Scaling Blockchain transactions using privacy preserving Payment Channel Network

Subhra Mazumdar

Cryptology and Security Research Unit
Indian Statistical Institute Kolkata, India

Email : subhra.mazumdar1993@gmail.com

1 Overview

Cryptocurrencies like Bitcoin [24] have recently become popular alternative method of payments, playing a major role in the economy. Blockchain forms the backbone of such decentralized network, which not only allows transacting parties to remain pseudonymous, but also guarantees reliability and security. The records stored in this distributed ledger are immutable and can be verified by anyone in the network. It is replicated across users who use consensus algorithms like Proof-of-Work [24], [25], [4], Proof-of-Stake [17], [18]) for agreement on state change in the ledger. But such consensus algorithms are quite computation and resource intensive, slowing down the performance and reducing scalability [5], [26]. Thus scaling blockchain transactions is an important concern which needs to be addressed, without compromising on the privacy.

Different blockchain scaling solutions like alternative blockchain consensus architectures, sharding, side-chains have been explored, but Layer two protocols stood out as a practically deployable answer to the scalability issue. With the advent of Layer-two protocols, built on top of (layer-one) blockchains, transactions are settled off-chain and blockchain is used only to settle dispute arising out misbehavior by counterparties involved. In the process transactions get completed in sub-seconds, allowing blockchains to scale.

Though there exist several Layer two protocols like Payment Channel, State Channel (generalization of Payment Channel) and Commit Chain, our main focus is on Payment Channel Network. In this proposal, several problems associated with Payment Channel Network like developing efficient routing algorithm for transfer of funds from payer to payee, developing payment algorithm ensuring fairness with minimum collateral at stake. We discuss possible solutions mitigating the problem stated with some interesting results. As part of our future work, we also discuss about *Atomic Cross Chain Swap* [14] using Payment Channel Network and fair exchange of digital asset using *Generalized State Channel Network [7].*

2 Background & Related Work

Payment Channel, used for off-chain transactions in cryptocurrencies enhances scalability, without recording every payment on blockchain. Any two users, with mutual consent, can open a payment channel by locking their funds. Users can perform several off chain payments between each other without recording the same on blockchain. This is done by locally agreeing on the new deposit balance, enforced cryptographically by smart contracts [26], key based locking [22] etc. If one party wants to close the payment channel, it broadcasts the transaction on blockchain with the final balance. If a counterparty tries to cheat by claiming payment for an old transaction, it will be penalized and lose out all the funds it had locked initially. Opening of new payment channel between parties which are not connected directly has its overhead where funds get locked for a substantial amount of time. This can be avoided by leveraging on the set of existing payment channels for executing a transaction, proving beneficial in terms of resource utilization. These set of payment channels form the *Payment Channel Network* or PCN [26]. Several P2P path-based transaction networks such as such as Lightning Network for Bitcoin [26], Raiden Network for Ethereum [2], SilentWhispers [20], InterLedger [33], Atomic-swap [1], TeeChain [19] etc. have been developed over the years. Perun [6] proposes a more efficient network structure which is built around payment hubs. An extension of such networks, State Channel Network [7], not only supports off chain payment but allows execution of complex smart contract.

The two main algorithms which assures multiple guaranteed off-chain payments in the network are *Routing* and *Payment*. The first one is responsible for finding set of paths between payer and payee having sufficient capacity for payment. The latter ensures settlement of funds on those paths without allowing an attacker to steal funds from honest intermediaries. The major challenge in designing any protocol for PCN is to ensure privacy of payer and payee and hiding the payment value transferred. No party, other than the payer and payee, should get any information about the transaction.

Several routing algorithms proposed for payment channel networks are as follows : Canal [34] - uses a centralized server for computing the path, Flare [27] - requires intermediate nodes to inform source node about their residual capacity, SilentWhispers [20] - a distributed PBT network without using any public ledger guaranteeing unlinkability of transaction, SpeedyMurmur [31] - a privacy preserving embedded based routing, extending Voute [30], depending on presence of landmark nodes. Elias et al. [29] proposed an extended push relabel for finding payment flow in the payment network. Later, a distributed approach for PCN routing, CoinExpress [36], was proposed for finding routes that fulfill payment with higher success ratio. A routing algorithm based on swarm intelligence, ant colony optimization [11] has been explored. Hoenisch et al. [16] proposed an adaptation of an Ad-hoc On-demand Distance Vector (AODV)-based routing algorithm which supports different cryptocurrencies allowing transactions across multiple blockchains.

