First Year Review

Orfeas Stefanos Thyfronitis Litos

University of Edinburgh
o.thyfronitis@ed.ac.uk

Primary Supervisor: Prof. Aggelos Kiayias
Secondary Supervisor: Dr. Kousha Etessami

Abstract. This research proposal concerns the study of economic trust in the decentralised, asynchronous setting and its applications on online commerce and blockchains. In particular, the aim of this project twofold. The first goal is to understand whether it is possible to achieve robust online commerce with minimal risk of fraud without trusted third parties and if so at what cost. The second goal is to provide a general model of off-chain payments and explain how a relaxation of the trustless element of current solutions can facilitate the creation of a scalable blockchain. Tools from the fields of Cryptography and Game Theory will be used to achieve the aforementioned targets.

1 Research Topic

The advent of bitcoin [1] has promised to revolutionise the way online commerce takes place, only to meet multiple hurdles in the way. Today there exists an entire ecosystem of cryptocurrencies, e.g. [2,3,4], each with is own benefits and drawbacks. Nevertheless, there are several fundamental questions that still remain unanswered. Some of the prominent issues are whether cryptocurrencies can become the global unit of account, whether it is possible to regulate and fairly distribute them, what are the exact benefits they provide when compared to fiat currencies, in which respects the latter outperform them and if it is possible to completely replace fiat currencies with decentralised cryptocurrencies for which the traditional measures of law enforcement are less applicable due to the global nature of blockchains.

This research proposal is crucially motivated by the following observation: The single most advertised advantage of permissionless blockchains is the lack of need for trust between the interacting individuals *and* for a mediating trusted third party [1]. Unfortunately, this benefit is limited to a certain type of digital assets that admit a cryptographically secure proof of ownership and can be provably transferred to another party. Additionally, the overhead for setting such a system in motion is too big for the entire economy to be realised in this way.

1.1 Decentralised E-Commerce

Going in more detail, blockchains provide us with novel capabilities. It is possible to immutably and truthfully record the fact that Alice paid Bob some cryptographic coins on a blockchain; it is possible for Bob to provably and irrefutably give her a digital asset in return (e.g. an ICO [5], or a sword in a future blockchain–backed game) and record the fact on a blockchain as well; additionally, it is possible to make the exchange atomic so that no trust between the two parties is needed. Unfortunately, all the above are impossible to accomplish if the asset is physical. Alice may pay first and then Bob can refuse to give the asset; or Bob may indeed hand in the asset, but Alice may then claim that it was not as advertised and demand her money back. The possibility of fraud is exacerbated if the exhange takes place online between parties who do not know each other personally. Crucially, an external observer cannot always resolutely decide which of the parties is right. Even if she can, it seems impossible for her to prove that she is not colluding with either player and that she is completely fair and impartial.

For comparison, if the payment is done in fiat currency, then there exists a mediator (usually a bank, often a court) who is charged with deciding what is the fair way to settle the dispute. Additionally, it is in the mediator's direct benefit to strive for the objectively fairest solution. Obviously the cryptographic security of this state of affairs is very low: it is entirely possible that the bank employee or the judge who has the final say is a friend of the fraudster. Nevertheless, it seems that currently the incentives are much better aligned when relying on the traditional legal system. As a result, transacting in fiat currency is, for practical and everyday purchases of physical goods and especially in the online setting, safer and more robust than using cryptocurrencies.

The cryptographic achievements of blockchains are currently realised only in a closed system, where everything can be proven mathematically. Nevertheless, the economy is not such a system. In order for decentralised blockchains to substitute the traditional fiat currencies, a mechanism for decentralised e-commerce with guarantees of minimal trust is needed. We propose to attempt the creation of such a mechanism or give plausible arguments why such a mechanism is unattainable. In case we manage to create such a system, it should ideally be able to constitute the backend of a decentralised ecommerce platform, similar in usage to existing platforms such as OpenBazaar [6].

In essence, such a mechanism would provide a decentralised reputation system. In current online marketplaces (e.g. ebay.com) there exists a central authority that aggregates ratings and reviews from past interactions between users and serves them to participants of future transactions with the aim of minimising fraud. We aspire to replace this central authority with a distributed protocol or prove that such an endeavour is impossible.

