How to profit from payments channels

Ersoy, Oğuzhan; Roos, Stefanie; Erkin, Zekeriya

Publication date
2020

Published in
Proceedings of the 24th Financial Cryptography and Data Security, Kota Kinabalu, Sabah, Malaysia

Citation (APA)
Ersoy, O., Roos, S., & Erkin, Z. (2020). How to profit from payments channels. In *Proceedings of the 24th Financial Cryptography and Data Security, Kota Kinabalu, Sabah, Malaysia*

How to profit from payments channels

Oğuzhan Ersoy, Stefanie Roos and Zekeriya Erkin

Delft University of Technology
{o.ersoy, s.roos, z.erkin}@tudelft.nl

Abstract. Payment channel networks like Bitcoin's Lightning network are an auspicious approach for realizing high transaction throughput and almost-instant confirmations in blockchain networks. However, the ability to successfully conduct payments in such networks relies on the willingness of participants to lock collateral in the network. In Lightning, the key financial incentive to lock collateral are low fees for routing payments of other participants. While users can choose these fees, real-world data indicates that they mainly stick to default fees. By providing insights on beneficial choices for fees, we aim to incentivize users to lock more collateral and improve the effectiveness of the network.
In this paper, we consider a node **A** that given the network topology and the channel details establishes channels and chooses fees to maximize its financial gain. Our contributions are i) formalization of the optimization problem, ii) proving that the problem is NP-hard, and iii) designing and evaluating a greedy algorithm to approximate the optimal solution. In each step, our greedy algorithm establishes a channel that maximizes the increase to **A**'s total reward, which corresponds to maximizing the number of shortest paths passing through **A**. Our simulation study leveraged real-world data sets to quantify the impact of our gain optimization and indicates that our strategy is at least a factor two better than other strategies.

1 Introduction

Payment channel networks [14] overcome the need to globally agree on every transaction in a blockchain. Instead, nodes can open and close *channels* that they can use to transfer coins directly. In the absence of disputes, transactions only require local communication between the parties involved in a transaction. Nodes without a direct payment channel can route payments via intermediaries to avoid the transaction fees and delays of channel opening. Thus, by moving transactions off-chain, payment channels have the potential to drastically increase the transaction throughput while reducing the confirmation times from tens of minutes to sub-seconds. The most notable examples of payment channel networks are Bitcoin's Lightning [19] and Ethereum's Raiden [2].

When opening a payment channel, nodes need to lock coins that they cannot use outside of the channel during the lifetime of the channel. This opportunity cost makes it unattractive to maintain payment channels. However, routing payments in a network requires that the network has well-funded channels [14].

The key incentives for locking collateral in a channel are i) frequent transaction with the other party [7] and ii) financial gain through routing fees [11], i.e., fees that nodes charge for routing payments as intermediaries. Our analysis of the Lightning network shows that the fees charged for routing are currently low and mainly equal to the default value [21]. We conjecture that the current payment channel networks primarily rely on the first incentive. However, research on the Lightning network suggests that this incentive entails networks of a low resilience with a few central hubs [22]. Analyzing the second incentives and show-casing that payment channels can entail financial profit is the most promising avenue of research to incentivize the participation in payment channel networks and fully leverage the potential of this promising blockchain scalability approach.

In this paper, we adapt a payment channel network (PCN) model based on Lightning. We assume a known topology and fees. Nodes select the cheapest path to conduct a payment. A node **A** aims to maximize its profit through routing fees by choosing both its payment channels and fees. The problem is challenging as higher fees indicate a higher profit if the node routes the payment but also a lower probability to be chosen for routing due to the transactions taking the cheapest path.

Despite the importance of fees in payment channel networks, the issue has been mainly ignored in past research. The majority of papers deal with cryptographic protocols for channel establishment and multi-hop payments (e.g., [6, 7, 10, 15, 17]) as well as algorithms for routing payments (e.g., [16, 20, 23]). There is some work on comparing routing fees to the on-chain fees of blockchains and presenting an economical analysis of the relation between the two fee types [5, 11]. It is interesting to note that routing fees are related to the payment value whereas on-chain blockchain fees usually relate to the size of the transactions. In contrast, Di Stasi et al. [24] evaluated the impact of routing fees on keeping channels balanced, i.e., ensuring that a channel is not used exclusively in one direction. The authors suggest a novel linear fee policy for each channel to improve channel balances. Most similar to our work, Avarikioti et al. [3] studied the optimal fee assignment of channels from the point of view of a payment service provider (PSP). The authors analyzed optimal channel fees of the whole network that maximizes the total reward of the PSP instead of focusing on a node, which defines our problem. However, the authors can only solve for tree-structured networks, which does not make the approach useful in practice.

We are hence the first to cover the aspect of maximizing fees in payment channel networks. More precisely, we formalize the problem of maximizing fees in a Lightning-inspired system model. We present an algorithm for solving the defined optimization problem heuristically. Our greedy algorithm iteratively i) adds channels and ii) selects fees such that each added channel increases the profit maximally for the previously selected channels. For this purpose, we leverage the concept of (edge) betweenness centrality, i.e., the fraction of cheapest paths a vertex or edge is contained in. We evaluate our algorithm for real-world data sets of the Lightning network. Our evaluation strongly indicates that our approach does not only greatly improve the profit in comparison to default fees but also

that leveraging betweenness centrality for selecting channels offers considerably better results than other network centrality measures. More preciously, our algorithm increases the profit by a factor 4 in comparison to default fee values and is at least a factor 2 better than other strategies. Our evaluation further demonstrates that nodes with already established channels can increase their profit by utilizing only our fee selection algorithm without establishing new channels.

2 Background

This section summarizes key concepts from the field of payment channels. Furthermore, as our algorithm relies on graph centrality metrics, this section defines these metrics and gives some intuition on their role.

