The challenge of decentralized marketplaces

Bas van IJzendoorn, 4024850

ABSTRACT

Categories and Subject Descriptors

1 [**A**]: B

; 2 [**C**]: D

Keywords

e-mails, demographic data, languages

1. INTRO

Commodities are traded on decentralized markets (Miao, J., 2005).

http://www.uniba.it/ricerca/dipartimenti/dse/seminari/seminari-2011/Schiraldi-al2011.pdf Rapson, D. (2011) Proof that transaction costs are less in decentralized markets and that

2. PROBLEM DESCRIPTION

Decentralized markets are hard to create. Buyers and sellers need to be matched to each other according to their preferences. A price should be negotiated and a trade deal should be made. The requirements vary among markets. Brunner, E. et al divides the economic requirements into four categories of parameters: basic, composed, complex and comments. Basic and composed parameters are simple values like price, volume and quantity. Composed parameters are more complex economic measurements that needs to be computed from more values like Return of Investment (ROI) and Price-earnings ratio. The last parameters are comments like quality or expert reviews. Policies on how these parameters should be created, altered and read needs to be specified for each market. Other research introduces the concept of contracts between peers called P2P contracts or smart contracts. These contracts allow to transfer user specified amounts against user specified conditions. For instance, ABN AMRO bank uses smart contracts in a case in which it only transfers money after a quality check has been done successfully (BRON). These conditions allow great flexibility in the economic parameters. Namely, all transactions conditions and requirements can be programmed as a smart contract. This allows to maintain money on the Internet without the need of an intermediate party (Fairfield, J., 2014). Brunner, E. *et al* also specifies time sensitive and historic information that should be made public to the user. Also privacy information of the public and private market and personal data of the user are considered parameters by Brunner, E. *et al.*

MEER info over preferences, tot nu toe alleen requirements.

3. SYSTEM MODEL AND ARCHITECTURE

There are a wide range of possibilities to architect the decentral market systems in which a lot of decisions have to be made. The most important problems are:

1) Trust and reputations In order to understand the problems with in computer markets the underlying economic mechanisms have to be studied. In particular the mechanisms that provide new problems when computerized. When actors go online to do business they don't necessarily have a social relation with the persons they are doing business with. The lack of social relations with other actors in the economies creates problems in communication and trust. Communication issues can be solved by providing well enough information about products, vendors, buyers etc. to actors in the markets. However, questions arise as to how much of the information should be made available to actors and to what extent should information be anonymous. For instance, a buyer in grain trading markets might be reluctant in sharing how much grain it wants to buy because this gives valuable information about the trading position of this actor. When other actors know the trading position of the buyer they can play economic games like only selling grain for a higher price to this buyer. What and how information should be presented to users depend on the structure and information demand in each market.

Trust among actors is another problem in computerized markets. In traditional economic theory of a perfect market their is no discussion for trust and the concept is kept outside the domain of economics. In the traditional market anonymous buyers and sellers come together to exchange standardized goods. It is assumed that buyers and sellers try to maximise their welfare. Because of the transparent nature of the perfect market their are no opportunities to be dishonest and so there is a natural trust among buyers and sellers. In recent research the concept of trust has become a part of economic theory and is evaluated in a number of economic theories. A new consensus among economic theorists is growing among economic theorists that emphasises the importance of social relations among economic transactions. In transaction theory literature it is suggested that more trust between actors lowers the transaction costs. Broader social relations among actors lowers the costs of transactions between actors and at the same time minimizes risk from opportunistic behavior in the marketplace. The insight that trust can lower the costs of exchange has pushed the concept of trust in the economic debate.