Privacy guarantee offered by PCN and its challenges has been extensively discussed in [3], [15], [12]. A payment along a path must be atomic - either it succeeds fully or it is aborted. Partial satisfaction of transaction may lead to loss of funds. As a solution, Hashed Timelock Contract [26] was proposed for Lightning Networks. It is compatible with Bitcoin script but has its own demerits. Bolt [10] states about a hub based payment construction retaining payment anonymity but it is restricted to just two-hop payment. TumbleBit [13] follows a similar approach assuring payer/payee privacy but suffers from the same shortcoming. Sprite [23], a variant of payment channel network, does not focus on privacy and concurrency issue. Malavolta et al. [21] had proposed a secure version of payment for multi-hop path based on zero-knowledge proof system `ZK-Boo` [9]. This solution uses HTLC as the backbone, working on one path at a time. *Multi-hop Locks*, defined in [22], are compatible with vast majority of cryptocurrencies. It is generic as well as interoperable, supporting both script and scriptless support

for PCN. *However drawback of these payment protocols is that even if the transaction is split into several microtransaction across several paths, locking of funds and subsequent payment is done on per path basis, which is quite inefficient.* Atomic multi-path payment in Bitcoin payment network as well as in cross-chain payment has been studied in [8], [14] but there is substantial leak of information violating transaction-level privacy. Recently, an efficient privacy preserving payment protocol based on Chameleon Hash Function [35] was proposed which is devoid of complex key management and zero knowledge proof. But in this protocol, honest intermediaries lying on a path are susceptible to key exposure attack.

While PCNs increases the transaction throughput by processing payments off-chain, they unfortunately require high collateral (i.e., they lock coins for a non-constant time along the payment path) and are restricted to payments in a path from sender to receiver. These issues have severe consequences in practice. One major attack causing maximum collateral damage is Griefing Attack, which has been defined in [8], [28]. It was conjectured previously that mitigating damage caused by locking of high collateral challenge cannot be solved without modifications to the Bitcoin script [23]. However the paper of Egger et al.[8] refutes the conjecture by providing a solution for Bitcoin-compatible PCNs and reducing the collateral cost. However their solution violates transaction level privacy. Any intermediate party lying on the path connecting the sender and receiver knows the identity of payer and payee as well as the fund being transferred. This makes the network vulnerable to a host of other attacks where a malicious node in the path may leak such sensitive information to the outside world. Based on the several such limitations, we formulate our problem and discuss possible solutions in the next section.

3 Problem Statement and Ongoing Work

We have identified the following problems in PCNs and have proposed solution for the same :

- It is not always possible to route the transaction across a single path as the value may be quite high compared to minimum capacity of the designated path. Hence it is better to find set of paths such that the total amount to be transferred is split across each such path. We define the problem as follows -

  Problem 1 *Given a payment channel network $G(V, E)$, a transaction request (s, r, val) for a source-sink pair (s, r), the objective is to find a set of paths $p_1, p_2, \ldots, p_m$ for transferring the fund from s to r such that p_1 transfers val_1, p_2 transfers $val_2, \ldots, p_m$ transfers $val_m : val = \Sigma_{i=1}^{m} val_i$ without violating transaction level privacy i.e. neither the sender nor the receiver of a particular transaction must be identified as well as hiding the actual transaction value from intermediate parties.*

  Proposed Solution (in submission) - We have proposed a privacy preserving distributed routing algorithm, *HushRelay*. It was implemented and its performance was compared with Speedy-Murmur [31] in terms of *success ratio* and *time taken to route (TTR)* a payment. Testing was done on real instances of Ripple like Network and Lightning Network and it was observed that *HushRelay* attains a success ratio of 1 in both the cases. However *SpeedyMurmur* attained a maximum success ratio of 0.9815 and 0.907 respectively, when number of landmarks is 6. The time taken to execute the routing algorithm in Ripple like Network and Lightning Network are 2.4s and 0.15189s for *HushRelay* but it takes 4.736s and 1.937s for *SpeedyMurmur*.

