A temporal network analysis reveals the unprofitability of arbitrage in The Prosper Marketplace

U. Redmond *, P. Cunningham

UCD CASL, Belfield Office Park, Clonskeagh, Dublin 14, Ireland

ABSTRACT

There is an increasing focus on methods for network data analysis that consider temporal aspects of the data. We propose a method of network analysis based on the idea of a time-respecting subgraph composed of paths of consecutive edge activations. We present an algorithm to identify these structures and apply the algorithm to a network comprising data from The Prosper Marketplace, an online peer-to-peer lending system. To examine the flow of funds in the network, we extract time-respecting subgraphs. In the larger time-respecting structures, some members act as both borrowers and lenders, possibly attempting to profit from the difference between interest rates of incoming and outgoing loans. We present an analysis of the distribution of time-respecting structures over the lifetime of The Prosper Marketplace and we examine some structures in detail to show that they do represent arbitrage.

1. Introduction

While much network data analysis has focused on static networks it is clear that, in many circumstances, consideration of temporal aspects of the data will improve insight (Holme and Saramäki, 2011). This is particularly true when the underlying data has an event structure. In a static analysis these events are aggregated into edges between nodes to form a static network.

A basic strategy for capturing some temporal information is to partition the network into time slices, which are then analyzed as if they were static (Greene et al., 2010). This method has the drawback of removing much potentially relevant topological information from each static snapshot. One cause of this problem is that, a priori, it is difficult to chose how to slice the network while maintaining the coherence of its function.

To avoid these issues, we propose an approach to temporal network analysis that remains faithful to both the temporal information provided and the entire topological structure. To this end, we introduce the notion of a time-respecting subgraph and present an algorithm for its identification. Such a subgraph is composed of paths of consecutive edge activations, which represent flow in the network – this flow may carry different types of data, depending on the application domain.

To illustrate the effectiveness of our methodology, we apply our algorithm to network data from The Prosper Marketplace (2012), an online peer-to-peer lending system. The presence of time-respecting subgraphs in this context is due to the existence of members (approximately 5% of the total user base), who act as both borrowers and lenders. In principle members should be borrowers or lenders but not both. Two possible explanations for the existence of large time-respecting subgraphs are money laundering and arbitrage. Our analysis suggests that arbitrage is the explanation as members with reasonable credit ratings borrow for the purpose of lending on at higher interest rates. It transpires that the arbitrage is rarely successful in the structures we uncover. We present the distribution of time-respecting subgraphs both before and after the temporary closure of the system, and analyze the behaviour of the intermediate members – there is a drop-off in activity as members realize that the arbitrage is rarely successful.

This paper is laid out as follows. Section 2 presents related work. Section 3 introduces our theoretical framework, defining a time-respecting subgraph and providing an algorithm to identify it. Section 4 discusses the network data to which our algorithm is applied. Section 5 presents the results of our analysis. Section 6 concludes the paper.

2. Related work

This paper draws on previous work on temporal network analysis, and is motivated by the potential for development in this area. Current approaches to the problem are discussed, with a view to how methods could be improved. A brief overview of network analysis applied to peer-to-peer lending is presented, to motivate our choice of application domain.
2.1. Temporal network analysis

Although temporal information may be available, much analysis is performed as if the network were static. This yields problems even for basic connectivity. In a static network, given directed edges \((u,v)\) and \((v,w)\), then \(u\) is connected indirectly to \(w\) via a path through \(v\). The same is not necessarily true in a temporal network context. In this case, if the edge \((v,w)\) activates before \((u,v)\), then \(u\) and \(w\) are disconnected, since nothing can propagate between them via \(v\).

In much of the literature, network data is segmented into time windows, within which the network is regarded as static. This approach suffers from the connectivity issues mentioned, as well as the difficulty in choosing the time window size correctly. Certain aggregations may fail to capture network properties of interest. For example, in a network of mobile telephone calls, networks aggregated over different time intervals yielded different insights (Krings et al., 2012). To counteract this, the use of sets of window lengths has been proposed, to provide an accurate multi-level view of temporal structure (Lijffijt et al., 2012).