2.1 Payment Channel Networks

Payment channel networks are one key approaches to scaling blockchains by moving transactions off-chain [14]. Two parties open a payment channel through an initial funding transaction on the blockchain that locks coins such that they can only be used for transactions between the two parties. After this initial funding transaction, the two parties can conduct payments without directly interacting with the blockchain. They commit to the latest balance of the channel, i.e., the distribution of the total number of locked coins over the two parties. For instance, let nodes u and v open a payment channel such that u locks x coins and v locks y coins. The initial *balance* of the channel is (x, y) and its total *capacity* is $x + y$. If u sends one coin to v, the balance changes to $(x - 1, y + 1)$.

In case of a dispute about the channel balance, the signed commitments documenting the state changes are published on the blockchain. The blockchain consensus then assigns the coins according to the latest valid channel state. Once the two parties decide to close their channel, they have to conduct a closing transaction on the blockchain. Afterward, they receive the coins locked in the channel with the number of coins per party corresponding to the channel balance at the time of the closure. In the absence of disputes, the intermediary transactions are almost instant and the number of transaction is merely bound locally by the bandwidth and latency of nodes.

Establishing a payment channel does not make sense if parties do not trade with each other regularly due to i) the on-chain fees for establishing the channel and ii) the opportunity cost caused by locking coins to the channel. Thus, most nodes will only establish a few channels with frequent trading partners. Routing payments via a path consisting of multiple channels nevertheless allows nodes to trade without having a direct channel. For instance, a node s can make a payment to a node r via two intermediary nodes u and v, meaning that the payment is routed via three payment channels: s to u, u to v, and v to r. The balances along all these channels change according to the transaction value.

The intermediary nodes charge fees for the use of their channels. For a channel Ch_i from u to v, these fees consist of a basic fee BF_{Ch_i} for using the channel

and fee rate FR_{Ch_i} per transferred unit. The overall fee of a transaction tx for the channel is hence

$$\mathbf{f}(Ch_i, tx) = BF_{Ch_i} + FR_{Ch_i} \cdot |tx|, \tag{1}$$

where $|tx|$ denotes the transaction amount. The fees are determined by and paid to u. The sender s has to pay the fees. Note that the fee calculation formula given in Equation 1 is specific to the Lightning network [1]. Still, the other payment or state channel networks have a similar structure.

2.2 Graph Centrality Metrics

In this work, we model a PCN network as a directed graph. In this manner, each node in the payment channel represents a vertex in the graph and each channel is represented by two directional edges between the nodes (one for each direction). The channel fees correspond to the weights of the edges.

As a consequence, we can make use of graph metrics that characterize the importance of certain nodes in a weighted directed graph. Our key metrics are (vertex) betweenness centrality and edge betweenness centrality.

Definition 1 (Betweenness Centrality). *The betweenness centrality of a vertex [12] v is proportional to the total number of shortest paths that pass through that vertex, i.e.,*

$$\mathbf{bc}(v) = \sum_{\substack{s \neq t \neq v \\ \sigma_{st} \neq 0}} \frac{\sigma_{stv}}{\sigma_{st}},$$

where σ_{st} denotes the number of shortest paths between s and t and σ_{stv} is the number of such shortest paths containing the vertex v.

Similarly, the edge betweenness centrality [13] of an edge relates to the total number of shortest paths that pass through that edge, i.e.,

$$\mathbf{e}([v_1 v_2]) = \sum_{\substack{s \neq t \\ \sigma_{st} \neq 0}} \frac{\sigma_{st[v_1 v_2]}}{\sigma_{st}},$$

where $\sigma_{st[v_1 v_2]}$ is the number of shortest paths passing through the edge $[v_1 v_2]$.

The analysis of this paper makes use of the following result about vertex betweenness centrality to assess the suitability of our greedy heuristic for selecting channel fees.

Theorem 1 ([4]). *For each vertex v, betweenness centrality function $\mathbf{bc}(v)$ is a monotone function for the set of edges incident to v.*

An important problem concerning the betweenness centrality is the *maximum betweenness improvement* (MBI) problem.

Definition 2 (MBI problem [4]). Maximum Betweenness Improvement *Problem: Given a directed graph G and a vertex v, find k edges incident to node v such that* $\mathbf{bc}(v)$ *is maximal.*

With the help of the following theorem concerning the MBI problem, we prove that our problem of maximizing the reward (MRI) is NP-hard.

Theorem 2 ([4]). *MBI problem cannot be approximated in polynomial time within a factor greater than* $1 - \frac{1}{2\epsilon}$*,unless* $P = NP$*.*

3 Our PCN Model

There are a number of PCNs with Lightning [19], Raiden [2], Perun [9] and Celer [8] being key examples. All of them use slightly different assumptions and properties. We base our system model on Bitcoin's Lightning network.

In the following, we first describe our PCN model **LN**. In this model, we then define the problem of an individual participant aiming to maximize their gain. We summarize the notation used in the paper in Table 1.

Table 1. Notation and Abbreviation Table

Symbol	Explanation
`CSF`	The channel selection function.
`CFF`	The channel fee function.
LN	The payment channel network.
$\mathbf{c}(X)$	The total amount of coins of X.
$\mathbf{f}(Ch, tx)$[1]	The charging fee of the channel Ch for a transaction of value tx.
$\mathbf{bc}(n, \mathbf{N})$[1]	The betweenness centrality of the node n in a network $\mathbf{N}$.
$\mathbf{e}(Ch, \mathbf{N})$[1]	The edge betweenness centrality of the channel Ch in a network $\mathbf{N}$.
$\mathbf{s}(Ch_i)$, $\mathbf{r}(Ch_i)$	The source and destination nodes of the channel Ch_i.
$ChCost$	The channel opening and closing on-chain cost.

3.1 Network Topology, Fees, and Routing

Nodes open and close payment channels through blockchain transactions. For simplicity, we assume that the cost $ChCost$ of opening and closing remains constant over time.

