Network Science (2019), 1–16
doi:10.1017/nws.2019.5

CAMBRIDGE UNIVERSITY PRESS

ORIGINAL ARTICLE

The evolution of a mobile payment solution network

Kjersti Aas* and Hanne Rognebakke

Norwegian Computing Center, P. O. Box 114, Blindern, N-0314 Oslo, Norway
*Corresponding author. Emails: Kjersti.Aas@nr.no, Hanne.Rognebakke@nr.no

Action Editor: Dr. Fernando Vega-Redondo

Abstract

Vipps is a peer-to-peer mobile payment solution launched by Norway's largest financial services group DNB. The Vipps transaction data may be viewed as a graph with users corresponding to the nodes, and the financial transactions between the users defining the edges. We have followed the evolution of this graph from May 2015 to September 2016. This is a unique data set, as information about transactions of individuals is usually not available for research. In this paper, we use an advanced statistical model where preferential attachment is combined with fitness. We show that the intrinsic quality of the nodes in the Vipps network plays an important part in the evolution of the network. This insight may, e.g., be used to identify influential nodes for viral marketing.

Keywords: mobile payment solution, network evolution, preferential attachment, fitness

1. Introduction

Vipps is a peer-to-peer mobile payment solution launched by Norway's largest financial services group DNB. It was released on May 30th, 2015 and is now the most downloaded app in Norway. The app is available to everyone with a Norwegian bank card, and by reaching 1 million users in November 2015, Vipps became Norway's largest payment application. Now, in 2018, Vipps has more than 3 million users. The application is designed for smartphones and gives the user the possibility to make payments to a receivers telephone number instead of a bank account number. Among other things, it makes it easier to split a restaurant bill, to do the settlement after a girl trip, or simply transfer money between friends.

The Vipps transaction data may be viewed as a graph with users corresponding to the nodes, and the financial transactions between the users defining the edges. In this paper, we have used the subgraph consisting of all nodes (users), but only having an edge between nodes A and B if user B was recruited by user A. This graph is unique in the sense that we may follow the evolution from the very first user. Hence, we may study how and why its topology changes over time. This is useful in many situations. The design of algorithms on complex networks, such as routing, scheduling, ranking, or recommendation, requires, e.g., a detailed understanding of the growth characteristics of the networks of interest. There is a growing literature analyzing the characteristics and dynamics of large complex networks, such as the web graph (Barabasi et al., 2000), social networks (Kunegis et al., 2013), scientific citation networks (Redner, 1998), and recommendation networks (Leskovec et al., 2006).

Information about financial transactions between individuals is however usually considered confidential. The only related work we are aware of is the empirical analysis of the Bitcoin network (Kondor et al., 2014), where a nonlinear preferential attachment (PA) model (Krapivsky et al., 2000) is used to model the network evolution. Recently, several papers have shown that interactions in real-world networks may be more complex than implied by a PA model. In this

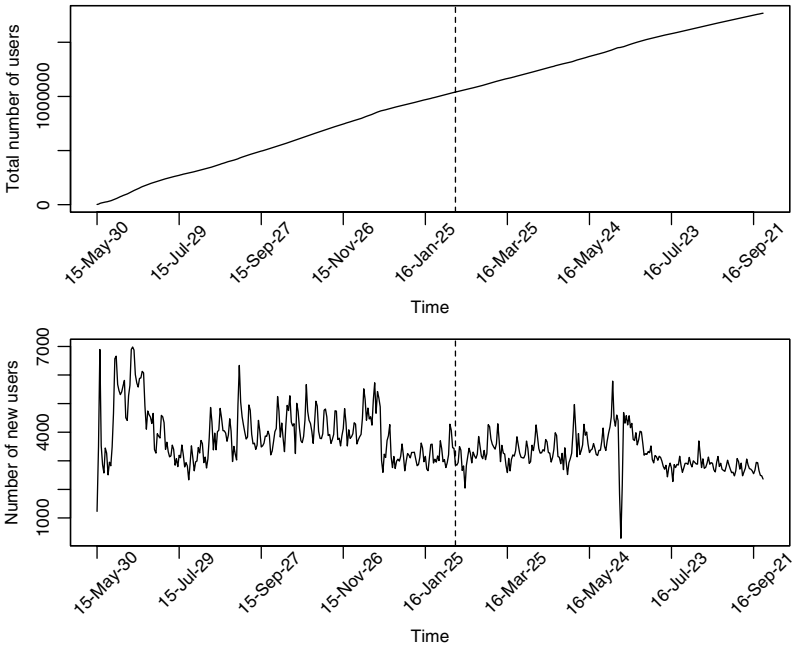

Figure 1. The aggregated total number of Vipps users (upper panel) and the number of new users (lower panel) every day in the observation period.

paper, we use a model proposed by Pham et al. (2016), in which the degree of a node is scaled by its intrinsic quality to determine its attractiveness. We study different aspects connected to the dynamic and static properties of the Vipps graph and indicate how this insight may be used to identify the influential users in the network.

Note that the behavior of people adopting an innovation like Vipps may also be interpreted as a spreading phenomenon throughout an underlying social network like in Iñiguez et al. (2017), where the underlying network is the largest connected component of the free Skype service network, and the product is a "buy credit" paid service. However, in our case, the underlying social network is unknown. It is actually the social network for which the nodes are the 5.3 million inhabitants of Norway. In the future, when Vipps is even more widespread than today, the final stage of the network may be used as a proxy for the underlying network. Currently, however, it is more relevant to view the Vipps adoption as the structural evolution of the network itself.

The rest of this paper is organized as follows. In Section 2, we describe the Vipps data set. Section 3 reviews the most common network evolution models, while the model studied in this paper is described in Section 4. In Section 5, we give the results obtained in our study, and finally, Section 6 contains some concluding remarks.

2. Data set

We have used transaction data from May 30th, 2015, to September 30th, 2016. This data set consists of 28,876,279 transactions (time and amount) for 1,769,142 different users. Figure 1 shows the aggregated and new number of Vipps users every day. During the period June 15th–17th, 2016, Vipps did not work properly due to technical problems, and therefore, we see a dip in the plots for these days. In the analysis described in Section 5, we have divided the data set into two periods: the first period, which is used for estimating the network evolution model, lasts from May 30th, 2015, to February 17th, 2016, and the latter, used for validating the model, lasts from