There is also other research that describe relationships between economic activity and trust that suggest it is hard for computers to generate trust and determine trust of agents. Williamson (1993) distinguishes six types of trust contexts that are important for economic activity: societal trust, political trust, regulatory trust, professional trust, network trust and trust in the corporates themselves. Agents take all these contexts into consideration before making an economic decision. These contexts are largely outside of the digital world an play a large role inside the social world. Other researchers argue that agents who operate economically have a bounded rationality. The number of possibilities to take into consideration before an economic decision is made are simply to large for an agent to rationally process. Therefore not all rules of thumb that an agent follows for economic decision making can be described as a rational process of cost minimization. Because it is hard for an agent to make a calculative rational economic decision it is also hard to calculate whether another agent is trustworthy or not (Furlong, D., 1996). As computers are purely rational decision makers, determining trust is hard for a computer.

However, there are a large number of positive examples where computer systems are trusted for economic activity. In these computer systems are alternative trust mechanisms in place like reputation systems for agents, anonymization systems and brand usage. There are a lot of successful examples of trust build on the internet in business to consumer electronic commerce. For instance: Amazone.com, bol.com and alibaba.com (BRON). Analysis have been done to measure the trust in these online business to consumer marketplaces. The amount of trust plays a central role in the technology acceptance model proposed by Corbitt et al (2003). Trust is solved in the Silk Road and other anonymous markets with vendor repuation systems and anonymization (SILK ROAD PAPERS, MEER UITLEG, DARKNET). In P2P file sharing are reputation systems in place to prevent users from freeriding behavior where users only download and not upload. Each user has a reputation, which in fact is a trust metric to test whether a user will upload data or not. EXAMPLES VAN DERGELIJKE SYSTEMEN GEVEN. Trust is solved in Uber. Airbnb with professional photography (BRON GEVEN).

There are also designs for trust systems for decentralized markets live BEAVER (BRON EN UITGEBREIDERE OMSCHRIJVING), P2P file sharing systems and payment for anonymous routing (BRON EN OMSCHRIJVING GEVEN).

3.1 Trust in P2P filesharing

3.2 Reputation systems

We will go into further detail of these systems. In P2P file sharing research there are a number of systems proposed with systems to prevent free riding. According to Moreton, T. (year) the major problem in P2P systems is the mutual distrust between peers. There are many pseudonyms or Sybil nodes that take up resources without providing resources to the network. These Sybils are run by agents which have a bad trust relationship with the other agents of the network. The behaviour of these agents is in P2P filesharing also denoted as freeriding. The problem was first described by Wilcox O'Hearn after his experiences with the deployment of the Mojo Nation file sharing system. O'Hearn describes as the biggest problem the distrust among nodes. The motivation between nodes to cooperate was not there. Nodes did not upload data to the network which made data availability a problem. There were even attacks on the network by which users altered their clients to gain more advantage for himself.

The main question in free-riding research in P2P file systems research is how to prevent nodes to free-ride and to architect a system that allows nodes to determine the trustworthiness of other nodes in the network. I will discuss some of the system proposals and their relation with decentralized markets. Vishnumurthy, V. (year) introduces a design of a P2P file sharing system that gives incentives to nodes to contribute resources to the global pool in the network. A currency is introduced in where a single value called KARMA represents the amount of resources a peer has contributed and consumed. This represents a users trustworthiness with regard to upload/download ration within the system. There are groups of k nodes called bank-sets that keep track of the KARMA of each user. There are mechanisms in place to make the KARMA system work. There are distributed hash tables (DHT's) that map nodes towards a bank set. When a node goes down, a new node becomes part of the bank set. It is impossible for nodes to adjust their KARMA level at will and KARMA can compensate bank nodes for participating in transactions with KARMA. There are also security mechanisms for replay attacks, malicious providers, malicious consumers, attacks against DHT routing, corrupt bank sets and denial of service attacks. However, KARMA does not protect against sybil attacks.

Tsuen-Wan et al (2003) proposed three solutions to the free-riding problem and to enforce sharing. Two of them are not suitable according to the authors. The third one introduces a method that involves the auditing of peer nodes. Each node maintains a usage file where it defines the amount of capacity it advertises and it also maintains the advertised capacities of all neighbours. A simple rule is added that says that a node can only download new data if its own advertised capacity is larger than the sum of the advertised capacity of all its neighbors. An auditing procedure is introduced that let nodes check on each other whether to tell whether they are trustworthy or not. The economics of the auditing model seems very unlikely to be successful. The required capacity needs to be very high to be able to download data.