1.2 Off–chain payment channels

One additional impediment to the more widespread use of decentralised blockchains is the issue of scalability. Decentralised consensus protocols based on Proof-of-Work [7] or Proof-of-Stake [4] make the assumption that the majority of the hashing power or of the stake respectively is controlled by honest parties, each of which processes every single transaction. To oversimplify, one can think that such protocols function only as long as every member of a crucial portion of their users processes all transactions. A great duplication of effort takes place in order to avoid trusting a particular set of users. It seems intuitively obvious that, no matter how much the parameters are tuned, such systems can only handle a limited amount of transactions per second and thus cannot compete traditional centralised solutions such as VISA in their current state [8].

One solution to this issue is to ensure that most payments happen off–chain. A variety of mechanisms (discussed in more detail in Section 5) enable cryptocurrency owners to temporarily lock some funds in specially crafted channels with one on–chain transaction and subsequently perform any number of transactions between them with these funds without touching the blockchain. One more on–chain transaction unlocks the funds, ensuring each participant receives exactly the funds they owned in the latest state of the channel. Surprisingly, no trust is needed between the parties. The established term for this type of mechanism is “payment channels”.

As of today there exists no definitive systematisation of what payment channels can achieve and at what cost. Each proposal provides different benefits and incurs different costs; the language and formalism of each proposal is largely incompatible with the rest. Most of the proposals lack formal security definitions and proofs. More importantly, it is unclear what are the exact inherent tradeoffs and whether an ideal balance exists. One of the targets of this research is to provide a suitable conceptual framework that encompasses the whole range of possibilities and limitations of such channels.

Two related issues that have not yet been tackled are the following. Firstly, currently all proposals offer trustless operation as a feature. Nevertheless, it seems plausible that a relaxation of this constraint may indeed allow for more efficient, faster and lighter payment channels. This is a new direction that this research will attempt to explore, ideally incorporating it in the general framework proposed above.

Lastly, current blockchains do not focus on how to facilitate payment channels, instead attempt to obviate their need. An interesting topic would be to provide a treatment on what properties should a blockchain have to aid the creation of lightweight payment channels.

2 Toolset and Strategy

2.1 Simulation–based Security

The two distinct research directions will both leverage tools and methods used in Cryptoghraphy and Game Theory. In order to formally model the interactions between marketplace participants, we will employ simulation–based security [9] and more specifically the Universally Composable Security framework [10]. The general strategy of simulation–based security is as follows: First we give some security definitions that formalise the desired properties our system should have. Subsequently we give an ideal–world functionality that satisfies these properties. This functionality acts as a centralised trusted party that executes the desired operation on behalf of the participants. Afterwards we describe a protocol that hopefully realises this functionality. Finally we prove that the proposed protocol indeed realises the functionality by showing that an external observer cannot distinguish an execution of the functionality from an execution of the actual protocol.

To make matters worse (i.e. closer to the real-world challenge the cryptographic protocol faces), the concept of the Adversary is introduced. This party is in control of all messages that the honest participants exchange and can change, delay or completely block them at will. A complete security proof of the fact that a given protocol realises a particular functionality states that for every efficient Adversary (polynomial algorithm) there exists a Simulator (polynomial algorithm) such that the distributions of the execution of the functionality as viewed from the perspective of each participant are statistically close to the respective distributions of the execution of the protocol.

The Universally Composable Security framework is a particular way of setting up and proving the security of a protocol. Achieving simulation of a protocol in the Universal Composition setting ensures that the protocol can be executed concurrently with other protocols that are secure in the Universal Composition setting. Consequently, managing to prove the security of our protocol in this setting ensures that it can coexist with other secure protocols in the same processing unit without suffering diminished security itself or damaging the security of the rest of the aforementioned protocols. Such an achievement would ensure that our protocol is more readily implementable whilst maintaining a high standard of security and would obviate the need of further security proofs for the case when it is executed in an interleaved fashion with other secure protocols.

