A Platform for Analyzing Payment Channel Networks in Supporting Real-world Payment Patterns

Marco Benedetti, Giuseppe Galano, Sara Giammusso, Matteo Nardelli
Banca d'Italia
{first name}.{last name}@bancaditalia.it, giuseppe.galano2@bancaditalia.it

ABSTRACT

Payment channel networks (PCNs) promise to overcome the scalability issues of blockchains by enabling fast, secure, and confidential transactions. Although the literature analyzes different aspects of PCNs, thus far it is not clear whether they can successfully handle real-world payment volumes, and if they can even go beyond that. In this paper, we outline our research activity toward a systematic investigation of PCN topologies, aiming to analyze and optimize the trade-offs between locked liquidity in channels, representing a cost, and rates of successfully routed payments.

Keywords

Lightning Network, Bitcoin, Simulation, Scalability.

INTRODUCTION

Background

The limited scalability of blockchain-based cryptocurrencies has generated an increasing interest in off-chain solutions, such as *payment channel networks* (PCNs). A PCN operates on top of a blockchain, achieving scalability with faster, cheaper, and higher volumes of transactions. Participants (nodes) can establish payment channels directly between them, by depositing a certain balance into the channel (i.e., capacity), which remains frozen throughout the entire channel's lifespan. Not all participants are connected; nonetheless, they can execute payments by using others as relays, essentially forming a network. Sending and receiving payments causes a shift of the balance to one side of the channel. One of the most famous examples of PCN is the Lightning Network (LN), a permissionless network layered on top of Bitcoin.

Motivation

As pointed out in [5], the PCN "ever-shifting balance sheet" feature is very similar to the concept of real-time gross settlement (RTGS)[1] and continuous linked settlement (CLS)[2] A significant difference though is that PCNs provide *instantaneous*, *peer-to-peer*, and *end-to-end encrypted* payments, enabling digital currency to flow with features comparable to those of physical cash (e.g., in terms of privacy). These interesting *"cash-like"* features raise the following questions: *Are PCNs scalable in terms of transactions per second (TPS)? Can they successfully handle the volumes of payments currently performed in national currencies, e.g., Euro, US Dollar?*

Problem statement

PCNs require exploiting the right trade-off between channel liquidity and payment success rate. Payments can succeed only if the routing algorithm finds a path of channels connecting the sender and receiver, where each channel has sufficient balance to complete the transaction. Moreover, for the entire duration of a single transaction, the *Hash-Time-Locked Contract* (HTLC) locks the balance needed to support that transaction, until the "secret passcode" is revealed. As a result, the bigger the capacity buffer, the higher the number of transactions that can be simultaneously supported by a channel, and thus the higher the probability of payment success. Whilst infinite capacity channels may be desired, liquidity generally involves costs (e.g., interest charges, opportunity costs). Although there are apparently no restrictions to the topology of a PCN, the liquidity needed for the channels allocation, and its associated cost, may push the network to a hub-and-spoke distribution of nodes and channels: a few big nodes, called *Liquidity Service Providers* (LSPs), incentivized by relay fees, open channels towards end users so to increase the users' inbound capacity and their reachability in the network.

In this paper, we outline our journey toward the investigation of PCNs. Using simulation, we want to analyze the efficiency of hub-and-spoke topologies, aiming to understand whether and how their *liquidity needs* can support volumes of payments comparable with those of *national currencies*.