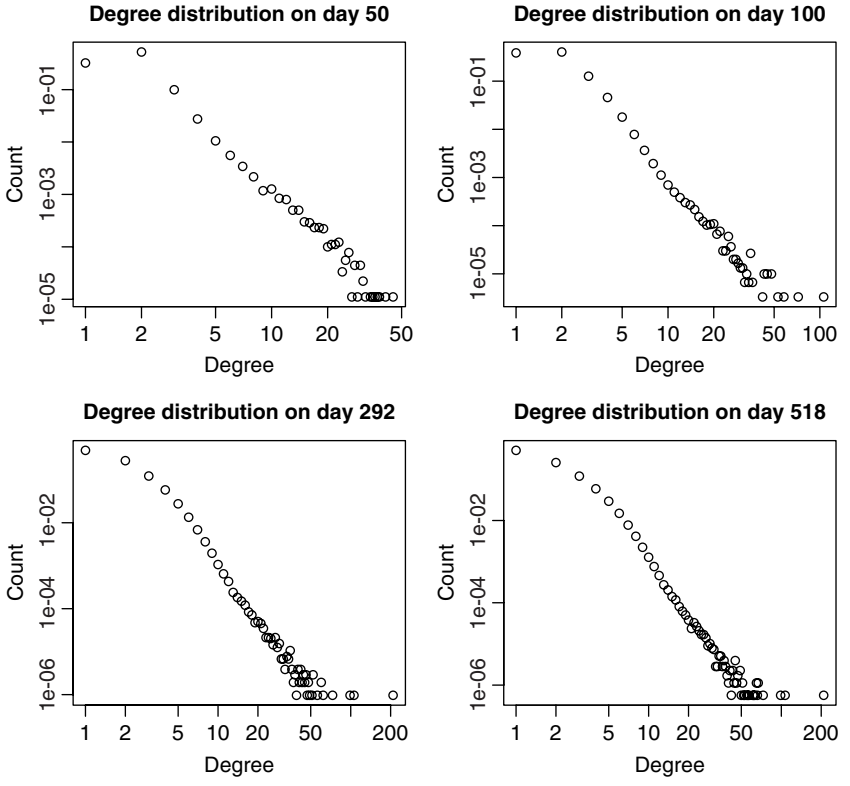

Figure 2. The recruitment graph: Degree distributions over time. All plots are on log–log scale. Days 292 and 518 correspond to February 17th, 2016, and September 30th, 2016, respectively.

February 18th, 2016, to September 30th, 2016. In Figure 1, the two periods are separated by a vertical dotted line. The numbers of users joining Vipps during the two periods were 1,038,997 and 730,145, respectively.

In the original `Vipps` graph, there might be several edges between each pair of nodes corresponding to multiple transactions. In the analysis described in this paper, our main aim is to identify the users who are most efficient in recruiting new users. Hence, we will mainly use a subgraph consisting of all nodes (users), but only having an edge between nodes A and B if user B was recruited by user A. We denote this graph "the recruitment graph." Here, we say that user B has been recruited by user A if (i) A started to use `Vipps` before B and (ii) the first transaction from/to B was to/from A. When presenting the results in Section 5.3 we will also use the term "friend.". By nodes A and C being "friends," we then mean that there has been at least one transaction between A and C during the whole time period from May 30th, 2015, to September 30th, 2016.

Figure 2 shows the development of the degree distribution for the recruitment graph over time, while various summary statistics for the same snapshots are given in Table 1. As can be seen from the figure, the data points form an approximate straight line on log–log scale, suggesting that the degree distribution of the recruitment graph is well approximated with a power-law distribution. The estimated exponent γ is 3.8. The recruitment graph is a connected graph that has no cycles, i.e., a tree. Every time a new user is added to the network it is connected to only one of the existing users.

For the sake of comparison, we have also included the same properties for the subgraph consisting of all nodes (users), but having edges between two nodes A and C if they are friends; see Figure 3 and Table 2.

Table 1. The recruitment graph: Summary statistics for different snapshots

Day	Number of nodes	Number of edges	Max degree	Connected component distribution				
				Min	25%	Mean	75%	Max
50	90,006	90,006	44	1	2	3.5	3	1,722
100	301,234	301,234	105	1	2	4.5	4	3,865
292	1,038,997	1,038,997	208	1	2	8.9	8	10,380
518	1,769,142	1,769,142	208	1	3	12.7	12	16,770

Table 2. The friends graph: Summary statistics for different snapshots

Day	Number of nodes	Number of edges	Max degree	Connected component distribution				
				Min	25%	Mean	75%	Max
50	90,006	90,094	102	1	2	4.1	3	31,320
100	301,234	424,491	241	1	2	8.7	3	204,400
292	1,038,997	4,002,626	711	1	2	102.3	2	1,016,000
518	1,769,142	12,955,189	836	1	2	477.9	2	1,761,000

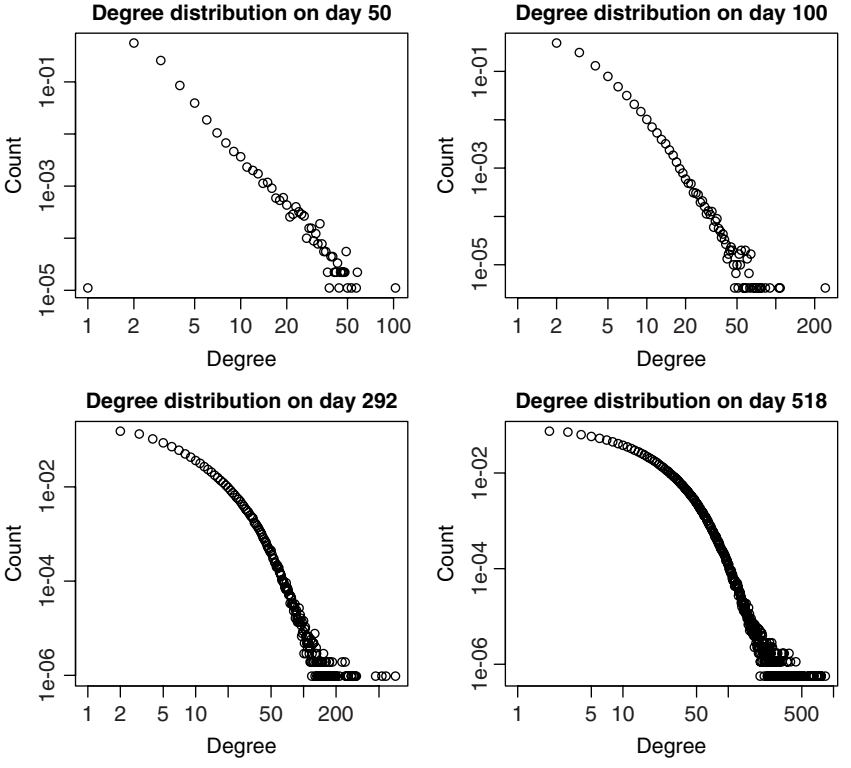

Figure 3. The friends graph: Degree distributions over time. All plots are on log–log scale. Days 292 and 518 correspond to February 17th, 2016, and September 30th, 2016, respectively.

3. Network evolution models

In recent years, there has been a convergence of ideas coming from computer science, social sciences, and economic sciences to model and analyze the characteristics and dynamics of large complex networks, such as the web graph, social networks, and recommendation networks.

Various mechanisms have been suggested, but models for network growth resulting in a scale-free distribution have received special attention. Scale-free networks have power-law degree distributions; i.e., the number of nodes with degree d is proportional to $d^{-\gamma}$, for a particular γ.

