Trust in Bitcoin Exchange Networks

Olivier Moindrot
olivierm@stanford.edu

December 11, 2017

Abstract

We consider two marketplaces to exchange bitcoins and dollars, Bitcoin OTC and Bitcoin Alpha. In these settings, trust is a fundamental necessity since any user can initiate a transaction, receive money from the other user and never send money back in the other currency. To achieve a web of trust, users rate each other after successful or unsuccessful trades. We analyze these datasets using the fairness and goodness method from [5]. These two measures can be used to predict the weight of a given edge (the rating that u gives to v) with good accuracy. We show that whenever the edge is reciprocated, social interaction becomes a better predictor than goodness, and users apply the talion law: "an eye for an eye".
We analyze the relationship between trust in these networks with the exchange price of bitcoin and find that there is no evident correlation between these two variables.

1 Introduction

Bitcoin's development in 2008 and 2009 introduced a radical new concept for money and currency [7]. Bitcoin was the first example of a valuable digital asset without any real world backing, or any central authority like a bank.
The model developed by Satoshi Nakamoto introduced a blockchain of transactions that could reach global consensus with Byzantine fault tolerance of up to 49% of the computational power. The key to achieve this consensus is that blocks are added by miners who solve the proof of work computational puzzle; this can be verified very easily and ensure that only one miner will add the next block most of the time.
Bitcoin as a currency has been the biggest recent revolution in the financial space. The price of one bitcoin has increased from virtually nothing to an all-time high above $18,000 in 2017 (see figure 1). Its introduction as a new programmable virtual money has led to the emergence of a whole cryptocurrency space, including Ethereum [9].

We don't study directly the bitcoin graph, but instead focus on two marketplaces to trade bitcoins against dollars. At the beginning of bitcoins, people even traded bitcoins against dollars in the street at some locations. The two marketplaces, Bitcoin OTC and Bitcoin Alpha are described in section 3. The advantage of using these smaller networks is that each edge corresponds to some weight, the rating from user u to user v. This forms a web of trust between users, and it allows two users who don't know each other to perform a transaction based on the aggregated trust that they each possess.

Fairness and goodness are two measures of nodes introduced by [5] in the context of weighted signed networks that capture how fair nodes are (do their ratings assess the goodness of users) and how good they are (how much can we trust them, according to the network). We conduct experiments in section 4.3 to try to predict the rating of a given edge, and we show that when two users u and v both rate each other, their ratings have a high correlation. Therefore, social interactions matter more than overall goodness in those cases.

We finally analyze the co-evolution of bitcoin price and trust in the marketplaces in section 5.3. We don't find any correlation between these two variables, and conclude that trust is not affected by the exchange price of bitcoin.

2 Related work

Properties of social networks are influenced by external events. For instance, a natural catastrophe will make friends call each other to make sure that everyone is safe. A big dump in the price of bitcoin might make people rush to sell before the price gets too low.

The authors of [8] investigate how price variations affect a network of messages between decision makers in a large hedge fund. Two types of nodes exist: insiders working at the company, and outsiders exchanging messages with insiders.
The authors define the notion of a "turtled up" and "open" networks. Using different metrics like clustering or proportion of border edges (insider-outsider), they examine how the network reacts to changes of the prices in the stock market.
When there is a big jump in price, in either direction, the network tends to turtle up. Insiders consolidate links with other insiders and talk less to outsiders. The hedge fund as a whole regroups itself to decide of their next move. However, this change in the network structure is not permanent, and the authors find that the network can be qualified as elastic since it goes back to its original state in two to three days on average.

As in the previous paper [8], the goal of [4] is to link properties of an underlying network (bitcoin) to the current market price of bitcoin in US dollar. Whereas the previous paper focused on network changes under stress from external events (like price drops), this paper focuses on the opposite: how to predict the price of bitcoin (external event) based on the evolution of the transaction network.

Our goal in this paper is to analyze the amount of trust in weighted signed networks like Bitcoin OTC and Bitcoin Alpha. We base a good amount of our analysis on [5], which defines fairness and goodness as two intertwined measures that describe each node in a weighted signed network. The exact definition and algorithm are explained in section 4.2.
This is the first paper to deal with networks with signed edges, and as such there is not previous literature on this topic. However, there has been a great amount of research in the edge weight prediction domain, such as EigenTrust [1] and TidalTrust [2].