Another problem is that the broader context outside of each time window, which may be extremely relevant, is lost. Also, independent processes may be grouped together in the same time-slice, which may not be useful. Our approach counters this effect by, instead of grouping interactions within a time window, grouping interactions that are part of the same process. This means that the broader context of the interactions is captured.

A review of network concepts that include temporal information is introduced by Holme and Saramäki (2011). Some of the key definitions are presented here. In a temporal graph, Kempe et al. (2002) define a time-respecting path as a sequence of contacts with non-decreasing times that connect sets of vertices. According to Nicosia et al. (2012), two vertices \(i\) and \(j\) are strongly connected if there is a directed, time-respecting path connecting \(i\) to \(j\) and vice versa. Vertices \(i\) and \(j\) are weakly connected if there are undirected time-respecting paths from \(i\) to \(j\) and vice versa. In this case, the directions of the contacts are not observed (Nicosia et al., 2012).

In a reachability graph, a directed edge exists between vertices \(i\) and \(j\) if there is a time-respecting path between them. The algorithm supplied by Moody (2002) to identify these structures reveals the vertices which are reachable from each other. Bearman et al. (2004) analyze a reachability graph constructed from a dating network of high-school students. Another interpretation of this type of graph is the associated influence digraph of a time-stamped graph, which encodes the ability of vertices to influence, or reach, each other (Cheng et al., 2003). Dynamic reachability sets have also been introduced (Macropol and Singh, 2012). In this formulation, the reachability set of a node is found by traversing edges outward from the node, with each step incrementally later in time. The edges in the set must occur within a given time interval.

In the formulation of Zhao et al. (2010), the lifespan of a piece of information is defined as the time between the end of one communication and the beginning of another. In another work, the relay time of an edge is defined as the time taken for a newly infected node to further spread the infection via the next interaction that the link participates in (Kivelä et al., 2012). Thus, limiting the time allowed between interactions is an important concept in a variety of network types.

Correspondingly in this work, we require the time delay between contacts on a time-respecting path not to exceed a threshold \(d\). To see why, consider a shortest path between vertices \(i\) and \(j\) via a vertex \(k\). The centrality of the vertex \(k\) is vulnerable, in that it depends on the interval between the communication from \(i\) to \(k\) and from \(k\) to \(j\). The longer this interval, the higher the chance that the information intended for \(j\) will be disrupted (Tang et al., 2010). Also, although it is possible that there is no causality relationship between any two adjacent communications, it appears that the closer in time they take place, the more likely they are to be about the same topic (Zhao et al., 2010).

2.2. Peer-to-peer lending

Peer-to-peer lending provides an online platform for the lending of money directly to borrowers, without the intermediation of traditional banks. There is not necessarily a prior relationship between lenders and borrowers, although members may join groups with similar affiliations or interests. Lenders choose loans in which to invest, based on the credit profile of borrowers and the potential return on investment.

Social lending among such peers is also influenced by network effects. Since there is very little contact between lenders and borrowers, the problem of information asymmetry is a factor. A borrower may offset this by being as descriptive as possible in their loan listing and joining a group (Lerner et al., 2011). Lenders also choose borrowers based on their friendship network and their endorsements from other members (Lin et al., 2009). This selection criterion sometimes leads to irrational behaviour, with lenders funding the loans of borrowers who have a higher risk of default. It is interesting to see from Lin et al. (2009) that members acting as both borrowers and lenders tend to outperform pure borrowers on loan repayment.

3. Theory

There have been varied approaches to the representation of temporal networks in studies thus far, as noted in Section 2. We choose the following definition, since it maintains the original topology of the network, while incorporating the available temporal information.

