Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Lightning Network Economics: Topology

Paolo Guasoni, Gur Huberman, Clara Shikhelman

To cite this article:
Paolo Guasoni, Gur Huberman, Clara Shikhelman (2025) Lightning Network Economics: Topology. Management Science 71(7):5477-5490. https://doi.org/10.1287/mnsc.2023.03872

https://pubsonline.informs.org/journal/mnsc

MANAGEMENT SCIENCE
Vol. 71, No. 7, July 2025, pp. 5477–5490
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

Lightning Network Economics: Topology

Paolo Guasoni,[a,b,*] Gur Huberman,[c] Clara Shikhelman[d]

[a] School of Mathematical Sciences, Dublin City University, D09 W6Y4 Dublin, Ireland; [b] Dipartimento di Statistica, Università di Bologna, 40126 Bologna, Italy; [c] Columbia Business School, New York, New York 10027; [d] Chaincode Labs, New York, New York 10017

*Corresponding author

Contact: paolo@guasoni.it, https://orcid.org/0000-0002-8562-3658 (PG); gh16@gsb.columbia.edu, https://orcid.org/0009-0002-9077-436X (GH); clara.shikhelman@gmail.com, https://orcid.org/0000-0002-0587-7181 (CS)

Received: November 26, 2023
Revised: July 18, 2024
Accepted: July 24, 2024
Published Online in Articles in Advance: October 7, 2024

https://doi.org/10.1287/mnsc.2023.03872

Abstract. By design, the Bitcoin protocol has a low throughput. The Lightning Network (LN) is a layer-two solution built to increase throughput by cryptographically securing commitments to transactions and only occasionally converting cumulative balances into on-chain transactions. LN channels enable payments between nodes connected by a path of channels. The payment flow through a channel determines its cost. Different channel topologies can support the same underlying flows but impose different costs. This paper obtains necessary conditions for cost-minimizing topologies by identifying local cost-reducing strategies. The first local strategy entails repositioning of channels. The second entails adding hubs to handle the flows of groups of nodes. The paper also evaluates the efficiency of a global configuration, obtaining bounds on the minimum cost topology and showing the unusual circumstances in which the cost minimal structure is a hub that connects to all other nodes.

History: Accepted by Joshua Gans, business strategy.

Funding: This work was partially supported by Science Foundation Ireland [Grants 16/IA/4443, 16/SPP/3347], Columbia-International Business Machines Center for Blockchain and Data Transparency, Chaire Fintech at University Paris Dauphine-Paris Sciences et Lettres, Algorand Foundation, and Simons Institute.

Keywords: payments • Lightning Network • channels • topology • cost minimization

1. Introduction

As originally conceived, the Bitcoin payment system records every transaction on its blockchain. This design feature, coupled with the blockchain's limited capacity, imposes a ceiling on the throughput of the Bitcoin payment system. That ceiling amounts to fewer than 10 transactions per second—a low number in comparison with other payment systems, such as Visa. However, that limited throughput can be expanded, even substantially so, by payers committing to transactions off-chain and only occasionally posting these commitments on-chain. The Lightning Network (LN) is such a device (Poon and Dryja 2015).

The Lightning Network consists of nodes and channels that connect pairs of nodes. Each channel is associated with a net payment flow through it. Two nodes are directly connected if there is a channel between them and are connected if there is a path of channels that enables flows between the nodes. Thus, any node on an LN can pay any other node in the same connected component of the LN. The LN is the leading example of a class of layer-two solutions called payment channel networks (PCNs). This paper's results apply also to other PCNs.

The cost of a channel in the LN arises from locking funds in the channel and from the fees that occasionally need to be paid to rebalance it.[1] At the leading order, the cost of a channel is an increasing function of the average net flow.[2] In particular, in a symmetric channel, the flows offset each other, and the overall cost is negligible in comparison with a channel with nonzero net flow. Thus, at the leading order, the cost of a channel between Alice and Bob is approximately the same, whether (i) Alice, on average, sends Bob one unit annually or (ii) Alice, on average, receives one unit from Bob annually or (iii) Alice, on average, sends Bob 11 units annually and Bob, on average, sends her 10 units annually. The cost-minimizing network is always connected because symmetric channels, which have negligible costs, can be added to connect all components as needed. (Guasoni et al. (2024) offer the details.)

The LN is designed to support a given set of flows between a given set of nodes. In principle, these flows could all be posted on the blockchain, but directing them through the LN is less costly. Starting with an arbitrary LN, the paper considers cost-reducing modifications of that LN.

The present paper identifies channel configurations that are excessively costly and offers procedures to replace them with less costly configurations. The first part of the paper shows that a cost-reducing reconfiguration is available if there are two distinct channels attached to the same node, which is the origin in one of the channels and the destination in the other. Direct corollaries of this result are (i) the presence of a cycle with an odd number of channels implies the availability of a cost-reducing channel reconfiguration and (ii) a cost-minimizing LN is always a bipartite graph. The common intuition of both findings is that, in a cost-minimizing configuration, all the flow should be from net payers to net payees, avoiding intermediate steps.

The second part of the paper considers adding a new node to the network, connecting a set of existing nodes to that new node and deleting all the other preexisting channels connecting these nodes to each other. In this new star-like configuration, all flows among the set of nodes go through the star's center, which is the newly added node. The paper's second part articulates (i) the conditions that the set of nodes needs to meet for the reconfiguration to be cost-reducing; (ii) a polynomial-time algorithm to find such a set of nodes in a given LN; (iii) a proof that such a set must exist if the average number of channels involving each node is large enough; (iv) a proof that such a set is very likely to exist if the network flows are chosen randomly, according to a distribution whose tails are not thicker than a power law.

The third and last part of the paper establishes bounds on the minimum possible cost of an LN that supports a given set of flows. The upper bound is at most twice as large as the lower bound. A star, in which a single hub connects all nodes, in general, is not the least costly topology, but is no more than twice as costly. Finding the exact cost-minimizing topology is an NP-complete problem in general. However, the concluding result describes a quadratic-complexity algorithm to construct a two-factor approximation of the minimum, which is robust to the improvements identified in the paper.

The paper's results consider the cost of LN topology from the viewpoint of aggregate welfare, abstracting from the issue of assigning such costs to individual nodes. The rationale for this analysis lies in the familiar Coase (1960) theorem, whereby aggregate cost savings can be achieved by rational participants who are willing to share them. Identifying protocols that encourage participants to share aggregate savings is an important topic that lies beyond the scope of this paper.

The next section briefly reviews the literature on the topology of the lightning network and its implications. The following section presents the model of the LN and the cost functions of channels. Section 3 contains the main results of the paper. Section 4 presents an algorithm that gives an LN that approximates the minimal cost possible, and Section 5 contains empirical results. Concluding remarks are in Section 6. Appendices A–D contain the proofs of the results. Appendix E provides an example of cost function that meets the definitions in the model, whereas Appendix F reports the statistics of the lightning network snapshots examined in the empirical section.

1.1. Related Work

Guasoni et al. (2024) study the economics of channels, (i) identifying conditions for two parties to optimally establish a channel, (ii) finding explicit formulas for channel costs, (iii) obtaining the optimal collateral and savings entailed, and (iv) deriving the resulting reduction in congestion of the blockchain. Brânzei et al. (2022) obtain partial results on the cost of channels.

Ersoy et al. (2020) discuss game-theoretic aspects of channel construction. In Sali and Zohar (2020), together with game-theoretic questions, the authors show that, under the assumption of all channels being symmetric, the cost of an LN with a star topology is at most twice the minimal possible cost. Theorem 4 and Corollary 4 in this paper generalize this result to any set of channels, symmetric or not.

Several papers study the current topology of the LN (e.g., Lin et al. 2020, Martinazzi and Flori 2020, Seres et al. 2020), observing that the topology is tending toward the centralized hub-and-spoke structure. In Bartolucci et al. (2020), the authors predict the future topology of the LN using tools from percolation theory, abstracting from economic incentives.

Network topology has significant security implications. Rohrer and Tschorsch (2020) introduce an attack that allows routing nodes to learn the destination of a payment—information meant to be private. Empirical results concerning privacy attacks in the LN are in Kappos et al. (2021). Channels may be rendered unusable by an adversary performing a denial-of-service attack. This and similar attacks are discussed in Rohrer et al. (2019). (For mitigation strategies, see Shikhelman and Tikhomirov 2022.) Attacks that aim to steal funds by flooding the base blockchain (Harris and Zohar 2020, Sguanci and Sidiropoulos 2023) become significantly easier under certain topologies. The general robustness of the network also heavily depends on its structure as shown in Lee and Kim (2020).

Liquidity management and throughput maximization in the LN influence both the costs of routing nodes

and the success probability of payment routing. Several papers (e.g., Pickhardt and Nowostawski 2020; Sivaraman et al. 2020; Papadis and Tassiulas 2022, 2023) study the state of the current network and suggest potential improvements to the protocol.

Payments in the LN and in PCNs in general often use a route between two nodes instead of opening a direct channel. Contemporary routing algorithms take into account not only fees, but also privacy, efficiency, success probability, and other parameters. Examples of such algorithms include Grunspan et al. (2020), Roos et al. (2017), Tang et al. (2020), Varma and Maguluri (2021), Wang et al. (2019), and Yu et al. (2018). The performance of these algorithms depends on the topology of the network.

