University of Piraeus
School of Finance and Statistics
Department of Banking and Financial Management
MSc. in Banking and Financial Management

«Auctions using blockchain technology»

MINASIAN ARTOUR
MXPH1911
a.minasiann@gmail.com

Supervisor: Assistant Professor Dimitris Voliotis
Evaluation Committee: Professor Christodoulos Stefanadis
Assistant Professor Nikolaos Englezos
Assistant Professor Dimitris Voliotis

PIRAEUS
FEBRUARY 2021

To Helen and Alexia
with Love.

Acknowledgments

"I would like to warmly thank my supervisor assistant professor Dimitris Voliotis who, despite the difficulties of these days, was always by my side throughout the course of my dissertation. His exceptional ethos and his great academic experience were a key factor in the implementation of this work.

I would also like to thank my undergraduate professor Veni Arakelian who was always there to help and advise me by guiding me in the right direction.

I would also like to thank my friends Giannis Banusis and Nikos Delis for their support, without them it would not be the same.

Finally, I would like to thank my family who is always by my side and constantly supports me to achieve all my goals."

Abstract

Auctions are especially useful in cases where there are no organized and standardized markets. They have a great application in commerce but also in finance. The usual way of conducting them involves an intermediary who must be fully trusted by all participants. The involvement of a third party in the auction process creates inefficiency for the other participants with high costs and often a lack of transparency and flexibility. For these reasons, we are considering the use of blockchain technology and its fundamental features in the auction process. After analyzing the technology and the existing relevant literature, we came to the conclusion that a properly designed blockchain can cover many of the existing problems of centralized auctions. In addition, we present a Double Auction (DA) implemented in a blockchain network, while in our effort to go one step further, we propose, based on the foundations we had laid earlier, a complex combinatorial auction (CA) for which we designed our own Generalized Algorithm (GCAS) to solve the problem of item allocation (General Combinatorial Auction Solver).

Keywords: #Auctions #Blockchain #Combinatorial_Auctions #Double_Auctions #GCAS

Table of Contents

Introduction

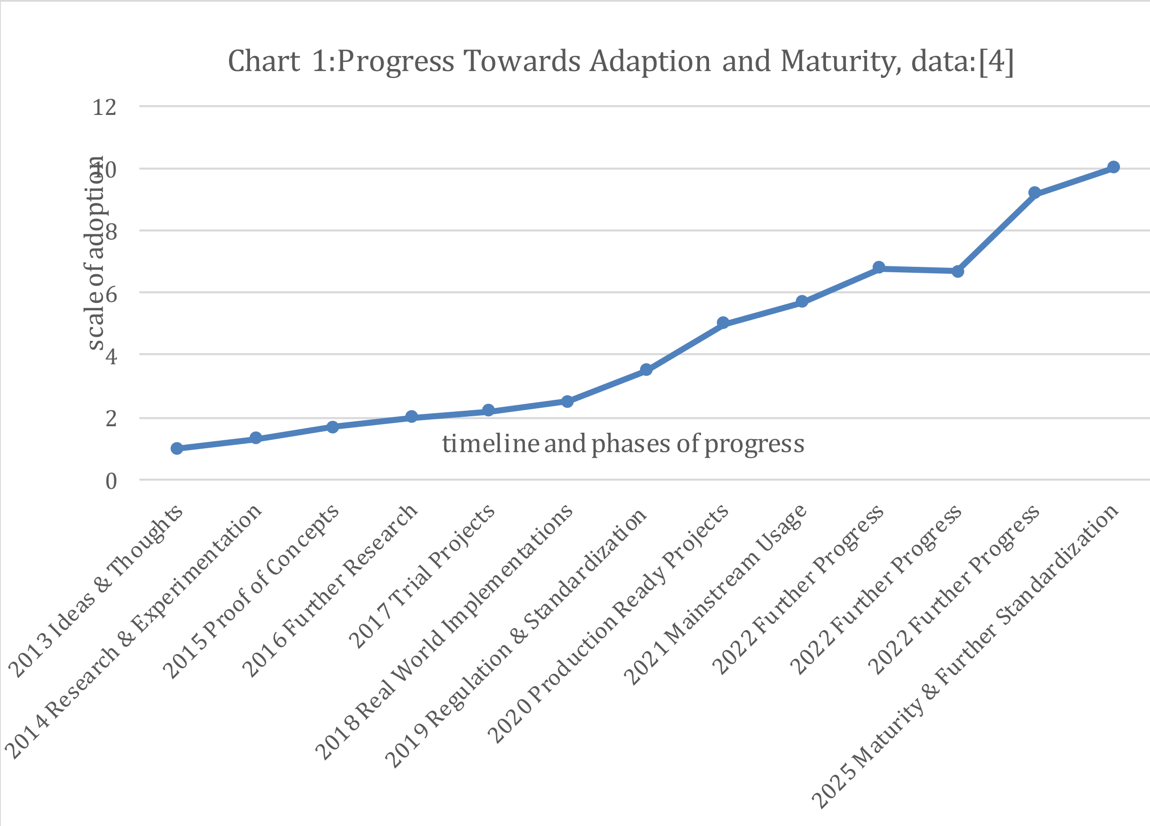
In the last decade the internet and its multiple uses have grown rapidly. Many processes that used to be done in traditional ways (offline) are now being modernized and transformed to digital form from which the whole society benefits. One of the most emerging technologies that promises to be influential on any kind of industry is the blockchain technology. Blockchain technology applies not only to IT but also finance, law, medicine, governance and more.

In chart 1 we can see the blockchain adaptation and its stages chronologically. From the same chart we can, also, observe that from 2018 we have real applications in the world while nowadays we have reached the point where we can create production ready projects based on blockchain. It is predicted that in 2025 the technology will have fully matured and stabilized. [13]

The interest attracted by research on blockchain technology is extremely important and this is recognized by the European grants which reached 340 million euros in 2020. At the same time, investments in tech start-ups reach 4.4 billion euros for USA (33%), 2.9 billion euros for Europe (22%), followed by China with 2.8 billion euros (21%) (data: europa.eu).

In this dissertation we will try to examine the application of blockchain technology in auctions which is one of the most popular aspects of financial markets and commerce. Most of today's auctions, whether digital or not, have one thing in common and that is that they are centralized and dependent on a third party. The system suffers from problems such as trust issues in the intermediary organization, increased costs due to high fees, inefficiency and lack of flexibility which we will inspect and see if it is possible that the replacement of the intermediary by the blockchain can provide a solution.

In the present dissertation, in the first part, we begin with the study of auction theory in order to formulate the basic ideas and problems around auctions. Then in the second part, the blockchain technology is presented without extensive technical analysis. The third section provides a literature overview of the application of blockchain in auctions, while the fourth section presents a DA model on blockchain. Based on this, a new combinatorial auction is proposed applied to the blockchain and at the same time our generalized code (GCAS) is presented, which gives a solution to the WDP of such auctions. In the end we report the conclusions that emerged from the overall study of this work.



Auction Theory

Being in an environment where it is hard to evaluate the buying price of an item-service is where auctions show their true value. Auctions can help us in answering questions like how much an old authentic painting or a television frequency costs. As Asunción Mochón and Yago Sáez wrote in their book "Understanding Auctions, 2015"

"We can define auctions as the market mechanism, operating under specific rules, that determines to whom one or more items will be awarded and at what price."

They are comprised of the sellers, the bidders and the auctioneer, who handles the procedure for a fee.

Before we proceed to analyze the bidders, it stands to reason that we should firstly refer to following quantities: value, selling price and bid. With the term value (v_i) we refer to the highest price that the i bidder would pay in order to obtain the item or service in question. Bid (b_i) is called the offer that the bidder places for the item or service in question whereas selling price (p_i) is the actual price that the successful player has to pay in order to acquire the item or service.

Consequently, there are three types of bids:
1) Underbidding, where $b_i < v_i$
2) Overbidding, where $b_i > v_i$
3) Sincere Bidding, where $b_i = v_i$

The highest bid that a bidder (b_i) will place is, also, the winning bid. Therefore, at this point the uncertainty from the seller's point of view is noted, caused by the lack of information about the bidders' personal valuations. Therefore, the selling price is a result of the auction.

In auction theory, three categories of bidders are recognized.

1) Private value: The personal valuation of each bidder is not affected by the personal valuations of the other bidders. An extreme category in which we assume that there is no secondary market (no resale).
2) Interdependent value: In case there is a secondary market, the information coming from other bidders' valuations will change the value v_i of the i bidder.
3) Common Value: An extreme case of interdependent value. Before the auction everyone has their personal valuation and after the end of the process everyone has the same (usually because after the sale there is an announcement of important information).

It is important to note the difference between the income of the bidder and the surplus of the bidder. After the bidder wins the auction by submitting the largest bid his income is equivalent to his valuation.

$$Income_i = v_i , \tag{1}$$

The difference between Income and expenditure (seller price) is the Surplus.

$$Surplus_i = Income_i - P^* \tag{2}$$

In case the bidder has not won, the income but also the surplus is zero. We should also note that the revenue of the seller in an auction is equivalent to the price payable by the winning bidder.

$$\text{Revenue} = P^* \tag{3}$$

The biggest challenge that bidders face is determining the bid they have to register. It is easy to notice that in case a bidder registers a higher bid automatically the chance of winning rises but at the same time reduces the profit margin from the process. On the contrary, if the bidder enters a lower bid this will reduce the chance of acquiring the item through the auction, while at the same time increasing the potential profit it may have from the process. The probability of winning the auctioned item is inversely proportional to the potential profit and for this reason the players have to make the appropriate decisions according to their risk profile. For example, a risk averse player would prefer to enter a higher bid in order to reduce the risk of losing the item which will of course cost him the expected profit, instead a risk lover player has a different approach as he would prefer a lower bid in which would increase his profit margin at the price of reducing the chances of winning the item. Somewhere between these cases would act a risk neutral player who would try to calculate the bid that would maximize his expected return.

Auctions can be divided into two major categories: single unit auctions and multiple units' auctions. At this point we will deal with standard single unit auctions. It is important in terms of design to dwell on the parameters with which the auctioneer will set the rules for the auction. The first parameter is the phases that the auction will have. The two most known are:

Dynamic auctions: in this category bidders are allowed to make multiple bids either in discrete rounds or in a continuous bidding process.
Single round auctions: the bidders have only one attempt to submit the best possible bid in order to win the auction (sealed - bid auction).

The main advantage of Dynamic Auctions is the transmission based on the information from the valuations of the other bidders as a result of which we have the reduction of the uncertainty as well as the effect of the winner's curse[*1]. On the contrary, the process' prolongation increases the cost of the auction and makes it more complicated while at the same time it does not favor the weak bidders who have lower valuations. On the other hand, in single round auctions the process is fast, simple and not so expensive with the negative point of the reduced seller revenue that may arise due to the threat of winner's curse.

The second specification to be considered by the auctioneer is the price rule, i.e., how the price to be paid by the winning bidder will be determined. The two main categories (while there are others) are the following:

First price is the rule that when applied, the winner pays the full amount of his bid.

$$P^* = b_i^* \tag{4}$$

Second price is the rule that when applied, the winner pays the amount equal to the second-best bid.

$$P^* = b_i^{**} \tag{5}$$

*1 Winner's Curse: It is the case in which the winner of the item has overrated the item that he won and as a result no one else can follow his bid. The larger the difference from the other bids the more likely the effect.

The first price rule, although a simple procedure, creates the problem of inefficient allocation of items. This is because an offer equivalent to the valuation results in zero profit, so bidders will resort to bid below the personal valuation. Additional enhancement of the winner's curse effect results from this rule. These two problems faced by the bidders are mitigated with the rule of second price, as now bidders can make an offer equivalent to their valuation since by paying the second-best price there will be a profit margin.