RELATED WORK

Different aspects of PCNs have been analyzed, including incentives, routing, rebalancing, confidentiality, as well as their topological properties and node attachment strategies. As pointed out in [5], assuming a fully private setting as in LN, the two main challenges in studying the network aspects of PCNs are (a) the lack of knowledge of channel balances and (b) the impossibility to measure the payment success rate because the transaction's outcome is visible only to the involved nodes in the path. As a result, simulations are used in many studies of PCNs. Lange et al. [4] analyze the impact of different attachment strategies on the trade-off between efficiency and decentralization of LN, under the assumption of three different transactions volumes, thus addressing the challenge (b) by simulating transactions of fixed amounts. Cordi [3] overcomes challenge (b) by simulating transactions from a partner bank database containing credit card users' payments. Finally, Beres et al. [1] evaluate the economic viability of transaction fee revenues in the LN, simulating transactions

[1]RTGS refers to national payment systems generally employed for large-value inter-bank fund transfers.
[2]CLS is an international multi-currency settlement system for financial exchange (FX) transactions.

DLT23: 5th Distributed Ledger Technology Workshop, May 25–26, 2023, Bologna, IT

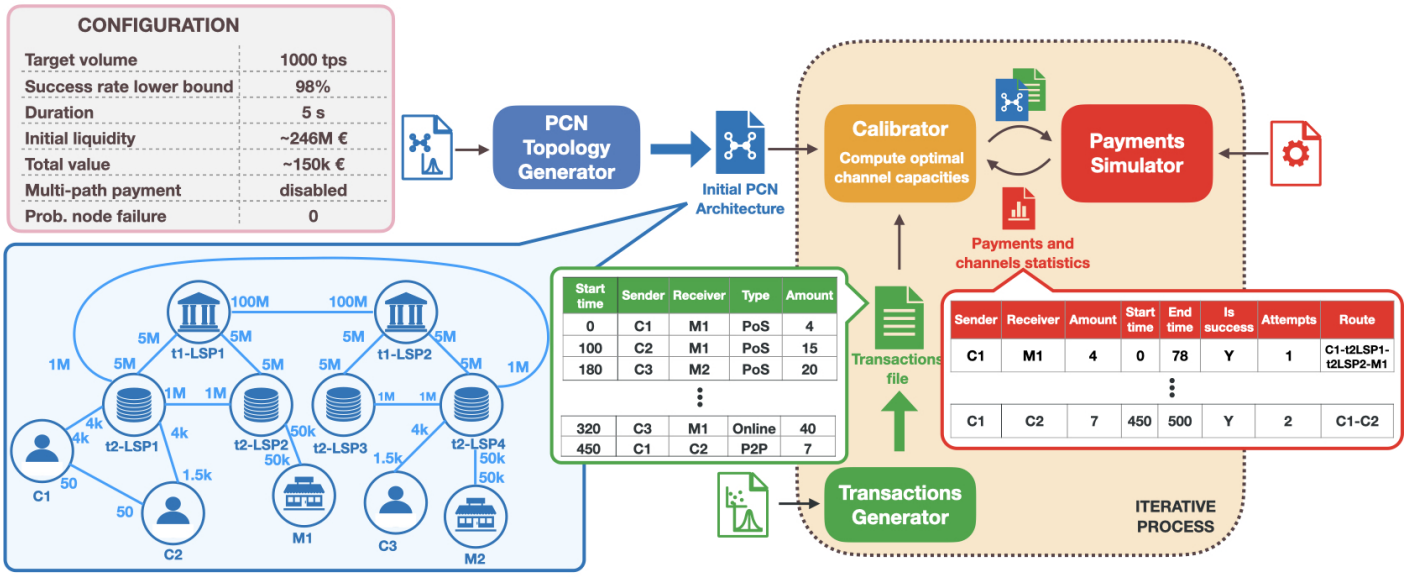

Figure 1. The system diagram includes four different components: (1) a PCN topology generator (blue), (2) a transactions generator (green), (3) a payments simulator (red) and (4) a calibrator (orange).

generated under assumptions based on certain blog posts of LN node owners and concluding that participation was economically irrational for the majority of large routing nodes at that time.

To the best of our knowledge, none of the previous works analyzed the PCN locked liquidity–payments trade-off.

RESEARCH QUESTIONS

The research questions that motivate our work are as follows.

RQ1. What would be the required LPS liquidity to support a given target of transaction/seconds with lower bounds on payments success rate?