The perhaps most well-known scale-free network model is the PA model (Yule, 1925). PA means that the more connected a node is, the more likely it is to receive new links. Nodes with higher degree have stronger ability to grab new links added to the network ("rich-get-richer effect"). Intuitively, the PA can be understood if we think in terms of social networks connecting people, where a link between A and B means that person A "knows" person B. Heavily linked nodes represent well-known people with lots of relations. When a newcomer enters the community, she or he is more likely to become acquainted with one of those more visible people rather than with a relative unknown. Similarly, on the web, new pages link preferentially to hubs, e.g., well-known sites like Google or Wikipedia, rather than to pages that hardly anyone knows.

The classical PA model for networks is the Barabasi–Albert model (Barabasi & Albert, 1999). It assumes a linear relationship between the number of neighbors of a node in the network and the probability of attachment. That is,

$$P(\text{Node } i \text{ receives a new edge}) \propto d_i,$$

where d_i is the degree of node i. This model, which implicitly assumes a network for which the number of edges grows linearly with the number of nodes, has later been generalized to

$$P(\text{Node } i \text{ receives a new edge}) \propto d_i^{\alpha} \tag{1}$$

where we have a sublinear model if $0 < \alpha < 1$ and a superlinear model if $\alpha > 1$. For the sublinear model, the network's degree distribution is stretched exponential (Dereich & Mörters, 2009) and the hubs are much smaller than in a scale-free network. If the model is superlinear, almost all nodes are connected to a few hubs instead (Krapivsky et al., 2000).

Recently, several papers have shown that interactions in real-world networks may be more complex than previously thought; see, e.g., Borgs et al. (2007), Kong et al. (2008), Kunegis et al. (2013), Pham et al. (2015, 2016). The central assumption of the PA model, stating that the popularity of the nodes depends only on their degree, means that the oldest nodes in the network are likely to have most links. In many situations, the growth rate of a node does not depend on its age alone. Instead web pages, companies or persons have intrinsic qualities ("fitness") that influence the rate at which they acquire links. Hence, several papers have proposed to combine PA with fitness models. The first scale-free network model introducing this heterogeneity of the nodes was the Bianconi-Barabási model (Bianconi & Barabási, 2001) that has been used to model the Internet and the World Wide Web. In this model, nodes acquire new links with a generalized PA rule that assigns higher probability of attracting new edges to high degree and high fitness nodes than to those with lower degree or lower fitness. In Bianconi & Barabási (2001), the definition of PA is restricted to that of the original Barabasi–Albert model, while in a recent work, (Pham et al., 2016) a model combining PA and node fitness is estimated without imposing any functional constraints. We have used the latter model, which is called the Generative Temporal (GT) model, to investigate the interplay between PA and node fitness in the `Vipps` network. The GT model includes several existing network models as special cases, e.g., Barabasi & Albert (1999), Callaway et al. (2001), Bianconi & Barabási (2001), Krapivsky et al. (2001) and Caldarelli et al. (2002) and hence allows for very flexible modeling of the network evolution. In Section 4, we provide a more thorough description of this model.

4. The GT model

The GT model (Pham et al., 2016) is nonparametric in the sense that it does not assume any particular form for either the PA function or the fitness distribution. According to this model, one starts from a seed network G_0 and then at each time step (in our case, day) t, n_t new nodes and m_t

new edges are added independently to G_{t-1} to form G_t. The new edges may emanate either from the new or from the existing nodes. When a new edge is added to the network at time t, it will connect to node i with probability

$$p_{i,t} = \frac{f(d_{i,t-1}) \times \eta_i}{\sum_{j=1}^{N_{t-1}} f(d_{j,t-1}) \times \eta_j} \tag{2}$$

where $d_{i,t-1}$ and η_i are the current (in-, out- or total-) degree and fitness of node i, respectively, and N_{t-1} is the total number of nodes at time $t-1$. Hence, if two nodes i and j both have degree d at time t and η_i is 2 η_j, the probability of a new node connecting to node i is twice as large as the probability of it connecting to node j. While $f(d_{i,t})$ represents an ability of node i to attract links that usually is increasing in time, the node fitness η_i represents something attractive about the node that is constant in time. The degree of a node grows faster if its fitness value is large, allowing nodes with high fitness to become even more "popular" than nodes that have stayed in the network for a much longer period. Note that both $f(d_{i,t})$ and η_i by definition are concerned with the ability of a node to *acquire* new edges.

Let D and N be the maximum degree of the network and the final number of nodes after T time steps, respectively (where T is the length of the estimation period), and let $z_{i,t}$ be the number of new edges that connect to node i at time t. In Pham et al. (2016), the problem of estimating $f(d); d = 1, \ldots, D$ and $\boldsymbol{\eta} = \{\eta_1, \ldots, \eta_N\}$ is formulated as the maximization of the log-likelihood function of the GT model with suitably added regularization terms to avoid overfitting. That is, the following objective function is maximized:

$$l^*(f, \boldsymbol{\eta}) = l(f, \boldsymbol{\eta}) + \text{reg}_f + \text{reg}_\eta, \tag{3}$$

where reg_f and reg_η are the regularization terms for the PA function and the node fitness, respectively, and

$$l(f, \boldsymbol{\eta}) = \sum_{t=1}^{T} \sum_{i=1}^{N} z_{i,t} \log\left(f(d_{i,t})\right) + \sum_{t=1}^{T} \sum_{i=1}^{N} z_{i,t} \log(\eta_i) - \sum_{t=1}^{T} \sum_{i=1}^{N} z_{i,t} \log\left(\sum_{j=1}^{N} f(d_{j,t})\, \eta_j\right)$$

See Appendix A for more details.

We have used the R package PAFit[1] (Pham et al., 2017) to fit the GT model to the `Vipps` data. Here, the maximization problem is solved using the Minorize–Maximization (MM) algorithm (Hunter & Lange, 2000). See Appendix B for more details. For more stable estimation of the $f(d)$-function, logarithmic binning is used. The degrees are divided into K bins, and the $f(d)$ function is estimated for each bin $k = 1, \ldots, K$ instead of each degree $d = 1, \ldots, D$. The logarithmic binning ensures small-width bins in low-degree regions with many data points, while large-width bins are created for higher degrees. Choosing the number of bins K is a trade-off between stability and accuracy. A small K means high stability at the risk of loosing fine details.

5. Experiments with Vipps data

As described in Section 2, we divide the data set into two periods; May 30th, 2015, to February 17th, 2016, and February 18th, 2016, to September 30th, 2016. The data from the first period are used to fit the GT model. Section 5.1 describes this process, while the characteristics of the estimated fitness values are discussed in Section 5.2. To check whether the estimated model also fits the data from a later period, we performed a simulation experiment in which we expanded the network at February 17th, 2016,with 730,145 nodes, corresponding to the new users joining Vipps during the period February 18th, 2016, to September 30th, 2016. The results from this experiment are treated in Section 5.3.