Using the parameters, we analyzed previously we can form the four basic models in the category of single unit auctions.

Ascending-Bid Auction

"In ascending-bid auctions, also called English auctions, the seller sets an initial price which is quite low and gradually increases until there is only one bidder left. It is important in English auctions that the winner pays based on the second price rule plus the bid increment. There are several ways to conduct an ascending bid auction."

Bidders increase the price: In this example the same bidders register their bids in different rounds increasing their bids each time and the auction ends when a bidder secures an offer that no one else can cover. Classic auction model for works of art and antiques, but also with applications in electronic auctions.
The auctioneer raises the price: The auctioneer raises the price from round to round and the bidders decide whether to bid.
Price increases continuously: The price increases automatically (with a steady increase) and the bidder decides whether he stops or not participating.

Let's study an ascending-bid auction case through an application. In this example (table 1) an item is auctioned by a seller to three interested bidders (b_1, b_2, b_3) who have formed their personal valuation for the item ($V_1 = 100, V_2 = 200, V_3 = 400$). Assume that they will bid sincerely ($b_i = v_i$) and will stop biding when the value of the round is greater than their valuation. The seller starts the auction at $t = 0$ with price $P_0 = 100$€ and continues to increase the price by 100€ each round. Bidders compare the price of each round with their personal valuation and if the price is less or equal with their personal rating then they continue participating in the auction. As can be seen from the table 1 in round t = 2 the price is € 300 and players 1 and 2 stop participating in the auction because $V_1 < V_2 < 300$, while player 3 continues as $V_3 > 300$. The process is completed with player 3 as the winner with,

winning bid: $200 + 100 = 300$€
profit: $500 - 300 = 200$€ and
sellers Revenue: 300€

Table 1: Ascending-Bid Auction

	Price P_t	$b_{1,t}$	$b_{2,t}$	$b_{3,t}$
Valuation / Round(t)		100	200	500
t = 0	100	100	100	100
t = 1	200	0	200	200
t = 2	300	0	0	300

The English auctions are a dynamic and simple process, easily understood by all which at the same time reduces the effect of the winner's curse as it applies elements from the second price rule.

Descending-Bid Auction

In case of descending-bid auctions or the so-called Dutch auctions, the process starts with a high price while it continues declining. The first bidder to accept a price is the auction winner. In this kind of auctions, the winner pays based on the first price rule, an amount equal to the price accepted. By extension of that, Dutch auctions maximize the winner's curse effect but also maximize the seller's revenue.

Having the information from the previous example we can study a corresponding case where 1 item is auctioned by a seller to three interested bidders. The seller starts the auction at $t = 0$ with an initial price of 700€ and in each round the price is reduced by 100€. Bidders compare the price of each round with their personal valuation and when the price is less than or equal to their valuation then they will bid. As we see in table 2 the auction starts with a price of 700€ but no one is interested as $V_1 < V_2 < V_3 < 700$€. For $t = 1$ we have the same results, but at $t = 2$ when the price drops to € 500 player 3 is interested as $V_3 = P_2$ and bids for $b_3 = 500$. Therefore, the auction is completed with player 3 as the winner with profit $V_3 - b_3 = 0$ and sellers Revenue = 500€

Table 2: Descending-Bid Auction

Round(t) \ Valuation	Price P_t	$b_{1,t}$	$b_{2,t}$	$b_{3,t}$
		100	200	500
t = 0	700	0	0	0
t = 1	600	0	0	0
t = 2	500	0	0	500

We observe that the players in Dutch auctions will try to make bids smaller than their valuation (underbidding) so that they can have a positive profit, which can cause inefficient allocation.

First-Price Sealed-Bid Auction

In the first-price sealed-bid auction the procedure provides just one round in which players register their best bid, with the bids being done simultaneously by all the bidders. The winner of the auction is defined as the one with the highest bid and he pays an amount equivalent to the bid he has submitted. In such auction models' bidders tend to follow an underbidding strategy to avoid the zero-surplus due to the first price rule.

Second-Price Sealed-Bid Auction

In the second-price sealed-bid auction model (Vickrey auction) the process is completed in a round where players register their best bids at the same time. The winner of the auction is the one with the highest bid and the amount you have to pay to get the item is equivalent to the second-best offer following the second price rule.

The case of the example we observed so far in a sealed-bid auction will take place as shown in table 3 where for first price rule player 3 will win with a bid of € 500 having 0$ profit and the seller a revenue of 500$, while for second price rule, again, player 3 will win with a bid of 500$ but will pay 200 $ and will have a profit of 500-200 = 300$ with the seller having a revenue of 200$. Unlike the first price rule, players can follow a sincere bidding strategy as they can have a positive profit with this strategy. It is obvious that if the seller applies the second price rule, he will achieve efficient allocation but at the same time he will not maximize his revenue.

table 3: First-Price/Second-Price Sealed-Bid Auction			
	$b_{1,t}$	**$b_{2,t}$**	**$b_{3,t}$**
Round(t) \ **Valuation**	100	200	500
t = 0	100	200	500

Double auctions

So far, we have referred to the part of the literature which analyzes auction cases involving many players and a seller that actually acts as a monopoly. But there are also cases in which many players and many sellers interact. These models are two-sided auctions called double auctions (DA). The two-sided auctions are categorized into sealed bids and Dynamic auctions.

Sealed-bid DAs

In this category buyers and sellers are asked to enter in one round the prices at which they buy and sell. We symbolize with b_i the bid of the i buyer while with $o_k$the price below which the k seller refuses to sell. The equilibrium price P^* is then formed through cumulative supply and demand. One question that arises is how much each buyer will pay and how much each seller will receive. The literature makes two proposals for this problem.

The first one refers to the uniform price rule according to which all winners, ie those who buy at a price equal to the equivalent equilibrium price, will pay P^*. At the same time the winning sellers who offer a price up to the equilibrium price will be paid P^*.

$$P^e = P^* \tag{6}$$

On the other hand, there is the proposal of discriminatory pricing rules in which either each bidder pays an amount proportional to the amount of his winning bid, or the price is determined by the offers of sellers.

Aggregate payments:

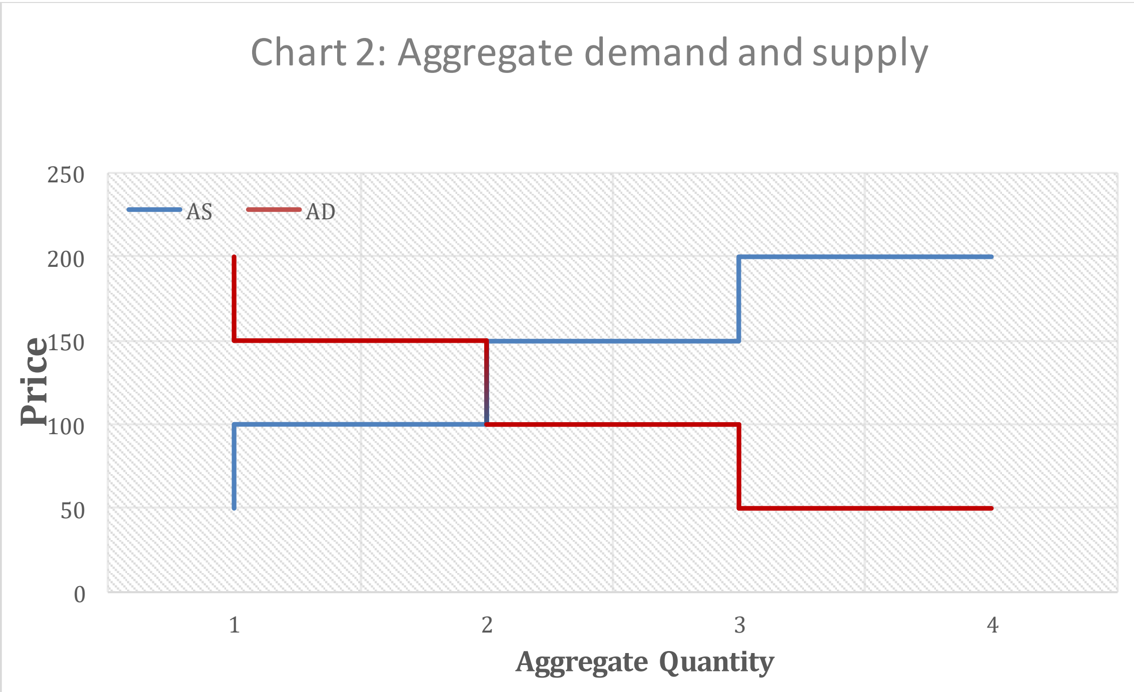
$$P^* = \sum\nolimits_{i\epsilon W} b_i^* \ , with\, W \;\; the\ set\ of\ winning\ bidders\ or, \qquad (7)$$

$$P^* = \sum\nolimits_{k\epsilon Z} o_k^* \ , with\, Z\ the\ set\ of\ winning\ sellers. \qquad (8)$$

Let's look at an example of a sealed-bid double auction to understand the process in practice. Suppose we have 8 participants, 4 buyers and 4 sellers trying to sell the same item*2. In the table 4 we have the bids of the participants. Also, in chart 2 we can see the aggregate demand and supply. At this point depending on the rule set by the auctioneer the process continues.

Table 4:Sealed-bid Das

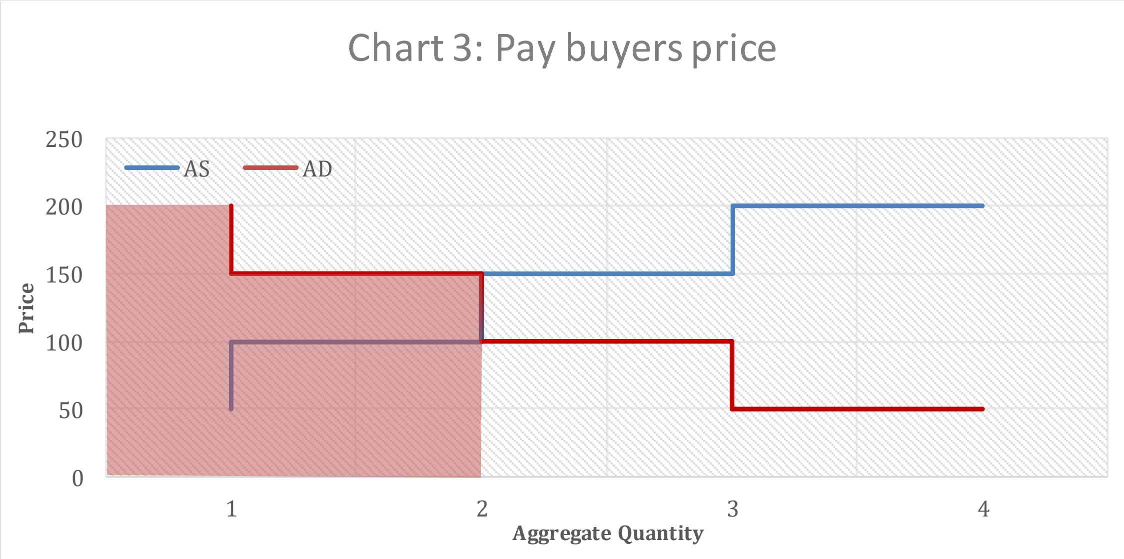
Participants	Price
b_1	50
b_2	100
b_3	150
b_4	200
s_1	200
s_2	150
s_3	100
s_4	50



*2 For reasons of simplification of the example we assume that the quantity of demand and supply for the item is one unit for everyone. Nevertheless, the process works normally for more units also.