**Definition 1.** A directed temporal graph \(G\) consists of a set \(V\) of vertices and a set \(E\) of ordered pairs of vertices representing interactions. An interaction \(e_i\) in \(E\) is represented by a four-tuple \(e_i = (u_i, v_i, t_i, \delta_i)\), in which \(u_i\) is the source vertex, \(v_i\) is the target vertex, \(t_i\) is the initiation time of the interaction and \(\delta_i\) is the duration of the interaction.

Thus, time is encoded as an explicit part of the representation. In this setting, the terms “edge”, “interaction” and “event” may be understood to mean the same thing. Each interaction begins at a given time, and lasts for a given duration. This allows us to generalize to many applications, such as in telephone contact networks (in which the durations of calls differ), or in networks of epidemics (in which a disease may render an individual infectious for variable amounts of time).

A path in this context is only meaningful if composed of edges whose activations follow each other in time. Thus, we introduce the notion of time-respecting edge pairs, which are the building blocks of larger time-respecting graph structures, such as paths and subgraphs. Fig. 1 illustrates the definition.

**Fig. 1.** A pair of edges \((e_i, e_j)\) which are time-respecting. Vertex \(u_i\) initiates an interaction at time \(t_i\). The interaction concludes after an amount of time \(\delta_i\) at time \(t_i + \delta_i\). After some time delay \(d\), interaction \(e_j\) begins at time \(t_j\). It can be seen that \(0 < t_i - (t_i + \delta_i) < d\). Thus, \(0 < t_i - t_j - \delta_i < d\).
Definition 2. Let $e_i$ and $e_j$ be edges in a temporal graph. The edges are time-respecting if $v_i = u_j$ and $0 \leq t_j - t_i - d_i \leq d$, for some threshold $d$.

A time-delay threshold between interactions, $d$, is incorporated to model a variety of real-world scenarios. For example, in an epidemic network, an individual is infectious for some time after the contact which resulted in their infection, during which time another contact may spread the infection further. In a communication network, a piece of information may be propagated further some time after it has been received. Various models for waiting times between two consecutive interactions have been proposed, including exponential and power law (Barabasi, 2005). For the purposes of this work, we specify a constant value.

To simplify our theoretical development, we set to zero the value of $d$ for each edge. This implies that the duration of each interaction is instantaneous. This is the correct interpretation for scenarios such as financial transactions between individuals, which we will see more of later.

Definition 3. Let $v$ and $w$ be two vertices in a temporal graph. A directed time-respecting path between $v$ and $w$ in the graph is a finite alternating sequence $v = v_0, e_1, v_1, e_2, \ldots, e_n, v_0 = w$ of non-repeating vertices and edges of the graph such that each pair of adjacent edges is time-respecting.

A path composed of time-respecting edge pairs is illustrated in Fig. 2. Each edge is labeled with the day on which the interaction took place. This path describes a non-decreasing sequence of edge activations, as introduced in Pan and Saramäki (2011).

A motivating example of a time-respecting subgraph in a temporal graph is illustrated in Fig. 3. Each directed path in the subgraph is time-respecting. Where the paths intersect at a vertex, any incoming edges occur before any outgoing edges.

Definition 4. A time-respecting subgraph $S = (V', E')$ of a temporal graph $G = (V, E)$ is composed of a vertex set $V' \subseteq V$, and a set of interactions $E' \subseteq E$ such that every edge pair $(e, e')$ in which $v_i = u_j$ is time-respecting.

From a reachability point of view, it can be seen in Fig. 3 that from any vertex in the time-respecting subgraph, at least one of the vertices on the rightmost frontier can be reached via a time-respecting path. The lengths of the paths that comprise a time-respecting subgraph are not constrained by their overall duration (Pan and Saramäki, 2011), but rather by the duration of edge pairs along each path, so that in theory a path may originate at the graph’s inception, and terminate at the last interaction. This avoids effectively time-slicing the graph, as in the case of the Dynamic Reachability Set (DRS) (Macropol and Singh, 2012). In the DRS setting, a threshold $\Delta$ specifies the longest permitted duration of a path in a set of vertices reachable from a given source vertex. In a DRS, consecutive events must occur in integer increments, with no further delay time allowed between the interactions. In our model, the encoding of the duration of an interaction and an inter-interaction waiting time allows for more general application. Henceforth, for ease of exposition, we abbreviate “time-respecting path” and “time-respecting subgraph” to “path” and “subgraph”, respectively.