2.2 Game Theoretic approach

Before the advent of blockchains, most cryptographic protocols solved problems where there were clear-cut lines between the aims of honest parties and the machinations of the Adversary. For example, in the case of public key encryption [11], the honest parties should be able to exchange messages encrypted in a fashion that only those who know the private key can decrypt, even though the adversary has the ability to employ any method conceivable (within some computational limits) to read the original contents of the message. Any protocol that claims to realise secure encryption should satisfy (the formal version of) this requirement. It is intuitively obvious that it is in the benefit of the honest parties to follow such a protocol.

On the other hand, blockchains made the state of affairs much more complex. Indeed, bitcoin was a concrete solution to an ill-defined problem; several years passed before a formal definition of the problem blockchains solve was formulated [12]. Nevertheless, this definition does not necessarily coincide with the desires of the participants to the protocol. The definitions demand that a blockchain realises an immutable ledger that is inexorably extended, however the participants' goals span a great variety, such as maximising their share of coins or preventing certain transactions from entering the blockchain. In this case, assuming that there exist only honest parties that execute the protocol exactly as described and an Adversary that employs every conceivable technique in order to break the security definitions is an oversimplification. It ignores slight but crucial deviations from the protocol (e.g. [13]) that may provide advantages not considered in the security definitions. Therefore the assumption of an honest majority can be contested.

When considering decentralised e-commerce, the problem is even more exacerbated. Formulating a simple definition for the characteristics of a healthy marketplace is highly non-trivial. The individual desires of parties should employ a central role in the analysis and not be necessarily restricted to a clear-cut protocol. Assuming that a portion of honest participants exists may be wrong, especially if other participants follow counter-strategies that make honest behaviour unprofitable or if honest players have incentive to profitably deviate from their strategy. Conversely, permitting arbitrary actions to the Adversary could complicate the analysis more than necessary, since it may be reasonable to assume that certain "unreasonable" strategies will never be followed by the Adversary.

This is where rational analysis and mechanism design [14] come into play. Game Theory [15] provides a series of tools and concepts that greatly support the study of multi-agent rational systems. The concept of Nash Equilibrium [16], fundamental to the study of Game Theory, will be used throughout this research. Indeed, our aim will be to describe a mechanism that resembles the real e-commerce setting as closely as possible—while abstracting away irrelevant complications—and find strategies that constitute Nash Equilibria.

Fortunately, we can leverage an arsenal of cryptographic tools to help us rule out some categories of undesired behaviours and thus design a system that achieves the goals of modern e-commerce whilst minimising the trust towards third parties. The guarantees given will not be necessarily cryptographic, but rather rational; that is, participants will refrain from acting badly not because they do not have the ability, but rather because it will be detrimental to their social standing, access to goods and services or other desirable attributes.

3 Progress to date

3.1 Modelling Decentralised E-Commerce

The main aim of this research is to construct a model that closely follows the dynamics of a marketplace. Following the Universal Composability framework, we define n interactive Turing Machines (ITMs) that represent the players that take part in the marketplace game. Their description is Π_{SAT}, the satisfaction protocol. These ITMs comprise the set $\mathcal{P}$. Furthermore, we define an ITM named Environment and represented by $\mathcal{E}$.

The latter ITM is the first to start functioning. Initially, its responsibility is to allocate a utility function to each player, drawn from a distribution on legal utility functions that is common knowledge [17] to the players. We will elaborate on the nature of the utility functions later on. Additionally, $\mathcal{E}$ provides players with an initial "endowment" of assets and money. Contrary to the utility functions, these are not sent only to the players, but additionally to the $\mathcal{G}_{\text{Assets}}$ and $\mathcal{G}_{\text{Ledger}}$ global functionalities respectively. These two functionalities exist with the sole purpose of keeping track of physical and digital assets respectively. We will later provide further explanation as to why we introduce these global functionalities.