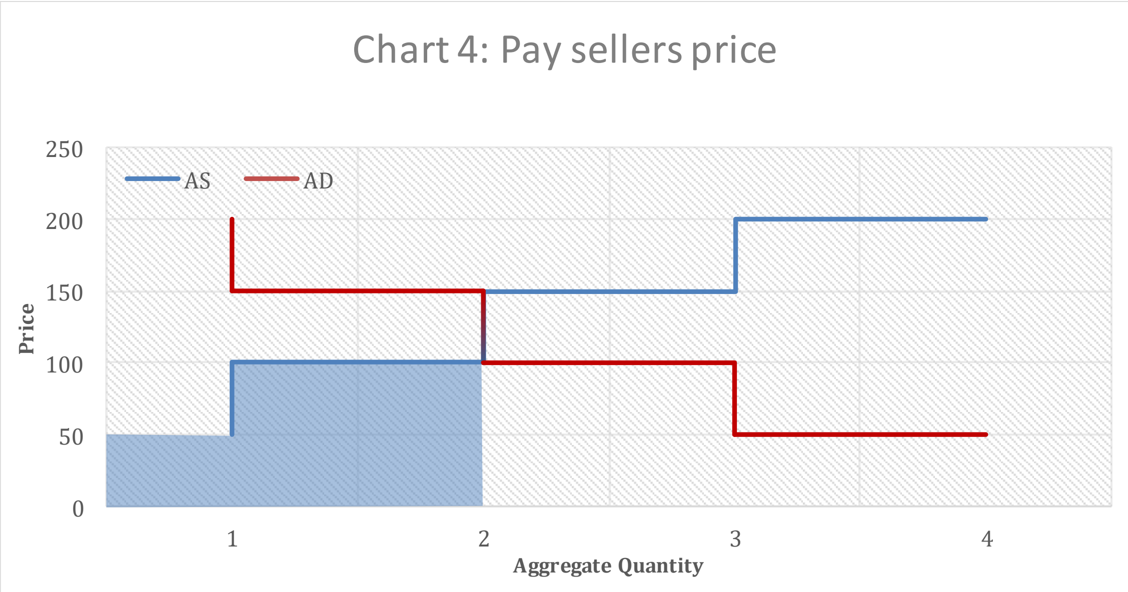
In case of pay buyer price:

$P^* = 200 + 150 = 350€$,*with* P^* *the aggregate price*



Also, pay sellers prices:

$P^* = 50 + 100 = 150€$, *with* P^* *the aggregate price*



Dynamic DAs

In Dynamic models, unlike sealed bids, players interact in a continuous process that allows them to buy and sell items in multiple rounds. The basic forms of these are Synchronized DAs and the Double Dutch auction method, but we will not expand further in their analysis as they are not necessary for the continuation of the dissertation.

Combinatorial Auctions

There are cases where players are interested in acquiring two or more items at the same time. When these items are auctioned in separate simultaneous auctions then players face the exposure problem. In other words, when players have complicated preference structures, they are exposed to greater risk (aggregation risk). But what do we mean by complicated preference? When players are interested in complementary or substitute items then the personal valuation process becomes more complex. Specifically, when interested in substitute items: "the value of a combination of items is lower than the sum of the individual values", while when interested in complementary items: "the value of a combination of items is greater than the sum of the individual values". [1]

Therefore, sellers need to design the auction in such a way that bidders have the opportunity to express their preferences clearly. This problem is solved by an auction model called 'combinatorial auctions' (grouped auctions). Combinatorial are multiple units' auctions models in which the seller in an auction offers multiple items, usually substitutes or complementary, while at the same time bidders have the right to bid individual items or combinations of more than one items.
Combinatorial auction models are the most effective way to avoid the exposure problem that bidders face. Below there is a table with an example to comprehend the exposure problem.

Table 5:Exposure problem

item packages	index	Valuation
Pen	A	6
Pencil	B	3
Paper	C	5
Pen + Pencil	AB	7
Pen + Paper	AC	14
Pencil + Paper	BC	10
Pen + Pencil + Paper	ABC	12

Next, we will analyze an example through which we will be able to highlight the exposure problem that a player may face. Suppose we have a case where three items are auctioned, a pen, a pencil and a piece of paper, while at the same time we have a player who is interested in the items as he intends to write a letter. In this case the pen and the pencil are substitute goods while these two items are complementary to the paper. The personal valuations of the player are mentioned in table 5 from which we observe that while the player valuates the pen separately for 6 and the pencil for 3 in case, he acquires them together his valuation is less than their sum. Contrariwise, in the case of the pen and the paper as well as the pencil and the paper, his valuation is higher than the sum of the separate objects. In this way the player has the opportunity to express exactly his needs. We also notice that if it was not possible to get the pen or pencil as a bundle with the paper and had to win each one separately the player would have to face the risk (exposure problem) to win only one of the two items. On the contrary, now in the form of CAs he has the ability to claim them at the same time (as a bundle) and not be in the awkward position of having won only one of the two items needed to achieve his goal.

After the bids are placed, follows the determination of the winners. Due to the complex process, the allocation problem appears that the seller must solve, also known as 'winner determination problem'(WDP). Combinatorial models include I = (1, 2, ..., i) players claiming J = (1, 2, ..., j) items. Each player can place as many bids as he wants either for separately items ether for a combination of items. These combinations are subsets of J (Z ⊆ J), while its value is expressed by the symbol v_i(Z) and bid by the symbol b_i(Z). Therefore, the seller collecting all the bids, provided that each item is given to only one bidder, must compute the winning bids that max revenue through this mathematical formula:

$$\max \sum_{i \in I} \sum_{Z \subseteq J} b_i(Z) w_i(Z)\,, where \tag{9}$$

$$1) \sum_{Z \supseteq j} \sum_{i \in I} w_i(Z) \leq 1, \forall\, j \in J \tag{10}$$

$$2) \sum_{Z \subseteq J} w_i(Z) \leq 1, \forall\, i \in I \tag{11}$$

$$3) w_i(Z) \;\in\; [0,1], \forall\, i \in Z \subseteq J, \forall\, i \in I \tag{12}$$

Limitation 1 arises from the fact that each item must be assigned to only one bidder

Limitation 2 arises from the fact that the bids are mutually exclusive.

Limitation 3 is the binary variable w_i that takes the value 1 when bidder i wins an item and the value 0 when the bidder does not win anything.

To consider a case of resolving WDP in CA we will study the following example (table 6). The seller auctions 3 items (A, B, C) so the bidders can claim the items. In the table 6 we see all the possible combinations that the bidders can claim the items as well as the corresponding bid for each player b_i (K). There are many ways to distribute items and the WDP solution will give us the combination that maximizes revenue below the restrictions we mentioned. We observe that the case of $\boldsymbol{b_3}(A) + b_2(B) + \boldsymbol{b_3}(C) = 1350€$ while offering max revenue does not follow the limitation of mutually exclusive bids as b_3 wins the A item and to C item separately. Therefore, the $b_1(A) + b_2(B) + b_3(C) = 1350€$ combination offers the max revenue covering all the constraints.

Table 6: Combinatorial Auction (WDP)

items combinations (K)	**b_1(K)**	**b_2(K)**	**b_3(K)**
A	450	300	450
B	200	450	200
C	200	200	450
AB	400	400	425
AC	500	600	525
BC	300	300	300
ABC	600	600	800

Of course, in reality, the number of items and bidders is much larger than our example. As a result, the WDP solution is complex and time consuming and may require a lot of computing power. In summary, the bidders apply the offers in combinatorial auction models and the seller determines the WDP from which the winning bids result. Next, it remains to define the pricing rule. The rule that prevails in these models is the first price while there is also the VCG mechanism. It is worth noting that regardless of the pricing rule the winners of the items do not change.

The winning players have to pay as much as their winning bid if the first price rule is confirmed.

$$P_i^* = b_i^*(Z) \tag{13}$$

while, the revenue of the sellers is equivalent to the total of the winning bids,

$$Revenue^* = \sum_{i \in WDP} P_i^*\text{, where WDP is the set of winning bidders.} \tag{14}$$

The first price rule, although quite simple to understand and apply, has a notable disadvantage and that is the fact that the bidders in order to create a positive surplus tend to underbid and this, as we have said before, causes inefficient allocation. To address this problem the VCG mechanism has been proposed but is not applied in practice because it can end up creating more problems.

Efficient and Optimal Auctions

It is clear that there are many auction designs models and rules by which auction procedures are conducted. However, it is obvious in the literature that there are two goals that sellers must achieve: maximizing their revenue and the efficient allocation of items that will take place through the auction. These two goals often clash as in the attempt to maximize its profit a seller ends up not selling to the buyer who had the highest personal validation (case of first price rule etc.). We call an auction optimal when the seller maximizes the expected revenue. On the other hand, the auction in which the items/services end up with the bidders with the highest personal valuation is called efficient auction. We summed up some of the characteristics of the four-basic types of auctions in table 7 and, also, we created the tree diagram of all auction categories that we discussed in the previous analysis (figure 1).

Table 7: Auctions characteristics

Auction Mode	Winner's Curse	Seller's Revenue	Efficient Auctions	Optimal Auctions
Ascending	-	-	+	+
Descending	+	+	-	-
First-Price Sealed-Bid	+	+	-	-
Second-Price Sealed-Bid	-	-	+	+

Common Centralized Auction Systems and Challenges

The majority of the auctions are centralized systems either online or offline. Examples of such a system in finance are banks that undertake to negotiate government bonds to set the price. Also, companies that deal with crowdfunding by acting as an intermediary between producers and investors. In these systems the basic precondition for conducting the process is the existence of a third party (auctioneer) who will be fully trusted by the sellers and the bidders. The problem of the corrupt third party is one of the biggest problems mentioned in the relevant literature. [24] The auctioneer may, for personal gain, influence the bidding process in various ways either by adding fake bids or by blocking specific bids. They may even be in consultation with some bidders or may award the item to someone who has not won through the process mentioned above. Another problem of the system is the cost and time spent by third parties to offer these services which are often really high, since in fact the bids and the items/services are much more and require complex calculations to determine the winners.

These problems are highlighted and intensified in case the central authority has lost its credibility and respect. The approach to these questions is the application of blockchain technology and the utilization of its fundamental features. Before analysing the relevant literature on conducting auctions with decentralized blockchain technology we will try to present the blockchain.