In Lightning, the complete topology of the network is known to every node. Nodes publicly announce on the blockchain that they establish or close a channel.

[1]For brevity in the notation, tx and $\mathbf{N}$ can be omitted unless they alter with time.

Furthermore, nodes willing to route payments announce their channels and fees to the complete network. Thus, we assume in our model that both the topology and the fees of all nodes are publicly known. For simplicity, we assume that the topology and routing fees of the nodes that do not strategically change them remain fixed over time. Otherwise, our fee selection strategy would require a model to anticipate the expected changes. Current research on payment channel networks does not provide such a model. Our analysis of the Lightning network data from $1ml.com$ indicates that fees are indeed usually the default value. As topology changes require on-chain transactions, which are costly in both time and on-chain fees, the topology also should not change considerably. Moreover, we assume that nodes apply source routing to find one cheapest path from source to destination, as is the case in the current implementation of Lightning.

3.2 Problem Definition

We represent a network $\mathbf{LN}$ as a graph $G = (V, E)$ of vertices V and edges E. A node $\mathbf{A}$ aims to maximize its revenue in running a node in a payment channel network. For this purpose, $\mathbf{A}$ opens channels with other nodes in the network, each channel having a total cost of $ChCost$ for opening and closing. We assume that $\mathbf{A}$ can strategically select the nodes it establishes channels with from all nodes in the network. After all, these nodes do not need to invest anything into the channel as $\mathbf{A}$ completely funds them and they will likely receive additional monetary gains through routing fees. Furthermore, $\mathbf{A}$ has a budget of $\mathbf{c}(\mathbf{A})$ coins to use as collateral for the channels in total.

Formally, let C be the set of channels established by $\mathbf{A}$. For each channel $Ch_i \in C$, we have the coins allocated to the channel $\mathbf{c}(Ch_i)$ and the channel fee $\mathbf{f}(Ch_i, tx)$ for a transaction value tx. Wlog, transaction values are integers between 1 and $\mathbf{T}_{max}$ following a distribution T. Let $X_i(tx, S, R)$ be the event that a transaction of value tx going from a node S to a node R passes through the channel Ch_i. Then the expected fee from that transaction is $\mathbf{f}(Ch_i, tx)Pr[X_i(tx, S, R)]$. Last, we require the distribution M that returns a sender-receiver pair. $\mathbf{A}$'s objective is to find C, f, and $\mathbf{c}()$ such that the overall expected gain of one transaction

$$\sum_{\substack{\forall S,R \in V \\ S \neq R \neq \mathbf{A}}} Pr(M = (S, R)) \sum_{j=1}^{\mathbf{T}_{max}} Pr(T = j) \sum_{Ch_i \in C} \mathbf{f}(Ch_i, j) \cdot Pr[X_i(j, S, R)] \quad (2)$$

is maximized while adhering to the constraint that $\sum_{Ch_i \in C} \mathbf{c}(Ch_i) \leq \mathbf{c}(\mathbf{A})$. Equation 2 computes the expected gain over the involved variables T and M. If the capacity of the channel $\mathbf{c}(Ch_i)$ is less than the transaction amount tx, $Pr[X_i(tx, S, R)] = 0$. Similarly, if there does not exist a shortest path from S to R that passes through Ch_i, $Pr[X_i(tx, S, R)] = 0$. Otherwise, $Pr[X_i(tx, S, R)]$ is equal to the number of shortest paths from S to R passing through Ch_i divided by the total number of shortest paths from S to R.

Note that Equation 2 ignores the cost of opening C channels, $|C| \cdot ChCost$. The impact of this cost depends on the number of transactions K that occur during the lifetime of a channel. Let max be the maximal value for Equation 2. The overall gain of the node is then the difference: $K \cdot max - |C| \cdot ChCost$. By increasing the lifetime of the channel arbitrarily, the impact of $|C| \cdot ChCost$ diminishes, which is why we disregard it for Equation 2. Our model furthermore disregards the opportunity cost caused by locking coins due to the absence of suitable models for such a cost.

4 Our Fee Strategy

We start by showing that maximizing the objective function given in Equation 2 is NP-hard. Afterwards, we present our greedy algorithm for approximating a solution. As our algorithm contains an equation for choosing channel fees without a closed-form solution, the last part of the section demonstrates a method for solving the equation numerically.

Our proof and algorithm act on a version of Equation 2 for specific distributions T and M. In the absence of real-world data for these distributions, we utilize two straight-forward distributions. Concretely, our work considers a fixed transaction value, i.e., the random variable T only takes one value tx. For the distribution M, which characterizes the likelihood of two nodes to trade, assuming that all nodes are equally likely to trade with each other is the most natural choice in the absence of a concrete alternative model. Thus, M is a uniform distribution over all pairs of nodes in the following.

For the design of our algorithm, we furthermore bound the maximal channel fee by f_{max}. Assuming a maximal channel fee does not reduce the generality of our approach. As nodes send payments along the path with the lowest fee, any channel fee that entails the channel is not contained in any such path can be disregarded.

4.1 NP-hardness of the Problem

Before presenting the actual proof, we rephrase Equation 2 to relate it to the concept of (edge) betweenness centrality.

Choosing M to be a uniform distribution implies that $Pr(M = (S, R)) = \frac{1}{(|V|-1)(|V|-2)}$[2] is a constant, which can disregarded for the optimization. Furthermore, choosing a constant transaction value tx removes the second sum in Equation 2. Hence our modified objective function is

$$\sum_{Ch_i \in C} \mathbf{f}(Ch_i, tx) \cdot Pr[X_i(tx, S, R)]. \tag{3}$$

The next step relates $Pr[X_i(\text{tx}, S, R)]$ in Equation 3 to the betweenness centrality. There are two important quantities to consider: the number of shortest paths including the channel and total fee reward gained from these paths.