3 Datasets and Representation

3.1 Bitcoin Price Dataset

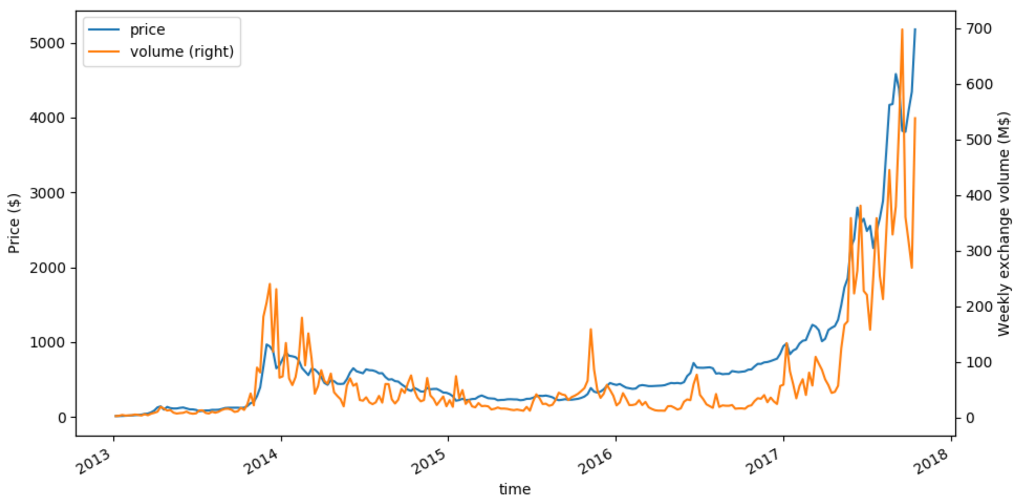

Figure 1: Price and exchange volume of bitcoin over time

Our first source of data is the historical exchange price of bitcoin released on Kaggle[1]. The data starts on January 1st, 2012 and ends on October 20th, 2017. We have values for the exchange price of bitcoin and the transaction volume at each minute between the start and end dates. We only used the exchange Bitstamp[2] a source of data since they have been running for the longest. We aggregate the values by taking the average price and the sum of transaction volume over each day.

The increase in price of Bitcoin is impressive from around $4 at the beginning of trading to almost $7,000 at the end of the data. In figure 1, we can see the evolution over time of the daily exchange price of bitcoin in USD. We also included the weekly exchange volume in USD.

3.2 Bitcoin OTC Trust network

Rating	Fraction	Guideline
10	2.1%	You trust this person as you trust yourself.
9	0.30%	
8	0.78%	Large number of high-value transactions, long period of association, very trustworthy.
6, 7	1.3%	
5	3.6%	You've had a number of good transactions with this person.
2, 3, 4	25.5%	
1	56.3%	One or two good transactions with this person
-1	1.7%	Person strikes you as a bit flaky. Unreasonable/unexpected delays in payment, etc.
-2 to -9	1.5%	
-10	6.78%	Person failed to hold up his end of the bargain, took payment and ran, fraudster.

Table 1: Rating guidelines from the OTC wiki and fraction of the overall ratings for each rating.

Our second source of data is the network of people using the Bitcoin OTC trusted exchange platform[3]. On Bitcoin OTC, people can exchange bitcoins and build up trust as they take part in more exchanges. Users rate each other in a scale of -10 (total distrust) to 10 (total trust). The guidelines for the rating are available in the wiki page for OTC[4] and are shown in table 1.

We can see that because there are only recommendations for ratings $-10, -1, 1, 5, 8$ and 10, we have a higher amount of ratings for these values. The most common rating is 1, which corresponds to the first exchanges between two users that go well. 6.78% of ratings are -10 which means that there is a significant number of fraudulent transactions where one user never sends the money.

The dataset is available on SNAP[5] and is the first public weighted signed network.

A weighted signed network is a directed weighted graph, where each edge (u, v) has a weight $w(u, v) \in [-1, 1]$. The weight of an edge (u, v) is associated with how much user u trusts user v. The higher a rating, the higher the trust.

You can see in figure 2 the degree distribution of the OTC network. The degree distribution follows a power law with estimated parameter $\alpha = 1.67$. We estimated the parameter with Maximum Likelihood Estimator.