3.1. Algorithm

The search for subgraphs in a temporal network is composed of repeated applications of a breadth-first-search (BFS) on the edges, as outlined in Algorithm 1. The search starts with an edge not part of a previously found subgraph, and expands from each edge $e$ via out-edges from the target vertex of $e$ which obey the time-respecting property, and via in-edges to the same target which happened at the same time. We allow the expansion via in-edges since these will also be time-respecting when compared with out-edges found in that step. These in-edges also allow for correct merging within the BFS, as illustrated in Fig. 4.

After all subgraphs have been found from repeated applications of BFS, a final merge step is enacted. This is required since each BFS begins with a single edge, while a subgraph may be initiated by an individual interacting with many other individuals within a short space of time. Hence, any nodes on the node-frontier of the subgraphs that match, and whose out-edges occur within time $d$ of each other, will lead to the merging of the subgraphs they inhabit. The remaining subgraphs are maximal (see Fig. 5).

A BFS may incorporate edges which have already been included in another subgraph, with which a merge is not permitted. Allowing an interaction to reside in multiple internally consistent subgraphs is important, since the individual initiating such an interaction may have been influenced by an interaction in any one of those subgraphs. Fig. 6 illustrates this situation. These two subgraphs should not be merged, since the in-edge at time 3 happens after the out-edge at time 2. The correct behaviour emerges since a BFS from the edge at time 3 will not merge with the first subgraph found, since the edge encountered (at time 4) is not on the edge-frontier of that subgraph.
Algorithm 1. Find maximal time-respecting subgraph \((G,e,d)\)

```plaintext
function bfs_augmented \((G,e,d)\)
    \(q \leftarrow e, s \leftarrow e\)
    while \(q\) is non-empty do
        \(t \leftarrow q\).dequeue()
        \(adj\_edges \leftarrow get\_out\_edges \((G,t,d)\) + get\_in\_edges \((G,t)\)\)
        for all \(o \in adj\_edges, o \neq s\) then
            if \(o\) is edge-frontier of some subgraph \(r\) then
                \(s \leftarrow merge(s,r)\)
            else
                \(q \leftarrow o, s \leftarrow o\)
            end if
        end for
    end while
    return \(s\)
end function

function get\_out\_edges \((G,e,d)\)
    out\_edges \leftarrow \emptyset, \(v_i \leftarrow e\).target()
    for all \(e_j \in G\).out\_edges\((v_i)\) do
        if \(0 \leq t_j - t_i - \delta_i \leq d\) then
            out\_edges \leftarrow \(e_j\)
        end if
    end for
    return out\_edges
end function

function get\_in\_edges \((G,e)\)
    in\_edges \leftarrow \emptyset, \(v_i \leftarrow e\).source()
    for all \(e_j \in G\).in\_edges\((v_i)\) do
        if \(0 \leq t_j - t_i - \delta_i \leq 0\) then
            in\_edges \leftarrow \(e_j\)
        end if
    end for
    return in\_edges
end function
```

4. Methods

This section introduces the data set which we examined using the methods described in Section 3. We also outline the implementation framework under which the results were gathered.

![Diagram](image)

**Fig. 5.** The final merging step, with \(d = 2\). Given that \(u_a\) and \(u_b\) are the same node, and that their out-edges occur within time \(d\) of each other, their subgraphs are merged.

![Diagram](image)

**Fig. 6.** A temporal graph, in which two subgraphs are found at \(d = 3\). The edge at time 4 is incorporated into both, since it is time-respecting when compared with both the edge at time 1 and that at time 3.