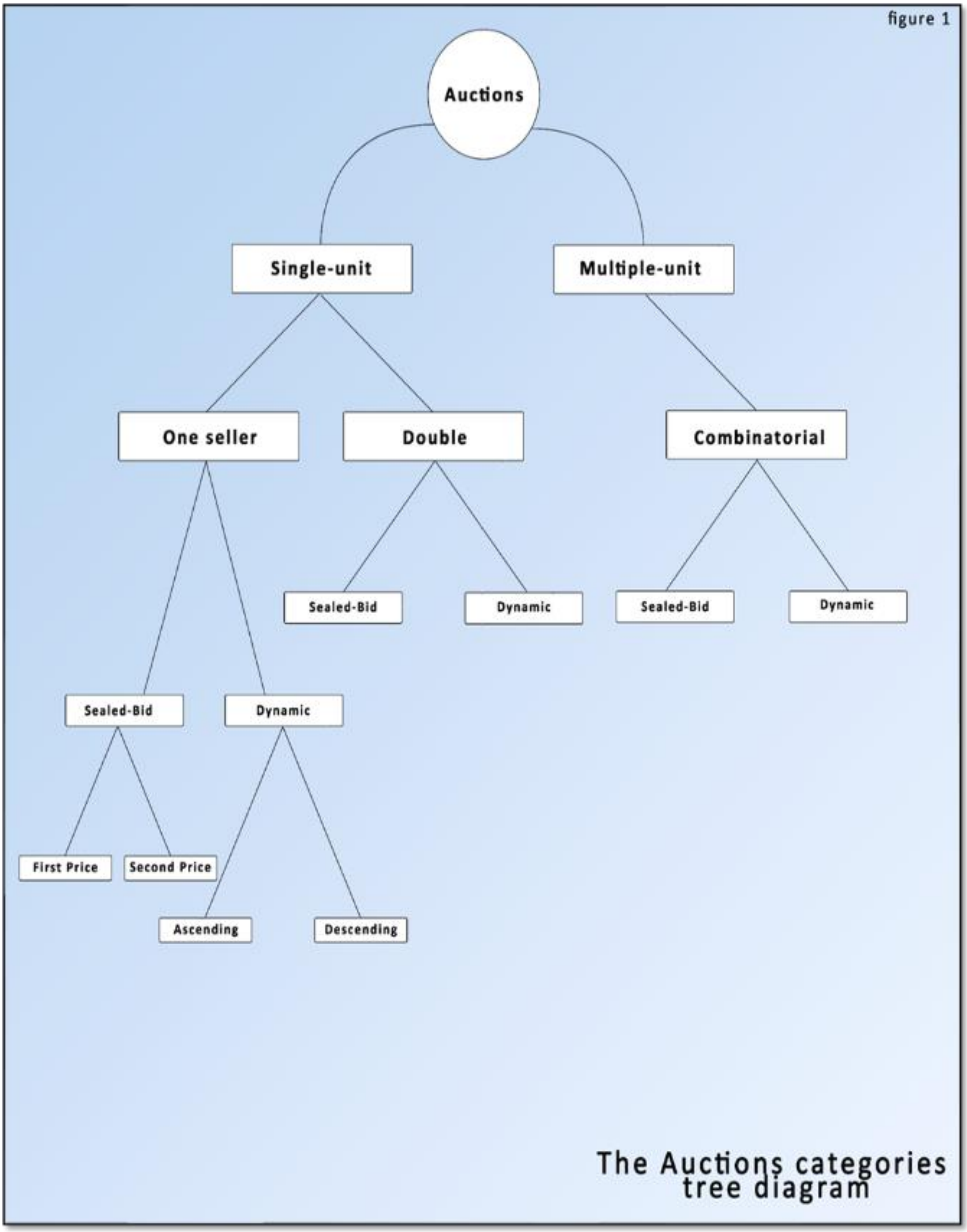


figure 1
Auctions
Single-unit
Multiple-unit
One seller
Double
Combinatorial
Sealed-Bid
Dynamic
Sealed-Bid
Dynamic
Sealed-Bid
Dynamic
First Price
Second Price
Ascending
Descending
The Auctions categories tree diagram

Blockchain Technology

In 2008 a remarkable article appeared with the title "Bitcoin: A Peer-to-Peer Electronic Cash System" under the pseudonym Satoshi Nakamoto. This article first mentions the concept of chain of blocks. The identity of the creator or group that created this article remains unknown to this day. As time went by, this technology became known as blockchain. There is particular difficulty in formulating a representative definition for blockchain as the technology is not yet fully mature and has not yet been sufficiently understood. From the study of the literature, we distinguish the following two definitions:

*Definition of Blockchain (Wattana Viriyasitavata, Danupol Hoonsopon (2018), "*Blockchain characteristics and consensus in modern business processes".

> *"A technology that enables immutability, and integrity of data in which a record of transactions made in a system are maintained across several distributed nodes that are linked in a peer-to-peer network".*

Definition of Blockchain (Imran Bashir, Mastering Blockchain Second Edition,2018, page 16)

> *"Layman's definition: Blockchain is an ever-growing, secure, shared record keeping system in which each user of the data holds a copy of the records, which can only be updated if all parties involved in a transaction agree to update.*
> *Technical definition: Blockchain is a peer-to-peer, distributed ledger that is cryptographically-secure, append-only, immutable (extremely hard to change), and updateable only via consensus or agreement among peers."*

The special fact of this paper is that at no point does it innovate and all its parts are pieces from previous work of other research. In fact, its creator had envisioned a new "electronic cash system fully peer to peer with no trusted party". In essence, Satoshi Nakamoto used the tools that already existed at the time and combined them to achieve his vision by creating the first blockchain in which the cryptocurrency Bitcoin was applied as a unit of measurement, transfer and storage of value.

An analysis of the technical terms referred to in the technical definition of blockchain will help in understanding blockchain technology.

Peer-to-peer: By peer-to-peer we mean that there is no central authority to control the network. Therefore, all network participants communicate with each other and trade without the direct intervention of a third-party intermediary.

Distributed ledger: This term refers to the ledger that is spread across all network participants. Therefore, all Peers have a copy of the ledger.

Cryptographically-secure: Blockchain is cryptographically-secure as it provides security that protects the ledger from malicious abuse and breach. This is achieved by encryption offering, among other things, non-repudiation, data integrity, and data origin authentication.

Append-only: Blockchain is append-only which means that data can be added to the network chain in time order or sequential order. When some data is added to the blockchain it is practically invariable except in a few rare cases. Therefore, the blockchain ledger cannot be changed.

Updateable via consensus: The most important technical feature of blockchain is the fact that it is updateable only via consensus. In other words, the network is decentralized and therefore there is no central authority to complement the ledge. In order for the blockchain to be updated, it will require unanimity from all participants (or 51% of them). Various algorithms are used to accomplish this in order for all the participants to agree on the final blockchain format based on its protocol.

Nodes: Network participants are represented by the hardware systems they own and are called nodes. All nodes have a memory and a processor while also having the same capabilities to create, send and receive data between them. The nodes involved in the blockchain can be miners creating new blocks or block signers and can be honest, damaged or malicious.

Illustration

For a further understanding of blockchain technology we can imagine the following simile (figure 2). A tower with floors, based on the internet, the main means of communication of all networks. Upstairs with the help of the internet, the peer-to-peer network runs which in turn hosts the blockchain functions such as transactions, blocks, smart contracts, consensus mechanism and others on the upper floors. Finally, on the top floor are the users or otherwise nodes that link to the blockchain and execute several procedures, including transactions, verifications of transactions, consensus and more.

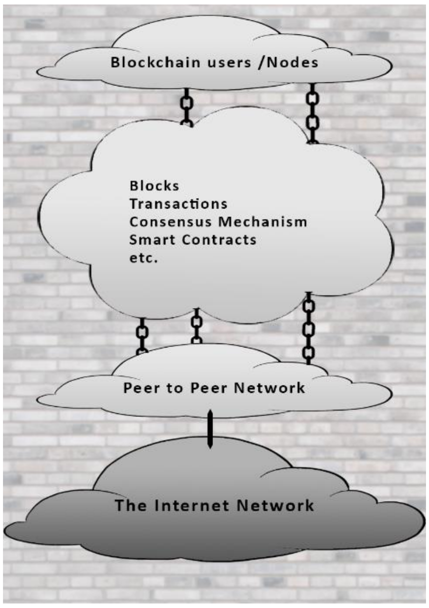


figure 2

Block, Nonce and Hash

Therefore, we have a distributed data storage consisting of blocks which are connected. But what exactly are these blocks? There is a selection of transactions grouped and organized, by transactions we mean a recording of an event such as the transfer of a monetary value from one person to another or a commodity exchange for money. The size of the blocks depends on the type of blockchain. Additionally, in blocks there is a reference to the previous block, unless it is the genesis block. We have blocks where each block knows the block before it, this builds up a chain where the order does matter. The basic structure of a block is: the block header which contains essentially the reference to the previous block, the timestamp which is a digital record of the time of occurrence of the block, the cryptographic nonce and the Merkle root (hash). After the header is the main part of the block which contains the list of transactions.

Cryptographic nonce is an arbitrary number which can be used only once and its purpose is to deal with problems such as replay attacks and authentication. Nonce is used in blockchain technology in Proof of Work (PoW). There is an extra level of difficulty in the process of finding the next block from the miners. As in addition to the transactions that they have to verify, they are also obliged to find the random number (nonce) that the block has. The characteristics of nonce are the means by which the level of difficulty is set for the miners to find a block and win the reward for this contribution to the network. This solves in this type of blockchain (PoW) the question of who will verify the next block and receive the reward.

Merkle root is a hash created from the hashes from all transactions that are listed in block. Hashing is the technique by which we convert any (input) text or data into a fixed number of text string characters (output) through functions. Thus, it is unlikely that two different input data will end up with the same hash output and at the same time two same inputs to always give the same output. A hashing application to be effective and beneficial must have some standard specifications regarding the security of the hash and the speed with which it occurs. In the blockchain all transaction information is converted to a hash and thus the transaction id is defined. Also, from all the transaction ids and the rest of the information that completes the block, the block hash emerges, as we have pointed out above, which is also, included in the next block (Merkel root) as a reference for the previous one. The new hash block is the hash that network miners are looking for in order to add the new block to the chain and win the reward. Hash is the mechanism that allows communication between blocks.

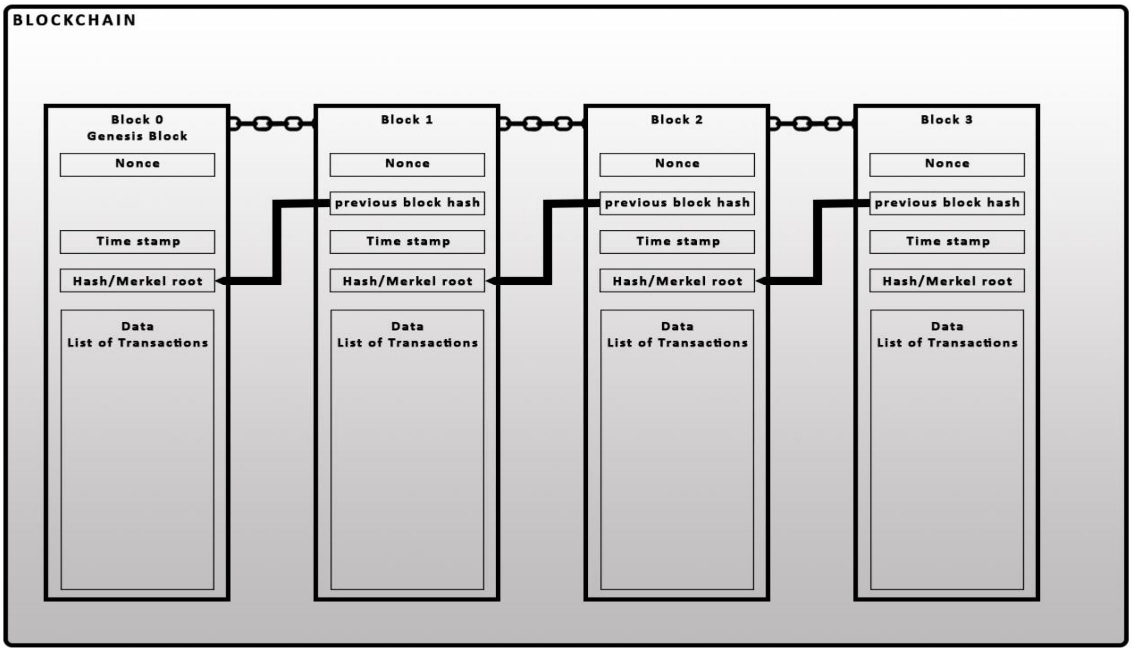


figure 3

Different Designs of Blockchain

Public Blockchain

Public blockchains are open to the world, meaning anyone can participate, but they belong to no one. All users have a copy of the ledger in their node and based on the consensus mechanism, which will be analyzed later, they update the ledger.