3.3 Bitcoin Alpha Trust Network

Our third source of data is the Bitcoin Alpha Trust Network, which is also available on SNAP. This dataset is similar in almost every way to the OTC bitcoin trust network. It is also a weighted directed graph, and also has ratings from -10 to 10. Although the OTC exchange is still active, the Alpha network is no longer active and has data up to 2016 January 22th.

You can see in figure 2 the degree distribution of the Alpha network. The degree distribution follows a power law with estimated parameter $\alpha = 1.63$. We estimated the parameter with Maximum Likelihood Estimator.

[1] https://www.kaggle.com/mczielinski/bitcoin-historical-data
[2] https://www.bitstamp.net
[3] https://www.bitcoin-otc.com/
[4] https://wiki.bitcoin-otc.com/wiki/OTC_Rating_System
[5] http://snap.stanford.edu/data/soc-sign-bitcoinotc.html

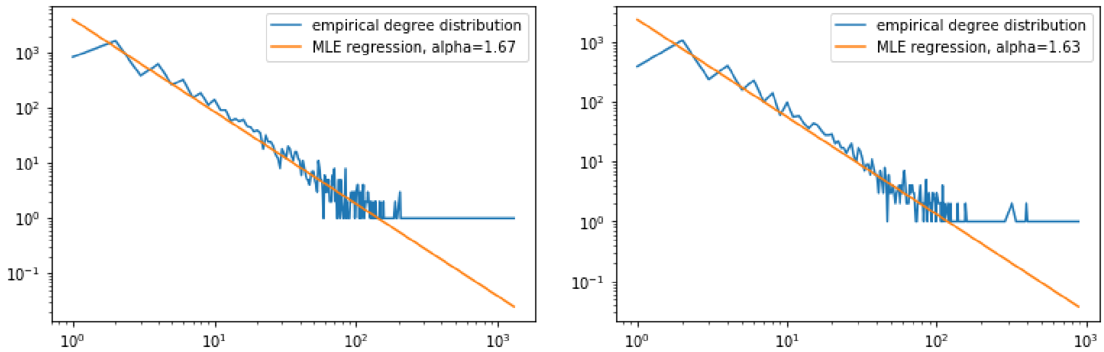

Figure 2: Degree distributions of the OTC network (**left**) and Alpha network(**right**).

4 Method

Our goal here is to study the two trust networks, and see if the price of bitcoin influences the behavior of users in these networks. In particular, we want to establish whether the price of bitcoin has an impact on the overall trust in the OTC and Alpha networks.

4.1 Price and transactions are linked

A medium of exchange like the US dollar or bitcoin can be valued through the **equation of exchange**, first introduced by John Stuart Mill in 1884 [6]. The equation links the total money supply M, the velocity of money V, the price level P and the transaction volume T in a simple equation:

$$MV = PT$$

If we rewrite the equation using $H = 1/V$ the time that each user holds a coin on average before doing a transaction, and $C = 1/P$ the actual price of one coin, we obtain:

$$MC = TH$$

The left term corresponds to the market capitalization of the currency. The right term corresponds to the amount of transactions each day T multiplied by the number of days each coin is held H.

This equation shows that the underlying price C of bitcoin is inherently linked to the behavior of the network, modeled by the amount of daily transactions T and the holding behavior of users H.

We can actually see in figure 1 that increases in price correlate with increases in transaction volume. Assuming that the total number of coins M stays the same over a short period (a month) and that people tend to hold coins for the same amount H over this period, we find a linear relationship between the price C and the volume of transactions T, which corresponds to our data.

4.2 Fairness and Goodness in a Weighted Signed Network

We introduce the *fairness* and *goodness* measures following [5]. These measures are defined on a weighted signed network, and can have applications to predict the weight of edges, as will be seen in section 4.3 .

Intuitively, the goodness of a node in a signed network measures how likely this node is to receive good ratings from other users. We want the goodness of a node to exactly represent how much the network trusts this node. However, we need to discard the ratings given by unfair nodes, because they deviate from the actual goodness of a node and mean less.
Therefore, we define the fairness of a node as how reliable its ratings are. The fairest node would be a node that always rate nodes with their intrinsic goodness.

We can see that fairness and goodness metrics are dependent on each other, like in Hubs and Authorities [3]. To obtain the fairness and goodness of every node, we need to run an algorithm that will make multiple iterations until the values converge for each node.

