and fixing virtual channels: Domino attack and donner,” in *Network and Distributed System Security (NDSS) Symposium*, 2023.

[74] X. Jia, Z. Yu, J. Shao, R. Lu, G. Wei, and Z. Liu, “Cross-chain virtual payment channels,” *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 3401–3413, 2023.

[75] T. Close and A. Stewart, “Forcemove: an n-party state channel protocol,” *Magmo, White Paper*, 2018.

[76] P. McCorry, S. Bakshi, I. Bentov, S. Meiklejohn, and A. Miller, “Pisa: Arbitration outsourcing for state channels,” in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, 2019, pp. 16–30.

[77] A. Miller, I. Bentov, S. Bakshi, R. Kumaresan, and P. McCorry, “Sprites and state channels: Payment networks that go faster than lightning,” in *International conference on financial cryptography and data security*. Springer, 2019, pp. 508–526.

[78] M. M. Chakravarty, S. Coretti, M. Fitzi, P. Gaži, P. Kant, A. Kiayias, and A. Russell, “Fast isomorphic state channels,” in *Financial Cryptography and Data Security: 25th International Conference, FC 2021, Virtual Event, March 1–5, 2021, Revised Selected Papers, Part II 25*. Springer, 2021, pp. 339–358.

[79] L. Aumayr, O. Ersoy, A. Erwig, S. Faust, K. Hostáková, M. Maffei, P. Moreno-Sanchez, and S. Riahi, “Generalized channels from limited blockchain scripts and adaptor signatures,” in *Advances in Cryptology–ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6–10, 2021, Proceedings, Part II 27*. Springer, 2021, pp. 635–664.

[80] L. Negka, A. Katsika, G. Spathoulas, and V. Plagianakos, “Origami: A flexible state channels design for public blockchain systems,” *arXiv preprint arXiv:2304.10313*, 2023.

[81] M. Green and I. Miers, “Bolt: Anonymous payment channels for decentralized currencies,” in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 473–489.

[82] E. Heilman, L. Alshenibr, F. Baldimtsi, A. Scafuro, and S. Goldberg, “Tumblebit: An untrusted bitcoin-compatible anonymous payment hub,” in *Network and distributed system security symposium*, 2017.

[83] R. Khalil, A. Zamyatin, G. Felley, P. Moreno-Sanchez, and A. Gervais, “Commit-chains: Secure, scalable off-chain payments,” *Cryptology ePrint Archive*, 2018.

[84] J. Lind, O. Naor, I. Eyal, F. Kelbert, E. G. Sirer, and P. Pietzuch, “Teechain: a secure payment network with asynchronous blockchain access,” in *Proceedings of the 27th ACM Symposium on Operating Systems Principles*, 2019, pp. 63–79.

[85] E. Tairi, P. Moreno-Sanchez, and M. Maffei, “A2l: Anonymous atomic locks for scalability in payment channel hubs,” in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 1834–1851.

[86] Z. Ge, J. Gu, C. Wang, Y. Long, X. Xu, and D. Gu, “Accio: Variable-amount, optimized-unlinkable and nizk-free off-chain payments via hubs,” in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS ’23. Association for Computing Machinery, p. 1541–1555.

[87] J. Zhang, Y. Ye, W. Wu, and X. Luo, “Boros: Secure and efficient off-blockchain transactions via payment channel hub,” *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 407–421, 2021.

[88] J. Du, Z. Ge, Y. Long, Z. Liu, S. Sun, X. Xu, and D. Gu, “Mixct: Mixing confidential transactions from homomorphic commitment,” in *European Symposium on Research in Computer Security*. Springer, 2022, pp. 763–769.

[89] J. He, W. Qiu, R. He, S. Zhuo, and W. Jie, “Turbo: A high-performance and secure off-chain payment hub,” in *International Conference on Machine Learning for Cyber Security*. Springer, 2022, pp. 67–75.

[90] X. Qin, S. Pan, A. Mirzaei, Z. Sui, O. Ersoy, A. Sakzad, M. F. Esgin, J. K. Liu, J. Yu, and T. H. Yuen, “Blindhub: Bitcoin-compatible privacy-preserving payment channel hubs supporting variable amounts,” in *2023 IEEE Symposium on Security and Privacy (SP)*, pp. 2462–2480.

[91] G. D. Bissias, A. P. Ozisik, B. N. Levine, and M. Liberatore, “Sybil-resistant mixing for bitcoin,” in *Proceedings of the 13th Workshop on Privacy in the Electronic Society, WPES 2014, Scottsdale, AZ, USA, November 3, 2014*, G. Ahn and A. Datta, Eds. ACM, 2014, pp. 149–158.

[92] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J. A. Kroll, and E. W. Felten, “Mixcoin: Anonymity for bitcoin with accountable mixes,” in *Financial Cryptography and Data Security - 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers*, ser. Lecture Notes in Computer Science, N. Christin and R. Safavi-Naini, Eds., vol. 8437. Springer, 2014, pp. 486–504.

[93] P. Fauzi, S. Meiklejohn, R. Mercer, and C. Orlandi, “Quisquis: A new design for anonymous cryptocurrencies,” in *Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8–12, 2019, Proceedings, Part I 25*. Springer, 2019, pp. 649–678.