2. The Model

The starting point is a network of n nodes (or users) with some desired rates of flow. Each user i has some required payment rate ϕ_{ij} (positive or negative) from each other user j. The flow $f_i = \sum_{j=1}^{n} \phi_{ij}$ represents the aggregate of such required payments. The problem is how to organize their execution so that each user's net flow is f_i. In other words, though the problem's original inputs may be the required payment rates, its solution only depends on aggregate net flows, which are the focus of the discussion that follows. For example, Alice might send at rate three to Bob and receive at rate one from Carol. Thus, her overall flow is $-3+1=-2$. If Bob receives three from Alice and does not send or receive from anyone else, his flow is three.

Denote the sequence of net flows as $\mathcal{F} = (f_i)_{i=1}^{n}$. Because each payment is debited to the payer and credited to the payee, the sum of all the flows is zero, that is,

$$\sum_{i=1}^{n} f_i = 0.$$

At the modeling level, this accounting identity means that f_i represents the aggregate flow to node i within the network, excluding outside means of payments, such as on-chain transactions or physical transfers.

To facilitate these flows over the LN, channels must be established. A flow between two nodes may or may not be supported by a channel connecting them. Rather, it can be supported by a path of connected channels through which the flow is routed with the flow starting in one node (the origin) and ending in the other (the destination). Thus, a node need not have a channel with every other node with which it interacts. If Alice has channels with both Bob and Carol, Bob can send funds to Carol through Alice even in the absence of a direct channel between Bob and Carol.

A channel's size is the amount locked in it. That amount entails a cost that depends on the expected flow through the channel. We assume that the parties to a channel choose the cost-minimizing size. Moreover, cost minimization implies that the amount locked by each party increases with the relative frequency of payments sent rather than received. (Guasoni et al. (2024) offer a formal development of these results.) Thus, the cost of the channel is a function of the flow. The cost function is general, representing the leading order of the cost of a channel and relying only on a few assumptions that capture some basic properties: it must be increasing in net flow, invariant to its direction, and null for channels with zero net flow (including absent channels).

2.1. Channel Network

Channels connect nodes to support the required net flows. For two nodes, say u and v, denote by $\lambda_{u,v}$ the net flow from u to v. If there is no channel between u and v, then $\lambda_{u,v}=0$. Thus, the matrix $\Lambda = (\lambda_{i,j})_{i,j=1}^{n}$ is, by definition, antisymmetric as $\lambda_{u,v} = -\lambda_{v,u}$. For example, if the channel is between u and v with u sending three and v sending one, then the net flow of the channel is two. Denote the above channel as ${}^{1}\overleftrightarrow{uv}^{3}$.

A given LN $(\lambda_{i,j})_{i,j=1}^{n}$ supports the underlying payment flow $\mathcal{F}$ if, for every node i, the sum of net flows in the channels involving i sums up to the net flow of i, that is,

$$\sum_{j=1}^{n} \lambda_{i,j} = f_i.$$

In particular, the matrix $(\lambda_{i,j})_{i,j=1}^{n}$ describes the flows along each edge, whereas the vector $\mathcal{F} = (f_i)_{i=1}^{n}$ describes the net flows to each node.

2.2. Cost Function

In general, the cost of a channel may depend on a variety of properties, including the statistical properties of arrival times and sizes of payments (as well as the cost of opening and closing a channel, which is common to all channels). Yet both models of money demand (Baumol 1952, Tobin 1956, Miller and Orr 1966) and recent models of Lightning channels (Brânzei et al. 2022, Guasoni et al. 2024) highlight that, at first order, a sufficient statistic of a channel's cost is its net average flow. In particular, the cost of channels with zero net flow is of second order.

Motivated by these observations, this paper offers a first order analysis in which a channel's cost is modeled as a function of its average flow. Rather than providing an analysis based on a specific cost function, the results are established for a general cost function $c:\mathbb{R}\to\mathbb{R}_+$ that satisfies the following structural conditions:

(MO) $c(x) > c(y)$ for $x > y \geq 0$ (monotonicity).
(SY) $c(-x) = c(x)$ (symmetry).
(CI) $c(0) = 0$ (costless inaction).

(SA) $c(x+y) < c(x)+c(y)$ for $x > y \geq 0$ (subadditivity).

Assumption (MO) stipulates that a channel's cost increases in its average flow. When the flow is unidirectional, it is clear that a larger flow requires a bigger channel so as to reduce the frequency of resets, and a bigger channel implies a larger opportunity cost (creating a channel requires depositing its full amount, which cannot earn interest or be used for other purposes). Further, if the same average flow arises from bidirectional payments, the required channel is even bigger because a party's balance may be depleted also because of random fluctuations in payments' directions.

Assumption (SY) specifies that a channel's cost depends on the average flow without regard to the identities of the two parties: if Alice pays Bob at unit rate or vice versa, the optimal channel's cost is the same. This property stems from the decentralized nature of PCNs, whereby the same rules and protocol apply to all users without distinction. Note also that the cost function refers to the total cost of operating a channel regardless of how such costs are allocated between the two parties.

Assumption (CI) has a dual meaning. First, it recognizes that the absence of a channel does not generate any costs. Second, it implies that the cost of an average zero flow, such as a symmetric random walk, is negligible in comparison with an average positive flow. Intuitively, the cost of symmetric flows stems entirely from random fluctuations, which occasionally deplete a channel's balance even in the absence of directional flow and are second order to average flow. This intuition originates from the comparison of unidirectional (Baumol 1952, Tobin 1956) and symmetric (Miller and Orr 1966) models of money demand and is confirmed in recent models of payment channels (Brânzei et al. 2022, Guasoni et al. 2024). This assumption is appropriate in this paper because it aims to study first order effects, which depend on average flows alone.

Assumption (SA) states that combining flows in a single channel is cheaper than handling each of them in a separate channel. This property is natural for three related reasons: First, merging the balances of two channels into one saves on reset costs, which are fixed, regardless of a channel's size. (Because the total balance is the same, opportunity costs do not vary.) Second, a single channel may be able to handle payments of larger size (two channels of size one and two each cannot process a payment of 2.5, but a channel of size three can). Third, if channels' balances are affected by random fluctuations, then a single channel can withstand more shocks before requiring a reset.[3] In the cryptocurrency context, this property also reflects the fixed costs that creating a channel entails regardless of its size. Such costs are instead related to the space occupied on the blockchain and play the same role as withdrawal costs in models of money demand.

The results below on hermetization (Theorems 2 and 3) also require a slightly stronger property than subadditivity:

(SA+) for some strictly increasing $g_+ : \mathbb{R}_+ \to \mathbb{R}_+$,

$$\frac{c(kx)}{k} < \frac{c(x)}{g_+(k)} \qquad x,k > 0.$$

Assumption (SA+) requires that channel costs exhibit economies of scale: as a channel's flow is scaled by a factor, its average cost (defined as the channel's cost divided by the factor) should decrease as the factor increases.[4] In practice, in the typical setting of $c(x) \sim x^\alpha$ for some $\alpha \in (0,1)$, this property is satisfied for $g_+(x) = cx^{1-\beta}$ for $\beta > \alpha$ and some constant $c > 0$. (Appendix E verifies that the cost function obtained in Guasoni et al. (2024) satisfies these properties.)

A channel ${}^{\mu_u}\overleftrightarrow{uv}{}^{\mu_v}$ maps to a flow of $\lambda_{u,v} = \mu_u - \mu_v$. Denoting by $c({}^{\mu_u}\overleftrightarrow{uv}{}^{\mu_v})$ the cost of the specific channel between u and v, it equals the cost function of the net flow:

$$c({}^{\mu_u}\overleftrightarrow{uv}{}^{\mu_v}) = c(\overrightarrow{uv}{}^{\mu_v-\mu_u}) = c(|\mu_u - \mu_v|) = c(|\lambda_{u,v}|) = c(\lambda_{u,v}).$$

Let G be an LN with net flows $(\lambda_{i,j})_{i,j=1}^n$, the cost of G is defined as the sum of the costs of all of the channels:

$$c(G) = \sum_{i<j} c(\lambda_{i,j}) = \frac{1}{2}\sum_{i,j=1}^{n} c(\lambda_{i,j}).$$

3. Main Results

This section contains the statements of the main results. Given an underlying payment flow $\mathcal{F} = (f_i)_{i=1}^n$ and an LN that supports it, consider different cost-reducing changes to the LN. The first change is to locally restructure channels among a subset of nodes in the LN. The second change is the addition of a new node that serves as a hub for several nodes that previously had channels between them. The problem is to understand which topology minimizes the overall cost among the topologies that support $\mathcal{F}$.

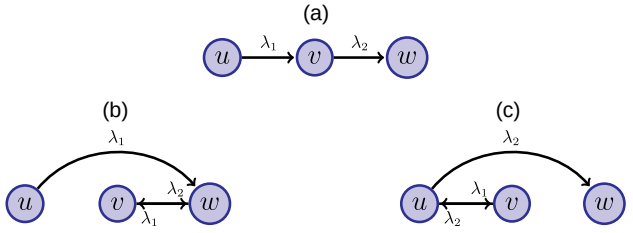
Note that the present analysis focuses on recurring costs, that is, costs per unit of time. Thus, the costs of replacing the channels of one topology with the channels of another topology, which are one off, are neglected. Of course, the rationale is that small recurring savings over time exceed their initial one-off costs.

3.1. Local Channel Reconfiguration

First, consider the case in which a given set of nodes reconfigures the channels between them to reduce the overall cost. In general, a smaller node set is more likely to reach an agreement on channel reconfiguration.

Theorem 1 states that, given three nodes u, v, and w, if there is a channel directed from u to v and a channel

Figure 1. (Color online) All Configurations Respect the Same Flow $f_u = -\lambda_1, f_v = \lambda_1 - \lambda_2, f_w = \lambda_2$



Notes. The first configuration costs more than either the second or the third one. (a) Not cost minimizing. (b) Cost minimizing if $\lambda_1 \le \lambda_2$. (c) Cost minimizing if $\lambda_1 \ge \lambda_2$.

directed from v to w, then this topology is not cost minimizing. It offers a cost-reducing improvement, which is summarized in Figure 1.