Fairness and goodness need to satisfy two equations:

$$g(v) = \frac{1}{|in(v)|} \sum_{u \in in(v)} f(u)w(u,v)$$

$$f(u) = 1 - \frac{1}{|out(u)|} \sum_{v \in out(u)} \frac{|w(u,v) - g(v)|}{2}$$

The first equation means that a score is as good as the mean of its ratings, weighted by fairness. The second equation creates a fairness score in $[0,1]$ for u given how well the ratings of u correlate with true goodness.

By iterating on values of f and g using the equations above, we can converge to the true fairness and goodness values for every node. The algorithm is detailed in figure 3.

Input: A WSN $G = (V, E, W)$
Output: Fairness and Goodness scores for all vertices in V
Let $f^0(u) = 1$ and $g^0(u) = 1$, $\forall u \in V$
$t = -1$
do
    $t = t + 1$
    $g^{t+1}(v) = \frac{1}{|in(v)|} \sum_{u \in in(v)} f^t(u) \times W(u,v)$, $\forall v \in V$
    $f^{t+1}(u) = 1 - \frac{1}{2|out(u)|} \sum_{v \in out(u)} |W(u,v) - g^{t+1}(v)|$, $\forall u \in V$
while $\sum_{u \in V} |f^{t+1}(u) - f^t(u)| > \epsilon$ or $\sum_{u \in V} |g^{t+1}(u) - g^t(u)| > \epsilon$
Return $f^{t+1}(u)$ and $g^{t+1}(u)$, $\forall u \in V$

Figure 3: Fairness and Goodness algorithm (from [5]).

4.3 Signed Edge Weight Prediction

The $F \times G$ score One of the direct applications of the *fairness and goodness algorithm* is to predict the weight of a missing edge.

Given two nodes u and v, we can predict that the weight of the directed edge (u,v) will be $f(u) \times g(v)$. The intuition behind this predicted score is that $g(v)$ represents how good node v is, and $f(u)$ represents how fair u is. If u is totally fair, then the score of (u,v) should be the goodness of v, $g(v)$. However, if u is unfair, then we cannot really predict which value it will give to the weight so we predict a value closer to zero by adding the factor $f(u)$. As in [5], we call this predicted score the $F \times G$ score of the edge.

A simple baseline [5] compared the $F \times G$ score to existing techniques to predict the signed edge weights. Here we will only consider the simplest baseline to predict a missing signed edge weight: the *reciprocal* method. Given a directed edge (u,v), we just predict the weight of (u,v) with the weight of (v,u) if it exists, or 0 otherwise.
This method performs surprisingly well when the reciprocal edge is available, as we will see in section 5.

4.4 Interaction between price and trust

Our last goal is to find whether there is correlation between price and trust in the network.

To do that, we compare the time series of the bitcoin price and the time series of the mean ratings given in the OTC and Alpha networks.

Results can be found in section 5.3.

Method	OTC	Alpha
Reciprocal	(0.325, 0.461)	(0.271, 0.469)
Goodness (G)	(0.274, 0.644)	(0.237, 0.582)
Fairness x Goodness (FxG)	(**0.275**, **0.653**)	(**0.238**, **0.585**)

Table 2: Leave-One-Out prediction for signed weights. Each cell contains the pair root-mean-square error (RMSE) and Pearson Correlation Coefficient (PCC).

5 Results

5.1 Signed edge weight prediction

We conduct a first experiment following [5] by leaving one edge out of the graph and trying to predict its weight. We report two metrics. First, the root-mean-square error (RMSE), which is the square root of the mean square error between the predictions and the true weights. Second, we report the Pearson Correlation Coefficient (PCC). We note that every experience led to p-values below 10^{-10}.

The results can be found in table 2. We observe that using goodness or fairness and goodness outperforms significantly our baseline, for both the OTC and the Alpha networks. Our results for the reciprocal method match closely those from [5], but our results for methods G and $F \times G$ are better than those reported in this paper.

We can observe that the prediction form method $F \times G$ outperforms method G by a very small margin. This seems to indicate that the goodness of the target node is an essential feature to predict the weight of the signed edge, but the fairness of the source node adds little value. This might be explained by the fact that if we know that the source node is unfair, there is no good way to predict the weight of the edge. The G method assumes that the unfair node will behave like a fair node on this case, while the $F \times G$ method assumes nothing and predicts a weight 0.