- Several privacy preserving payment protocols [21], [22] have been implemented which works perfectly for single route. But this is not always possible as the transaction value may be quite high compared to minimum capacity of a payment channel on a single path. Hence it is better to find set of paths such that the total amount to be transferred is split across each such path. Considering this mode of routing, we define the problem as follows -

  Problem 2 *Given a payment channel network $G(V, E)$, a transaction request (s, r, val) for a source-sink pair (s, r) uses set of paths $p_1, p_2, \ldots, p_m$ in order to transfer the fund such that p_1 transfers val_1, p_2 transfers $val_2, \ldots$, p_m transfers $val_m : val = \Sigma_{i=1}^{m} val_i$. The objective is to design an efficient payment protocol ensuring atomic payment across all the paths without violating transaction level privacy i.e. neither the sender nor the receiver of a particular transaction must be identified as well as hiding the actual transaction value from intermediate parties.*

  Proposed Solution (in submission) - We have proposed a privacy preserving atomic off-chain payment protocol, *SplitPayLock*, for secure transaction across multiple paths in payment channel network. The setup phase of the payment protocol is devoid of zero knowledge proof, complex key management process and involves less computation. The scheme was implemented on real instances - Ripple-like Network [21] and Lightning Network[32]. *SplitPayLock* takes around 10 s to complete the payment with communication overhead of less than 1.5 MB compared to Multihop HTLC, which takes around 65 s to complete the protocol and incurs a communication overhead of 26 MB. In an instance of Lightning Network, it takes around 485 ms and a communication overhead of 0.16 MB as compared to 10.6 s and communication overhead of 24 MB by Multihop HTLC.

- **Problem 3** *Given a payment request (s, r, v) in a bidirectional payment channel network $G(V, E)$, the sender s probes for a feasible route for initiating the payment. Assuming the path from s to r be P, where P comprises n nodes such that $U_0 = s, U_1, \ldots, U_{n-1} = r$, where each pair of adjacent vertices $(U_i, U_{i+1}), \forall i \in [0, n-2]$ forms a payment channel have funds locked. While transferring fund, each channel locks an amount val' which is the required amount val plus the additional fee charged by each intermediate vertex as processing fee. Existing payment protocol requires each channel to lock their fund for a certain timeout period. If the path length is $n-1$ and incremental timeout period is Δ then each channel (U_i, U_{i+1}) locks fund for period of $(n-i-1)\Delta, \forall i \in [0, n-2]$. Plus each channel locks a fund of value approximately val. Hence the collateral in terms of timelocked fund across path P is $\mathcal{O}(n^2 \Delta val)$. If any of the party acts maliciously and deviate from the payment protocol for causing a failure in payment, then this is the collateral damage incurred. This is known as* Griefing Attack. *We intend to propose an atomic multi channel-update protocol for payment channel network in general which will reduce the collateral cost to $\mathcal{O}(n\Delta val)$. By atomic channel update we mean that even if one party misbehaves while transfers fund then state of channel remains unchanged and transaction is deemed as invalid.*

  Proposed Solution - Yet to come up with a solution as the problem is quite non trivial.

4 Future Direction

As a part of our future work, we would like to explore cross chain trading [1] in lightning network using the principles of Atomic Cross-chain Swap. Apart from that, we will be exploring State Channel

Network which is a generalized version of Payment Channel Network. State Channel Network supports arbitrary contract terms and condition. Hence it can be used for faster digital asset exchange guaranteeing fairness to both payer and payee. The challenging part is that even though we can split data into chunks like we did for payment, there still lies the risk of data loss or corruption of the chunk. To view the entire content, all the chunks need to be recombined properly and a protocol must provide proof of correctness for each chunk of data transfer.

References

[1] "Atomic cross-chain trading." `https://en.bitcoin.it/wiki/ Atomic_cross-chain_trading`, July 2017.

[2] "Raiden network," `http://raiden.network/`, July 2017.

[3] K. Atlas, "The inevitability of privacy in lightning networks, 2017," *URL https://www. kristovatlas. com/the-inevitability-of-privacy-in-lightning-networks/.[Online.*

[4] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis, "Consensus in the age of blockchains," *arXiv preprint arXiv:1711.03936*, 2017.

[5] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer *et al.*, "On scaling decentralized blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 106–125.

[6] S. Dziembowski, L. Eckey, S. Faust, and D. Malinowski, "Perun: Virtual payment hubs over cryptographic currencies," IACR Cryptology ePrint Archive 2017, Tech. Rep., 2017.