What's interesting about the paper is that the concept of an auditing procedure by other peers is introduced. By this way the network maintains its own reputation.

KARMA is an example of a P2P system design that is a combination of a reputation system and a payment protocol. Each node has their own reputation (KARMA) and bank nodes are compensated for their contribution when they participate in transactions. A paper that tries to capture the essence of this combination is the stamp trading model by Moreton. Moreton describes that payment protocols operate using a currency. Nodes receive payments when they complete interactions succesfuly and they can spend the currency tokens elsewhere. The trustworthiness of a node is determined by how much of the token currency is received by a node. In reputation system the trustworthiness of nodes is determined by the "reputation" a node has among other nodes in the network. The reputation of a node is not maintained by the node itself, but arises as other nodes send recommendations about the nodes trust value. In both types of protocols nodes have incentive to contribute to the network. With payment protocols a node can only obtain service if it is able to pay enough tokens to other nodes. In reputation protocols nodes have incentive to higher their reputation in a network to obtain service.

3.3 Payment systems

All of the reputation systems described so far are impractical and are impossible to implement for various reasons. There are other directions in research that focus on payment systems between nodes. Another direction focusses on P2P contracts. Several payments systems to let nodes pay each other are introduced like micro-payments and ppay (Yang et al). These payment systems do not rely on a central broker but let peers pay each other with their own currency. The research on payment systems is used by other researchers such as Ham et al that introduce a credit system where credit is the uploaded bytes minus the downloaded bytes in a system. A higher credit of a peer means a higher trustworthiness of that peer. Ham et al provides solutions to the start-up deadlock problem and starvation. UITZOEKEN WAT DIT IS. The design of the system by Ham et al also tries to provide solutions to different types of cheating by peers such as the blackmailing of peers to each other.

Another proposed payment system is the stamp trading protocol proposed by Moreton. Moreton introduces stamps that can be traded between nodes and can later redeemed at a node for service. In this payment protocol the stamps have a variable value and are traded based on this value. It is assumed there is a centralized exchange rate mechanism which can observe all interactions between node and thus provide perfect valuations to the stamps' value. This assumption has practical issues. In the first place it is hard to observe all interactions between nodes and secondly the centralized exchange rate node has to be trusted fully. If this central nodes gets compromised by an adversary, all interactions can be observed and the whole network is compromised. In the paper multiple price valuation methods are proposed with different properties. The schemes have to be both token-compatible and trust-compatible. A scheme is token-compatible if the total value of the stamps in the network is bounded. A scheme is trust-compatible if failure by a node to redeem a stamp never increases the total value of its stamps. In four of the proposed methods for pricing the system can be flooded with requests by nodes with a higher bandwidth to artificially obtain a higher trust. In the last method called Bounded Redemption Rate (BRR) the value of the stamp is chosen in such a way that flooding the network with stamps causes a node's total stamp value to approach zero value. In this way the BRR method becomes trust-compatible. It is also proven that BRR is also token-compatible. BRR can resist Sybil attacks because when a nodes becomes flooded with requests of pseudonyms, the total stamp value of a node approaches zero. However, stamp trading still has the following open problems: double spending, cryptographically signing stamps, audit trails of stamps, the token exchange problem which is now fixed with the central node assumption and limited knowledge on both the stamp-trading economies and attacks. Thus altough stamp trading is resistent against sybil attacks it has many open problems which makes is impractical to implement in the real world.

3.4 P2P contracts

In an early paper where a contract between two peers is named is the paper by Ghosal et al (2005). The idea of an exchange between two peers based on a single value is questioned. Instead there is an exchange with relation to an amount of service S provided by a peer. The service S a peer can offer is actually a vector that contains different service specifications. For instance, in file sharing S can contain the amount of data that is shared and the available bandwidth for each file. The peers exchange money for a service level that can be specified differently for each type of service and each peer.