5.2 An eye for an eye: are users taking revenge?

We are now interested in testing whether users are impacted by how other people rate them. More precisely, if a user u gives a rating w to user v (so that w is the weight of edge (u, v)), does it impact the rating v will give to u, i.e. the weight of edge (v, u).

To find out, we consider the subset of edges which are reciprocated: all edges (u, v) such that there is also an edge (v, u). We compare the performance of the $F \times G$ method and the *Reciprocal* method. The $F \times G$ method is agnostic to the social interactions between u and v, and only consider the characteristics of node u and v. The *reciprocal* method, on the contrary, only uses social clues and retaliate on a rating by giving the same one.

Method	OTC	Alpha
Fraction of reciprocated edges	0.792	0.832
Reciprocal	(**0.223**, **0.611**)	(0.213, **0.583**)
Goodness (G)	(0.225, 0.473)	(**0.209**, 0.452)
Fairness x Goodness (FxG)	(0.226, 0.474)	(0.210, 0.451)

Table 3: Leave-One-Out prediction for signed weights on **reciprocated edges**. Cells are as in table 2.

The results can be found in table 3. We first observe that a majority of edges are reciprocated in the network. As OTC and Alpha are networks of trust between users exchanging bitcoins, it seems logical that most users rate each other after a transaction, so most edges are reciprocated.

We observe that on these edges, the *reciprocal* method outperforms both G and $F \times G$ methods. We can see that the two methods using fairness and goodness actually a lower Pearson Correlation Coefficient on this subset of the edges. On the contrary, the *reciprocal* method greatly improves its RMSE and PCC on the subset of reciprocated edges. This indicates that when a user u rates another user v that already rated u, the weight of (v, u) matters more than the fairness and goodness of u and v.

Another interesting observation is that the three methods have comparable RMSE, but the PCC of the *reciprocal* method is way higher than the PCC of G and $F \times G$. This indicates that users tend to vote with the same sign as the reciprocal vote. If u rated v with a negative weight, then v will likely rate u with a negative weight, so both votes are highly correlated. Another explanation might be that as interactions result from a bitcoin exchange, the rating from both users will be caused by how well the exchange go and how well users know each other.