Theorem 1. *Let u, v, and w be nodes and assume that there is a flow of $\lambda_1 > 0$ from u to v and a flow of $\lambda_2 > 0$ from v to w. Then, the structure $\overrightarrow{uv}^{\lambda_1}, \overrightarrow{vw}^{\lambda_2}$ is not cost minimizing. A less costly structure is either $\overrightarrow{uw}^{\lambda_1}, {}^{\lambda_1}\overleftrightarrow{vw}^{\lambda_2}$ if $\lambda_1 \le \lambda_2$ or ${}^{\lambda_2}\overleftrightarrow{uv}^{\lambda_1}, \overrightarrow{uw}^{\lambda_2}$ if $\lambda_1 \ge \lambda_2$.*

An important message of this result is that cost minimization fosters disintermediation in that it is inefficient for a node to merely forward others' payments. Instead, Figure 1 shows that a natural partner is someone with strong personal flow with one party who can benefit from offsetting some of this flow by forwarding payments in the opposite direction, thereby extending a channel's life and abating reset costs. Such a partner is not a traditional intermediary for two reasons. First, payments forwarded on behalf of others are less than the partner's own flow. Second, in a competitive equilibrium, the partner would not charge any fee to forward such payments and would even accept a small negative fee—if it were possible[5]—because forwarding payments to offset personal flows reduces one's costs.

A simple example helps illustrate this point: imagine that Alice's employer (u) pays Alice's bank (v), which pays Alice (w). This arrangement is not optimal: if Alice's salary λ_1 is greater than Alice's net payments λ_2 (because she uses some of the salary to buy goods and services), then the employer should pay Alice in relation to her net flow, keeping a channel with her bank. However, if Alice's net payments are higher than her salary (because she has additional income in excess of expenses), the employer is better off paying her gross salary, leaving her to independently manage unrelated flows.

Note also that Theorem 1 does not require the subadditivity condition (SA). In this sense, disintermediation is a very basic feature of payment-channel networks, independent of economies of scale (or lack thereof) that underpin subadditivity. Instead, disintermediation results from avoiding multiple transfers in the same direction.

Two corollaries follow from Theorem 1: (i) an odd cycle is not cost minimizing, and (ii) a cost-minimizing LN must be a bipartite graph.

Corollary 1. *Let $(u_i)_{i=1}^k$ be a cycle of length k; that is, a channel connects u_j to u_{j+1} for every $1 \le j < k$ and u_k to u_1. (The flow in the channels can go in either direction.) If the cycle has an odd length, then it is not a cost-minimizing structure.*

A stylized implication of Corollary 1 is that a cost-minimizing topology should have no triangles, and therefore, its clustering coefficient should be zero. (The clustering coefficient is the fraction of connected triples that are also triangles. See Section 5 for details.) Section 5 examines empirically this testable prediction and finds that the LN has indeed an unusually low clustering coefficient in comparison with other payment networks.

Corollary 2. *If an LN is cost minimizing, then it is a bipartite graph $G = (A \cup B, E)$ with all net flows from A to B.*

3.2. Adding a Routing Node

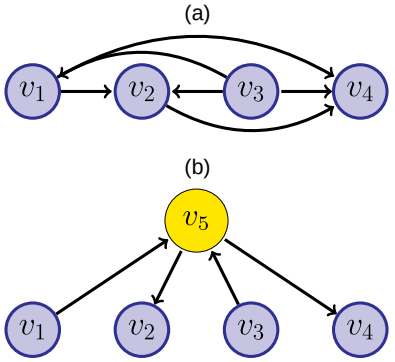
Centralization is a natural means to simplify a network by routing flows through a single node, thereby requiring no more than a channel for each user. In the case of the Lighting Network, the question is whether a low number of channels also translates in low aggregate costs. A broader goal is to understand whether economic forces, such as collective cost minimization, may progressively drive the network's topology toward a central structure. This section examines the extent to which centralization may reduce costs.

Consider a set of nodes that can benefit from deleting all the channels between them and connecting the nodes to a new node that serves as a single central hub. Such sets are defined as hermetizable. (See Figure 2.) An LN is hermetizable if it contains a hermetizable set.

Theorem 2 gives sufficient conditions for the existence of a hermetizable set in an LN and presents an efficient algorithm to find such a set if it exists. Theorem 3 proves that, under mild assumptions on the distribution of channel costs in the LN, the probability of the LN being hermetizable is high.

Note that, in principle, the central hub does not have to be a new node; it could be one of the existing nodes. When thinking about real-world examples, this

Figure 2. (Color online) An Example of a Set of Nodes Before and After Hermetization



Notes. (a) Original configuration. (b) Hermetized configuration.

prospect is not very likely. Nodes represent a specific individual or business and process the transactions to and from their represented entity plus the transactions that transit through them on behalf of others. Being a central hub is another specialization, which demands the continuous maintenance of several channels through refills, resets, and other liquidity-management measures. Thus, we conservatively assume that regular nodes are not amenable to take such a role, leaving it instead to specialized nodes. In principle, if one of the existing nodes became the hub, the cost of the LN would be slightly lower as its own traffic would not be required to pass through another hub. For this reason, both Theorems 2 and 3 hold even if an existing node were used as hub instead of a new node.

Note also that hermetized configurations are not optimal in general even if the central hub is chosen among existing nodes. Unless the central hub is the sole payer or the sole payee, one of the spokes is a payee and another a payer, resulting in payment forwarding, which is suboptimal according to Theorem 1. The main message of Theorems 2 and 3 is that, under mild assumptions, hermetization leads to lower—not minimal—costs.

To state Theorems 2 and 3 formally, the following definitions are necessary.

Definition 1. Let $G = (V, E)$ be a graph and $v \in V$ a node. Denote by $d(v)$ the degree of v, which is the number of channels that involve v, that is, $d(v) = |\{w \in V : \lambda_{v,w} \neq 0\}|$, and let $\overline{d}(G)$ be the average degree $\overline{d}(G) = \sum_{v \in V} d(v)/|V|$.

Definition 2. Let $G = (V, E)$ be an LN. For every node v, let $s(v)$ be the smallest flow of a channel adjacent to v and $S(v)$ be the largest, that is, $s(v) = \min\{|\lambda_{v,j}| : j \in V : \lambda_{v,j} \neq 0\}$ and $S(v) = \max_{j \in V} |\lambda_{v,j}|$.

Define the discrepancy of G as

$$disc(G) = \max_{v \in V} \frac{c(S(v))}{c(s(v))}.$$

Definition 3. The hermetization of a set of nodes U in an LN is the graph obtained by

i. Deleting all channels among nodes in U.

ii. Adding a new node v_0.

iii. Connecting all nodes in U to v_0 with channels that aggregate their flows toward other nodes in U.

Definition 4. A set of nodes U is hermetizable if hermetizing it reduces its cost. An LN is hermetizable if there exists a set of nodes U in the LN that is hermetizable.

Theorem 2 offers a sufficient condition for the existence of a hermetizable set and a polynomial-time algorithm to find it.

Theorem 2. *Assume a cost function with the properties (MO), (SY), (CI), (SA+). If G is an LN with $\overline{d}(G) > 2g_+^{-1}(2disc(G))$, then it is hermetizible, and there is a polynomial-time algorithm to find a hermetizable set U.*

A natural question is whether the conditions of Theorem 2 are met by a typical LN. Theorem 3 shows that, if the LN is chosen randomly using a distribution with tails that are not thicker than a power, then, with high probability, its cost can be reduced by hermetization as in Theorem 2, indicating that cost reduction is the norm, not the exception.

Theorem 3 examines the situation in which flows $\lambda_{i,j}$ are chosen randomly (and independently) with the same distribution. If the distribution of the resulting costs $c(\lambda_{i,j})$ in the LN has a tail thinner than power and each $c(\lambda_{i,j})$ is either zero or has a uniform lower bound, then the LN is hermetizable with high probability.

Theorem 3. *Assume a cost function with the properties (MO), (SY), (CI), (SA+) and that the size of each channel $\lambda_{i,j} \neq 0$ is distributed according to a law such that, for all channels $ij \in E$,*

i. *There exist $a > 1$, $b > 0$ such that $\mathbb{P}(c(\lambda_{i,j}) > x) < bx^{-a}$ for all $x > 0$.*

ii. *There exists $\varepsilon > 0$ such that $\mathbb{P}(c(\lambda_{i,j}) > \varepsilon) = 1$.*

Then, the LN is hermetizable with probability greater than or equal to

$$1 - (b\varepsilon/2)^{-a} \sum_{1 \leq i \leq |V|} d(i)(g_+(d(i)))^{-a}.$$

3.3. Bounds on the Global Minimal Cost of an LN

Finally, consider the minimal possible cost of an LN that supports the flow $\mathcal{F} = (f_i)_{i=1}^n$ and denote it by

$$c_{\min}(\mathcal{F}) = \min\{c(G) | G \text{ supports } \mathcal{F}\}.$$

The following bounds hold for any flow.

Theorem 4. *For any* $\mathcal{F} = (f_i)_{i=1}^n$,

$$\max\left(\sum_{f_i>0} c(f_i), \sum_{f_i<0} c(f_i)\right) \le c_{\min}(\mathcal{F}) \le \sum_{i\in V} c(f_i) - \max_{i\in V} c(f_i).$$

Recall that a star topology corresponds to a graph in which one node has edges with all others, and no other edges exist. The star topology is a natural candidate for the globally cost-minimal topology because it has the minimal number of edges for a connected graph. The global bounds in Theorem 1 clarify the conditions under which a star is optimal and the extent to which it is close to optimal.