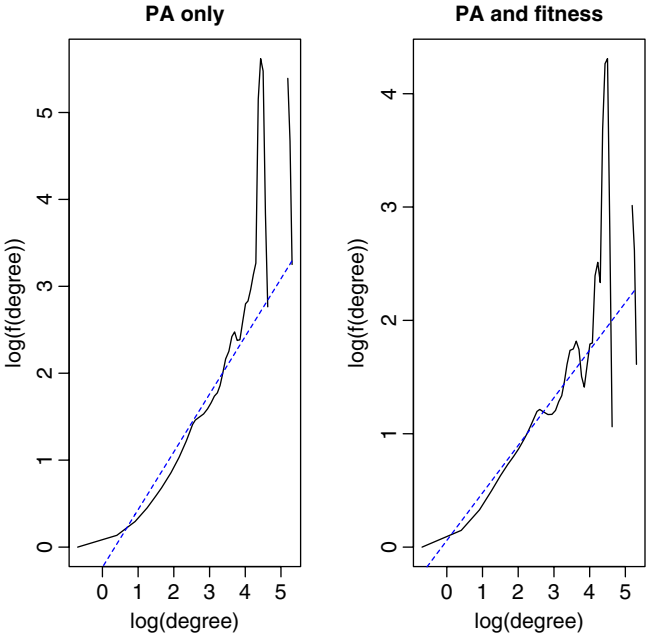

Figure 4. Left: The estimated $f(d)$-function assuming that all fitness are one. Right: The estimated $f(d)$-function when the fitness effect is taken into account. Both plots are on a log–log scale.

5.1 Estimation

The data set from the period May 30th, 2015, to February 17th, 2016, consists of 1,038,997 nodes. We first fitted the GT model to this data set fixing all η_i's to 1, i.e., ignoring fitness. The resulting $f(d)$-function is shown to the left in Figure 4 (the plot is on a log–log scale). As can be seen from the figure, the $f(d)$-function shows a strange behavior for large degrees. We believe that this is due to too few data points in this area (there are, e.g., only 7 persons who have recruited more than 55 users). For log(degrees) smaller than 4, the logarithm of the estimated $f(d)$-function is quite linear. Hence, we fitted a regression line to this part. The slope of this line is 0.66, clearly indicating the existence of the rich-get-richer phenomenon.

Next, the full GT model was fitted, with $K = 50$ bins and regularization parameters $\lambda = 0.5 \sum_{k=1}^{K-1} w_k$ and $s = 10$. The regularization parameters were determined by cross-validation, splitting the training data into two sets; a learning set and a validation set, where the learning set consisted of data from the first 189 days of the training period, while the validation set consisted of the last 74 days. The cross validation was performed as described in Pham et al. (2016). For many different combinations of λ and s the $f(d)$-function and fitness parameters were estimated using the learning data, and then the likelihood of these parameters were computed for the validation data. The solid line to the right in Figure 4 shows the logarithm of the estimated $f(d)$-function obtained when using $\lambda = 0.5 \sum_{k=1}^{K-1} w_k$ og $s = 10$. Comparing it to the one to the left in the same figure, we see that the rich-get-richer effect becomes smaller when the fit-get-richer effect is taken into account. The slope of the dotted blue line is in this case 0.42.

Because the estimated fitness values may have been influenced by the strange behavior of the $f(d)$-function for large degrees, we reestimated the fitness values keeping the $f(d)$-function fixed to the best linear fit, shown to the right in Figure 4, namely

$$f(d) = \exp\left\{0.42 \log(d) + 0.06\right\}. \tag{4}$$

The resulting fitness distribution is shown in Figure 5 and its properties are given in Table 3. As can be seen from the figure, almost all fitness values are concentrated around the mean, which is 1.

Table 3. Properties of the fitness distribution for the users recruited before February 18th, 2016

Median	Mean	99%	Max
0.99	1.00	1.36	11.92

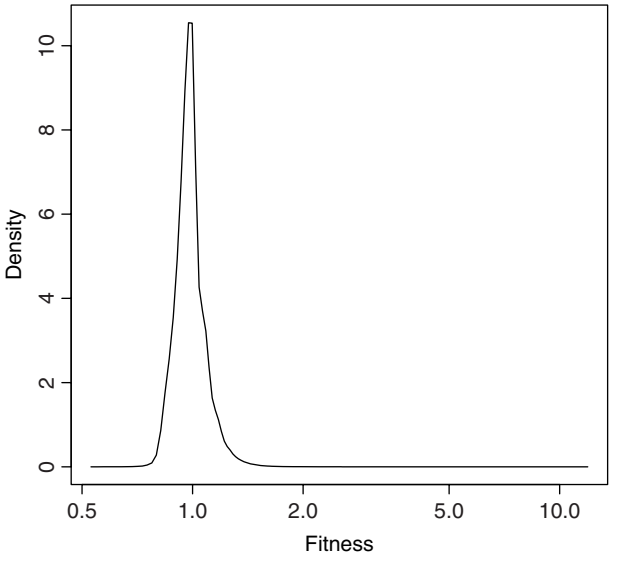

Figure 5. Fitness distribution. The *x*-axis is on log scale.

There are however some users with significantly higher fitness values, indicating that the fit-get-richer phenomenon is clearly present in this data set.

5.2 Fitness characteristics

Figures 6 and 7 show examples of degree growth curves for nodes with high and low fitness. As can be seen from the figure, there is a tendency of nodes with high fitness having very steep degree growth curves, while nodes with lower fitness having more moderate growth curves. Based on this, one would think that the fitness represents the ability of the `Vipps` user to rapidly acquire contacts. However, this is not the whole picture. Figure 8 shows the degree growth curves for two nodes that both ended up at degree 35 at February 17th, 2016. The first node enters the network at May 30th 2015 and after 27 days, its degree raises very rapidly to 32. The second node enters the network 103 days after the first and its degree steadily increases until February 17th, 2016. Based on these evolutions, one would assume that the fitness of the first is larger than that of the latter. However, on the contrary the two fitness values are 2.06 and 3.61, respectively.

This may be explained by having a closer look at Equation (2). We see that the probability of a node acquiring new edges is not only dependent on its fitness value and current degree, but also on *the corresponding quantities of all other nodes in the network*. In the early phase, when the network is small, there is a relatively small number of nodes competing for the new edges. When time goes by, however, and the size of the network increases, it becomes much harder for a certain node to attract links added to the network. Consequently, the nodes that arrive late and end up at a high degree will be the ones with the highest estimated fitness values in the GT model. This may be verified studying the equation for updating the fitness of node i:

$$\eta_i = \frac{\text{Final degree at time } T}{\sum_{t=1}^{T} f(d_{i,t}) \cdot \text{totalNumEdges}(t) \cdot \left(1/\sum_{j=1}^{N_t} \eta_j f(d_{j,t})\right)}$$