5.3 Influence of price

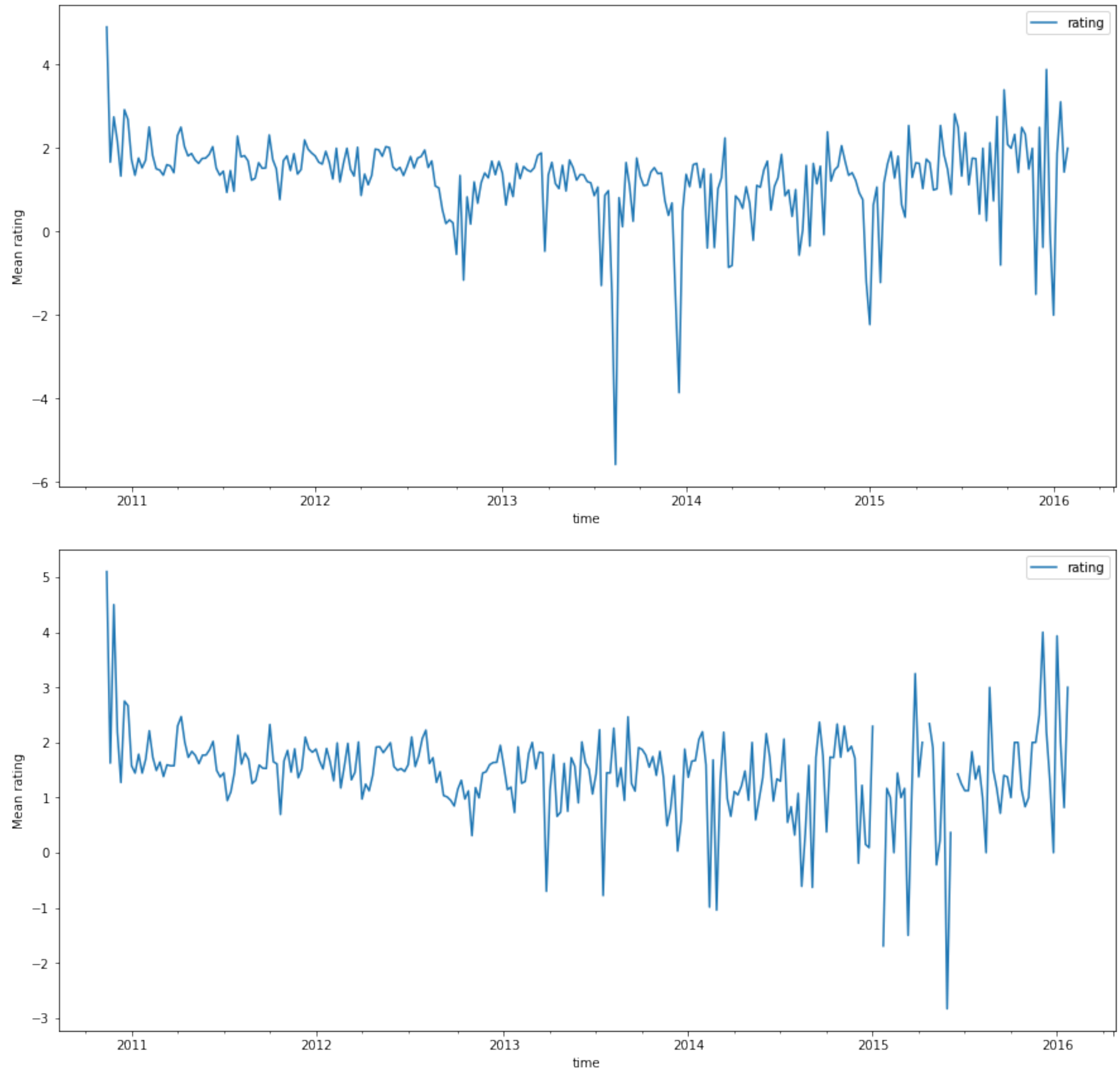

Figure 4: Weekly aggregated ratings over time of the OTC (**top**) and Alpha (**bottom**) networks.

We first plot the evolution of the ratings over time. We group ratings by taking the weekly mean and look at its evolution in figure 4.

Looking at the figures, we can't see any obvious evolution of trust over time. However, it seems like both plots evolve in a similar way so there might be some hidden correlation.

To determine if the two plots are correlated, we compute the pearson correlation coefficient of the two time series after centering and normalizing them. We don't find any correlation between the plots, as the p-value returned is 1.0 which means that random data could have the same correlation.

Next, we try to see if the mean rating is correlated with the price of bitcoin. We normalize the price data and truncate it to match the dates of the OTC and Alpha networks. We compute the Pearson Correlation Coefficient between OTC and price, and between Alpha and price. However for both pairs we find a non significant p-value of 1.0, which means there is no apparent correlation between price and trust.

6 Conclusion

Our paper has reproduced experiments in [5] on fairness and goodness used for predicting edge weights. However, we have found that what matters the most in a trust network is the user-to-user interaction. If user u rates v, the rating that v gave to u is more important than the overall goodness of v.
In the context of marketplaces based on trust, this can be explained by the fact that both ratings depend on an underlying event: the actual trade bitcoin / dollar. Furthermore, on the OTC network users can change their ratings $w(u, v)$ at any time, so there is not any "double-blind review" where users u and v have to rate each other before seeing the ratings of the other (technique used often in marketplaces based on trust like Airbnb).

Our second finding is a negative result on the correlation of price and trust in the bitcoin network. Despite a very volatile price of bitcoin that could make users trust each other less, we don't find any significant correlation between the exchange price of bitcoin and the amount of trust in the OTC and Alpha networks.

References

[1] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651. ACM, 2003.

[2] Y. Katz and J. Golbeck. Social network-based trust in prioritized default logic. 2006.

[3] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[4] D. Kondor, I. Csabai, J. Szüle, M. Pósfai, and G. Vattay. Inferring the interplay between network structure and market effects in bitcoin. *New Journal of Physics*, 16(12):125003, 2014.

[5] S. Kumar, F. Spezzano, V. Subrahmanian, and C. Faloutsos. Edge weight prediction in weighted signed networks. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 221–230. IEEE, 2016.

[6] J. S. Mill. *Principles of political economy.* D. Appleton, 1884.

[7] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system.

[8] D. M. Romero, B. Uzzi, and J. M. Kleinberg. Social networks under stress. *CoRR*, abs/1602.00572, 2016.

[9] G. Wood. Ethereum: A secure decentralised generalised transaction ledger.