Clearly, if there is only one net payer or only one net payee then a star with such a user as the star's center minimizes costs because it achieves the lower bound in Theorem 1. The next corollary shows that this condition is also necessary, namely, if there are at least two payers and two payees, then the star does not minimize costs.

Corollary 3. *Given* $\mathcal{F} = (f_i)_{i=1}^n$, *if there are distinct* i_1, i_2, j_1, j_2 *such that* $f_{i_1}, f_{i_2} > 0 > f_{j_1}, f_{j_2}$, *then a star is not cost minimizing.*

In addition, another implication of Theorem 4 is that the star topology is at most twice as costly as the minimum.

Corollary 4. *For any flow* $\mathcal{F}$, *a star topology costs at most* $2 \cdot c_{\min}(\mathcal{F})$.

The results so far suggest that a few iterations of the improvements described above could lead to an exact minimal-cost topology. If so, the previous results would imply an exact optimization algorithm with quadratic complexity. In general, this is not the case. The main observation is that the cost-minimization problem for general cost functions nests combinatorial optimization problems that are known to be NP-complete.

Theorem 5. *For any cost function* $c(x)$ *satisfying the structural conditions (MO), (SY), (CI), (SA), the network cost-minimization problem is NP-complete.*

The intuition of this result is relatively simple and is exemplified by a minimal example: if there are many payers and only two payees (or vice versa) and it is possible to assign each payer to a single payee so that both payees receive their flow, then finding the matching subset for a payee is equivalent to identifying (or determining the absence of) a subset of numbers with a given sum. This is the subset-sum problem, which is known to be NP-complete, because its solution requires us to examine nearly all subsets of payers (Karp 1972).

Algorithm 1 (Two-Factor Approximation of Minimal Cost)

input: A vector of flows $\mathcal{F} = (f_i)_{i=1}^n$ such that $\sum_{i=1}^n f_i = 0$.
output: A LN network $\Lambda = (\lambda_{i,j})_{i,j=1}^n$ supporting the input flow.
$\lambda_{1,1} := 0$; $\lambda_{1,i} := -\lambda_{i,1} := f_i$ for $2 \le i \le n$; $\lambda_{j,i} := 0$ for $2 \le i, j \le n$. ▷ Create a star centered at 1.
repeat
  $P :=$ GETPAIRS(Λ) ▷ Find suboptimal pairs $j \to 1 \to i$.
  for (i,j) in P **do**
    $\Lambda :=$ DISINTERMEDIATE($\Lambda, (i,j)$) ▷ Eliminate suboptimal pair.
until P empty
return Λ.
function GETPAIRS(Λ)
  posindex[i] := negindex[i] := 0 for i in 1 to n. ▷ Initialize sequence of indexes.
  pos := 0; neg := 0; P = ∅. ▷ Initialize iterators.
  for i in 2 to n **do**
    if $\lambda_{1,i} > 0$ **then**
      pos += 1; posindex[pos] := i; ▷ Record position of outflow from 1.
    else if $\lambda_{1,i} < 0$ **then**
      neg += 1; negindex[neg] := i; ▷ Record position of inflow to 1.
  for i in 1 to min(pos,neg) **do** ▷ Add outflow-inflow pair to list.
    add (posindex[i],negindex[i]) to P.
  return P.
function DISINTERMEDIATE($\Lambda, (i,j)$) ▷ Improvement in Theorem 1 to $j \to 1 \to i$.
  if $|\lambda_{1,i}| > |\lambda_{1,j}|$ **then** ▷ If the flow between i and 1 is greater than the flow between j and 1.
    $\lambda_{j,i} := -\lambda_{i,j} := \lambda_{j,1}$; ▷ j routes through i.
    $\lambda_{1,i} := -\lambda_{i,1} := \lambda_{1,i} + \lambda_{1,j}$; ▷ The channel between 1 and i accounts for the flow from j.
    $\lambda_{1,j} := -\lambda_{j,1} := 0$ ▷ j no longer has a channel with 1.
  else ▷ The flow between i and 1 is at most the flow between j and 1.
    $\lambda_{j,i} := -\lambda_{i,j} := \lambda_{1,i}$; ▷ i routes through j.
    $\lambda_{j,1} := -\lambda_{1,j} := -(\lambda_{1,i} + \lambda_{1,j})$; ▷ The channel between 1 and j accounts for the flow from i.
    $\lambda_{1,i} := -\lambda_{i,1} := 0$; ▷ i no longer has a channel with 1.
  return Λ.

4. Algorithm

The NP-completeness of the exact cost-minimization problem and the previous results on cost reduction motivate the development of an algorithm that approximates the minimum, keeping computational complexity polynomial. In this spirit, the following quadratic-time algorithm yields a two-factor

approximation: a network whose cost is less than twice the theoretical lower bound.

Algorithm 1 proceeds by first creating a star centered on node 1. Then, the function GETPAIRS assigns to each other node a payer or payee number, depending on the sign of its flow. Then, payer–payee pairs are created (first payer with first payee, etc.): as each such pair is suboptimal by Theorem 1, the function DISINTERMEDIATE replaces it by a cheaper pair. The identification replacement continues until further improvements are no longer possible.

Theorem 6. *For any* $\mathcal{F} = (f_i)_{i=1}^n$, *Algorithm 1*

i. *Has quadratic complexity.*

ii. *Returns a network that cannot be improved by Theorems 1 and 2.*

iii. *Has cost less than twice the minimal lower bound in Theorem 4.*

5. Lightning Network Structure

The preceding sections establish and study attributes of a cost-minimal Lightning Network, thus motivating the empirical examination of the visible parts of the actual LN. The goal is to understand the connectivity properties of the Lightning Network and evaluate the extent to which they are consistent with the implications of the above results.

The LN is decentralized. Its nodes use the gossip protocol to broadcast to each other newly created and deleted public channels. Thereby the information is synchronized across the nodes. (Synchronization delays of a few seconds are possible.) The Lightning Network does not allow entities that do not own a channel to learn anything beyond the channels' existence and their size and fee policies. Enhanced privacy is one of the main features of the LN, and improvements are regularly implemented to reduce any leakage of information. Such built-in privacy features entail that nodes' flows cannot be observed, thereby placing limits on testable implications. Yet a key necessary condition within the observable purview is the absence of cliques, which is evaluated quantitatively as low clustering. This section examines clustering in the LN over time and in comparison with other payment networks.

The data set in Table F.1 summarizes 23 monthly snapshots of public channels in the LN from October 2020 to July 2023 obtained from a Lightning node (Decker 2023). (Observations contain gaps from November 2021 to January 2022 and from August 2022 to April 2023. Data in the months immediately before and after the gaps may be incomplete but are, nevertheless, included for the sake of integrity.)

Publicly available information about each channel includes its size and its parties but not their respective balances or balance updates (i.e., payments from one party to another). Moreover, even nodes that act as payment intermediaries in chains of transfers do not know the origin and the destination of the payments they facilitate, but merely the identity of the two nodes from and to which they transfer funds.

Yet even the limited amount of publicly available information allows us to probe some implications of our results. Specifically, Corollary 1 states that a cost-minimizing network should not contain cycles of odd length and, in particular, triangles, that is, cycles of three nodes. In the parlance of graph theory, this property means that the clustering coefficient should be zero as now explained.

Defining a triplet as three nodes i, j, k connected by at least two channels, the clustering coefficient is the fraction of triangles (i.e., triplets connected by three channels) out of all triplets (i.e., connected by at least two channels). It is formally defined as[6]

$$Cl(G) = \frac{\sum_{v\in V} |\{\{u,w\} \subset V, \{u,w\}, \{v,w\}, \{u,v\} \subset E\}|}{\sum_{v\in V} d(v)(d(v)-1)}.$$

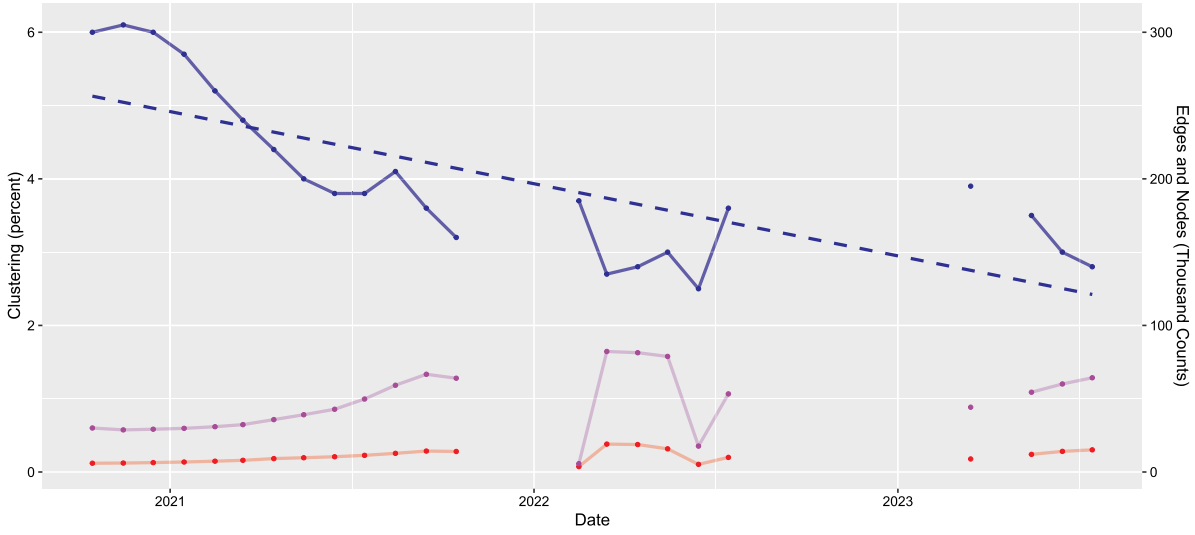
Thus, in theory, Corollary 1 implies that it is suboptimal for three channels to exist in any triplet; hence, the numerator in this equation should be zero as should be the clustering coefficient.