After providing the utility functions and the endowments, the main part of the game begins. $\mathcal{E}$ can send a message to Alice $\in \mathcal{P}$ that contains a desire d to be satisfied. To satisfy this desire, Alice has to buy an asset that satisfies d from some other player. The first and most important task Alice faces is who to trade with. For that purpose, she consults with the functionality $\mathcal{F}_{\text{Trust}}$, which hopefully responds with a trustworthy vendor, say Bob. Then Alice asks Bob whether he is willing to satisfy her desire with a satisfying asset and, if so, at what price. Given that Bob offers a reasonable price, Alice instructs the functionality $\mathcal{F}_{\text{Trade}}$ to pay Bob the designated price in exchange for the satisfying asset. The functionality pays Bob and crucially asks him if he wants to complete the exchange honestly or cheat. It then completes the trade as instructed by Bob by interacting with the two global functionalities appropriately and reports back to Alice the result. Finally Alice reports the result to $\mathcal{E}$ and $\mathcal{F}_{\text{Trust}}$. The latter updates the trust properties connected to Bob depending on Alice's report. Just like Bob could cheat and not deliver the satisfying asset, Alice has the ability to misreport her experience with Bob.

The current state of this part of the research can be found in https://github.com/OrfeasLitos/UC-Trust/blob/master/what_is_trust.pdf.

We will now go into further detail regarding the several parts of the construction.

$\mathcal{G}_{\textbf{Assets}}$

This functionality keeps track of which player owns which physical assets. Only $\mathcal{E}$ can add assets to it. Alice can remove assets she owns. $\mathcal{F}_{Trade}$ can transfer an asset that belongs to Alice to another player on behalf of Alice. Both Alice and $\mathcal{E}$ (and no one else) can query the quantity of a certain asset that Alice owns.

Observe that the last rule makes it impossible for players to definitively tell whether another player owns a particular asset by just querying the functionality. This is useful, as it reflects the real-world situation where an arbiter cannot decide who of two disagreeing players is entitled to a physical asset without additional information by third parties.

$\mathcal{G}_{\text{Ledger}}$

This functionality keeps track of the digital assets, i.e. coins. For the purposes of this work, all interactions are similar to those of $\mathcal{F}_{\text{Assets}}$, except for the fact that Alice can prove ownership of some funds to an arbiter. This reflects the fact that, through the use of blockchains, Alice can prove that she owns a particular key that has the ability to spend a particular amount of coins.

This difference is fundamental. Blockchains allow for consensus on who owns which coins because the whole history is distributed and immutable. This is not the case for physical assets. Conflicting accounts of past events can happen in such a way that no third party can tell which version is true; crucial events may be hidden (on purpose or because nobody observed them) and consensus may be unreachable. Furthermore, it seems impossible to record the state of the entire physical world on the blockchain without trusting a third party. It is our intent to capture this tension between digital and physical assets and thus we employ two separate global functionalities.

$\mathcal{F}_{\text{Trade}}$

This functionality attempts to abstract away the details of how a trade takes place once Alice has decided to buy from Bob. It checks that Alice indeed has the necessary coins and pays Bob. Then it asks Bob whether to proceed with transferring the agreed asset or cheat and acts accordingly.

This functionality does the “plumbing” of the system and is not central to the design. We have chosen to keep this functionality simple, so it does not make any important decisions; it simply executes a trade according to the wishes of the involved parties. We have provided a protocol $\mathit{\Pi}_{\text{Trade}}$ that realises the functionality. We have not yet proved that the latter indeed realises the former, but we will certainly do so in the future.

$\mathcal{F}_{\text{Trust}}$

This last functionality is probably the most crucial — it is certainly the main missing component in order to model a functional, secure marketplace. $\mathcal{F}_{\text{Trust}}$ is responsible for choosing a suitable vendor to satisfy a particular desire. More precisely, it receives a desire and a list of possible vendors from Alice and returns a single vendor that is the most trustworthy for Alice to buy from.