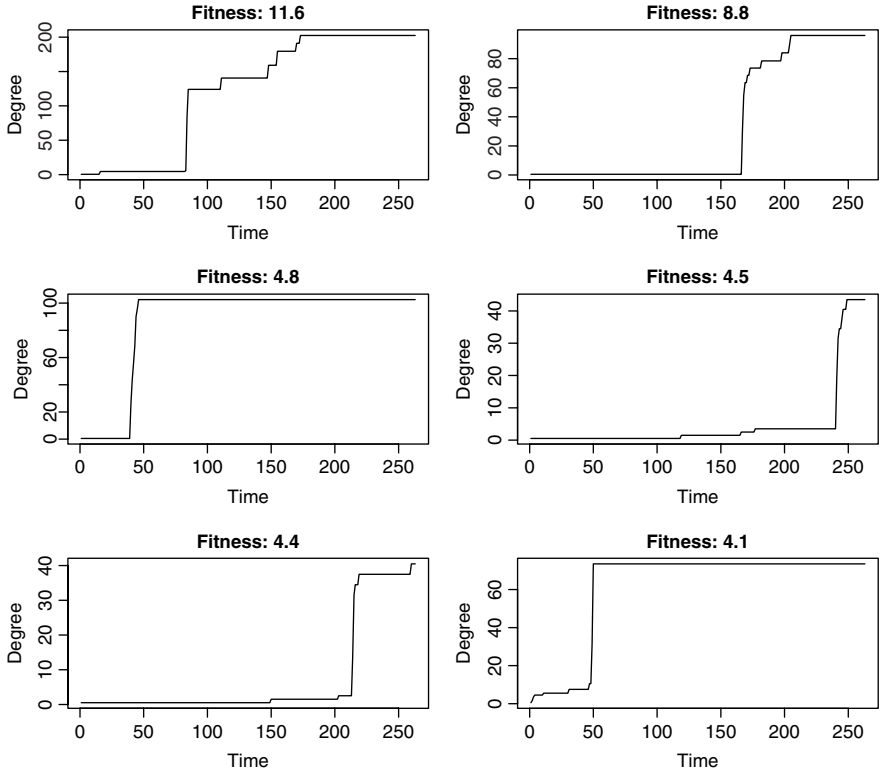

Figure 6. Degree growth curves for selected nodes with high estimated fitness.

From this equation, it is evident that of two nodes that end up with the same degree, it is the one that arrives last that get the highest fitness.

One may also view this in a different way. Assume that we have two nodes with fitness η_i and η_j and that their degrees at time t are $d_{i,t}$ and $d_{j,t}$, respectively. Assume further that $f(d) = b\,d^{\alpha}$. Then, for the probabilities $p_{i,t}$ and $p_{j,t}$ to be equal, we must have that

$$d_{i,t} = d_{j,t}\left(\frac{\eta_j}{\eta_i}\right)^{1/\alpha}$$

With $\alpha = 0.42$ like in Equation (4), η_j being twice as large as η_i means, e.g., that the degree of node i must be 5 times larger than the degree of node j for the probabilities of attracting new edges to be equal.

Returning to our example in Figure 8, at day 27 there are only 123,741 nodes competing for 3,123 new edges. Hence, the gray node "does not need" a large fitness value to attract many links. However, 103 days later when the black node enters the network, the total number of nodes in the network is approximately 500,000, while the number of new edges is still approximately 3,000. Nevertheless, this node manages to acquire 35 new contacts during the consecutive 163 days, while the corresponding number for the first node is 3. Hence, the fitness of the black node must be much higher than that of the gray.

5.3 Simulation study

During the period from February 18th, 2016, to September 30th, 2016, 730,145 new users joined `Vipps`. To check whether the model estimated for the period May 30th 2015, to February 17th,

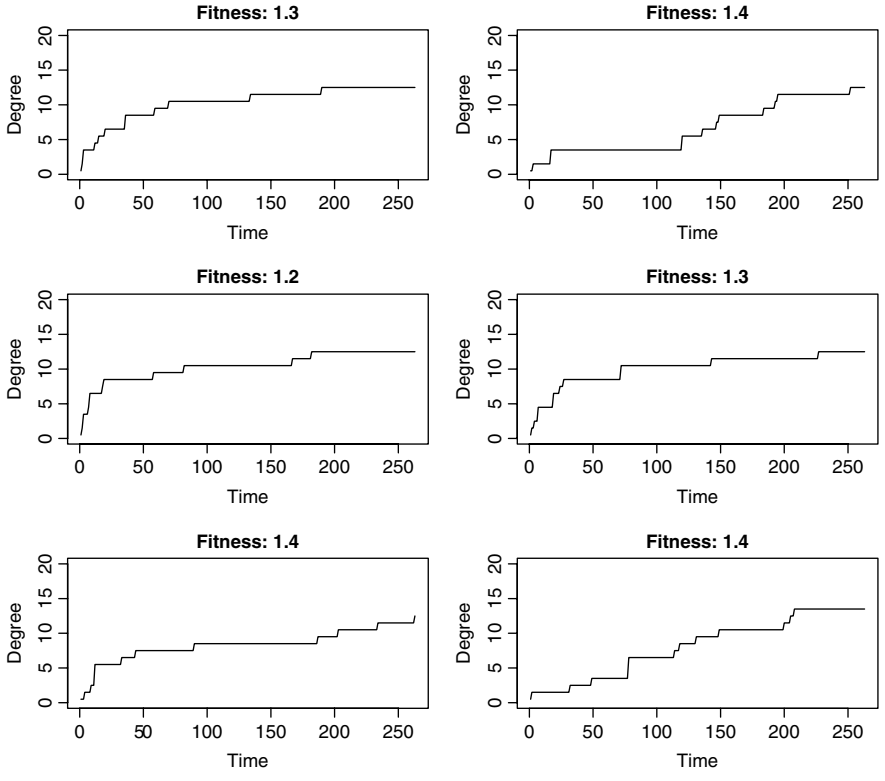

Figure 7. Degree growth curves for selected nodes with low estimated fitness.

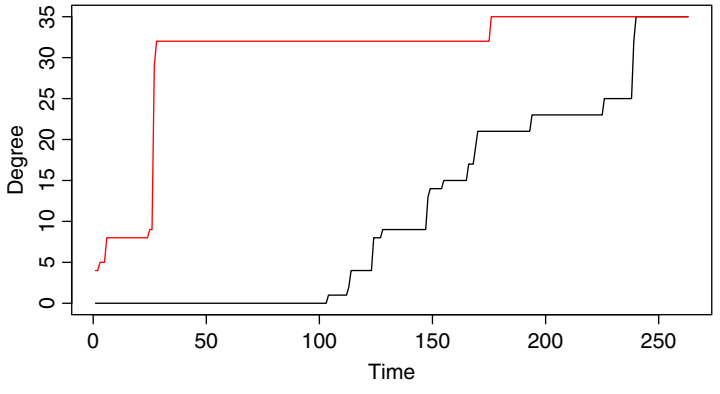

Figure 8. Degree growth curves for two nodes with fitness values equal to 3.61 (black) and 2.06 (red), respectively.

2016, also fits the data from the later period, we decided to perform a simulation experiment in which we expanded the network from February 17th with 730,145 nodes. The daily growth of the `Vipps` network depends on several factors, e.g., on external marketing. Our main aim is to study the mechanism that governs the growth, not the size of it. Hence, we decided to use the same number of new users every day in the simulated data set as what was observed in the actual data set.