In practice, the clustering coefficient need not be zero for multiple reasons. First, a channel may have its balance mostly shifted to one party, making the usual payer unable to make payments and indifferent to closing the channel. Yet the balance's main owner may delay closure to times when on-chain transactions are cheaper (such as the weekend). Second, a channel may have already been closed but unilaterally, thereby requiring a delay (time lock) of several days to finalize. Third, the channel may serve other purposes than payments, such as liquidity management or redundancy (Papadis and Tassiulas 2020), which are not modeled here.

Notwithstanding these caveats, the broad message of Corollary 1 is that triangles are generally not cost-efficient and, therefore, should be infrequent in comparison with other payment networks with different cost structures. Figure 3 plots the clustering coefficient (along with the number of nodes and edges) of the LN snapshots over time. Two features are apparent despite the gaps in observations: the clustering coefficient is low and steadily decreases over time. (The slope of the regression line in Figure 3 is -0.98 and is significantly negative; $p < 0.001$.)

The clustering coefficient never exceeds 6% in any of the snapshots considered: choosing three random nodes connected by two channels, it is very unlikely that also a third channel is present among them. In fact, even 6% is a rather low figure for a payment network. As a comparison, consider Venmo, a payment network owned by Paypal, in which users can send funds to each other. Venmo has both differences and similarities to the

Figure 3. (Color online) Clustering Coefficient (Top Solid, Left Scale), Its Linear Regression Against Time (Top Dashed, Left Scale), Number of Edges (Middle, Right Scale), and Nodes (Bottom, Right Scale) for Each of the Monthly Snapshots of the Lightning Network in Decker (2023)



LN: as in the LN, funds held in Venmo accounts do not earn interest. However, unlike the LN, Venmo balances are held in a single user account rather than in the bilateral channels typical of the LN, and there is no cost in connecting to another user. Also, payments among users are handled centrally rather than routed though nodes.

Absent any incentive to minimize their number of connections, Venmo users create as many of them as their payees. Thus, the resulting network can be considered as a reference case in which channel costs are null. Empirically, Bhattacharya et al. (2020) and Zhang et al. (2017) report that the clustering coefficient in Venmo is 14.7%—more than twice the LN clustering of 6% in October 2020 and more than five times its 2.8% clustering in July 2023.

As, in the LN, clustering is several times smaller than it is in payment networks with costless connections, it is tempting to ascribe such difference to efficiency choices rather than to payment relationships. More broadly, one might interpret the declining clustering coefficient as a gradual evolution of the LN toward a more efficient structure, fueled by growth in its size, reach, and sophistication of participants. Yet these macroscopic network properties do not provide direct observation of the network's payment efficiency per se because payment flows cannot be observed publicly in the LN.[7] Thus, these topological changes could also be related to a natural lower clustering of active payment relationships in general, of Lightning adopters in particular, or may be due to other phenomena.

6. Conclusion

Blockchains are at the heart of multiple protocols designed to manipulate ledgers securely in a decentralized fashion. Low throughput is a weakness shared by many of these protocols. A way to address this weakness is to relegate transaction processing to a device outside the main blockchain, secure these transactions cryptographically off-chain, and periodically post a summary of the transactions on the blockchain, thereby incorporating these originally off-chain transactions into the consensus. The Lightning Network is one of the most popular layer-two solutions. It supports Bitcoin payments.

The term "Lightning Network" often refers to collections of paths that support payments between nodes that are not directly connected. Intermediate nodes of an LN can and often charge fees to facilitate the intermediation. Viability of the LN protocol requires cryptographically guaranteed communication between nodes along the payment route. (For details, see Guasoni et al. (2024).)

This paper focuses on cost reduction through modifications of the LN. LN modifications amount to modifications of the LN topology, which affects various performance aspects, such as payment success rate, node privacy, and censorship resistance. Thus, reconfiguration of the LN to reduce cost can have wide-ranging implications beyond mere cost reduction.

The paper shows that computing the general global minimum cost configuration is an NP-complete problem and then describes a quadratic-time algorithm that is guaranteed to achieve a two-factor approximation. It also addresses the intuitive conjecture that a star-like structure is cost minimal. It is not.

Although the LN protocol treats all nodes equally, each node can differentiate itself from others by offering competitive fee rates for forwarding payments, thereby

introducing a layer of heterogeneity in the network and creating the potential for centralization.

Acknowledgments

For their helpful comments, the authors thank Carla Kirk-Cohen, Jacob Leshno, Jiasun Li, Maarten van Oordt, Julien Prat, Fahad Saleh, and seminar participants at Harvard University, SIAM FME, CEBRA 2021, Chicago SAFM 2023, Amamef 2023, Simons Institute, the Technion, and the 2023 CBER Conference. Special thanks to Joshua Gans (the editor) and four anonymous reviewers who helped strengthen the paper with their insights.

Appendix A. Proofs: Local Channel Reconfiguration

This section states and proves the local change results.

Proof of Theorem 1. To prove that the structure $\overrightarrow{uv}^{\lambda_1}, \overrightarrow{vw}^{\lambda_2}$ is not cost minimizing, consider different structures that respect the flow and have an overall cost lower than $c(\overrightarrow{uv}^{\lambda_1}) + c(\overrightarrow{vw}^{\lambda_2}) = c(\lambda_1) + c(\lambda_2)$. Note that $f_u = -\lambda_1, f_v = \lambda_1 - \lambda_2, f_w = \lambda_2$.

To construct less costly structures, consider the two cases $\lambda_1 > \lambda_2$ and $\lambda_2 > \lambda_1$. (In the unlikely case of $\lambda_1 = \lambda_2$, both of the constructions below are less costly than $\overrightarrow{uv}^{\lambda_1}, \overrightarrow{vw}^{\lambda_2}$.)

In the case $\lambda_1 > \lambda_2$, observe that the structure ${}^{\lambda_2}\overleftrightarrow{uv}^{\lambda_1}, \overrightarrow{uw}^{\lambda_2}$ supports the flows and has a lower overall cost. This channel structure supports the flows because

$$u\colon \lambda_{vu} + \lambda_{vw} = (\lambda_2 - \lambda_1) + (-\lambda_2) = -\lambda_1 = f_u, \tag{A.1}$$

$$v\colon \lambda_{uv} + \lambda_{uw} = (\lambda_1 - \lambda_2) + 0 = \lambda_1 - \lambda_2 = f_v, \tag{A.2}$$

$$w\colon \lambda_{wu} + \lambda_{wv} = 0 + \lambda_2 = \lambda_2 = f_w. \tag{A.3}$$

And its cost is

$$c({}^{\lambda_2}\overleftrightarrow{uv}^{\lambda_1}) + c(\overrightarrow{uw}^{\lambda_2}) = c(\lambda_1 - \lambda_2) + c(\lambda_2) \overset{(MO) \text{ and } \lambda_1 > \lambda_2}{<} c(\lambda_1) + c(\lambda_2) = c(\overrightarrow{uv}^{\lambda_1}) + c(\overrightarrow{vw}^{\lambda_2}).$$

A symmetric argument shows that, if $\lambda_2 > \lambda_1$, the structure $\overrightarrow{uw}^{\lambda_1}, {}^{\lambda_1}\overleftrightarrow{vw}^{\lambda_2}$ is superior. See Figure 1 for an illustration.

Proof of Corollary 1. Assume the cycle is of length $2m+1$ for some $m \in \mathbb{N}$. Then, either there are at least $m+1$ channels directed clockwise or at least $m+1$ channels directed counterclockwise. For the $m+1$ channels not to share any nodes, $2m+2$ nodes are necessary. As there are only $2m+1$ nodes, two of the $m+1$ channels must share a node. As both channels are directed either clockwise or counterclockwise, the node shared by both has an incoming and an outgoing channel. Thus, by Theorem 1, the graph is not cost minimizing.

Proof of Corollary 2. By Kőnig's (1931) theorem (see also Wilson 1979), a graph is bipartite if and only if it does not contain an odd cycle. By Corollary 1, a graph that is cost minimizing does not have an odd cycle; thus, a cost-minimizing graph is bipartite. Furthermore, let A be the set of all nodes i such that $f_i < 0$, and $B = V \setminus A$ the rest. Then, all of the flows are from A to B.

Appendix B. Proofs: Adding a Routing Node

Throughout this section, assume that the cost function c has all five properties (MO), (SY), (CI), (SA+).

B.1. Proof of Theorem 2

The proof of Theorem 2 requires several lemmas.