There are several important things to note here. First of all, in contrast to a real trusted third party (e.g. ebay), this functionality cannot be “held accountable” if it recommends a bad vendor. No legal action can be taken against it. Contrary to a real-world enterprise, it does not have any incentives, it just follows its specification; the mechanism it uses to choose suitable vendors is common knowledge to all players, thus security through obscurity (as employed by real-world enterprises that closely guard their reputation system mechanism) is out of the question. An advantage that this functionality has is that we can choose to provide it with oracle access to the utilities of all players; this is however a luxury that is necessarily absent from any protocol that will attempt to realise $\mathcal{F}_{\text{Trust}}$. Last but not least, in the real world players have the option to employ a variety of fundamentally different mechanisms in order to choose a suitable vendor, with some being word of mouth, search engines, rating websites and the reputation systems of online marketplaces. $\mathcal{F}_{\text{Trust}}$ should ideally encompass the possibilities of all these methods and more. Only then will we be able to safely assume that all coalitions of players will have incentives to keep using $\mathcal{F}_{\text{Trust}}$, instead of employing a different method.

As of now, we have built an $\mathcal{F}_{\text{Trust}}$ that, having oracle access to the utilities of all players, chooses a vendor in such a way that:

- The vendor finds it beneficiary not to cheat.
- Amongst all possible vendors that conform to the previous constraint, the additional utility the buyer will enjoy after the trade is maximised.

Utility functions properties

There is no game theoretic analysis without utility functions. In our case, each player can act upon receiving a message only if she has been given a utility function by $\mathcal{E}$. Any such utility function is defined for every moment in time, every possible asset ownership and every amount of coins owned:

$$u_{\text{Alice}} : \text{Time} \times \text{Assets} \times \text{Money} \rightarrow \mathbb{R} \ \ .$$

Some properties that we may assume for the utility functions if needed are quasi-concavity, continuity and non-satiation [18], at least regarding a restriction of the utility function for a particular moment in time.

The ideal market functionality: $\mathcal{F}_{\text{SAT}}$

Following the simulation-based security paradigm and in order to help us conceptualise how an ideal market should function, we have additionally designed a satisfaction functionality $\mathcal{F}_{\mathrm{SAT}}$. In the world where this functionality exists, the protocol players are dummy, i.e. they simply forward all messages they receive from $\mathcal{E}$ to $\mathcal{F}_{\mathrm{SAT}}$ and vice versa. Therefore, $\mathcal{F}_{\mathrm{SAT}}$ knows the utilities of all players and can complete optimal trades without ever having to show trust to other actors.

More precisely, the current version of $\mathcal{F}_{\mathrm{SAT}}$ behaves as follows: upon receiving a desire from Alice, it creates a list of vendors, assets and prices that constitute possible trades, each of which would be beneficial both to Alice and the implicated vendor. It then sends this list to the Adversary $\mathcal{A}$; the Adversary sends back the trade that will take place and $\mathcal{F}_{\mathrm{SAT}}$ executes it. If the Adversary does not respond with a valid message, then $\mathcal{F}_{\mathrm{SAT}}$ chooses the best trade for Alice and executes that. Note that this is the first time we have implicated the Adversary in any way.

Putting it all together

Our current approach is as follows: In the real world, each player may follow any protocol. At an arbitrary moment in time $\mathcal{E}$ stops the game and evaluates the utility functions of all the players. This way we do not restrict the players to optimization protocols. We will attempt to find specific protocols that achieve desirable properties. More specifically, our main concern is to find a protocol that realises $\mathcal{F}_{\mathcal{SAT}}$. It may make sense to leverage previous work [19] to argue on the extent to which such a protocol constitutes a Nash Equilibrium, but the correct approach in this respect is not yet clear. We will further discuss future work in Section 4.

Results

Let $\mathcal{F}'_{\mathrm{Trust}}$ be an alternative trust functionality that simply chooses a vendor at random. We have proved that there exists an Environment that can provide suitable utility functions and initiate a trade such that it always distinguishes whether it is interacting with the real-world protocol $\mathit{\Pi}_{\mathrm{SAT}}$ or with the ideal-world functionality $\mathcal{F}_{\mathrm{SAT}}$. This proof confirmed our intuition that the trust functionality is of crucial importance if we wish to realise $\mathcal{F}_{\mathrm{SAT}}$.