In the simulation procedure, we assumed the log-linear PA-function from Equation (4). As far as the fitness values were concerned, we used the estimated values for the users already present

Table 4. Different characteristics for the six Vipps users recruited before February 18th, 2016, having highest estimated fitness

User	Estimated fitness	Total number of friends	Day joining Vipps	Total number recruit	Test number recruit	Pos number recruit
1	11.92	581	13	208	0	0
2	8.76	712	165	98	0	0
3	4.82	289	39	106	1	10
4	4.52	314	118	42	0	2
5	4.37	730	149	56	16	48
6	4.15	424	162	54	15	35

in the network. For the new users, we first tried to fit a lognormal and a gamma distribution, respectively, to the empirical fitness distribution in Figure 5. However, none of these distributions provided an adequate fit to the estimated fitness values. Hence, we decided to simulate fitness values for the new users by bootstrapping from the observed distribution in Figure 5.

To check whether the simulated network evolution during the period from February 18th, 2016, to September 30th, 2016, has the same characteristics as the actual network evolution, we computed the mean absolute error (MAE)[2]:

$$MAE = \frac{1}{N_{old}} \sum_{i=1}^{N_{old}} |final\_deg\_real(i) - final\_deg\_est(i)|$$

where N_{old} is the number of users already in the system at February 18th, 2016, and $final\_deg\_real(i)$ and $final\_deg\_est(i); i = 1, ..., N_{old}$ are their actual and simulated degrees at September 30th, 2016. The simulated degrees are obtained as follows. At day t in the simulation period J_t new nodes $j = 1, ..., J_t$ are inserted into the network. For each of these new nodes, one and only one edge is produced to one of the nodes existing at day $t - 1$. The probability of choosing a specific existing node i is given in Equation (2). From this formula, we see that this probability is based on the fitness of node i as well as its degree at day $t - 1$ (numerator) and the fitness and current degree of all nodes that existed at day $t - 1$ (denominator). Hence, the order in which the new nodes enter the network at day t will not matter.

This MAE was computed both for our estimated GT model and for the GT model with $\alpha = 0.66$ and all η_i^s fixed to one. The resulting values were 0.718 and 0.732, respectively. Hence, the difference between the two models is not very large and probably not significantly different from zero. By having a closer look at the real data, the following may be observed. First, most of the users having a large degree and/or fitness at the end of the training period recruit zero or very few new users during the testing period. This might be due to the fact that they almost have reached their full potential during the training period, i.e., that most of their friends already have been recruited by February 18th, 2016.[3] Table 4, containing different figures for the six `Vipps` users from Figure 6 shows that this is not very far from the truth. For each user, the table shows the following quantities: (i) its estimated fitness value, (ii) its total number of friends,[4] (iii) the day at which the user was recruited, (iv) the total number of other users being recruited by this user, (v) the number of other users being recruited by this user during the test period, and (vi) the number of friends who potentially could have been recruited during the test period. The two users in Table 4 with the highest estimated fitness values do not recruit any new users during the test period. This is not strange, since they already have reached their full potential. All their friends have either been recruited by themselves or by others.

Estimating the fitness distribution for the users recruited after February 18th, 2016, using the same framework as described in Section 5.1, we get the properties shown in Table 5. By comparing the figures in this table to the ones in Table 3, we see that none of the new users have very high

Table 5. Properties of the fitness distribution for the users recruited after February 18th, 2016

Median	Mean	99%	Max
0.98	1.00	1.24	3.17

Table 6. Different characteristics for the six Vipps users recruited after February 18th, 2016, having highest estimated fitness

User	Estimated fitness	Total number of friends	Day joining Vipps	Total number recruit	Test number recruit	Pos number recruit
1	3.17	694	399	24	24	77
2	3.14	376	304	25	25	96
3	2.93	594	292	23	23	119
4	2.86	718	333	22	22	34
5	2.65	637	315	19	19	124
6	2.65	148	366	19	19	41

Table 7. Properties of the fitness distribution for the users recruited before August 9th, 2015

Median	Mean	99%	Max
0.98	1.00	1.39	7.27

fitness values. Table 6 shows the properties for the six users recruited after February 18th, 2016, with highest estimated fitness values. The numbers in the third column of this table show that these users actually have more friends than the users in Table 4 on average. However, from the last column in the same table it is evident that the majority of their friends already have been recruited by someone else. Hence, even if these users might be as efficient in recruiting new users as the ones in Table 4, they will not be able to reach the same level, simply because the number of possible "prospects" is smaller.

In addition to the above simulation experiment, we also tried to reduce the training set, to check whether the model correctly predicts the evolution of the network in this case as well. More specifically, we fitted the GT model to data from May 30th, 2015, to August 8th, 2015, only. This data set consists of 301,235 nodes. The $f(d)$ function was then estimated to[5]

$$f(d) = \exp\left\{0.52 \log(d) + 0.10\right\} \tag{5}$$

and the properties of the fitness distribution were as shown in Table 7. Having estimated the model, we performed a simulation experiment in which we expanded the network from August 8th, 2015, with 1,467,907 nodes. Finally, we computed the MAE:

$$MAE = \frac{1}{N_{old}} \sum_{i=1}^{N_{old}} |final\_deg\_real(i) - final\_deg\_est(i)|$$

where N_{old} now is the number of users already in the system on August 8th, 2015, and $final\_deg\_real(i)$ and $final\_deg\_est(i)$, $i = 1, ..., N_{old}$, are their actual and simulated degrees at September 30th, 2016. The resulting MAE was 1.81, i.e., significantly larger than for the original training set. By having a closer look at the smaller training data set, we observe that only 21% of the nodes have a degree which is larger than 1 at the end of the training period. Hence, this training period seems to be too short to get proper estimated fitness values. The increase in MAE may also be partly due to the fact that during most of the test period, there are far more nodes with simulated fitness values than with estimated ones. The simulated fitness values are generated from the distribution given by Table 7. Based on the information in Table 3, we believe that many

of these values are likely to be smaller than the true ones, meaning that the original nodes will have less competition in the simulation study than in real life. This is verified by comparing the total number of simulated edges connecting to the original nodes during the test period (507,572) to the corresponding true number of edges (416,567).

6. Summary and discussion

The peer-to-peer mobile payment solution `Vipps`, which was launched in May 2015, is now the number one downloaded app in Norway. The `Vipps` transaction data may be viewed as a graph for which the users correspond to the nodes, and the transactions between the users define the edges. In this paper, we have used the subgraph consisting of all nodes (users), but only having an edge between nodes A and B if user B was recruited by user A. The `Vipps` graph is unique in the sense that we may follow the evolution from the very first user up to now. By fitting a combined PA and fitness model to this data set, we have shown that the intrinsic quality of the nodes in the `Vipps` network plays an important part in the evolution of the network.

The results in this study may be used for viral marketing. Viral marketing refers to marketing techniques that use social networks to try to produce increases in brand awareness or to achieve other marketing objectives such as product sales through self-replicating viral processes, analogous to the spread of viruses. One way of encouraging positive word-of-mouth is by distributing reduced price or free products to target customers (seed users), who then hopefully will encourage their friends to buy the product (Stonedahl et al., 2010). If the bank in the future wants to launch a new solution which is similar to `Vipps`, a smart strategy might be to select the persons with the highest Vipps fitness values as seed users, since these persons are the ones who seem to recruit most other users in shortest time.

Funding. This work was supported by the Norwegian Research Council grant 237718 (Big Insight).