[94] E. Heilman, F. Baldimtsi, and S. Goldberg, “Blindly signed contracts: Anonymous on-blockchain and off-blockchain bitcoin transactions,” in *Financial Cryptography and Data Security - FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers*, ser. Lecture Notes in Computer Science, J. Clark, S. Meiklejohn, P. Y. A. Ryan, D. S. Wallach, M. Brenner, and K. Rohloff, Eds., vol. 9604. Springer, 2016, pp. 43–60.

[95] P. Moreno-Sanchez, T. Ruffing, and A. Kate, “Pathshuffle: Credit mixing and anonymous payments for ripple,” *Proc. Priv. Enhancing Technol.*, vol. 2017, no. 3, p. 110, 2017.

[96] T. Ruffing, P. Moreno-Sanchez, and A. Kate, “Coinshuffle: Practical decentralized coin mixing for bitcoin,” in *Computer Security - ESORICS 2014 - 19th European Symposium on Research in Computer Security, Wroclaw, Poland, September 7-11, 2014. Proceedings, Part II*, ser. Lecture Notes in Computer Science, M. Kutylowski and J. Vaidya, Eds., vol. 8713. Springer, 2014, pp. 345–364.

[97] T. Ruffing and P. Moreno-Sanchez, “Valueshuffle: Mixing confidential transactions for comprehensive transaction privacy in bitcoin,” in *Financial Cryptography and Data Security - FC 2017 International Workshops, WAHC, BITCOIN, VOTING, WTSC, and TA, Sliema, Malta, April 7, 2017, Revised Selected Papers*, ser. Lecture Notes in Computer Science, M. Brenner, K. Rohloff, J. Bonneau, A. Miller, P. Y. A. Ryan, V. Teague, A. Bracciali, M. Sala, F. Pintore, and M. Jakobsson, Eds., vol. 10323. Springer, 2017, pp. 133–154.

[98] M. Tran, L. Luu, M. S. Kang, I. Bentov, and P. Saxena, “Obscuro: A bitcoin mixer using trusted execution environments,” in *Proceedings of the 34th Annual Computer Security Applications Conference, ACSAC 2018, San Juan, PR, USA, December 03-07, 2018*. ACM, 2018, pp. 692–701.

[99] L. Valenta and B. Rowan, “Blindcoin: Blinded, accountable mixes for bitcoin,” in *Financial Cryptography and Data Security - FC 2015 International Workshops, BITCOIN, WAHC, and Wearable, San Juan, Puerto Rico, January 30, 2015, Revised Selected Papers*, ser. Lecture Notes in Computer Science, M. Brenner, N. Christin, B. Johnson, and K. Rohloff, Eds., vol. 8976. Springer, 2015, pp. 112–126.

[100] J. H. Ziegeldorf, F. Grossmann, M. Henze, N. Inden, and K. Wehrle, “Coinparty: Secure multi-party mixing of bitcoins,” in *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy, CODASPY 2015, San Antonio, TX, USA, March 2-4, 2015*, J. Park and A. C. Squicciarini, Eds. ACM, 2015, pp. 75–86.

[101] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage, “A fistful of bitcoins: characterizing payments among men with no names,” in *Proceedings of the 2013 conference on Internet measurement conference*, 2013, pp. 127–140.

[102] G. Maxwell, “Coinswap: transaction graph disjoint trustless trading (2013),” *URL: https://bitcointalk. org/index. php*, 2013.

[103] R. Khalil and A. Gervais, “Revive: Rebalancing off-blockchain payment networks,” ser. CCS ’17. Association for Computing Machinery, 2017, p. 439–453.

[104] L. M. Subramanian, G. Eswaraiah, and R. Vishwanathan, “Rebalancing in acyclic payment networks,” in *2019 17th International Conference on Privacy, Security and Trust (PST)*. IEEE, 2019, pp. 1–5.

[105] N. Awathare, V. J. Ribeiro, U. Bellur *et al.*, “Rebal: channel balancing for payment channel networks,” in *2021 29th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE, 2021, pp. 1–8.

[106] Z. Avarikioti, K. Pietrzak, I. Salem, S. Schmid, S. Tiwari, and M. Yeo, “Hide & seek: Privacy-preserving rebalancing on payment channel networks,” in *International Conference on Financial Cryptography and Data Security*. Springer, 2022, pp. 358–373.

[107] Z. Hong, S. Guo, R. Zhang, P. Li, Y. Zhan, and W. Chen, “Cycle: Sustainable off-chain payment channel network with asynchronous rebalancing,” in *2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2022, pp. 41–53.

[108] Z. Ge, Y. Zhang, Y. Long, and D. Gu, “Shaduf: Non-cycle and privacy-preserving payment channel rebalancing,” *Cryptology ePrint Archive*, 2022.

[109] Z. Avarikioti, S. Schmid, and S. Tiwari, “Musketeer: Incentive-compatible rebalancing for payment channel networks,” *Cryptology ePrint Archive*, 2023.

[110] W. Chen, X. Qiu, Z. Cai, B. Tang, L. Du, and Z. Zheng, “Graph neural network-enhanced reinforcement learning for payment channel rebalancing,” *IEEE Transactions on Mobile Computing*, vol. 23, no. 6, pp. 7066–7083, 2024.

[111] ACINQ, “Trampoline fee insufficient,” https://bit.ly/4fvJN0A.