Private Blockchain

A private blockchain is a permission-based system, and as its name implies, unlike the public blockchain it is not open to the general public while its goal is to offer a private ledger to an organization or a group. Although very large projects do not yet exist in the form of private blockchain one of the applications it could possibly have is a kind of information sharing between government ministerial systems.

Tokenized Blockchain

Tokenized types of blockchain are the most basic type where Cryptocurrencies are created on the blockchain as a result, either of the consensus mechanism performed by the miners (as a reward) or directly as an initial coin offering (ICO) (appendix 1), or even, with an initial distribution backed to another asset (airdrops). The most recognized blockchains of this type are Bitcoin and Ethereum.

Tokenless blockchain

Tokenless blockchains are developed for the use of distributed ledger where transaction value is not the goal of the network. As we understand from their name, they do not have a value transfer unit.

Consensus Mechanism

We mentioned the Consensus mechanism previously and it is worth focusing on it as it is one of the most important features of blockchain, because it allows it to remain decentralized. This mechanism, in other words, enables a peer-to-peer system without any authority to make decisions while at the same time it is a key reason why malicious attacks on the network are ineffective. There are different types and categories of consensus mechanisms but they all have the same purpose, to ensure that records are true and honest. However, they vary in the way the consensus reached. Some of the best technical researches on this concept are the references [2],[9]. Below we will explain two of the most well-known mechanisms but before we analyzing them, it is essential to go back to the roots of the problem.

In 1982 Leslie Lamport wrote a computer science article describing the byzantine general problem [17]. The issue addressed in the paper is that stable computer systems should be able to operate efficiently in the presence of unreliable parts that transmit contradictory data to various components of the system. Although this situation is relevant in computer networks in Lamport's paper an attempt is made to describe it with the following example. Suppose the Byzantine army is outside a city. The Army is divided into three groups and each group has a general and the groups are scattered and not close to each other. The Army must form a coordinated action plan and the only way to communicate is through messages. Suppose that one of the generals is corrupted-traitor and will try to distort the messages in order to weaken the plan. The problem is how they will be able to execute the coordinated plan without being influenced by the actions of the malicious general.

Many attempts have been made to answer this question using mathematics and game theory. But the first practical solution was given through the paper of Satoshi Nakamoto [22] with the proof of work mechanism of Bitcoin. In the case of proof of work, generals are the nodes which are the connected dots that form a network in which can create as well as receive or send data.

Proof of Work

The proof of work system is one of the most effective in practice and is used in the blockchain of Bitcoin and Litecoin. In this model the transaction data is stored inside the blocks which are validated by the nodes. To decide which node will validate the block, the nodes compete with each other to solve a complex mathematical problem (mathematical puzzle) that is attached to the block (hash). This process is called mining and to achieve it super computing power is used by the nodes. The winner of the process wins the mining reward that differs accordingly to the blockchain. After validation, the winning node sends to the rest of the network the message of the block it found to validate it in turn (which happens much faster as they have the entire blockchain history at their disposal along with the new hash) so that to be registered in the blockchain and to continue the race for the next block. In other words, they spend energy in electrical form to solve the problem with the motivation of the reward and the fees of the transactions that are in the block.

The following example will help us to understand the system easier. Suppose we have a box locked with a padlock that contains a five-digit code. Participants are asked to find the code which opens the box. The way we achieve this is through multiple tests performed by the machines that represent them. The first who opens the box is the winner of its contents (rewards & fees).

Proof of Stake

In the case of proof of stake, the algorithm adopted by blockchains such as Eos and Cardano is validated by the validators, who in order to be selected, deposit a certain amount of coins into the network as stake (security deposit). The probability of selection of the validator is determined mainly by the size of the deposit that is insured in the network, since the larger the stake is, the greater the probability of selection, but also other factors depending on the blockchain. The final validator earns the fees from the transactions contained in the block, while in case of malicious action in the block transactions will be revealed by the rest of the network which in turn will try to validate the block to be added to the blockchain. As a result, the validator who tried to cheat lost a very large part of his stake. An example is given below for a better understanding of Proof of Stake.

We assume again that we have a box which this time is awarded based on a lottery system. Individuals depending on the size of their stake can buy their corresponding lotteries. With this mechanism, those who have the biggest stake have the highest chances of winning the box. This idea was based on the fact that the validation of the block must be done by a user who has invested enough in the network so that a malicious attack could not create greater advantages.

Smart Contract

Smart contracts are one of the most interesting and useful features of blockchain. They were first mentioned in the literature by Nick Szabo 1997 long before Satoshi Nakamoto's article was published. Nick Szabo's attempt was to create a distributed network that would store contracts. As we understand from their name, they are contracts, with the difference that smart contracts are digital and can be stored in the blockchain. In fact, smart contracts are computer programs that are stored in the blockchain. They usually contain some business logic and limited data.

The business logic stated in the contract is executed if certain criteria are met (if this - then that).

An example of implementing a smart contract are cases where a crowdfunding company enables product teams to advertise their idea to the public and set a minimum goal for its implementation. Then the public evaluates the ideas and decides whether to invest in any of them or not. In this case, both parties must trust the company that if the goal is achieved then it will distribute the funds to the production teams and in case the minimum goal is not met to return the funds to the public. Smart contracts have the potential to replace intermediaries and perform this service. Also, smart contracts are on the distributed network and they share blockchain principles like immutability and disreputability. For this reason, no one can change the smart contracts once they are programmed and not only that, but, also, the funds are controlled by no one. Smart contracts can find application and be useful in many areas such as banking (loans) and insurance (claims process), while the largest platform based on them is Ethereum. Another platform based on them is Binance Smart Chain (BSC).

The blockchain transaction validation and block creation

Following the descriptions of the blockchain and the technical definitions we will try to approach the functional actions of the blockchain regarding transaction validation and block creation. The general plan of generating blocks in the chain:

1. Initially participants create the transaction, and with the help of the private key, signs it (digital signing).
2. Then the transaction spreads, using a spreading protocol called gossip, to the other nodes (peers), which based on certain criteria validate the transaction.
3. In case the transaction is validated then it is added to the block which in turn spreads to the network. After these three stages, the transaction is considered approved.
4. After adding the block to the chain the next block to be added will be linked to the previous one and so the transaction will get the second confirmation.
5. Therefore every time a new block is created the transaction receives an additional confirmation. Depending on the blockchain, a different number of confirmations is required for a transaction to be considered final.

Advantages, Challenges and Questions About Blockchain

Blockchain technology has been proposed and used in many industries and leading companies. Some of the main advantages mentioned for the technology are the following:

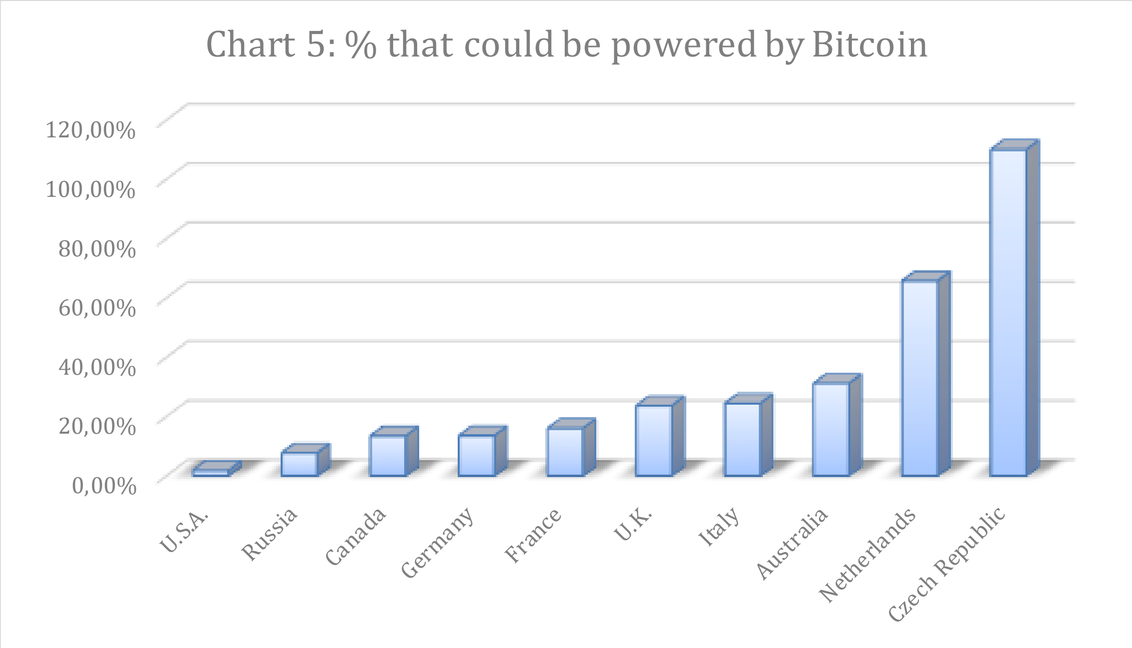
Decentralization: The basic idea of blockchain is based on the fact that it does not require a trusted third party but a consensus mechanism to update transactions.
Transparency and trust: In cases where recovery of trust is required is where blockchain shines as it is freely accessible to everyone and enables the system to be transparent.
Immutability: It is extremely difficult to change the ledger as it is almost impossible to change what is written.
Highly secure: The blockchain and its transactions are encrypted.
Cost reduction: With blockchain technology, intermediaries in transactions are removed, where along with them, the huge commissions for their services (fees) are eliminated.

Other advantages mentioned in the literature are *faster processes*, *fewer intermediaries*, *high availability* etc.

There are questions and challenges in blockchain technology but also in every attempt for change. Some of the most important questions analyzed in the literature are the following

Environmental Cost

The use of blockchain presupposes the use of a huge amount of electricity and especially in the mining process where users convert electricity into computing power through their machines to perform the verification of transactions. More specifically, the consumption of the Bitcoin network (which is the largest blockchain network) as shown in the chart 5 could cover 110% of the electricity needs of the Czech Republic and 65.7% of the Netherlands according the 2020 data [7]. Although the problem of the level of energy consumption in all data management mechanisms is a real issue, where in some cases there is a green approach through recyclable energy sources and in others not. Nevertheless, blockchain currently has a very large carbon footprint due to its energy consumption (main fossil fuel).



Regulation

The DLT (distributed ledger technology) as we have already mentioned is a relatively immature technology which, however, grows over time. Nonetheless, the lack of regulations is obvious and this is an obstacle for many large companies to be able to use it. Of course, we have now seen many regulators express their intention in this direction [12]. An important challenge in blockchain regulation is the balance between security and system stability while encouraging the development of this innovative technology.

Scalability

The issue of scalability in blockchain is a technological problem that the Developer Community must solve since as its size and use increase, the transactions that a blockchain network must complete are multiplied exponentially. Until the problem is solved effectively there will be black spots regarding the speed but also the cost of transactions. To solve this problem several solutions have been proposed, including on-chain, off-chain, side-chain solutions etc. [23]. Worth mentioning is, also, the proposal for an off-chain network the so-called Lightning network [14] which aims to reduce transactions' time and fees. Therefore, a major challenge is to find the ideal ratio between network security and scalability.