### 4.1. The Prosper Marketplace

The Prosper Marketplace (henceforth Prosper) opened to the public in February 1996. It closed for regulatory reasons in November 2008 but relaunched in July 2009. The Prosper website provides a nightly snapshot of all data pertaining to listings, bids, users, groups and loans, in order to facilitate the statistical analysis of the system. As of September 2011, there were 8916105 bids on 401180 listings between 1207418 members. Of those listings, 43576 were accepted as loans.

A member of the Prosper system is a registered user, who may have roles including that of borrower, lender, group leader or trader. A borrower creates a listing in order to solicit bids. If enough bids are received to reach the amount requested, the listing becomes a loan after the listing period ends. A lender creates a bid, specifying an amount and a minimum rate required, should the bid win the auction and the listing become a loan. The possible statuses of a loan include current, late, paid, defaulted upon and cancelled. For a further discussion on the institutional background of social lending on Prosper, the interested reader is referred to Lin et al. (2009).

### 4.2. Network data

A network composed of Prosper members and their loans may be constructed. The vertex set consists of members who received or contributed to loans. The edge set captures the flow of funds from members who acted as lenders to those who acted as borrowers for the purpose of the transaction. The edge is time-stamped with the origination date of the loan, which marks the time when the borrower received funds and amortization began. For further analysis, also included in the edge data is the status, amount and grade or rating of the loan, along with the lender rate and borrower rate of interest. The network takes the form of a directed graph with parallel edges, but no self-loops.

The data we analyze comes from transactions occurring from November 2005 to September 2011. Since the Prosper system was temporarily closed for regulatory reasons, we have split the data set into a pre- and post-closure network. Before the closure there were 1995399 edges and 72334 vertices. Afterwards, there were 1399580 edges and 32191 vertices.

In order to present a meaningful comparison, without seasonal variation, between behaviour before and after the temporary closure, we selected a calendar year of activity from each time frame from which to begin the algorithm. The algorithm explored the network via BFS until termination, which may have occurred at any date within the overall time frame. This allows each time-respecting subgraph to represent an entire process, without being artificially restricted to a given time window. We compare 2007 (169162 edges) with 2010 (91435 edges), so as to avoid the initial periods in which the user bases were not yet firmly established.

### 4.3. Implementation

To facilitate our later analysis of arbitrage strategies in social lending, we select a threshold for the time-delay between interactions which is meaningful in this context. Since the first repayment on a loan is due one month after the origination date, we require that reinvestment occurs before this month has elapsed. This makes it more likely that the member will cover the cost of borrowing with the money earned on investment. Since transactions occur instantaneously, our value for \(\delta\) is set to zero.

The Python programming language (Python Software Foundation, 2012), which provides the NetworkX Developers (2010) and matplotlib (Hunter et al., 2011) libraries, was used to generate
5. Results

The application of our algorithm to the Prosper data set yielded interesting results. The number and size of subgraphs found before the closure are for the most part greater than those found afterwards. This result corresponds to our observation in previous work that there are significantly more structures composed of directed paths before than after the closure (Redmond et al., 2012). To illustrate this we consider two statistics, the subgraph size (number of edges) and the subgraph diameter (length of the longest shortest path).

After identifying the time-respecting subgraphs we analyze the behaviour they represent. As mentioned in the introduction, two possible explanations are money laundering and arbitrage. In Section 5.2 we show that the explanation is arbitrage, although the attempts are not very successful.

5.1. Distribution of time-respecting paths and subgraphs

The distribution of subgraph diameters before and after the closure is illustrated in Fig. 7. Subgraphs from 2007 have longer paths, probably since they have larger subgraphs.

Our initial suspicion was that directed paths in the network were the result of money laundering operations. If this were the case, it should be that the intermediate members in these loan chains would default on their loans. In fact, while the incidence of defaulting in these chains is high, many loans are repaid, suggesting that money laundering is not the objective.