[2] $(|V| - 1)(|V| - 2)$ is the number of pairs of nodes when not including $\mathbf{A}$

Maximizing the number of shortest paths passing through a channel or node corresponds to the edge or vertex betweenness centrality (BC), respectively, as defined in Section 2. However, maximizing the BC does not necessarily imply maximal revenue. As fees represent edge weights, the shortest path here is a path whose edges have the minimal sum of weights. Choosing low fees hence increases the probability to be contained in the shortest path but low fees also indicate a low gain from each transaction.

Rather, the expected reward of a channel Ch_i is equal to the probability of the transaction passing through that channel times the fee. Note that each channel needs to have a capacity of at least tx for the payment to choose this path. Thus, an optimal solution for Equation 3 will only create channels of sufficient capacity and we can exclude the capacity aspect from $Pr[X_i(tx, S, R)]$. With $\mathbf{e}(Ch_i)$ denoting the edge betweenness centrality of a channel Ch_i with fees $\mathbf{f}(Ch_i)$[3], the formal expression for the expected reward of Ch_i is

$$\mathsf{ER}(Ch_i) = \mathbf{f}(Ch_i) \cdot \mathbf{e}(Ch_i). \tag{4}$$

As a consequence, the total expected reward of $\mathbf{A}$ from Equation 3 is

$$\mathsf{ER}(\mathbf{A}) = \sum_{Ch_i \in C} \mathsf{ER}(Ch_i). \tag{5}$$

Now, we can formally define the problem from Equation 2 as the *maximum reward improvement* (MRI) problem.

Definition 3 (MRI Problem). Maximum Reward Improvement *problem: For a payment channel network $\mathcal{LN}$ and a node n, find k channels incident to node n such that $\mathsf{ER}(n)$ is maximized.*

The following theorem states that it is not possible to design an algorithm `CSF` that finds the optimum solution within polynomial time, unless $P = NP$.

Theorem 3 (MRI Approximation Theorem). MRI *problem cannot be approximated in polynomial time within a factor greater than $1 - \frac{1}{2\epsilon}$, unless $P = NP$.*

Proof. To prove this theorem, we reduce our MRI problem to the MBI problem presented in Definition 2. Using Equation 5, we can formulate the MRI problem as follows:

$$\mathsf{MRI}(\mathbf{LN}, n, k) \rightarrow \mathcal{CH}_M = \underset{\substack{|\mathcal{CH}| \leq k \\ \mathbf{s}(Ch_i) = n \\ \mathbf{f}(Ch_i) \in [1, f_{max}]}}{\operatorname{argmax}} \left(\mathsf{ER}(n) = \sum_{Ch_i \in \mathcal{CH}} \mathsf{ER}(Ch_i) \right).$$

We introduce a subproblem, namely MRI_FF, where the upper limit of the fee f_{max} is equal to 1, which means that all the channel fee are equal to 1. Using the Equation 4, MRI_FF can be formulated as:

[3]For the rest of section, we drop the transaction amount tx from the channel fee formula $\mathbf{f}(Ch_i)$ as it is fixed.

$$\mathsf{MRI\_FF}(\mathbf{LN}, n, N_c) \rightarrow \mathcal{CH}_M = \underset{\substack{|\mathcal{CH}|\leq k \\ \mathbf{s}(Ch_i)=n}}{\operatorname{argmax}} \left(\sum_{Ch_i \in \mathcal{CH}} \mathbf{e}(Ch_i) \right) \tag{6}$$

$$\overset{(*)}{=} \underset{\substack{|\mathcal{CH}|\leq k \\ \mathbf{s}(Ch_i)||\mathbf{r}(Ch_i)=n}}{\operatorname{argmax}} (bc_n) \overset{(**)}{=} \mathsf{MBI}(\mathbf{LN}, n, k),$$

which reduces to the MBI problem. Here, the first equality $(*)$ holds because the summation of the all shortest paths passing from out-going edges is equal to the total number of shortest paths passing through that node. In other words, the summation of edge betweenness centrality of all out-going edges of a node is equal to betweenness centrality of that node. The second equality $(*)$ follows from the definition of the MBI problem given in Definition 2.

Now, we can prove our theorem by contradiction. Let assume there exists an approximation to MRI problem within a factor greater than $1 - \frac{1}{2\epsilon}$. Then, the same approximation would hold for the subproblem of MRI, MRI_FF with a certain maximal fee, namely 1. However, in Equation 6, we showed that MRI_FF problem is equivalent to the MBI problem. This contradicts Theorem 2. Therefore, MRI problem cannot be polynomially approximated within a factor greater than $1 - \frac{1}{2\epsilon}$, unless $P = NP$. □

4.2 Channel Selection Function

We present a *greedy* algorithm `CSF` to approximate the MRI problem. `CSF` takes the PCN and the requested number of channels as input and outputs the set of nodes to whom channels are created. It internally calls `CFF`, the algorithm for deciding the fee of a channel. Formally, we have

$$\begin{aligned}
&\texttt{CFF}(\mathcal{CH} \cup Ch) \rightarrow R_{Ch} : \\
&R_{Ch} = \mathsf{TotalER}(\mathcal{CH} \cup Ch, f) \text{ where } f = \underset{f_i \in [1, f_{max}]}{\operatorname{argmax}} \left(\mathsf{TotalER}(\mathcal{CH} \cup Ch, f_i) \right), \\
&\mathsf{TotalER}(\mathcal{CH} \cup Ch, f_i) = \mathsf{ER}(Ch)_{\mathbf{f}(Ch)=f_i} + \sum_{Ch_j \in \mathcal{CH}} \mathsf{ER}(Ch_j).
\end{aligned} \tag{7}$$

As detailed in Algorithm 1, our greedy algorithm for `CSF` consists of the following five key steps:

1. Start with an initial PCN of nodes and channels.
2. At each step, try all possible channels between our node and other nodes.
3. Compute the maximum reward of the channel by using `CFF`.
4. Connect to the node who gives the maximum reward and update the PCN.
5. Go to step (2) until the desired number of channels is established.