Self-Responsibility

An issue that emerges with applications such as DLT blockchain is the fact that users need to further develop their personal responsibility to ensure the secure use of blockchain functions (eg personal wallet). The reason is obvious as behind most decentralized applications there is no company to contribute to it as we are used to from traditional applications. Therefore, for example, users should be able to open wallets on their own and keep the keys safe so as not to be faced with unpleasant events.

Privacy

One of the key features of blockchain is its transparency. In other words, anyone who has some basic knowledge about blockchain has the ability to find the chain of transactions and study it. Therefore, since this data is often sensitive (eg wallet balance, detailed transactions) this feature could not be pleasant and stands in the way for many users.

Interests in blockchain failure from big players

The blockchain comes to make all the areas in which it can be implemented better and more effective with the ultimate goal of the benefit of society as a whole. Many times, however, it comes to replace the intermediaries in the market, which are usually of colossal size (eg Banks, etc.). This results in conflicts of interest and various underground obstacles to its further development.

Literature review

The [“Trevathan J., Read W. and Ghodosi H”, 24] was published with the purpose to highlight the main problems in the design of E-Auctions. It refers to trust issues, anonymity issues, bid authentication issues, price determination and payment problems but does not mention blockchain because it was published in 2005 before its existence, however it is very helpful in distinguishing and better understanding the main issues around the auction design.

The auctions section is a popular topic in the literature and there are many references such as anonymization [“Chin-Chen Chang, Ya-Fen Chang”, 6] and encryption [“Kazue Sako”, 16] for the solution of privacy issues that arise in types of auctions where there is a risk of information leakage and at the same time approach the trade of between efficiency and revenue.

In recent years we can find relevant studies dealing with auction issues by approaching them with decentralized systems as in [“Chiara Braghin, Stelvio Cimato, Ernesto Damiani and Michael Baronchelli”, 5] where an attempt to develop an online auction system for the four basic auction models based on Ethereum smart contracts is studied, as well as a cost and time analysis that takes place. Despite the problems in the privacy of the users, the authors that was mentioned above concludes that the auctions that are designed with smart contracts are superior to the traditional ones in transparency, non-repudiation, integrity and no need trusted third party.

In [“Li-Hsing Yen, Guang-Hong Sun”, 18] protocols are created for the application in CAs which have a decentralized character with the main feature being the fact that it is auctioneerless. In the system the participants (bidders) decide for themselves the winners, as they locally determine their bids and then they communicate that message to other participants in order to reach a consensus. They prove through their study that with the method they finally propose the result stabilizes and confirms the winner’s determination algorithm.

Another study is presented in [“Yi-Hui Chen, Shih-Hsin Chen, Iuon-Chang Lin”, 28] which highlights two of the main problems that traditional E-Auctions have. Firstly, it targets the decentralized system they adopt and the fees they charge that increase the transaction costs by a lot. In addition, it emphasizes the fact that in some type of auctions that is crucial to keep bidding information’s secret (sealed bids), there is no way to reassure participants about this as opposed to the traditional centralized model. In an effort to find a solution to these issues, a decentralized system based on blockchain technology is being developed using Smart Contracts to host the digital bidding process of the auction.

In [“Jung-San Lee, Chit-Jie Chew, Ying-Chin Chen, Kuo-Jui Wei”, 15], in order to dismiss the assumption that a trusted third party has to exist to ensure the auction system an auction model based on the blockchain technology was designed. They managed to create a Fair and Liberty (as they call it) auction system using the fundamental properties of decentralization, transparency but also with Smart contracts.

Although we are still at an early stage in terms of blockchain applications in finance, we can now study one of the most comprehensive proposals for the issuance of bonds through blockchain technology [“Malamas, Vangelis and Dasaklis, Thomas and Arakelian, Veni and Chondrokoukis, Gregory”, 19]. In addition, the whole process of issuing bonds is presented digitized, including the auctions that take place at the heart of the process in order to make all the process efficient and to overcome the issue of the untrusted third party (banks) and the transaction charges because of the high fees. An obstacle to proposals like this is the fact that the technology as we have mentioned has not yet matured and stabilized and as a result there are legal and regulation problems in its applications for securities [12].

The implementation of auctions, which is one of the most successful new projects on the market, could not be left out of our attention. In February 2020, a non-profit organization named WISE was formed to operate in the emerging innovative financial sector of DiFi (decentralized finance). The company's plan is through the new token (wise) to create a decentralized financial product (staking) similar to that of bonds or CDs (certificates of deposits). Unlike traditional ICOs, in order for Wise to distribute Wise tokens, conducted a new auction system through which the initial distribution took place. The process started on November 11 and was successfully completed on December 30. This practice was very successful and was characterized as fair distribution. It was very useful to us as it was a complete auction application integrated with blockchain technology. [WISE, 26]

Double Auction Model Based on Blockchain Technology

Introduction

In this chapter we will try to present in detail a complete and complex model the Double Auctions (DA), while at the same time we will study step by step the process of integrating the traditional auction in the technology of decentralized blockchain. We will analyze the model through an example of an application that we believe could be implemented with multiple benefits. In this example we will use blockchain technology to replace the third-party intermediary that brings producers and investors together. Producers present their business plan and look for investors to carry out their project with their funds. In practice, producers/companies through the initial coin offering [Appendix 1 ICO] have as their main goal their project's finance. Prospective investors, on the other hand, are looking for projects to invest their funds most effectively. Through the auction that we will present, a decentralized market will be created through which funds can be transferred from investors to producers/companies, but also to decide which project will be implemented and which will fail. In other words, the companies issue new coins and through the auction the price at which the investors buy them is formed. If the funds from the sale of coins meet the minimum requirements of the project then the ICO succeeds.

In the beginning the users of the network have to choose between being a patron or participants. Patron is the one who will launch an auction on the main auction blockchain. Users who choose to be participants will define their status as producers / companies (seller) or investors / consumers (buyer). Participants then enter their bids based on which the WDP result that determines the winners is calculated. Finally, the process of exchanging money with digital coins takes place (figure 4). We can therefore note the three basic auction procedures:
1) Creation of an auction by the patron.
2) Bidding process and WDP calculation
3) Merchandise

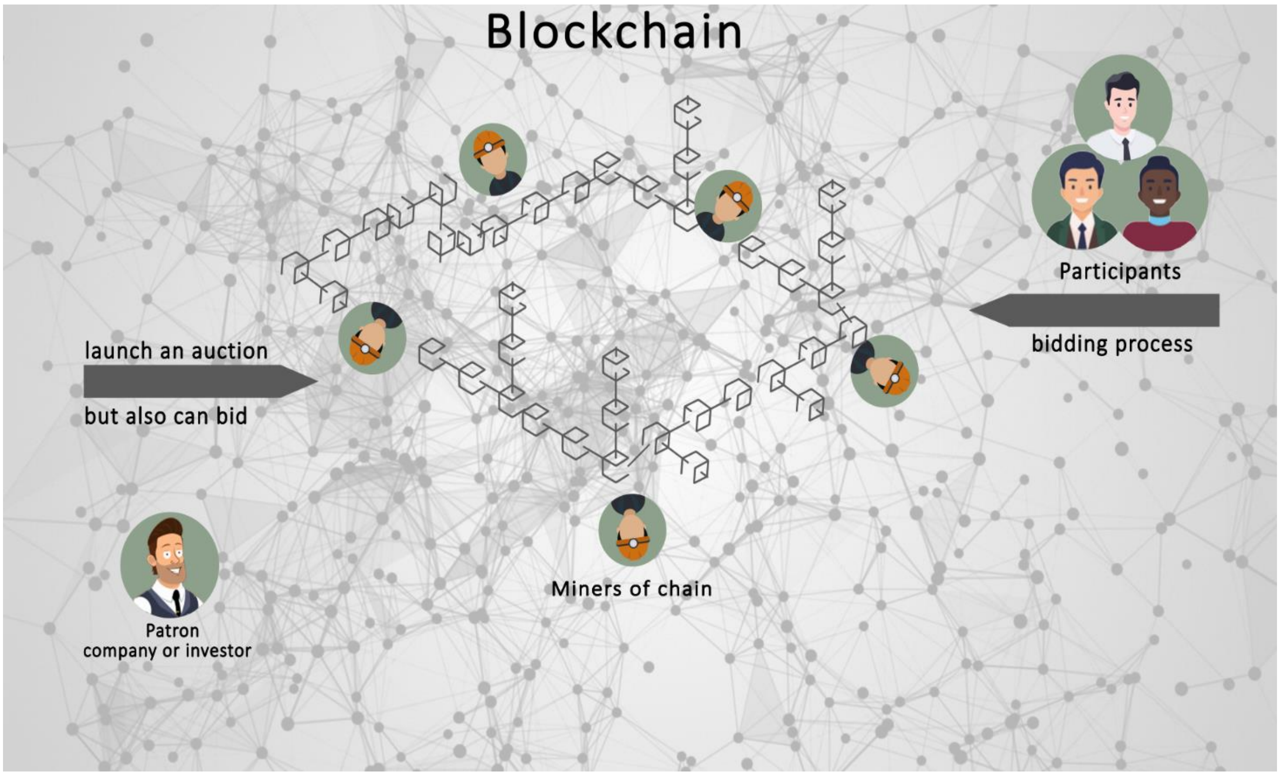


Figure 4

We will continue in this chapter presenting first the blockchain design then the user ranking system, the auction mechanism and finally the optimal auction allocation mechanism.

The Blockchain Design

In our model the blockchain will consist of three different types of blocks. First appears the beginning of the chain with the genesis block which contains the rules of the auction, the requirements for the creation of an auction by a patron and finally the standing orders for the participants. The Genesis block is followed by the main chain consisting of the auction blocks. Each block represents an auction that has been created. Each block is connected to a side chain that includes the transaction blocks in which are recorded all the transactions that take place during the bidding phase. In the following figure we visualize the chain.

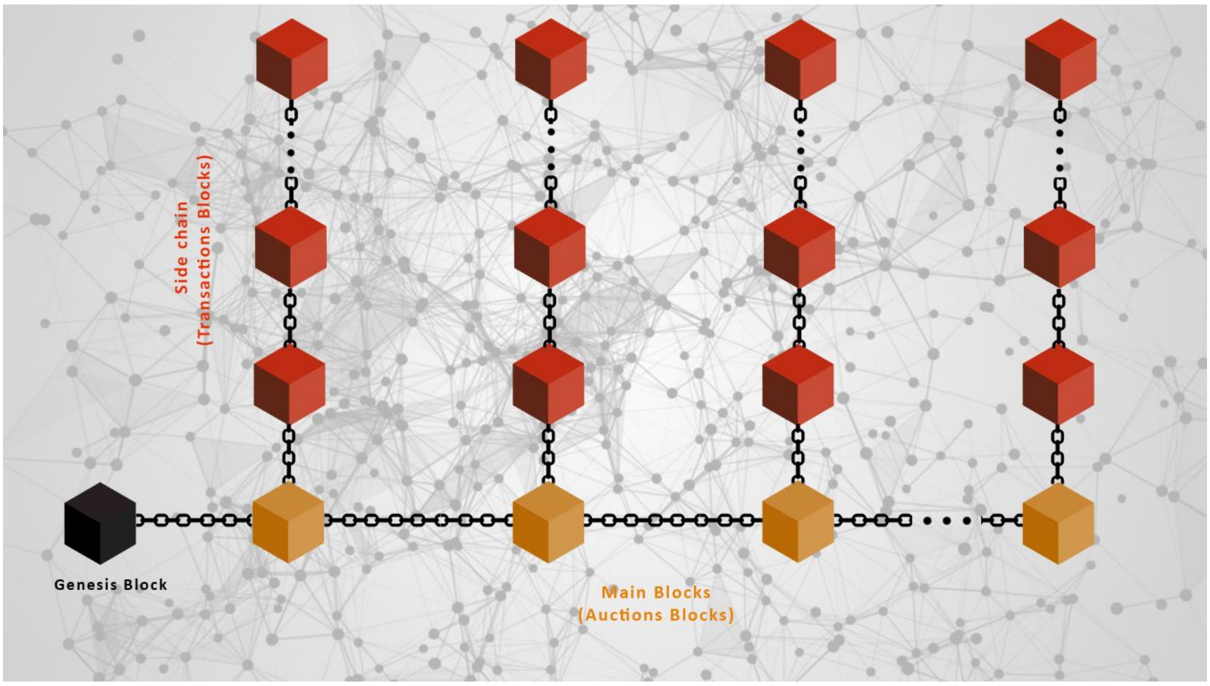


Figure 5

User Ranking System (URS)