Subgraphs are composed of intersecting time-respecting paths. The distribution of subgraphs of different sizes is illustrated in Figs. 8 and 9. The two size ranges are compared separately since there is a clean and expansive division between subgraphs with less than 1600 edges and those with more than 200000 edges. It is clear that the very largest subgraphs occur more frequently in the 2007 data than in the 2010 data.

5.2. Detecting arbitrage

The practice of arbitrage aims to capitalize on the price imbalance between matching assets, from which a profit is made on the price difference. In our case, the assets are loans, which do not trade at the same price for all borrowers. A high-risk borrower will ordinarily pay a higher rate of interest, while a low-risk borrower will have lower borrowing costs. The practice is permissible, with some members joining groups that declare arbitrage as their aim.

Strictly speaking, arbitrage is the possibility of making a profit at zero cost, with no risk. Transactions must occur simultaneously, in order to avoid the risk of prices changing for one before the other is complete. Within Prosper, borrowers may default on loans, thereby introducing risk. Also, the incoming and outgoing transactions do not occur simultaneously, since a borrower who wishes to subsequently lend must spend time selecting a set of borrowers.

There are certain network-based features which correspond to the definition of an arbitrageur. These include incoming and outgoing loans associated with an individual, which are manifested as a positive in- and out-degree for a vertex. In terms of timing, the member will first receive a loan, and then use this amount to fund investment, as shown in Fig. 12. The simplest case occurs when a member gets a single loan and reinvests the same amount within our prescribed waiting time. The intention of the member is more difficult to discern in the case where multiple loans are received...
and an amount in reinvested, all within the period covered by our chosen time delay. In this case, we identify a pool of money available to the individual from the in-loans, and analyze them with respect to the out-loans that occurred within the same time period.

Many members who experiment with arbitrage on the Prosper system do not reinvest exactly the amount that they borrowed. For the purposes of this study, we require that a member should reinvest at least 80% of the amount borrowed, and borrow at least 80% of what is invested. Otherwise, we assume the member is experimenting with different roles on the system for other reasons.

Positive or negative quantities earned from an arbitrage attempt correspond to successful or unsuccessful strategies, respectively. The amount earned is the return on investment from the outgoing loans less the amount and interest owed for the initial loan. Each outgoing loan is evaluated such that if it defaulted, that amount is lost, otherwise that amount plus its interest rate is gained.

Fig. 10 shows that there is a strong correlation between subgraph size and the number of arbitrage attempts. In fact, with only one exception, no arbitrage attempts were found in the smaller subgraphs. Thus, the large subgraphs that exist before the closure are the parts of the network in which arbitrageurs operate. It is interesting to note that the decline in arbitrage attempts mirrors the decline in the number and size of subgraphs found. An analysis of this outcome follows.

5.3. Distribution of gains and losses

Since most of the loans issued in 2010 are still current at the time of this writing, it is not reasonable to project successful or unsuccessful outcomes for arbitrageurs operating in the post-closure network. Thus, the results here are for the pre-closure network. Our analysis of the time-respecting subgraphs we have identified showed that there were 357 attempts at arbitrage before the closure, but only 38 attempts afterwards. This shows how little of the practice was continued when Prosper reopened. Fig. 11 is a clear illustration of why this might be the case, with negative outcomes dominating.

5.4. Winning and losing strategies

Given that many loans which originated in 2010 are still current, the strategies analyzed here are from before the closure. In

Fig. 10. Comparing the number of arbitrage attempts with the size of the subgraph within which the attempt was identified.

Fig. 11. Returns from attempted arbitrage before the closure. Only four of the attempts show a positive return as defaults push the majority of attempts into negative territory.

Tables 1 and 2 represent successful and unsuccessful arbitragees are shown along with the sum of all amounts bid which are associated with the loans.

Table 1 describes a winning strategy from 2007. The arbitrageur has an excellent credit rating, and can thus get a loan at a very low interest rate. Over the course of the next month, the arbitrageur reinvests the borrowed money in the loans of members with excellent credit ratings but higher interest rates. For each of these loans, a relatively small amount is invested, thus spreading the risk. All of the members repay their loans, so the arbitrageur makes a profit.