3.2 Research on Payment Channels

In this part of the research we have mainly focused on understanding the general framework within which the various realisations of payment channels can be classified, compared and evaluated. To achieve this, we reviewed the relevant literature, which has rapidly grown to be quite extensive and contains both concrete systems [**?**] and attempts for the formalisation of payment channels [20]. Section 5 provides a summary of the field.

The inquiry into the nature of payment channels amounted to the following (possibly incomplete) list of characteristics any payment channel has:

–

4 Plan

– Missing pieces:
  - What does the Adversary do?
  - What are the desired security properties?
  - How can F__Trust be substituted by Trust is Risk? Is a wrapper around Trust is Risk needed?
  - What are the correct security definitions?

– Future expansions:
  - arbitrary digital assets
  - connect with real G__Ledger
  - decide whether and how to realize G__Assets
  - allow dishonest seller to send alternative asset as well

5 Literature Review

References

1. Nakamoto S.: Bitcoin: A Peer-to-Peer Electronic Cash System (2008)
2. Wood G.: Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper, 151, 1-32 (2014)
3. Zhong M.: A faster single-term divisible electronic cash: ZCash. Electronic Commerce Research and Applications: vol. 3–4(1), pp. 331–338 (2002)
4. Kiayias A., Russell A., David B., Oliynykov R.: Ouroboros: A provably secure proof-of-stake blockchain protocol. pp. 357–388: Springer, Cham: Annual International Cryptology Conference (2017)
5. block.one: EOS.IO Technical White Paper v2. https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md (2018)
6. OpenBazaar.org. https://openbazaar.org
7. Back A.: Hashcash - A Denial of Service Counter-Measure. http://www.hashcash.org/papers/hashcash.pdf (2002)
8. Vermeulen J.: Bitcoin and Ethereum vs Visa and PayPal âĂŞ Transactions per second. https://mybroadband.co.za/news/banking/206742-bitcoin-and-ethereum-vs-visa-and-paypal-transactions-per-second.html (2017)

9. Lindell Y.: How To Simulate It – A Tutorial on the Simulation Proof Technique. Tutorials on the Foundations of Cryptography: pp. 277–346 (2017)
10. Canetti R.: Universally composable security: A new paradigm for cryptographic protocols. In Foundations of Computer Science: pp. 136–145: IEEE (2001)
11. Diffie W., Hellman M.: New Directions in Cryptography. IEEE transactions on Information Theory: vol. 22(6), pp. 644–654 (1976)
12. Garay J., Kiayias A., Leonardos N.: The Bitcoin Backbone Protocol: Analysis and Applications. In Annual International Conference on the Theory and Applications of Cryptographic Techniques: pp. 281–310: Springer (2015)
13. Eyal I., Sirer E. G.: Majority is not enough: Bitcoin mining is vulnerable. In International conference on Financial Cryptography and Data Security: pp. 436–454: Springer (2014)
14. Nisan N., Roughgarden T., Tardos E., Vazirani V. V.: Algorithmic game theory: vol. 1. Cambridge University Press Cambridge (2007)
15. Leyton-Brown K., Shoham Y.: Essentials of game theory: A concise multidisciplinary introduction. Synthesis Lectures on Artificial Intelligence and Machine Learning: vol. 2(1), pp. 1–88 (2008)
16. Daskalakis C., Goldberg P. W., Papadimitriou C. H.: The complexity of computing a Nash equilibrium. SIAM Journal on Computing: vol. 1(39), pp. 195–259 (2009)
17. Fagin R., Halpern J. Y., Moses Y., Vardi M.: Reasoning about knowledge. MIT Press (2004)
18. Arrow K. J., Debreu G.: Existence of an equilibrium for a competitive economy. Econometrica: Journal of the Econometric Society: pp. 265–290 (1954)
19. Garay J., Katz J., Maurer U., Tackmann B., Zikas V.: Rational protocol design: Cryptography against incentive-driven adversaries. In Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on: pp. 648–657: IEEE (2013)
20. Dziembowski S., Faust S., Hostáková K.: Foundations of State Channel Networks. https://eprint.iacr.org/2018/320 (2018)