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

[8] C. Egger, P. Moreno-Sanchez, and M. Maffei, "Atomic multi-channel updates with constant collateral in bitcoin-compatible payment-channel networks," in *26th ACM Conference on Computer and Communications Security*. ACM, 2019.

[9] I. Giacomelli, J. Madsen, and C. Orlandi, "Zkboo: Faster zero-knowledge for boolean circuits." in *USENIX Security Symposium*, 2016, pp. 1069–1083.

[10] 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*. ACM, 2017, pp. 473–489.

[11] C. Grunspan and R. Pérez-Marco, "Ant routing algorithm for the lightning network," *arXiv preprint arXiv:1807.00151*, 2018.

[12] L. Gudgeon, P. Moreno-Sanchez, S. Roos, P. McCorry, and A. Gervais, "Sok: Off the chain transactions," *IACR Cryptology ePrint Archive*, vol. 2019, p. 360, 2019.

[13] 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.

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

[15] J. Herrera-Joancomartí and C. Pérez-Solà, "Privacy in bitcoin transactions: new challenges from blockchain scalability solutions," in *Modeling Decisions for Artificial Intelligence*. Springer, 2016, pp. 26–44.

[16] P. Hoenisch and I. Weber, "Aodv–based routing for payment channel networks," in *International Conference on Blockchain*. Springer, 2018, pp. 107–124.

[17] S. King and S. Nadal, "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake," *self-published paper, August*, vol. 19, 2012.

[18] W. Li, S. Andreina, J.-M. Bohli, and G. Karame, "Securing proof-of-stake blockchain protocols," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, 2017, pp. 297–315.

[19] J. Lind, I. Eyal, F. Kelbert, O. Naor, P. Pietzuch, and E. G. Sirer, "Teechain: Scalable blockchain payments using trusted execution environments," *arXiv preprint arXiv:1707.05454*, 2017.

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

[21] 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*. ACM, 2017, pp. 455–471.

[22] G. Malavolta, P. Moreno-Sanchez, C. Schneidewind, A. Kate, and M. Maffei, "Multi-hop locks for secure, privacy-preserving and interoperable payment-channel networks," in *Network and Distributed System Security Symposium*, 2019.

[23] A. Miller, I. Bentov, R. Kumaresan, and P. McCorry, "Sprites: Payment channels that go faster than lightning," in *Twenty-Third International Conference on Financial Cryptography and Data Security 2019*, 2019.

[24] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[25] K. J. O'Dwyer and D. Malone, "Bitcoin mining and its energy footprint," 2014.

[26] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," *See https://lightning. network/lightning-network-paper. pdf*, 2016.

[27] P. Prihodko, S. Zhigulin, M. Sahno, A. Ostrovskiy, and O. Osuntokun, "Flare: An approach to routing in lightning network," *White Paper (bitfury.com/content/5-white-papers-research/whitepaper
_flare_an_approach_to_routing_in_lightning_n etwork_7_7_2016. pdf)*, 2016.

[28] Rohrer, Elias, J. Malliaris, and F. Tschorsch, "Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks," in *Proceedings of the IEEE European Symposium on Security and Privacy Workshops*. IEEE, 2019, pp. 347–356.

[29] E. Rohrer, J.-F. Laß, and F. Tschorsch, "Towards a concurrent and distributed route selection for payment channel networks," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, 2017, pp. 411–419.

[30] S. Roos, M. Beck, and T. Strufe, "Voute-virtual overlays using tree embeddings," *arXiv preprint arXiv:1601.06119*, 2016.

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

[32] I. A. Seres, L. Gulyás, D. A. Nagy, and P. Burcsi, "Topological analysis of bitcoin's lightning network," *arXiv preprint arXiv:1901.04972*, 2019.

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

[34] B. Viswanath, M. Mondal, K. P. Gummadi, A. Mislove, and A. Post, "Canal: Scaling social network-based sybil tolerance schemes," in *Proceedings of the 7th ACM european conference on Computer Systems*. ACM, 2012, pp. 309–322.

[35] B. Yu, S. K. Kermanshahi, A. Sakzad, and S. Nepal, "Chameleon hash time-lock contract for privacy preserving payment channel networks," in *International Conference on Provable Security*. Springer, 2019, pp. 303–318.

[36] 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 *2018 27th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2018, pp. 1–9.