Table 2 shows one of many losing strategies from 2007. The arbitrageur has lent large sums of money to a small number of borrowers. Thus, the risk is not well spread, making the possibility that any borrower defaults a real concern. Most of the borrowers funded by the arbitrageur have indeed defaulted on their loans. (Charge-off is a term used by Prosper for a loan which is in default
caused by their very low credit grade. Fig. 12 illustrates the part of the arbitrageur has chosen borrowers with very high interest rates, or for which payment is so late that any future repayment is as-
in Table 2. The day on which each loan originated is displayed on each edge. Fig. 12. The time-respecting ego-centric subgraph of the losing strategy presented

Table 2
This table shows a losing strategy from before the closure, in which the arbitrageur lost $1657.34, given a return on investment of $13.85 and a borrowing cost of $1671.19.

<table>
<thead>
<tr>
<th>Loan</th>
<th>Amount</th>
<th>Rate</th>
<th>Status</th>
<th>Grade</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>In</td>
<td>1505</td>
<td>0.1100</td>
<td>Paid</td>
<td>A</td>
<td>2007-10-03</td>
</tr>
<tr>
<td>Out</td>
<td>100.00</td>
<td>0.2900</td>
<td>Charge-off</td>
<td>E</td>
<td>2007-10-09</td>
</tr>
<tr>
<td></td>
<td>400.00</td>
<td>0.2400</td>
<td>Paid</td>
<td>C</td>
<td>2007-10-12</td>
</tr>
<tr>
<td></td>
<td>100.00</td>
<td>0.2000</td>
<td>Charge-off</td>
<td>C</td>
<td>2007-10-15</td>
</tr>
<tr>
<td></td>
<td>325.00</td>
<td>0.2315</td>
<td>Charge-off</td>
<td>D</td>
<td>2007-10-16</td>
</tr>
<tr>
<td></td>
<td>215.81</td>
<td>0.2300</td>
<td>Paid</td>
<td>D</td>
<td>2007-10-17</td>
</tr>
<tr>
<td></td>
<td>100.00</td>
<td>0.2400</td>
<td>Charge-off</td>
<td>HR</td>
<td>2007-10-22</td>
</tr>
<tr>
<td></td>
<td>122.59</td>
<td>0.2000</td>
<td>Charge-off</td>
<td>D</td>
<td>2007-10-22</td>
</tr>
</tbody>
</table>

In the early stages of peer-to-peer lending, people were excited about the potential for earning big returns through arbitrage. As it turned out, a significant amount of effort was involved for a relatively small return. Success was not likely, since borrowers could default at will. This fact is illustrated in Fig. 11. By the time Prosper was relaunched, it had already been established that social lending arbitrage was not lucrative, so members partook less. This is also reflected in the decline in the sizes of subgraphs, whose backbone is composed of members who may be attempting arbitrage.

6. Conclusions

This paper has developed the idea of examining a network with temporal information as an explicit property of the edges. This approach maintains the original network topology, unlike other methodologies which slice the network in various manners.

In order to explore sections of the network which facilitate flow, we defined a time-respecting subgraph, composed of sequences of consecutive edge activations. We presented an algorithm to iden-

tify these subgraphs, which is based on a breadth-first-search of the network edges.

The setting chosen for the application of our algorithm is a network derived from data provided by The Prosper Marketplace. In this context, the existence of large time-respecting subgraphs relied on the presence of members who partook in both borrowing and lending. Their motivation for doing this seems to have been the prospect of capitalizing from arbitrage. However, predominantly due to the tendency of borrowers to default, this turned out to be an unprofitable enterprise.

Since the introduction of time-respecting subgraphs reveals those parts of the network that are temporally connected, the scope for further network analysis within these structures is broad. The study of centrality and role analysis, the possibility of identifying temporal communities, and the opportunity to examine temporal motifs appear as worthy candidates of further exploration.

References


NetworkX Developers. (2010). NetworkX.