Lemma B.1. *Let $G = (V, E)$ be an LN, and let $U \subseteq V$ be a subset of vertices. U is hermetizable if*

$$d(i) > g_+^{-1}\left(2\frac{c(S(i))}{c(s(i))}\right) \text{ for all } i \in U.$$

Proof. Define U' to be the result of hermetizing U. Denoting the new vertex by zero, and the new channels are $(\lambda'_{i,j})_{i,j\in U'}$, where

$$\lambda'_{k,l} = \begin{cases} \sum_{j=1}^{n} \lambda_{k,j} & \text{if } l = 0, k \neq 0 \\ \sum_{i=1}^{n} \lambda_{i,l} & \text{if } k = 0, l \neq 0 \\ 0 & \text{if } k \neq 0, l \neq 0 \\ 0 & \text{if } k = l = 0. \end{cases}$$

To show that $c(U') < c(U)$, it is enough to show that the cost that each node contributes to U is greater than the cost it contributes to U'. Note that

$$c(U) = \frac{1}{2}\sum_{i,j=1}^{n,n} c(\lambda_{i,j}) = \sum_{i=1}^{n}\frac{1}{2}\sum_{j=1}^{n} c(\lambda_{i,j})$$

$$c(U') = \frac{1}{2}\sum_{i=1}^{n}(c(\lambda'_{i,0}) + c(\lambda'_{0,i})) = \sum_{i=1}^{n} c(\lambda'_{i,0}) = \sum_{i=1}^{n} c\left(\sum_{j=1}^{n}\lambda_{i,j}\right).$$

Thus, to show that $c(U') < c(U)$, it is enough to show that, for every i,

$$\frac{1}{2}\sum_{j=1}^{n} c(\lambda_{i,j}) > c\left(\sum_{j=1}^{n}\lambda_{i,j}\right). \tag{B.1}$$

Note that

$$\frac{1}{2}\sum_{j=1}^{n} c(\lambda_{i,j}) > \frac{1}{2}\sum_{j:\lambda_{i,j}\neq 0} c(s(i)) = \frac{1}{2}d(i)c(s(i)),$$

and

$$c\left(\sum_{j=1}^{n}\lambda_{i,j}\right) < c(d(i)S(i)) \overset{(SA+)}{<} \frac{d(i)}{g_+(d(i))}c(S(i)).$$

As g_+ is increasing, the inverse function g_+^{-1} exists and is also increasing. From the above inequalities, it follows that (B.1) holds if

$$\frac{1}{2}d(i)c(s(i)) > \frac{d(i)}{g_+(d(i))}c(S(i))$$

$$\iff g_+(d(i)) > 2\frac{c(S(i))}{c(s(i))} \iff d(i) > g_+^{-1}\left(2\frac{c(S(i))}{c(s(i))}\right), \tag{B.2}$$

which completes the proof.

Lemma B.2. *Denote by $\delta(G) = \min_{v\in V} d(v)$ the minimum degree of a graph $G = (V, E)$. A set U is hermetizable if $\delta(U) \geq g_+^{-1}(2disc(U))$.*

Proof. Remembering that $\delta(U) = min_{v\in V} d(v)$, $disc(U) = \max_{v\in V} \frac{c(S(i))}{c(s(i))}$, and g_+^{-1} is a monotone increasing function, it follows that the assumptions of Lemma B.1 hold. Indeed, for every $i \in V$,

$$d(i) \geq \delta(U) \geq g_+^{-1}(2disc(U)) > g_+^{-1}\left(2\frac{c(S(i))}{c(s(i))}\right).$$

One of the main implications of Lemma B.2 is that a possible business model for an entrepreneur is to find subgraphs G with $\delta(G) \geq g_+^{-1}(2disc(G))$ and offer the service of adding a new vertex. In doing so, the entrepreneur can charge a part of the users' savings. The following lemma shows how to find such subgraphs.

Lemma B.3. *There is an algorithm that, given a graph G, finds a subgraph H with $\delta(H) \geq g_+^{-1}(2disc(G))$ if there exists one or, alternatively, returns an empty graph if there is no such H. The running time of the algorithm is $O(V + E)$.*

Proof. The algorithm is as follows: at each step, delete all the vertices of degree up to $g_+^{-1}(2disc(G))$ until there are no more vertices to remove either because the minimum degree of the graph becomes greater than $g_+^{-1}(2disc(G))$ or because all vertices are deleted.

Lemma B.4. *Every graph G with average degree $2d$ has a subgraph G' with $\delta(G') > d$.*

Proof. Finding G' requires the following steps: define $G_0 = G$ and let G_i for $i \geq 1$ be defined recursively as follows. If $\delta(G_i) > d$ or G_i is the empty graph, define $G' = G_i$. If not, let v_i be a vertex with the smallest degree in G_i and define $G_{i+1} = G_i \setminus v_i$.

It remains to show that the process stops before all of the vertices are deleted. By contradiction, suppose that all of the vertices in the graph were deleted. Let n be the number of vertices in G.

On one hand, each vertex deleted has degree at most d; thus, the number of edges in the graph is at most $(d-1)n$. On the other hand, the average degree is at least $2d$, so the number of edges is at least dn, which is a contradiction.

Thus, the process could not have terminated with an empty graph, and there must be some subgraph G_i of G with $\delta(G_i) > d$.

The proof of the main theorem now follows.

Proof of Theorem 2. By Lemma B.4, if the LN G has average degree greater than $2g_+^{-1}(2disc(G))$, then it has a subgraph with minimal degree $g_+^{-1}(2disc(G))$, and by Lemma B.3, there is an efficient algorithm to find it. By Lemma B.2, this subgraph is hermetizable, and hence, G itself is hermetizable.

B.2. Proof of Theorem 3

Proof. In view of Lemma B.1, the graph is hermetizable with a probability of at least

$$\mathbb{P}\left(\cap_{1\leq i\leq |V|}\left\{d(i) > g_+^{-1}\left(2\frac{C(S(i))}{c(s(i))}\right)\right\}\right) \tag{B.3}$$

$$= 1 - \mathbb{P}\left(\cup_{1\leq i\leq |V|}\left\{d(i) \leq g_+^{-1}\left(2\frac{C(S(i))}{c(s(i))}\right)\right\}\right) \tag{B.4}$$

$$\geq 1 - \sum_{1\leq i\leq |V|}\mathbb{P}\left(d(i) \leq g_+^{-1}\left(2\frac{C(S(i))}{c(s(i))}\right)\right). \tag{B.5}$$

To estimate the last probability, note that

$$\mathbb{P}\left(d(i) \leq g_+^{-1}\left(2\frac{c(S(i))}{c(s(i))}\right)\right) \tag{B.6}$$

$$= \mathbb{P}\left(g_+(d(i)) \leq 2\frac{c(S(i))}{c(s(i))}\right) \tag{B.7}$$

$$= \mathbb{P}\left(\frac{c(s(i))}{2}g_+(d(i)) \leq c(S(i))\right) \tag{B.8}$$

$$\leq \mathbb{P}\left(\frac{\varepsilon}{2}g_+(d(i)) \leq c(S(i))\right). \tag{B.9}$$

In addition, for any $x > 0$,

$$\mathbb{P}(c(S(i)) \geq x) = \mathbb{P}\left(\max_{j:\lambda_{i,j}\neq 0} c(\lambda_{i,j}) \geq x\right) \tag{B.10}$$

$$= \mathbb{P}(\cup_{j:\lambda_{i,j}\neq 0} c(\lambda_{i,j}) \geq x) \tag{B.11}$$

$$\leq \sum_{j:\lambda_{i,j}\neq 0}\mathbb{P}(c(\lambda_{i,j}) \geq x) \leq d(i)bx^{-a}. \tag{B.12}$$

Thus, it follows that, for some constant k,

$$\mathbb{P}\left(c(S(i)) \geq \frac{\varepsilon}{2}g_+(d(i))\right) \leq kd(i)(g_+(d(i)))^{-a},$$

whence the claim.

Appendix C. Proofs: Bounds on the Global Minimal Cost

Proof of Corollary 3. Consider a star LN; that is, there is a special node h such that all of the other nodes have a single channel to it. Consider the nodes i_1, i_2, j_1, j_2, and note that one of them could be h. Out of the four nodes without loss of generality assume that $i_1 \neq h$; then, as $f_{i_1} > 0$, the channel is $\overrightarrow{hi_1}^{f_{i_1}}$. For similar reasons, there is a channel $j_1\overrightarrow{h}^{(-f_{j_1})}$. By Theorem 1, the structure $j_1\overrightarrow{h}^{(-f_{j_1})}, \overrightarrow{hi_1}^{f_{i_1}}$ is not cost effective.

Proof of Theorem 4. To prove the upper bound, for any given flow $\mathcal{F}$, it suffices to construct a network that supports $\mathcal{F}$ and has a cost of at most $\sum_{i\in V} c(f_i) - \max_{i\in V} c(f_i)$. Assume without loss of generality that $f_0 = \max_{i\in V} f_i$. For each node $i > 0$, create a channel between node i and node 0 with $\lambda_{0,i} = f_i$.

First, note that this network supports $\mathcal{F}$ because, for each $i > 0$,

$$\sum_{j=0}^{n}\lambda_{j,i} = \lambda_{0,i} + \sum_{j=1}^{n}\lambda_{j,i} = f_i + 0 = f_i.$$

As for $i = 0$,

$$f_0 = -\sum_{i\in V} f_i = \sum_{i\in V}\lambda_{i,0}.$$

Second, the cost of this network is $\sum_{i,j\in V} c(\lambda_{i,j}) = \sum_{i\in V} c(\lambda_{i,0}) = \sum_{i\in V} c(f_i) = \sum_{i=0}^{n} c(f_i) - c(f_0) = \sum_{i\in V} c(f_i) - \max_{i\in V} c(f_i)$, and so the upper bound holds.

As for the lower bound, examine first the flow of a single node i. A network supports $\mathcal{F}$ if and only if $\sum_{j\in V}\lambda_{i,j} = f_i$. Thus, the cost resulting from the flow of f_i is $\sum_{j\in V} c(\lambda_{i,j}) \geq c(\sum_{j\in V}\lambda_{i,j}) = c(f_i)$, where the inequality holds because of the subadditivity of the cost function.