In consumer products service is exchanged in an online system where consumers cannot bargain. With smart contracts people may be able to execute trades through Trustless public ledgers (TPLs). TPLs allow a restructering of power relations between parties and intermediaries. TPLs enable parties to store digital assets online without the need of banking intermediary who charges a fee. In addition to that they also allow parties to transfer digital assets directly to each other on their own terms. The conditions of the terms can be programmed in a "smart contract": "an automated program that transfers digital assets within the block-chain upon certain triggering conditions". Smart contracts do not require an institution as an intermediary exchange. Smart contracts also solve the longstanding problem of e-commerce courts to refuse to protect consumer contract terms. With smart contracts consumers can express their own wishes for the contractual terms and negotiate with other parties on their own. The way this is implemented is via automated consumer purchasing agents that can be used throughout the whole web. Smart contracts provide a standard online infrastructure on which consumers and providers can negotiate on their terms. (Fairfield, 2014).

A practical implementation of smart contracts is the Ethereum system (White paper Ethereum). In the Ethereum system money is traded with smart contracts using its own currency: "Ether". The underlying transactions of the smart contracts are done with BlockChain Technology by Satoshi Nakamoto's (2009). BlockChain does not only provide an infrastructure for digital payments, but also provides a distributed consensus for the rightness of the payments and prevents double spending attacks. Ethereum is a fully fledged Turing-complete programming language that can a wide range of financial applications like smart contracts, digital currencies for exchange and also programmable decentralized autonomous organizations (DAOs).

Ghosal, FairField, Ethereum, Recht. PEER CONTRACT MOVEMENT UITWERKEN.

3.5 Implemented examples of trust systems

The P2P research group by Pouwelse, J. has developed Tribler: an open source P2P file sharing system. Tribler is a fully implemented P2P system that operates in the real world and is used for research towards P2P systems. In previous versions of Tribler, BarterCast is the system used to that tracks reputation and therefore the trustworthiness of nodes. BarterCast is an example of a reputation system to generate trust in P2P systems as decribed above. BarterCast is fully decentralized and does not depend on central servers to track reputation. Thus if one node gets compromised by an adversary this does not immediately effect the whole system. With a centralized component, the reputation or trustworthiness of nodes is immediately exposed to an adversary when compromised. However, the main limitation of BarterCast is that it assumes that the majority of the nodes is honest and follows the protocol. Malicious nodes can create reputation records without a limit or punishment (Norberhuis, S., 2015). In a decentralized market situation it cannot be assumed that nodes are always honest. If a node gets a chance to cheat the system in order to make money at the cost of other nodes it will do that. The system needs to be attack resistant against nodes with dishonest intentions. Therefore BarterCast is not a suitable system for the decentralized market.

Because of the limitations of BarterCast, Norberhuis, S. (2015) developed a new method to track reputation in Tribler. His method is a payment system and based on BlockChain Technology. The MultiChain system tracks the amount of uploaded and downloaded data of a node. When the amount is above a certain threshold a payment will be done to add or substract from the wallet of the node. The main difference between MultiChain and Original BlockChain technology is the way the individual blocks are setup and how the transaction history is managed. In MultiChain a transaction history is managed for every peer instead of a full transaction history for each coin. A full and global transaction history will make it hard for the system to scale because all transactions are saved in one chain. With MultiChain the scaling is solved, however the double spending of a coin is not prevented but is punished by nodes in the network. In the MultiChain version proposed by Norberhuis, S. (2015) the punishment is only done by the node that detected the fraud. There is no implementation yet of a global fraud announcement system to inform all other nodes a certain node is not trustworthy. Such an implementation is for future work.