Acknowledgments. The authors are grateful to Thong Pham for valuable advice concerning the use of the PAFit R package. We also thank the referees and Action Editor for their help to improve this paper with their constructive comments and suggestions.

Conflict of interest. The authors Kjersti Aas and Hanne Rognebakke have nothing to disclose.

Notes

1 https://cran.r-project.org/web/packages/PAFit/index.html
2 Note that we ignore all the users joining Vipps after February 18th, 2016, when computing the MAE.
3 Nodes A and B being "friends" here mean as previously stated that there has been at least one transaction between A and B during the whole time period from May 30th, 2015, to September 30th, 2016.
4 We assume that the network has reached its saturation point at September 30th, 2016.
5 We used the same regularization parameters as for the longer training period.

References

Barabasi, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, *286*, 509–512.
Barabasi, A.-L., Albert, R., & Jeong, H. (2000). Scale-free characteristics of random networks: The topology of the world-wide web. *Physica A: Statistical Mechanics and Its Applications*, *281*, 69–77.
Bianconi, G., & Barabási, A. L. (2001). Competition and multiscaling in evolving networks. *Europhysics Letters*, *54*, 436–442.
Borgs, C., Chayes, J., Daskalakis, C., & Roch, S. (2007). First to market is not everything: An analysis of preferential attachment with fitness. In *Proceedings of the thirty-ninth annual acm symposium on theory of computing*. STOC '07 (pp. 135–144). New York, NY, USA: ACM.

Caldarelli, G., Capocci, A., De Los Rios, P., & Muñoz, M. A. (2002). Scale-free networks from varying vertex intrinsic fitness. *Physical Review Letters*, *89*, 258702-1–258702-4.
Callaway, D. S., Hopcroft, J. E., Kleinberg, J. M., Newman, M. E. J., & Strogatz, S. H. (2001). Are randomly grown graphs really random? *Physical Review E*, *64*, 041902.
Cole, S. R., Chu, H., & Greenland, S. (2014). Maximum likelihood, profile likelihood, and penalized likelihood: A primer. *American Journal of Epidemiology*, *179*, 252–260.
Dereich, S., & Mörters, P. (2009). Random networks with sublinear preferential attachment: degree evolutions. *Electronic Journal of Probability*, *14*, 1222–1267.
Hunter, D., & Lange, K. (2000). Quantile regression via an MM algorithm. *Journal of Computational Statistics and Data Analysis*, *9*, 60–77.
Iñiguez, G., Ruan, Z., Kaski, K., Kertész, J., & Karsai, M. (2017). *Service adoption spreading in online social networks*. arXiv preprint, arXiv:1706.09777.
Kondor, D., Posfai, M., Csabai, I., & Vattay, G. (2014). Do the rich get richer? An empirical analysis of the bitcoin transaction network. *PLOS ONE*, *9*, e86197.
Kong, J. S., Sarshar, N., & Roychowdhury, V. P. (2008). Experience versus talent shapes the structure of the web. *Proceedings of the National Academy of Sciences*, *105*(37), 13724–13729.
Krapivsky, P. L., Redner, S., & Leyvraz, F. (2000). Connectivity of growing random networks. *Physics Review Letters*, *85*, 4629–4632.
Krapivsky, P. L., Rodgers, G. J., & Redner, S. (2001). Organization of growing networks. *Physical Review E*, *63*, 066123-1–066123-14.
Kunegis, J., Blattner, M., & Moser, C. (2013). *Preferential Attachment in Online Networks: Measurement and Explanations*. Presented at WebSci'13 Conference, Paris.
Leskovec, J., Singh, A., & Kleinberg, J. (2006). Patterns of influence in a recommendation network. In *Proceedings of the 10th pacific-asia conference on advances in knowledge discovery and data mining*. PAKDD'06 (pp. 380–389). Berlin, Heidelberg: Springer-Verlag.
Pham, T., Sheridan, P., & Shimodaira, H. (2015). PAFit: A statistical method for measuring preferential attachment in temporal complex networks. *PLOS ONE*, *9*, e0137796.
Pham, T., Sheridan, P., & Shimodaira, H. (2016). Joint estimation of preferential attachment and node fitness in the evolution of complex networks. *Nature Scientific Reports*, *6*, 1–13.
Pham, T., Sheridan, P., & Shimodaira, H. (2017). *PAFit: An R Package for Estimating Preferential Attachment and Node Fitness in Temporal Complex Networks*. arXiv preprint, arXiv:1704.06017.
Redner, S. (1998). How popular is your paper? an empirical study of the citation distribution. *The European Physical Journal B - Condensed Matter and Complex Systems*, *4*, 131–134.
Stonedahl, F., Rand, W., & Wilensky, U. (2010). Evolving Viral Marketing Strategies. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*.
Yule, G. U. (1925). A mathematical theory of evolution, based on the conclusions of Dr. J. C. Willis, F.R.S. *Philosophical Transactions of the Royal Society B*, *213*, 21–87.

Appendix A: GT model: More details

According to the GT model, one starts from a seed network G_0 and then at each time step t, n_t new nodes and m_t new edges are added independently to G_{t-1} to form G_t. The likelihood of the data at time step t is given by

$$P(G_t|G_{t-1}, \boldsymbol{\theta}_t, f, \eta) = P(m_t, n_t|G_{t-1}, \boldsymbol{\theta}_t)\, P(G_t|G_{t-1}, m_t, n_t, f, \boldsymbol{\eta})$$

where m_t and n_t are, respectively, the actual number of edges and nodes that appear between times $t-1$ and t, and $\boldsymbol{\theta}_t$ are the parameters in the simultaneous distribution of m_t and n_t. This distribution is assumed not to depend on $f(d)$ and $\boldsymbol{\eta}$, meaning that we can ignore $\boldsymbol{\theta}_t$. Hence, the log-likelihood function of the whole data set may be written as

$$l(f, \boldsymbol{\eta}) = \sum_{t=1}^{T} \log P(G_t|G_{t-1}, m_t, n_t, f, \boldsymbol{\eta})$$

Let $z_{i,t}$ be the number of new edges that connect to node i at time t. Given m_t, the quantities $z_{1,t}, \ldots, z_{N,t}$ follow a multinomial distribution. Hence, the log-likelihood function may be written as

$$l(f, \boldsymbol{\eta}) = \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t}\ \log\left(f(d_{i,t})\right) + \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \log\left(\eta_i\right) - \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \log\left(\sum_{j=1}^{N} f(d_{j,t})\, \eta_j\right)$$

Note that when computing $z_{i,t}$, the edges corresponding to nodes that appear the same day as the node itself are not taken into account. This is due to the fact that within each day we do not know the order in which the users were recruited. Hence, we do not want to introduce spurious effects by randomizing ties.[6]

As stated in Section 4, regularization is used to avoid overfitting, meaning that the following objective function is maximized:

$$l^*(f, \boldsymbol{\eta}) = l(f, \boldsymbol{\eta}) + \text{reg}_f + \text{reg}_\eta$$