Without loss of generality, assume that $f_i > 0$ if and only if $i \le k$ for some fixed k. By Theorem 1, if $i, j \le k$, it cannot be that there is a flow from i to j as this would imply that either i or j is a node that is both the origin and the destination of two different channels. The same holds if $i, j > k$. Thus, $\lambda_{i,j} = 0$ if $i, j \le k$ or $i, j > k$.

Using the above, the lower bounds for the cost are

$$\sum_{i<j} c(\lambda_{i,j}) = \sum_{i=1}^{k} \left(\sum_{j=k+1}^{n} c(\lambda_{i,j}) \right) \tag{C.1}$$

$$\ge \sum_{i=1}^{k} c\left(\sum_{j=k+1}^{n} \lambda_{i,j} \right) = \sum_{i=1}^{k} c(f_i). \tag{C.2}$$

Repeating the same steps, it also follows that

$$\sum_{i<j} c(\lambda_{i,j}) \ge \sum_{j=k+1}^{n} c(f_j), \tag{C.3}$$

completing the proof.

Proof of Corollary 4. Without loss of generality, assume that $f_0 = \max_i f_i$ and connect each node $i > 0$ to the node 0 with a channel such that $\lambda_{0,i} = f_i$. As $\sum_i f_i = 0$, it follows that $f_0 = -\sum_{i>0} f_i$, and thus, this network supports the underlying flow. Call this network G and note that $c(G) = \sum_i c(f_i) - \max_i c(f_i)$.

Let $M = \max(\sum_{f_i>0} c(f_i), \sum_{f_i<0} c(f_i))$. Then,

$$\sum_{i \in V} c(f_i) - \max_{i \in V} c(f_i) \le \sum_{i \in V} c(f_i) \tag{C.4}$$

$$= \sum_{f_i>0} c(f_i) + \sum_{f_i<0} c(f_i) \le 2M. \tag{C.5}$$

Remembering that M is an upper bound on the minimal cost of an LN that supports the given flow, $c(G) \le 2M$ as needed.

Proof of Theorem 5. Given a function satisfying the structural conditions, it suffices to construct flows $(f_i)_{i=1}^n$ for which the problem is NP-complete. Suppose that $f_i, f_j > 0$ for some $i \ne j$, whereas $f_k < 0$ for $k \ne i, j$, and that there exists a subset $S \subset \{f_k : f_k < 0\}$ such that $-\sum_{k \in S} f_k = f_i$. By Corollary 2, the cost-minimal network is bipartite with nodes i, j in one class and all other nodes in the other class. By Theorem 4, the total cost satisfies the lower bound

$$\sum_{f_k<0} c(f_k) \le c_{\min}(\mathcal{F}),$$

and by assumption, the bound is attained establishing channels of size f_k between i and each $k \in S$ and between j and each $k \notin S$, thereby obtaining a network with cost $\sum_{f_k<0} c(f_k)$, hence minimal. Thus, solving the cost-minimization problem is equivalent to finding a subset $S \subset \{f_k : f_k < 0\}$ for which $-\sum_{i \in S} f_k = f_i$. But this is the subset-sum problem, which is known to be NP-complete (see Karp 1972; Korte and Vygen 2011, corollary 15.27).

Appendix D. Proof of Theorem 6

Proof of Theorem 6. To prove (i), note that the function DISINTERMEDIATE preserves the direction of channels with 1 in that, if $\lambda_{1,i} > 0$ in its output (respectively, < 0), then also $\lambda_{1,i} > 0$ in its input (< 0).

Let p_l denote the number of pairs identified in the lth iteration of GETPAIRS and DISINTERMEDIATE. The algorithm stops at the first k for which $p_k = 0$. Because the degree of node 1 after the kth iteration is $n - 1 - \sum_{m=1}^{k} p_m$ and $p_m \ge 1$ for $m \le k - 1$, it follows that $k \le n$. Because each iteration invokes once GETPAIRS and DISINTERMEDIATE, which both have linear complexity, it follows that the algorithm has quadratic complexity.

To prove (ii), first note that Theorem 1 cannot improve the network because the algorithm stops precisely when the set of improvable pairs P is empty. Second, note that the algorithm starts with a connected graph on n nodes with $n - 1$ edges and remains so after each iteration because Theorem 1 does not change the number of edges. Thus, the graph is always a tree, and Lemma D.1 implies that hermetization cannot improve a tree.

To see (iii), recall that a star is already no more than twice as costly as the minimum by Theorem 4 even if DISINTERMEDIATE is never executed (as in the case in which node 1 is the sole payer or payee).

Lemma D.1. *If G is a forest, hermetization does not reduce costs.*

Proof of Lemma D.1. By contradiction, suppose there exists a set on nodes U that can be hermetized to reduce costs. To calculate the cost of the original LN, separate U into disjoint sets of vertices, say $L_0, L_1, .., L_k$, such that each $v \in L_i$ has at most one neighbor in $\cup_{j>i} L_j$. Let L_0 be the set of leaves in G. Let $G_0 = G \setminus L_0$, and L_1 is the set of leaves in G_0. Continue the process by defining $G_i = G_{i-1} \setminus L_i$, and L_{i+1} is the set of leaves in G_i. As the number of vertices becomes smaller at each step, the processes must end. For each node $i \notin L_k$, denote by $l(i)$ its single neighbor when it was a leaf. Note that each channel in the LN is between i and $l(i)$, and hence, the cost before the hermetization is $\sum_{i \in U \setminus L_k} c(\lambda_{i,l(i)})$. Going back to the original LN, denote by $N(i)$ the set of neighbors of node i. Note that $\{l(i)\} \subseteq N(i)$. The cost after hermetization is

$$\sum_{i \in U} c\left(\sum_{j \in N(i)} \lambda_{i,j} \right) \ge \sum_{i \in U \setminus L_k} c\left(\sum_{j \in N(i)} \lambda_{i,j} \right).$$

Because $c(\sum_{j \in N(i)} \lambda_{i,j}) \ge c(\lambda_{i,l(i)})$ for all $i \notin L_k$, it follows that hermetization did not improve the set U, thereby completing the proof.

Appendix E. An Example of a Cost Function

Guasoni et al. (2024) examine the costs of LN channels in a simple bilateral flow, obtaining the following result for the cost of a channel between two nodes. Guasoni et al. (2024, figure 4) show that first order expansions are very accurate for interest rates of up to 20%; therefore, this section focuses on such expansions for the sake of tractability.

Theorem E.1 (Guasoni et al. 2024). *Let λ_1 be the rate of transactions sent over the channel by node 1, and let λ_2 be the rate of transactions sent by node 2, and assume that $\lambda_2 \ge \lambda_1$. Denote by r the market interest rate, by B the cost of refunding a Lightning channel, and assume that the cost of opening a Lightning channel is $B/2$. Then, the first order approximation of a channel's cost is*

$$\begin{cases} 2\left(\frac{B(\lambda_2 - \lambda_1)}{r}\right)^{1/2} + O(\log(1/r)) & \text{for } \lambda_2 > \lambda_1 \\ 3\left(\frac{2B\lambda}{r}\right)^{1/3} & \text{for } \lambda_1 = \lambda_2 = \lambda. \end{cases}$$

And so, given a net flow of λ in the channel, the first order approximation of the cost function in Guasoni et al. (2024) is

$$c(\lambda) = 2\left(\frac{B|\lambda|}{r}\right)^{1/2}. \tag{E.1}$$

It is straightforward to check that this cost function indeed satisfies properties (MO), (SY), (CI), (SA):

(MO): $c(x) = 2\left(\frac{B|x|}{r}\right)^{1/2} > 2\left(\frac{B|y|}{r}\right)^{1/2} = c(y)$, for $x > y > 0$.

(SY): $c(x) = 2\left(\frac{B|x|}{r}\right)^{1/2} = 2\left(\frac{B|-x|}{r}\right)^{1/2} = c(-x)$.

(CI): $c(0) = 2\left(\frac{B\cdot 0}{r}\right)^{1/2} = 0$.

(SA): As $\sqrt{x+y} < \sqrt{x} + \sqrt{y}$ for any $x, y > 0$,

$$c(x+y) = 2\left(\frac{B(x+y)}{r}\right)^{1/2} < 2\left(\frac{Bx}{r}\right)^{1/2} + 2\left(\frac{By}{r}\right)^{1/2} = c(x) + c(y).$$

(SA+): Let $g(n) = n^{1/2-\epsilon}$ for some $\epsilon > 0$, then

$$c(nx) = 2\left(\frac{Bnx}{r}\right)^{1/2} = n^{1/2} 2\left(\frac{Bx}{r}\right)^{1/2} < \frac{n}{g(n)} c(x).$$

Appendix F. Lightning Network Snapshots

Table F.1. Summary Statistics of LN Snapshots (Decker 2023, August 23, 2022 to September 24, 2023)

Date	Nodes	Edges	Clustering coefficient (%)
2023-07-15	15,110	64,284	2.8
2023-06-15	14,071	60,059	3.0
2023-05-15	12,024	54,422	3.5
2023-04-15	NA	NA	NA
2023-03-15	8,887	44,191	3.9
2023-02-15	NA	NA	NA
2023-01-15	NA	NA	NA
2022-12-15	NA	NA	NA
2022-11-15	NA	NA	NA
2022-10-15	NA	NA	NA
2022-09-15	NA	NA	NA
2022-08-15	NA	NA	NA
2022-07-15	9,963	53,291	3.6
2022-06-15	5,217	17,658	2.5
2022-05-15	15,807	78,835	3.0
2022-04-15	18,739	81,390	2.8
2022-03-15	19,032	82,232	2.7
2022-02-15	3,681	5,669	3.7
2022-01-15	NA	NA	NA
2021-12-15	NA	NA	NA
2021-11-15	NA	NA	NA
2021-10-15	13,985	63,967	3.2
2021-09-15	14,289	66,672	3.6
2021-08-15	12,720	59,178	4.1
2021-07-15	11,362	49,776	3.8
2021-06-15	10,411	42,808	3.8
2021-05-15	9,727	39,064	4.0
2021-04-15	9,145	35,732	4.4
2021-03-15	7,955	32,307	4.8
2021-02-15	7,374	30,896	5.2
2021-01-15	6,806	29,768	5.7
2020-12-15	6,412	29,179	6.0
2020-11-15	6,104	28,729	6.1
2020-10-15	5,982	30,015	6.0

Endnotes

[1] This paper focuses on the aggregate costs for the network and, therefore, abstracts from payments that are costs for a user but benefits for others, such as routing fees.