Another system that is implemented in Tribler that deals with the trustworthiness of peers is the decentralized credit mining system by Capota et al (2015). The system aims to earn trustworthiness of peers in other swarms. In the paper by Capota et al (2015) this is described as earning credit in other swarms on behalf of the user. The system is part of the Tribler P2P client and is implemented for every peer and therefore completely decentralized. The system selects swarms on its upload potential and start to upload data to these swarms. In this way the peer gains trust in that swarm. Information is frequently updated to maximise upload to swarms and there are also spam detection and duplicate content detection to further enhance the upload process. The system is also tested to show that trust is gained in other swarms with the system. The underlying mechanism to gain trust in the paper is simple. The peers simply behave cooperativly by uploading data to proof that they are not free riders and thus to proof their trustworthiness.

There are also decentralized markets in development using the Tribler system. The first example is Tsukiji, a first implementation by The,M. and Reinbergen, H. (2013). It is a simple implementation where decentralized nodes act as traders. The traders can place bid and ask offers and respond to an offer such that a trade can be established. The discovery of peers is also implemented but there is no real money traded and there also isn't a working user interface.

An improvement on the design of Tsukiji is the Decentral market design by Olsthoorn, M.J.G. and Winter, J. (2016). Instead of peer discovery bid and ask prices together with quantities are distributed across the network with ticks when a peer bids or asks a certain quantity. Secondly, there is a simple matching engine implemented that matches bid and ask quantity amounts with the highest and lowest prices. Then when a match is made real money is traded. MultiChain coins of Tribler peers are traded against BitCoins in a single transaction where both wallets of both traders are updated. The design is successfully implemented in Tribler, constructed with Dispersy and tested.

4. THE SYBIL ATTACK ON TRUST

Focus on Sybil attacks with personalized hitting time. Pim Otte research. There appears to be no system that is up against Sybil attacks. So far, this is proven to be the hardest problem to solve in P2P networks.

4.1 Trust with anonymity

TRUST in electronic commerce (Ratnashingham, 1999).

SCREENSHOTS MAKEN VAN VERSCHILLENDE TYPEN MARKTEN.

https://www.ids.ac.uk/files/Wp35.pdf

https://pdfs.semanticscholar.org/d490/3a683c7b60a27a0c19c28d0a77
http://www.emeraldinsight.com/doi/pdfplus/10.1108/1066224981023
Trust in electronic commerce.

ONDERZOEK DOEN NAAR PROBLEMS IN ELECTRONIC MARKETS.

Importance of trust in electronic commerce: http://www.emeraldinsig
Importance of perceived trust, security and privacy in online trading systems. https://www.researchgate.net/profile/Juan$_{G}arcia95$
$shoppinghttp : //download.springer.com/static/pdf/565/art$

http://dspace.unive.it/bitstream/handle/10579/7203/830275-1190055.pdf?sequence=2 2) Market structure Impact on market according to Bichler with broker services. However, time has proven that the market still requires the broker. Example van Olsthorn et al, just buy out the bid prices.

In economic theory the market structures are elaborated around the research to "two-sided markets".

As markets can obtain a variety of characteristics it is important to notice that for each market a different market mechanism is required. To reason easier about markets the following concepts are described in the paper by Hatfield and Kominers for market mechanism design. 1) Stability: There is no blocking pair for a match. A blocking pair is a match with a higher utility function than the original match. e.a. the blocking pair match is a better match than the original match. Thus a stable match is the best match available. If a match is stable this implies a future match offer will never be better (Niederle, Yariv, 2008, Gale and Shapley, 1962). Gale and Shapley (1962) showed that any market has a stable matching and provided an algorithm that identifies one in the deferred acceptance algorithm. 2) Strategy-Proofness: When a matching mechanism is implemented there might be strategies that disrupt the market. For instance, a person might BETER OPZOEKEN in two sided matching literature (Niederle, Yariv, 2008). Roth and Sotomayer have an example of a market where agents have an incentive to misstate its preferences even tough the optimal match is chosen by the implemented mechanism. 3) Substitutability: The definition of substitutability is as follows. Lets assume two group of agents G and H that are matched. An agent $a \in G$ chooses $b \in H$ as its optimal match. If b is also chosen as the optimal match from $H' \cup w$ where subset $H' \subset H$ than the preferences of a are substitutable. When b is chosen from a set, it is also chosen from a smaller set. (Echenique, F, Oviedo, J., 2006). SO a CAN ALSO CHOOSE ANOTHER WORKER. http://people.hss.caltech.edu/ fede/published/echen-oviedo-TE.pdf STRONG SUBSTITUTABILITY OOK NOG ERBIJ DOEN. 4) The Law of Aggregate demand: (Condition) If the choice set of contracts for an agent increases, the agent chooses a bit more contracts.