In order to have an indicator for untrustworthy users we use a ranking system. The rank (t) value is representing the level of trust of a user. Everyone starts with a starting value $rank_t = 0{,}7$ with rank $\in [0{,}1]$. The rank of the winners is renewed each time an auction is completed.

$$Rank_t = (w_1 \times Rank_{t-1})\left(w_2 \times \overline{Rank}\right) \quad [15] \tag{15}$$

$Rank_{t-1}$: the most recent participant rating

$\overline{Rank}$: the rating obtained by the other trader after the end of the auction.

w_1, w_2 : are the weights through which we control the ranking system. These values are equal to $w_1 = 0{,}7$ *and* $w_2 = 0{,}3$ if $Rank_{t-1} \leq \overline{Rank}$,while in the case of $Rank_{t-1} \geq \overline{Rank}$ then the weights are equal to $w_1 = 0{,}3$ *and* $w_2 = 0{,}7$.

A security deposit rule applies to participants who want to register a bid (either sellers or buyers), in order to avoid problems with bid authentication. Deposits are defined as follows:

$$dep_{bidder} = \overline{dep} \times RF + bid \tag{16}$$

$$dep_{seller} = \overline{dep} \times RF \tag{17}$$

with
dep_{bidder}: amount of the security deposit of a buyer participant
dep_{seller} : amount of the security deposit of a seller participant
$\overline{dep}$: constant defined by the patron
RF: an amount based on the Rank that each user has to pay (table 8).

Table 8: User RF [15]

Rank	Rank Fraction (RF)
[0 , 0.25)	8
[0.25 , 0.5)	4
[0.5 , 0,75)	2
[0.75, 1]	1

Obviously, this mechanism rewards participants with lower deposits with a higher level of trust. Once the process is over all security deposits are returned to the participants.

Auction Mechanism

In this part of the chapter, we will analyze the stages of the auction process.

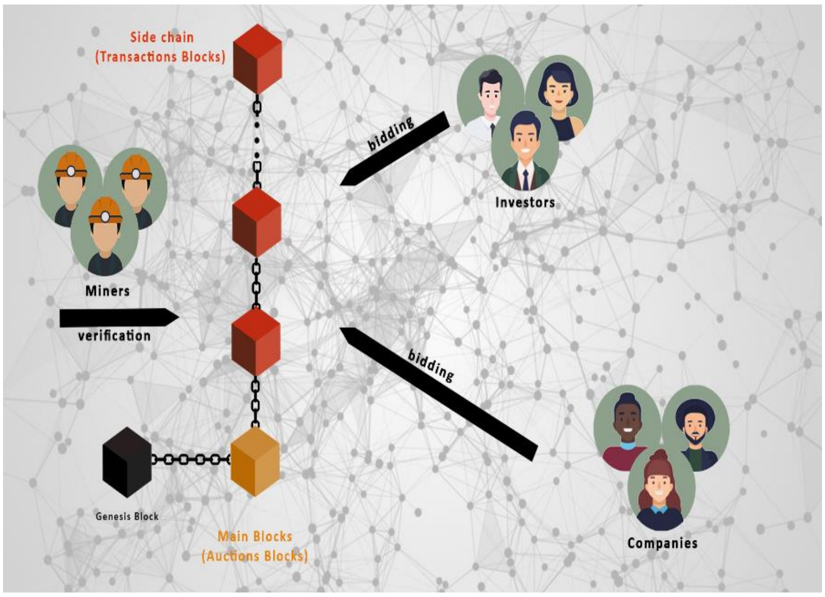
Stage 1. Create an auction

In order for an auction to start, a patron must begin by creating it based on the specifications set by the Genesis block. After consulting the Genesis block completes the block with the characteristics of the auction: Title, expired time, deposit, minimum number of participants, etc. as shown in the example of the table 9. After this process is done, the space is now open and framed with specific rules ready to receive participants. The first stage is completed with the miners verifying the existence and the correct syntax of the block.

Table 9: Characteristics of the Auction	
Title	DeFi Project
Deposit	5 €
Min. No. of participants	4
expired time	10/3/2021

Stage 2. Bidding process

Figure 6



At this stage the participants define their identity for the specific auction and then register their bids according to their personal needs by forming bidding data (figure 6/ table 10) in the side chain (transaction chain). In addition, bidders (buyers, sellers) before registering their bids must pay the deposit depending on the personal ranking level. Similar to the main chain in the transaction chain, after the bids are registered, the miners act to discover the blocks that contain all the information and to verify them so that they can be added to the side chain.

Table 10: Bidding Data

Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Project	Project 1	Project 1	Project 1	Project 1	Project 1	Project 1
Price	600	400	8000	500	800	3000
Quantity	100	100	1000	100	100	500
Project	Project 2	Project 2	Project 2	Project 2	Project 2	Project 2
Price	600	0	0	200	150	220
Quantity	1000	0	0	400	200	400
Project	Project 3	Project 3	Project 3	Project 3	Project 3	Project 3
Price	2300	2000	4000	0	0	8000
Quantity	2000	1000	5000	0	0	8000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Project	Project k	Project k	Project k	Project k	Project k	Project k
Price	p_{1k}	p_{2k}	p_{3k}	c_{1k}	c_{2k}	c_{3k}
Quantity	$Q\_inv_{1,k}$	$Q\_inv_{2,k}$	$Q\_inv_{3,k}$	$Q\_pro_{1,k}$	$Q\_pro_{2,k}$	$Q\_pro_{3,k}$

Stage3.The Winner Determination Mechanism

One of the key points in the auction process is the mechanism from which the winners emerge. The WDP (Winner Determination Problem) calculation process is undertaken by the miners. After they find the blocks and verify them, they start this process. First, they extract the bidding data (table 9) and continue calculating the WDP with which they end up with the allocation problem result through the Genetic Algorithm that we will study below. After calculating the result and the fitness of the result, the universal verification by the whole network follows and finally the completion of the smart contract of the auction that is in the auction block.

Stage 4. Settlements Mechanism

The last operation of the auction follows where the winners must proceed with the execution of the transactions and the settlement. To perform this process the participants first receive the validated message with the allocation result consequently, the sellers (producers/companies) transfer the coins to the winning buyers based on WDP. This is followed with the confirmation process by the buyers (investors/consumers) but also the evaluation of the sellers by the buyers with whom they had a transaction within the smart contract that is located on the block. Afterwards the winning sellers draw the income price and their security deposit and in turn evaluate in the smart contract the buyers with whom they had a transaction. Finally, the winning buyers and the other participants who did not win get their deposits back.

Optimal Solution of the WDP

The optimal allocation mechanism is the function by which winners of the auction are determined under some specific restrictions. This calculation is undertaken by the network miners. The first step for the calculation, as shown in table 10, is to separate the bidding data into different processes ($\text{process}_1, \text{process}_2, \ldots, \text{process}_k$) with K the number of projects. After categorizing the bids, the Genetic Algorithm will be used for the WDP_k solution and a quality index will be calculated at the same time called adaptive value (α_k) for each project separately. With the usage of the adaptive value, it will be ensured that the results of the WDPs follow the mathematical formulas below. Finally, after calculating all the WDP_k for each project we merge them to form the final WDP result of the auction. We use the binary variables to denote the successful bids of the sellers by 1 and the failed ones by 0. Accordingly, we denote the successful bids of the buyers by 1 and the failed ones by 0. The table 11 represents the example for better understanding.

Math formulas (all the variables are shown on table 14) [15]

$$1)\ Spread_k = \sum_{j=1}^{INV} P_{jk} b_{jk} - \sum_{i=1}^{PRO} c_{ik} s_{ik} \tag{18}$$

$$2)\ q_k = \sum_{i=1}^{PRO} Q\_PRO_{ik} s_{ik} - \sum_{j=1}^{INV} Q_{INV\,jk} b_{jk} \tag{19}$$

$$3)\, NoW_k = \begin{cases} \sum_{j=1}^{INV} b_{jk} + \sum_{i=1}^{PRO} s_{ik}, if \;\; q_k \geq 0 \\ -\left(\sum_{j=1}^{INV} b_{jk} + \sum_{i=1}^{PRO} s_{ik}\right), if \; q_k < 0 \end{cases} \tag{20}$$

$$4)\, \alpha_k = NoW_k \times 10^{*3} + Spread_k \tag{21}$$

$$5)\; \alpha = \sum_{k=1}^{K} \alpha_k \tag{22}$$

Table 11: Bidding data processing

Project 1						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	600	400	8000	500	800	3000
Quantity	100	100	1000	100	100	500
Project 2						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	600	0	0	200	150	220
Quantity	1000	0	0	400	200	400
Project 3						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	2300	2000	4000	0	0	8000
Quantity	2000	1000	5000	0	0	8000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Project k						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	p_{1k}	p_{2k}	p_{3k}	c_{1k}	c_{2k}	c_{3k}
Quantity	$Q\_inv_{1,k}$	$Q\_inv_{2,k}$	$Q\_inv_{3,k}$	$Q\_pro_{1,k}$	$Q\_pro_{2,k}$	$Q\_pro_{3,k}$

In our example we can observe (table 12) that for $process_1$ from WDP_1 the winners are $investor_1$ and $producer_1$ while the rest did not succeed. Forming the binary vector "100100" with,

$Spread_1 = 600 - 500 = 100$

$q_1 = 100 - 100 = 0$

$NoW_1 = 1 + 1 = 2$

$\alpha_1 = 20 + 100 = 120$

Regarding WDP2 the result is printed in binary vector "100111" with $Spread_2 = 600 - 200 - 150 - 220 = 30$

$q_2 = 400 + 200 + 400 - 1000 = 0$
$NoW_2 = 1 + 1 + 1 + 1 = 4$
$\alpha_2 = 40 + 30 = 70$
Also, WDP3 is captured in binary vector "111001" with,
$Spread_3 = 2300 + 2000 + 4000 - 8000 = 300$
$q_3 = 2000 + 1000 + 5000 - 2000 = 0$
$NoW_3 = 1 + 1 + 1 + 1 = 4$
$\alpha_3 = 40 + 300 = 340$
Finally, the base of the previous ones is formed the final WDP of the auction through the matrix in table 13 with, $\alpha = 120 + 170 + 340 = 530$

Table 12: WDP_k

Project 1						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	600	400	8000	500	800	3000
Quantity	100	100	1000	100	100	500
WDP_1	1	0	0	1	0	0
Project 2						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	600	0	0	200	150	220
Quantity	1000	0	0	400	200	400
WDP_2	1	0	0	1	1	1
Project 3						
Identity	investor 1	investor 2	investor 3	producer 1	producer 2	producer 3
Price	2300	2000	4000	0	0	8000
Quantity	2000	1000	5000	0	0	8000
WDP_3	1	1	1	0	0	1

Table 13:Final WDP

1	1	1
0	0	1
0	0	1
1	1	0
0	1	0
0	1	1

However, in our example we have 6 participants while in real conditions it is on a much larger scale. For this reason, the WDP solution requires more computing power. This problem as mentioned earlier is solved by the GA of miners. The algorithm consists of six chain steps. (Figure 7)
First the algorithm generates 10 different allocation results in binary vector format for project 1. Then it performs a crossover where pairs of vectors are selected randomly from which swap elements with each other as in the figure (7). The elements of the vectors that emerged from the crossover are

then randomly selected to mutate (that is, to be converted from 0 to 1 and vice versa). After these algorithmic steps in cases where the spread is negative, but also in cases where α_1 is less than the original vector, the generic algorithm regenerates the vectors and returns to the first step. In cases where the vector that has emerged after the mutation has a greater α_1 than the original then it remains the same. Therefore, the vector with the highest adaptive value after a large number of consecutive iterations (e.g., 100-150) is defined as the optimal allocation result of the project1.