Next, we ascertain that channel additions cannot reduce the expected revenue, indicating that nodes should add all channels they can fund. Here, it is

Algorithm 1 Channel Selection Function

Input: $\mathbf{LN}$ and N_c
Output: $\mathcal{CH}$

function CSF($\mathbf{LN}, N_c$)
  $\mathcal{CH} \leftarrow \emptyset$
  while $|\mathcal{CH}| < N_c$ **do**
    $maxRew \leftarrow 0, selectednode = None$
    for Each node $n_i \in \mathbf{LN}$ **do**
      Create a channel between (n, n_i): $\mathbf{LN}_i \leftarrow AddEdges(\mathbf{LN}, [n, n_i])$
      Calculate the reward $R_{n_i} \leftarrow$ CFF($\mathbf{LN}_i, \mathcal{CH} \cup [n, n_i]$)
      if $maxRew \leq R_{n_i}$ **then**
        $maxRew = R_{n_i}$
        $selectednode = n_i$
      end if
    end for
    $\mathcal{CH} \leftarrow \mathcal{CH} \bigcup \{selectednode\}$
    $\mathbf{LN} \leftarrow AddEdges(\mathbf{LN}, [n, selectednode])$
  end while
  return $\mathcal{CH}$
end function

important to note that we do not take into account the channel opening cost $ChCost$. Thus, if the marginal reward improvement of a new channel is zero, there is no point in add the channel.

Theorem 4 (Monotonicity). *The objective function of Algorithm 1 is a monotone non-decreasing function.*

Proof. A function $\mathcal{F} : \Omega \rightarrow \mathbb{R}$ is a monotone function if it satisfies the following condition:

$$\forall S \subseteq T \subseteq \Omega, \quad \mathcal{F}(S) \leq \mathcal{F}(T). \tag{8}$$

In our case, we have to show that CFF($\mathcal{CH} \cup [n, n_i]$) $\geq$ CFF($\mathcal{CH}$) for any solution $\mathcal{CH}$ and node n_i such that $[n, n_i] \notin \mathcal{CH}$ where $\mathcal{CH}$ is the current channel list of node n.

Note that CFF checks for all possible fee values to maximize the total reward. In that sense, it would be enough to show that for the maximum fee value f_{max}, which can be formulated by using Equation 7 (with $\mathbf{LN}_0 = \mathbf{LN} \cup \mathcal{CH}$ and $\mathbf{LN}_i = \mathbf{LN} \cup \mathcal{CH} \cup [n, n_i]$):

$$\texttt{CFF}(\mathbf{LN}, \mathcal{CH} \cup [n, n_i]) \geq \mathsf{TotalER}(\mathbf{LN}, \mathcal{CH} \cup [n, n_i], f = f_{max}) \overset{?}{\geq} \texttt{CFF}(\mathbf{LN}, \mathcal{CH})$$

$$\iff \mathsf{ER}(Ch, \mathbf{LN}_i)_{f=f_{max}} + \sum_{\forall Ch_j \in \mathcal{CH}} \mathsf{ER}(Ch_j, \mathbf{LN}_i) \overset{?}{\geq} \sum_{\forall Ch_j \in \mathcal{CH}} \mathsf{ER}(Ch_j, \mathbf{LN}_0)$$

$$\iff \mathsf{ER}(Ch, \mathbf{LN}_i)_{f=f_{max}} \overset{?}{\geq} \sum_{\forall Ch_j \in \mathcal{CH}} \mathsf{ER}(Ch_j, \mathbf{LN}_0) - \mathsf{ER}(Ch_j, \mathbf{LN}_i)$$

$$\iff \mathbf{e}([n, n_i], \mathbf{LN}_i) \cdot f_{max} \overset{?}{\geq} \sum_{\forall Ch_j \in \mathcal{CH}} (\mathbf{e}(Ch_j, \mathbf{LN}_0) - \mathbf{e}(Ch_j, \mathbf{LN}_i)) \cdot \mathbf{f}(Ch_j)$$

$$\overset{(*)}{\Longleftarrow} \mathbf{e}([n, n_i], \mathbf{LN}_i) \overset{?}{\geq} \sum_{\forall Ch_j \in \mathcal{CH}} (\mathbf{e}(Ch_j, \mathbf{LN}_0) - \mathbf{e}(Ch_j, \mathbf{LN}_i))$$

$$\iff \mathbf{e}([n, n_i], \mathbf{LN}_i) + \sum_{\forall Ch_j \in \mathcal{CH}} \mathbf{e}(Ch_j, \mathbf{LN}_i) \overset{?}{\geq} \sum_{\forall Ch_j \in \mathcal{CH}} \mathbf{e}(Ch_j, \mathbf{LN}_0)$$

$$\overset{(**)}{\Longleftrightarrow} \mathbf{bc}(n, \mathbf{LN}_i) \overset{?}{\geq} \mathbf{bc}(n, \mathbf{LN}_0).$$

Here, $(*)$ condition is true since for all channels $\mathbf{f}(Ch_i) \leq f_{max}$ by the definition. Also, each term $\mathbf{e}(Ch_j, \mathbf{LN}_0) - \mathbf{e}(Ch_j, \mathbf{LN}_i)$ is non-negative as new channels of node n cannot increase the number of shortest paths passing through existing channels of the same node. Thus, the multiplication with a positive number preserves the inequality. $(**)$ is satisfied since the summation of edge betweenness centrality of all out-going edges of a node is equal to betweenness centrality of that node. At the end, $\mathbf{bc}(n, \mathbf{LN}_i) \geq \mathbf{bc}(n, \mathbf{LN}_0)$ holds because betweenness centrality is a monotone function, see Theorem 1. □

4.3 Efficient Search Algorithm for the Channel Fee Function

No closed-form formula finds the best fee amount maximizing the expected reward due to the term $\mathbf{e}(Ch)$ for a channel Ch. Here, we analyze Equation 4 to minimize the computational cost by discarding some parts of the search space. First of all, since $\mathbf{e}(\mathbf{LN})$ is not affected by changes to the fees of channels, the denominator is irrelevant for optimizing the $\mathsf{ER}(Ch)$. Therefore, $\texttt{CFF}$ can be seen as a function of the edge betweenness centrality of the channel $\mathbf{e}(Ch)$ and its fee $\mathbf{f}(Ch)$. Secondly, $\mathbf{e}(Ch)$ is negatively affected by $\mathbf{f}(Ch)$ because increasing the fee means an increase in the weight of the edge that results in a lower chance of being in the shortest paths.