A contract language is developed to describe the effects of varying contract language on stability and substitutability.

Verschillen tussen many-to-many and many to one markets.

CONCLUSIONS VERY USEFUL OF PAPER.

3) Matching engine The matching engine needs to be strategy proof. No obvious strategies to fool people should be in the market. (GIVE EXAMPLE OF POSSIBLE STRATEGIES). Olsthorn counterexampelen. TOR anonymity can be used as a tool to provide a better matching engine. A manual matching is also an option.

The markets are called matching markets. We have many-to-many markets. Meaning that they have substitutable contracts. Strategy Proofness in Harvard Paper.

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.359.3617rep=rep1type=pdf

Contract design and stability in markets (Harvard, Hatfield, 2011).

4) Price discovery mechanism Is fixed in matching engine. According to Bichler, M. dynamic pricing mechanisms can be implemented such that market prices match the market conditions and therefore creating an optimal outcome for both buyer and seller. In physical markets, the high transaction costs of auctions have made it impossible to implement these price mechanisms. With information technology it might be possible to implement auctions and change the way how the markets are operated. Ebay has already proven itself to be successful in online auctions. An example of an auction is where buyers send their bid prices to suppliers. The suppliers can then accept the bid prices as a contract. Electronic exchanges can focus on the buyer side or the seller side. The actor that has the least market power usually takes the initiative. There are also auction techniques on which over multiple attributes of the contract are negotiated to allow complex products (Bichler, 2001). In other markets there is also a need for dynamic pricing models. There is research done in multiple markets to find suitable price discovery mechanisms that suits each market. For instance, in the cloud computing market Anandasivam, A. and Prem, M. (2009) introduce a dynamic pricing model for price determination in the cloud computing market In cloud computing systems, sometimes the demand is high and sometimes the demand is low. The price is changed when the demand level changes. This price change is calculated in a mathematical model. Another example of the need for a dynamic pricing mechanism is in modern electric power grids. ELECTRONIC POWER GRID UITWERKEN.

Methods: Auction from Bichler, Auction from Lee,

Various possibilities on matching engine and price discovery mechanism

http://link.springer.com/article/10.1007/s12599-009-0071-2/fulltext.h
Current cloud computing solutions lack pricing mechanisms, but there are movements to bring this into the business world (Weinhardt, C.)

https://pdfs.semanticscholar.org/85e2/69c8b6a9d791424e16747a6d39
Auction as a dynamic price mechanism in e-commerce (Lee, J.)

https://books.google.nl/books?hl=nllr=id=-lhLmmSM–4Coi=fndpg=
Book on matching (Bichler, M.)
file:///C:/Users/Lenovo/Pictures/wilson-market-architecture.pdf
Economisch paper over markets (Wilson, R.) http://www.emeraldinsig
Importance of trust in economic commerce (Pauline Ratnasingham) http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.5
Commodity trading using an auction (Preist, C.). http://people.bu.edu
Search model centralized and decentralized trade (Miao, J.). (Matching engine)

http://scholarlycommons.law.wlu.edu/wlulr-online/vol71/iss2/3/?utm
$scholarlycommons.law.wlu.eduSmartcontracts(Fairfield)$

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4536461
Requirements and architecture decentralized information system (Brenner)

http://www.sciencedirect.com/science/article/pii/S002205318471074X Equilibrium mechanisms in decentralized market (Peters, M.)

ToDo:

Solutions: SOA, Blockchain, microservices.

4) Sybil attack resilience

5. REFERENCES