The regularization term for the PA function is given by

$$\text{reg}_f = -\frac{\lambda}{\sum_{d=1}^{D-1} w_d} \sum_{d=1}^{D} w_d \Big(\log\big(f(d+1)\big) + \log\big(f(d-1)\big) - 2\ \log\big(f(d)\big)\Big)^2 \tag{A1}$$

where λ determines the amount of regularization of the $f(d)$-function. This means that reg_f penalizes the second-order differentiation of $\log(f(d))$, encouraging the form $f(d) = d^\alpha$. The latter may also be verified as follows. If $f(d) = d^\alpha$, then $\log(f(d))/\log(d) = \alpha$, and we also have that $\log(f(d+1))/\log(d+1) = \alpha$ and $\log(f(d-1))/\log(d-1) = \alpha$. This again implies that $\log(f(d+1))/\log(d+1) - \log(f(d))/\log(d) = \log(f(d))/\log(d) - \log(f(d-1))/\log(d-1)$, which is equivalent to

$$\log\big(f(d+1)\big)/\log(d+1) + \log\big(f(d-1)\big)/\log(d-1) - 2\ \log\big(f(d)\big)/\log(d) = 0$$

For moderately large values of d we have that $\log(d+1) \approx \log(d) \approx \log(d-1)$, meaning that the last equation may be written as

$$\log\big(f(d+1)\big) + \log\big(f(d-1)\big) - 2\ \log\big(f(d)\big) = 0$$

The weights w_d may be arbitrarily chosen. We follow Pham et al. (2016) and set them to

$$w_d = \sum_{t=1}^{T} m_{d,t}$$

where $m_{d,t}$ is the number of edges that connect to a degree-d node at time t. With this choice, one balances the strength of the regularization and the observed data.

The regularization term for the node fitness is given by

$$\text{reg}_\eta = \sum_{i=1}^{N} \big\{(s-1)\ \log(\eta_i) - s\,\eta_i\big\} \tag{A2}$$

Multiplying the likelihood by a penalty function is equivalent to assigning a Bayesian prior distribution to the unknown parameters; see, e.g., Cole et al. (2014). The regularization term (A2) is equivalent to a Gamma prior on η_i, with mean and variance equal to 1 and $1/s$, respectively. The larger the value of s, the smaller the variance of the fitness distribution. When $s \to \infty$, all η_i's will be equal to 1. Hence, the special case of the classical PA model is obtained for the combination $\lambda = \infty$, $s = \infty$.

Appendix B: GT model: MM algorithm

As stated in Section 4, the maximization of the penalized log-likelihood function is performed using the MM algorithm (Hunter & Lange, 2000). The MM algorithm is an iterative optimization method which works by specifying a surrogate function that majorizes or minorizes the original objective function. Optimizing the surrogate function will drive the objective function upward or downward until a local optimum is reached. A minorize function $Q(f, \boldsymbol{\eta})$ for $l(f, \boldsymbol{\eta})$ should satisfy the following two requirements:

$$\begin{aligned} Q(f, \boldsymbol{\eta}) &< l(f, \boldsymbol{\eta}) \text{ for all } f \text{ and } \boldsymbol{\eta} \\ Q(f^q, \boldsymbol{\eta}^q) &= l(f^q, \boldsymbol{\eta}^q) \text{ for all iterations } q \end{aligned}$$

It can easily be shown that an appropriate minorize function for $l(f, \boldsymbol{\eta})$ then is

$$\begin{aligned} Q(f, \boldsymbol{\eta}) = &\sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t}\ \log\big(f(d_{i,t})\big) + \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \log(\eta_i) \\ &- \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \log\left(\sum_{j=1}^{N} f^q(d_{j,t})\,\eta_j^q\right) - \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \frac{\sum_{j=1}^{N} f^q(d_{j,t})\,\eta_j^q}{\sum_{j=1}^{N} f(d_{j,t})\,\eta_j} + \sum_{t=1}^{T}\sum_{i=1}^{N} z_{i,t} \end{aligned}$$

Let A_k be the value of $f(d)$ for bin k, and let $B(i,t)$ be the bin of node i at time t. We maximize Q with respect to $\boldsymbol{A} = \{A_1, A_2, \ldots, A_K\}$ and $\boldsymbol{\eta} = \{\eta_1, \eta_2, \ldots, \eta_N\}$ by solving the equations $\partial Q/\partial \boldsymbol{A} = 0$ and $\partial Q/\partial \boldsymbol{\eta} = 0$ obtaining

$$A_k = \frac{\sum_{t=1}^{T} \sum_{B(i,t)=k} z_{i,t}}{\sum_{t=1}^{T} \frac{\sum_{i=1}^{N} z_{i,t}}{\sum_{j=1}^{N} A_{B(j,t)} \eta_j} \sum_{B(i,t)=k} \eta_i} \quad \text{for } k = 1, \ldots, K \tag{B1}$$

$$\eta_i = \frac{\sum_{t=1}^{T} z_{i,t}}{\sum_{t=1}^{T} \frac{\sum_{i=1}^{N} z_{i,t} A_{B(i,t)}}{\sum_{j=1}^{N} A_{B(j,t)} \eta_j}} \quad \text{for } i = 1, \ldots, N \tag{B2}$$

Since A_k and η_i appear on both sides of these equations, we must use an iterative procedure. Starting from some initial values $\boldsymbol{A}^0 = \{1, A_2^0, \ldots, A_K^0\}$ and $\boldsymbol{\eta}^0 = \{1, \eta_2^0, \ldots, \eta_N^0\}$ at iteration $q = 0$, this algorithm iteratively calculates $\boldsymbol{A}^q$ and $\boldsymbol{\eta}^q$ until some convergence condition is met. For each iteration, the A_k's, $k = 1, \ldots, K$, are first updated. Then, the η_i's, $i = 1, \ldots, N$, are updated using the updated values for the A_k's. It should be noted that parallel processing may be used both when updating the A_k's and the η_i's.

Optimizing the penalized likelihood function $l^*(f, \boldsymbol{\eta})$ instead of $l(f, \boldsymbol{\eta})$, Equation (B2) is slightly modified to

$$\eta_i = \frac{\sum_{t=1}^{T} z_{i,t} + s - 1}{\sum_{t=1}^{T} \frac{\sum_{i=1}^{N} z_{i,t} A_{B(i,t)}}{\sum_{j=1}^{N} A_{B(j,t)} \eta_j} + s} \quad \text{for } i = 1, \ldots, N \tag{B3}$$

The new formula for A_k is however no longer available in closed form. Instead it is the solution of a univariate equation that is obtained by first combing the function Q above with a minorize function for the regularization term reg_f from Equation (A1). If the new minorize function is denoted $Q_A(\cdot)$, the next step is then solving the equation $\partial Q_A/\partial \boldsymbol{A} = 0$. The minorize term for reg_f may be found in the supplement to Pham et al. (2016). This is chosen in such a way that solving the equation $\partial Q_A/\partial \boldsymbol{A} = 0$ may be separated into K univariate problems $\partial Q_A/\partial A_k = 0$, which may be easily solved in parallel.

Cite this article: Aas K., and Rognebakke H. The evolution of a mobile payment solution network. *Network Science* https://doi.org/10.1017/nws.2019.5