Two observations give rise to an efficient search algorithm for finding the most suitable fee. The first observation utilizes the fact that edge betweenness centrality is a monotone decreasing function concerning the channel fee. Let the expected reward of a channel for chosen fees $f_3 > f_1$ be $r_1 = e_1 \cdot f_1$ and $r_3 = e_3 \cdot f_3$, respectively. If $r_3 > r_1$, let

$$f_2 = f_1 \cdot \frac{r_3}{r_1} = f_3 \cdot \frac{e_3}{e_1}. \tag{9}$$

It can be seen that the expected reward r_α for any fee f_α where $f_1 < f_\alpha \leq f_2$ is at most r_3:

$$r_\alpha = e_\alpha \cdot f_\alpha \leq e_1 \cdot f_\alpha \leq e_1 \cdot f_2 = e_3 \cdot f_3 = r_3. \tag{10}$$

In other words, there is no need to compute the expected reward values for the fees in between f_1 and f_2 as they cannot be optimal values.

The second observation is that increasing the fee of an out-going channel Ch cannot decrease the edge betweenness of another out-going channel Ch' of the same node. Such an increase can only reduce the edge betweenness of channels that are on a path containing Ch by removing the path from the set of shortest paths. However, as shortest paths cannot have loops, two out-going channels of the same node cannot be on the same shortest path. Now, let $\mathcal{CH}$ be the set of previously selected channels. Let r'_1 and r'_3 be the sum of the expected fees of all channels $Ch' \in \mathcal{CH}$ for fees f_1 and f_3 with $f_3 > f_1$. By the above observation, we have $r'_3 \geq r'_1$.

Our recursive algorithm divides the space of all possible fee values from 1 to f_{max} into d intervals. For each interval i, let $r_i = \mathsf{ER}(Ch, \mathbf{f}(Ch) = f_i)$ be the expected reward of Ch and $r'_i \leftarrow \sum_{Ch' \in \mathcal{CH}} \mathsf{ER}(Ch', \mathbf{f}(Ch'))$ be the total reward of the other channels. By the first observation, the maximal increase in r_i is $\frac{f_{i+1}}{f_i}$ and by the second observation $r'_{i+1} \geq r'_i$ as $f_{i+1} > f_i$. Thus, the maximum possible reward value for interval i is $\widetilde{R}_i = r_i \cdot \frac{f_{i+1}}{f_i} + r'_{i+1}$. If $\widetilde{R}_i$ is greater than the current maximum reward value, the algorithm recursively searches for a maximum in the interval, otherwise discards the interval.

This completes the description of our algorithm, which we evaluate in the following in comparison to other approaches based on common centrality metrics.

5 Evaluation

In this section, we evaluate our proposed fee strategy for a real-world topology. Our evaluation quantifies the total reward gained by $\mathbf{A}$ when using our greedy algorithm.

To emphasize the high effectiveness of our solution, we compared it with other channel and fee selection algorithm. For the channel selection, we considered random nodes as well as connecting to nodes with a high centrality for three centrality metrics: i) degree, i.e., connecting to the nodes with the most connections, ii) betweenness centrality, and iii) pagerank [18]. For the fee strategy, we compute the results for both cases where the channel fees are the default values and they are determined by `CFF`.

5.1 Model

In Lightning network, the upcoming transactions and current balances of channels are not known. Thus, we need to model the network and transactions.

Transactions. Like Section 4, our evaluation assumes that all source-destination pairs are equally likely. Furthermore, we categorize the transactions into three groups based on the amounts:

- *Micro payments* are the transactions involving a very small amount of coins. To represent this category, we use the transaction amount of 100 Satoshi, which is about one cent[4]. An example of a use case would be the streaming services where you pay small amounts per service.
- *Medium payments*: are the transactions spent for daily living expenses like buying a coffee, which is represented with 10000 Satoshi.
- *Macro payments*: are transactions of high amounts, which is represented with 1000000 Satoshi. The amount of these transactions are in the order of 100 Euros.

From these categories, it is most likely that micro payments are usually restricted to nodes that have a direct channel. Otherwise, the base fee for the payment greatly exceeds the actual payment value. Therefore, our target transactions are medium and macro payments, which are analyzed separately.

Network. Following our system model in Section 3, networks are represented as weighted directed graphs. The weights of the edges in the graph model are calculated according to the fee rate and base fees of the channels. Since the fee rate depends on the transaction amount, the weights of the same edges for medium and macro payments will be different. The graph generated for the medium (macro) payments is called medium (macro) graph.

5.2 Setup

We obtained a snapshot of the Lightning Network (LN) data from $1ml.com$ on July 10 2019, which contains 4618 nodes and 68729 edges in total. When we delete the edges with insufficient capacity, the medium graph has 68697 edges and the macro graph has 32193 edges.

As a node requires at least two connections to be contained in any shortest paths, we first connected **A** to the two nodes with the highest degree (, which happen to have the highest pagerank as well). For these two connections, we use the default fee rate and base fee values in both directions of the edges. Based on this initial scenario, we now connect **A** to additional nodes.