RQ2. How would some liquidity optimization techniques (e.g., multipath payments) impact the liquidity needs and payments success rate?

RQ3. What would be the impact of node failures on payment success rate?

RQ4. Given a PCN topology and the total volume of payments, how does changing payment load distribution impact on payment success rate?

RQ5. What kind of privacy challenges would such an almost-fixed topology need to consider?

RESEARCH APPROACH

To address these research questions, we design a system enabling flexible investigation of PCN behavior. As shown in Fig. 1, our system requires the development of four main components: (1) a PCN topology generator, (2) a transactions generator, (3) a payments simulator, and (4) a calibrator.

To answer RQ1, we build a 2-layered hub-and-spoke *PCN topology generator* that considers three types of nodes: (1) *t1-LSP*, a large tier-1 LSP that provides liquidity to multiple lower-layer LSPs; (2) *t2-LSP*, which participates in the second layer and opens channels toward multiple end-users; (3) *end-user*, representing either a merchant or a regular user. The PCN topology generator assumes different models for each internal subnetwork.

The *payment generator* creates a load of end-to-end transactions among end users. The amount and type (i.e., PoS, P2P, or online) of transactions follow the statistics provided by the 2022 ECB SPACE Study on payment attitudes [6].

Payments are then simulated using our extension of CLoTH [2], a *PCN simulator* that mimics the routing and HTLC mechanics used in LNs. It provides performance measures, including payment success rate and average payment time. Our extension of CLoTH exposes a larger number of channel-related metrics to support our analysis.

These three components interact with a *calibrator*, which aims to optimize the PCN by identifying the minimum channels' liquidity that satisfies a given lower bound payment success rate. Once the balances are optimized to reach the target performances in terms of payments success rate and transactions per second, the required total system liquidity can be analyzed.

With such a system in place, RQ2 and RQ3 can be addressed by enabling additional CLoTH features: multi-path payment and node failures. To answer RQ4, we plan to use our simulator to quantitatively investigate different payment load configurations. On the other hand, answering RQ5 is more tricky as it first requires a deeper literature review, and then a formal investigation of leaked information in fixed topologies.

Overall, we do believe that these results can enrich the world revolving around PCNs. Notably, we would provide a better understanding of PCN scalability. Also, our studies aim to shed light on the feasibility of using a PCN as a possible retail CBDC implementation, where central banks and commercial banks could play the role of LSPs.

REFERENCES

[1] Ferenc Beres, Istvan Andras Seres, and Andras A. Benczur. 2019. A Cryptoeconomic Traffic Analysis of Bitcoin's Lightning Network. (2019). DOI: http://dx.doi.org/10.48550/ARXIV.1911.09432

[2] Marco Conoscenti, Antonio Vetrò, and Juan Carlos De Martin. 2021. CLoTH: A Lightning Network Simulator, Vol. 15. SoftwareX, 100717. DOI: http://dx.doi.org/10.1016/j.softx.2021.100717

[3] Christopher Neal Cordi. 2017. Simulating high-throughput cryptocurrency payment channel networks. (2017). https://hdl.handle.net/2142/99319

[4] Kimberly Lange, Elias Rohrer, and Florian Tschorsch. 2021. On the Impact of Attachment Strategies for Payment Channel Networks. In *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 1–9. DOI: http://dx.doi.org/10.1109/ICBC51069.2021.9461104

[5] Nikolaos Papadis and Leandros Tassiulas. 2020. Blockchain-Based Payment Channel Networks: Challenges and Recent Advances. *IEEE Access* 8 (2020), 227596–227609. DOI: http://dx.doi.org/10.1109/ACCESS.2020.3046020

[6] ECB Surveys. 2022. Study on the payment attitudes of consumers in the euro area (SPACE). (2022). https://www.ecb.europa.eu/stats/ecb_surveys/space/html/ecb.spacereport202212~783ffdf46e.en.html