[2] The leading order refers to a Taylor expansion around a zero interest rate. As shown in Guasoni et al. (2024, figure 4), the leading order approximation is accurate even for interest rates of 20%.

[3] For example, imagine two channels, one of size three with balances one and two for each party, the other of size seven with balances three and four for the same parties. If the first party makes payments of three, then the reset of either channel is necessary. Instead, a single channel of size 10 with balances of four and six, respectively, does not require a reset before the first party's payments reach four.

[4] We are grateful to an anonymous reviewer who provided this clearer interpretation of the (SA+) property.

[5] Current implementations of the Lightning Network require fees to be positive.

[6] The denominator is the sum of the number of ordered pairs of neighbors of each node and, hence, written in terms of nodes' degrees. The numerator cannot be written in terms of degrees only.

[7] Public information can be obtained from the Bitcoin blockchain, which records the opening and closing of each channel and its balance. A lightning node may also be able to observe the flow passing through its channels but not its origin and destination.

References

Bartolucci S, Caccioli F, Vivo P (2020) A percolation model for the emergence of the bitcoin lightning network. *Sci. Rep.* 10(1):4488.

Baumol WJ (1952) The transactions demand for cash: An inventory theoretic approach. *Quart. J. Econom.* 66(4):545–556.

Bhattacharya S, Sinha S, Roy S (2020) Impact of structural properties on network structure for online social networks. *Procedia Comput. Sci.* 167:1200–1209.

Brânzei S, Segal-Halevi E, Zohar A (2022) How to charge lightning: The economics of bitcoin transaction channels. Domínguez-García A, Raginsky M, eds. *2022 58th Annual Allerton Conf. Comm. Control, Comput. (Allerton 2022)* (Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ), 453–460.

Coase RH (1960) The problem of social cost. *J. Law Econom.* 3(1):1–44.

Decker C (2023) Lightning network research—Topology datasets. Accessed September 19, 2023, http://dx.doi.org/10.5281/zenodo.4088530.

Ersoy O, Roos S, Erkin Z (2020) How to profit from payments channels. Bonneau J, Heninger N, eds. *24th Internat. Conf. Financial Cryptography Data Security, FC 2020* (Springer, Berlin), 284–303.

Grunspan C, Lehéricy G, Pérez-Marco R (2020) Ant routing scalability for the lightning network. Preprint, submitted February 20, https://arxiv.org/abs/2002.01374.

Guasoni P, Huberman G, Shikhelman C (2024) Lightning network economics: Channels. *Management Sci.* 70(6):3827–3840.

Harris J, Zohar A (2020) Flood & loot: A systemic attack on the lightning network. Meiklejohn S, Shelat A, eds. *AFT '20: Proc. 2nd ACM Conf. Adv. Financial Tech.* (Association for Computing Machinery, New York), 202–213.

Kappos G, Yousaf H, Piotrowska A, Kanjalkar S, Delgado-Segura S, Miller A, Meiklejohn S (2021) An empirical analysis of privacy in the lightning network. Borisov N, Diaz C, eds. *25th Internat. Conf. Financial Cryptography Data Security, FC 2021* (Springer, Berlin), 167–186.

Karp RM (1972) Reducibility among combinatorial problems. Miller RE, Thatcher JW, eds. *Complexity of Computer Computations* (Plenum Press, New York), 85–103.

Kőnig D (1931) Gráfok és mátrixok. *Matematikai és Fizikai Lapok 38*, 116–119.

Korte B, Vygen J (2011) *Combinatorial Optimization*, vol. 1 (Springer, Berlin).

Lee S, Kim H (2020) On the robustness of lightning network in bitcoin. *Pervasive Mobile Comput.* 61:101108.

Lin JH, Primicerio K, Squartini T, Decker C, Tessone CJ (2020) Lightning network: A second path toward centralisation of the bitcoin economy. *New J. Phys.* 22(8):083022.

Martinazzi S, Flori A (2020) The evolving topology of the lightning network: Centralization, efficiency, robustness, synchronization, and anonymity. *PLoS One* 15(1):e0225966.

Miller MH, Orr D (1966) A model of the demand for money by firms. *Quart. J. Econom.* 80(3):413–435.

Papadis N, Tassiulas L (2020) Blockchain-based payment channel networks: Challenges and recent advances. *IEEE Access* 8: 227596–227609.

Papadis N, Tassiulas L (2022) Payment channel networks: Single-hop scheduling for throughput maximization. Mao S, ed. *41st IEEE Conf. Comput. Comm., INFOCOM 2022* (Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ), 900–909.

Papadis N, Tassiulas L (2023) Deep reinforcement learning-based rebalancing policies for profit maximization of relay nodes in payment channel networks. Pardalos P, Kotsireas I, Knottenbelt WJ, Leonardos S, eds. *Mathematical Research for Blockchain Economy. MARBLE 2023*, Lecture Notes in Operations Research (Springer, Cham, Switzerland), 1–27.

Pickhardt R, Nowostawski M (2020) Imbalance measure and proactive channel rebalancing algorithm for the lightning network. Plataniotis K, ed. *2nd IEEE Internat. Conf. Blockchain Cryptocurrency, ICBC 2020* (Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ), 1–5.

Poon J, Dryja T (2015) The bitcoin lightning network: Scalable off-chain instant payments. Satoshi Nakamoto Institute, Austin, TX.

Rohrer E, Tschorsch F (2020) Counting down thunder: Timing attacks on privacy in payment channel networks. Meiklejohn S, Shelat A, eds. *AFT '20: Proc. 2nd ACM Conf. Adv. Financial Tech.* (Association for Computing Machinery, New York), 214–227.

Rohrer E, Malliaris J, Tschorsch F (2019) Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks. Plataniotis K, ed. *4th IEEE Eur. Sympos. Security Privacy Workshops, EUROS PW 2019* (Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ), 347–356.

Roos S, Moreno-Sanchez P, Kate A, Goldberg I (2017) Settling payments fast and private: Efficient decentralized routing for path-based transactions. Preprint, submitted September 18, https://arxiv.org/abs/1709.05748.

Sali Y, Zohar A (2020) Optimizing off-chain payment networks in cryptocurrencies. Preprint, submitted July 18, https://arxiv.org/abs/2007.09410.

Seres IA, Gulyás L, Nagy DA, Burcsi P (2020) Topological analysis of bitcoin's lightning network. *Mathematical Research for Blockchain Economy* (Springer, Berlin), 1–12.

Sguanci C, Sidiropoulos A (2023) Mass exit attacks on the lightning network. Mnaouer B, Stiller, Karray, eds. *2023 IEEE Internat. Conf. Blockchain Cryptocurrency (ICBC)* (Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ), 1–3.

Shikhelman C, Tikhomirov S (2022) Unjamming lightning: A systematic approach. Cryptology ePrint Archive.

Sivaraman V, Venkatakrishnan SB, Ruan K, Negi P, Yang L, Mittal R, Fanti G, Alizadeh M (2020) High throughput cryptocurrency routing in payment channel networks. Bhagwan, Porter, eds. *17th USENIX Sympos. Networked Systems Design Implementation* (Association for Computing Machinery, New York), 777–796.

Tang W, Wang W, Fanti G, Oh S (2020) Privacy-utility tradeoffs in routing cryptocurrency over payment channel networks. Chaintreau, Golubchik, Zhang, eds. *Proc. ACM Measurement Anal. Comput. Systems*, vol. 4, issue 2 (Association for Computing Machinery, New York), 29:1–29:39.

Tobin J (1956) The interest-elasticity of transactions demand for cash. *Rev. Econom. Statist.* 38(3):241–247.

Varma SM, Maguluri ST (2021) Throughput optimal routing in blockchain-based payment systems. *IEEE Trans. Control Network Systems* 8(4):1859–1868.

Wang P, Xu H, Jin X, Wang T (2019) Flash: Efficient dynamic routing for offchain networks. Mohaisen, Zhang, eds. *15th Internat. Conf. Emerging Networking Experiments Tech. (CoNEXT '19)* (Association for Computing Machinery, New York), 370–381.

Wilson RJ (1979) *Introduction to Graph Theory* (Pearson Education India, Chennai, India).

Yu R, Xue G, Kilari VT, Yang D, Tang J (2018) Coinexpress: A fast payment routing mechanism in blockchain-based payment channel networks. *27th Internat. Conf. Comput. Comm. Networks* (IEEE, Piscataway, NJ), 1–9.

Zhang X, Tang S, Zhao Y, Wang G, Zheng H, Zhao B (2017) Cold hard e-cash: Friends and vendors in the Venmo digital payments system. Zhang, Tang, Zhao, Wang, Zheng, Zhao, eds. *Proc. Eleventh Internat. AAAI Conf. Web Social Media (ICWSM 2017)* (PKP Publishing Services Network, Burnaby, BC), 387–396.