The experiments use $ChCost = 8192$ Satoshi, which reflects the fluctuating Bitcoin transaction fee estimates[5]. When establishing a new channel, our simulation added edges in both directions. The base fee and the fee rate of the in-coming edge corresponded to the default value to model that i) most users currently stick to the default values and ii) **A** has no control over the in-coming channel fees as they are determined by the other party. For the outgoing edges, we utilize either `CFF` to determine the best fee value or use default values. When

[4] https://awebanalysis.com/en/convert-satoshi-to-euro-eur/

[5] https://bitcoinfees.info/

using `CFF`, we set $f_{max} = ChCost$. Otherwise, the total fee cost of the transaction in the payment network is higher than the cost in the Bitcoin network and the sender is hence unlikely to proceed with the payment.

5.3 Experimental Results

Figures 1 and 2 show the performance of our greedy algorithm in comparison to the other approaches in terms of the total reward improvement per new channel connections. The x-axis shows the number of connections added and the y-axis represents the total reward of node **A**. Since, for each case, we started with the same two connections, the total reward values have the same offset.

Figure 1 displays the result for the medium graph. When using default values, the reward was consistently lower than for our fee selection algorithm. More precisely, for centrality-based selection of channels, fee optimization increased the reward by a factor of roughly 2. Selecting channels strategically doubled the gain further in comparison to using Pagerank centrality, which was the most beneficial one of the centrality-based selection methods. Figure 2 shows the results of macro graph. The results were similar to the case of medium payments, though the overall gain was slightly higher.

In terms of fee computation efficiency, our experimental results show that the recursive algorithm described in Section 4.3 reduced the search space of fees in the magnitude of 10–100.

5.4 Discussion

From the experimental results, it can be seen that our greedy algorithm outperformed other centrality metrics. Furthermore, the beneficial effect of the fee selection function was evident when comparing the results with and without it.

Note that adding new connections to the nodes with the highest centrality metrics did not increase the total reward in comparison to random selection much, in particular for betweenness centrality. The reason here is that connecting to nodes with many shortest path passing them does not imply that the newly added channels offer shorter paths. Instead, directly focusing on the betweenness centrality of **A** results in larger improvements.

Figure 1 and 2 furthermore show few but notable differences between medium and macro payments. First, the overall gain was higher for macro payments as expected due to the higher transaction value and hence increased revenue for a similar fee rate. However, the base rate, which is 1000 Satoshi by default[6] in comparison to a default rate of 0.001, dominates the fee value, so that the 100-fold increase in the transaction value does not translate to a similar increase in gain. Secondly, the differences between various centrality measures are more distinct for macro payments, see Figure 2.

[6]The default fee values may change regarding the imported implementation. Our analysis on dataset shows that 33177 out of 68733 edges use the defaults we adopted.

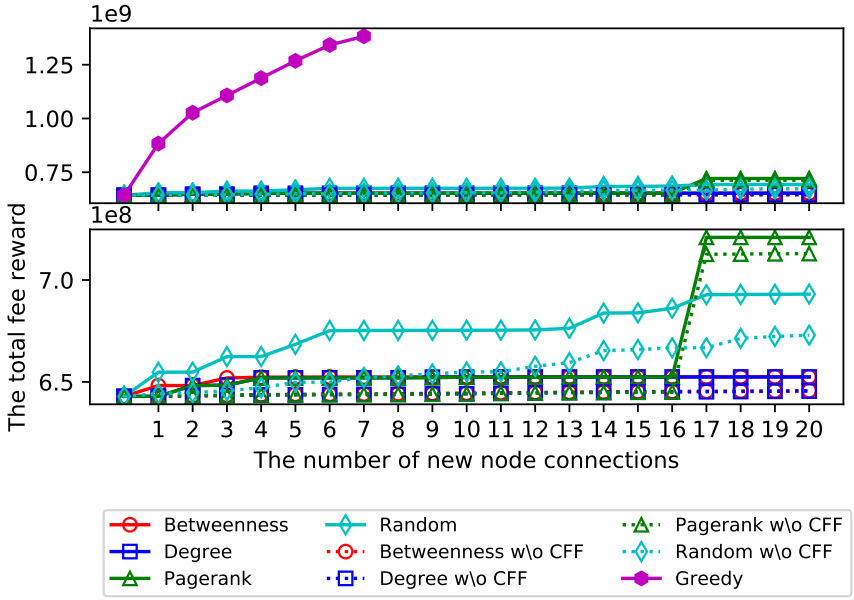

Fig. 1. Total fee reward of our node in medium graph. The bottom figure excludes the greedy results to present a clear comparison of the rest.

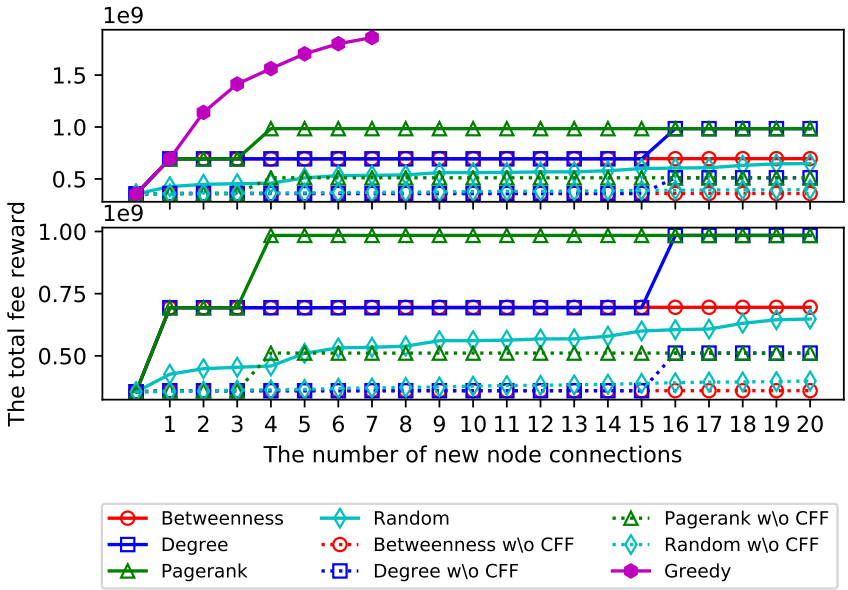

Fig. 2. Total fee reward of our node in macro graph. The bottom figure excludes the greedy results to present a clear comparison of the rest.

Overall, our greedy algorithm promises higher fees for individual nodes. Even if nodes cannot or do not desire to select their channels, they can still gain an advantage by using our more sophisticated fee selection algorithm for already established channels.

One key limitation of our design is that it does not consider channel capacities as such. When all transactions have the same known value, **A** will only establish channels with sufficient collateral. However, in practice, **A** does not have such information and routing may fail due to a lack of capacity. Thus, integrating capacity information into both our model and our evaluation is clearly necessary in the future.

6 Conclusion

In this paper, we formalized an optimization problem for maximizing fees in payment channel networks, presented a heuristic algorithm for solving the problem, and evaluated our algorithm on real-world data sets. Our work demonstrates that routing fees can be a strong incentive for locking coins in payment channels. Fees as incentive hence have the potential to motivate rational users to fund payment channel and hence increase the ability of these networks to route payments.

In this work, we focused on one individual node aiming to optimize its profit. Future work should design a game-theoretical framework for networks containing only rational nodes aiming to maximize their profit. For the continued usage of payment channel networks, incentives should ensure that strategies for optimizing profit locally also optimize the overall network health in terms of the availability of cost-effective paths. It remains an open question if the current fee model is a suitable incentive to further collaboration and network health.

Acknowledgments

This work was partially supported by Ripple's University Blockchain Research Initiative.

References

1. Basis of lightning technology, available at: https://github.com/lightningnetwork/lightning-rfc/blob/master/00-introduction.md
2. AG, B.T.: Raiden network (2019), available at: https://raiden.network/
3. Avarikioti, G., Janssen, G., Wang, Y., Wattenhofer, R.: Payment network design with fees. In: García-Alfaro, J., Herrera-Joancomartí, J., Livraga, G., Rios, R. (eds.) ESORICS 2018 International Workshops, DPM 2018 and CBT 2018. Lecture Notes in Computer Science, vol. 11025. Springer (2018)
4. Bergamini, E., Crescenzi, P., D'Angelo, G., Meyerhenke, H., Severini, L., Velaj, Y.: Improving the betweenness centrality of a node by adding links. ACM Journal of Experimental Algorithmics 23 (2018)

5. Brânzei, S., Segal-Halevi, E., Zohar, A.: How to charge lightning. CoRR abs/1712.10222 (2017), http://arxiv.org/abs/1712.10222
6. Decker, C., Russell, R., Osuntokun, O.: Eltoo: A simple layer2 protocol for bitcoin (2018), available at: https://blockstream.com/eltoo.pdf
7. Decker, C., Wattenhofer, R.: A fast and scalable payment network with bitcoin duplex micropayment channels. In: Symposium on Self-Stabilizing Systems. Springer (2015)
8. Dong, M., Liang, Q., Li, X., Liu, J.: Celer network: Bring internet scale to every blockchain. arXiv preprint arXiv:1810.00037 (2018)
9. Dziembowski, S., Eckey, L., Faust, S., Malinowski, D.: Perun: Virtual payment channels over cryptographic currencies. In: Symposium on Security and Privacy (2019)
10. Egger, C., Moreno-Sanchez, P., Maffei, M.: Atomic multi-channel updates with constant collateral in bitcoin-compatible payment-channel networks. In: CCS (2019)
11. Engelmann, F., Kopp, H., Kargl, F., Glaser, F., Weinhardt, C.: Towards an economic analysis of routing in payment channel networks. In: Proceedings of the 1st Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers, SERIAL@Middleware 2017. ACM (2017)
12. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry 40(1), 35–41 (1977), http://www.jstor.org/stable/3033543
13. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99(12), 7821–7826 (2002)
14. Gudgeon, L., Moreno-Sanchez, P., Roos, S., McCorry, P., Gervais, A.: Sok: Off the chain transactions. IACR Cryptology ePrint Archive 2019, 360 (2019)
15. Hearn, M.: Micro-payment channels implementation now in bitcoinj (2013), available at: https://bitcointalk.org/index.php?topic=244656.0
16. Hoenisch, P., Weber, I.: Aodv–based routing for payment channel networks. In: International Conference on Blockchain. Springer (2018)
17. Miller, A., Bentov, I., Bakshi, S., Kumaresan, R., McCorry, P.: Sprites and state channels: Payment networks that go faster than lightning. In: International Conference on Financial Cryptography and Data Security. Springer (2019)
18. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Tech. rep., Stanford InfoLab (1999)
19. Poon, J., Dryja, T.: The bitcoin lightning network: scalable off-chain instant payments (2016), available at: https://lightning.network/lightning-network-paper.pdf
20. Prihodko, P., Zhigulin, S., Sahno, M., Ostrovskiy, A., Osuntokun, O.: Flare: An approach to routing in lightning network (2016), available at: https://bitfury.com/content/downloads/whitepaper_flare_an_approach_to_routing_in_lightning_network_7_7_2016.pdf
21. Project, E.: Lightning-getroute (2019), available at: https://github.com/ElementsProject/lightning/blob/master/doc/lightning-getroute.7
22. Rohrer, E., Malliaris, J., Tschorsch, F.: Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks. In: Security & Privacy on the Blockchain (2019)
23. Roos, S., Moreno-Sanchez, P., Kate, A., Goldberg, I.: Settling payments fast and private: Efficient decentralized routing for path-based transactions. In: Network and Distributed Systems Security (2018)
24. Stasi, G.D., Avallone, S., Canonico, R., Ventre, G.: Routing payments on the lightning network. In: iThings/GreenCom/CPSCom/SmartData 2018 (2018)