This operation is repeated for K projects and at the end the table that forms the final WDP is created from all WDPk vectors.

Table 13: Variables Used

Sign	Definition
pro	the number of providers
inv	the number of investors
N	the number of projects
a	final adaptive value
a_k	project k adaptive value
q_k	remaining quantity of the kth project according to WDP_k
$spread_k$	spread between bid and ask in the auction
NoW	the number of the winners in an auction
NoW_k	the number of the winners of kth project
p_{jk}	bidding price of investor j for the project k (pay)
c_{ik}	bidding price of producer i for the project k (cost)
s_{ik}	binary "1" provider i bid succeed for the project k and "0" failed
b_{jk}	binary "1" investor j bid succeed for the project k and "0" failed
Q_pro_{ik}	quantity of the kth project for the provider i
Q_inv_{jk}	quantity of the kth project for the investor j
10^{*3}	positive variable, based on our data, that allows the correct operation of the model

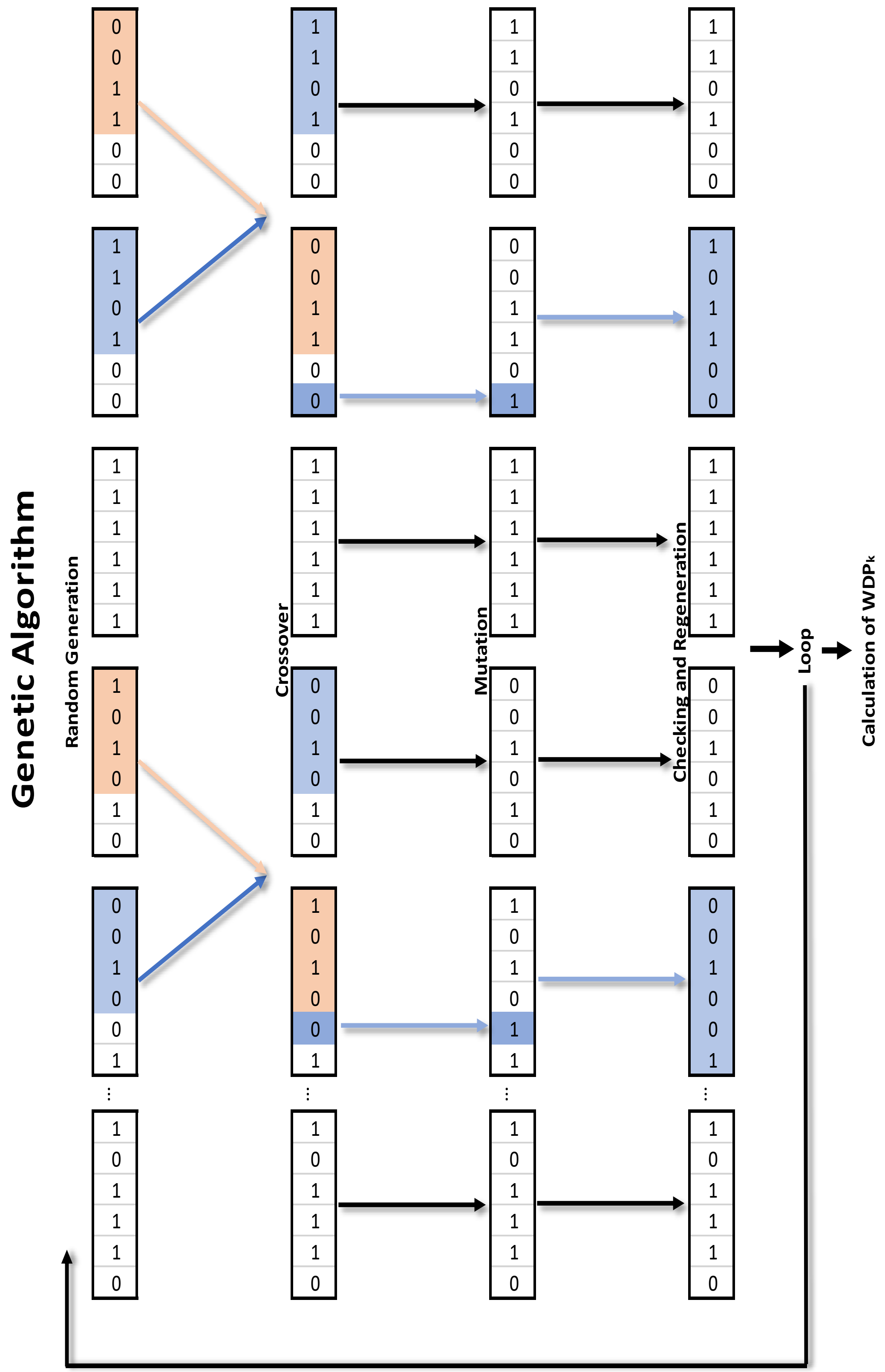

Genetic Algorithm
Random Generation
Crossover
Mutation
Checking and Regeneration
Loop
Calculation of WDPk

Figure 7

Combinatorial Auction Model on Blockchain

Although the model we analyzed in the previous chapter has the ability to manage and execute Double Auctions cases, it is unable to handle situations where there is a need for grouped bids from buyers, the so-called Combinatorial Auctions. An example of a Combinatorial Auction in finance is the Central Counterparties (Appendix 2 CCP) auctions under default management.
In this chapter we will try to build a model that can manage grouped bids from the buyers. Based on the previous model we presented we will try to make some changes to enable it to manage Combinatorial Auctions.

Proposed Model

This model will be an application of a Combinatorial Sealed-Bid First Price Auction with one seller and multiple buyers in a blockchain network. To describe the model, we will use an example of a European CCP having to manage a bankruptcy situation of a client who had a portfolio of 3 securities (e.g., interest rate swaps). The CCP, in order to deal with this unfavorable situation, will try to auction the three securities to three bidders.

As a result, we have a CA case with a seller and three buyers where the seller (CCP1) enters the system as a patron and can create the auction based on the Genesis block, while the other three players as buyers (bidders) aim to claim the securities.
The blockchain design will remain in the structure we presented in the previous model, figure [5] so the users ranking system (URS) will be implemented, as it meets the goals we want to achieve and protects us from attacks on the network.

Combinatorial Auction Mechanism

The mechanism of the auction of securities by the CCP (seller) is as follows:
First the CCP will create the main chain auction according to the Genesis block. Then interested participants will be able to express their needs through the bidding process of a round where they will have the opportunity to claim all possible combinations of securities while paying the security deposit depending on the URS as described for DA.

Thereupon, after the round of bids is over, it is the miners' turn to process the data and solve the complex winner's determination problem (WDP) of the auction and display the winning bids based on the general principle of maximizing the seller's revenue. Finally, after the winners are determined, the process of exchange and payment as well as evaluation is done in the same way as the DA model.

Optimal Allocation Mechanism

In the mechanism of optimal allocation our proposed model differs from the DA presented earlier as we have designed from scratch our own General Combinatorial Auctions Solver which the miners will use to solve the WDP. Initially the miners in order to solve the WDP of the auction receive all the bidding data of each player through the side chain (transaction chain) in the form of vectors. Then the table with the total data of the auction based on the personal vectors is formed. Once the overall data table is created, the algorithm begins the process of resolving the WDP and announcing the winners of the auction. The WDP solution results from trying to maximize the seller's revenue with two
limitations:
i) Each item must be sold to a maximum of one player (feasible)
ii) At most, each bidder acquires one successful offer. (mutually exclusive)

The generalized GCAS algorithm is designed based on the Heap theorem [11] and in a custom-made algorithm. More specifically, the Heap algorithm generates all possible variants that can result from n items. To better understand it we will use the example we studied in the chapter of CA (table 6). In this case we have three items and three bidders claiming them in a CA. Therefore, our algorithm calculates all possible local packages that can result from 3 items (table 15). Then the local WDP comes to solve for all local packages. To achieve this, we design an auxiliary table for each local package which contains binary variables x ~ [0,1] where with 1 we symbolize the winning bid, while with 0 the lost bids. Consequently, by applying the heap algorithm and the custom made one in the auxiliary table, we find (table 16) all the possible distributions (under the two restrictions mentioned earlier) for each local package and at the same time we calculate the seller's revenue for each distribution. After these calculations are made, we choose as the winning distribution for each local package the one that offers the highest revenue. After the end of these calculations for all local packages we end up with a table that contains the winning distributions for each of the local packages (table 17). From this point on the process is simple as GCAS chooses to distribute the items to the winners of the local package who offer the seller the highest revenue. In cases of tie in revenues we set the rule 'fair lottery' according to which we run a draw with equal chances for the draws through which the winner emerges.

Table 15: Local packages for 3 items
[[A, B, C]]
[[A], [B, C]]
[[A, B], [C]]
[[A, C], [B]]
[[A], [B], [C]]

The important thing we have achieved through this algorithm is to offer a generalized efficient and automatic mechanism for calculating the winners of a CA for N items/securities with K number of bidders.

Table 16:All Possible Auxiliary Matrices

[[A, B, C]]

items	$b_1(K)$	$b_2(K)$	$b_3(K)$
ABC	X_{abc1}	X_{abc2}	X_{abc3}
	1	0	0
	0	1	0
	0	0	1

[[A, C], [B]]

items	$b_1(K)$	$b_2(K)$	$b_3(K)$
AC	X_{ac1}	X_{ac2}	X_{ac3}
B	X_{b1}	X_{b2}	X_{b3}
	1	0	0
	0	1	0
	1	0	0
	0	0	1
	0	1	0
	1	0	0
	0	0	1
	1	0	0
	0	1	0
	0	0	1
	0	0	1
	0	1	0

[[A, B], [C]]

items	$b_1(K)$	$b_2(K)$	$b_3(K)$
AB	X_{ab1}	X_{ab2}	X_{ab3}
C	X_{c1}	X_{c2}	X_{c3}
	1	0	0
	0	1	0
	1	0	0
	0	0	1
	0	1	0
	1	0	0
	0	0	1
	1	0	0
	0	1	0
	0	0	1
	0	0	1
	0	1	0

[[B, C], [A]]

items	$b_1(K)$	$b_2(K)$	$b_3(K)$
BC	X_{bc1}	X_{bc2}	X_{bc3}
A	X_{a1}	X_{a2}	X_{a3}
	1	0	0
	0	1	0
	1	0	0
	0	0	1
	0	1	0
	1	0	0
	0	0	1
	1	0	0
	0	1	0
	0	0	1
	0	0	1
	0	1	0

[[A], [B], [C]]

items	$b_1(K)$	$b_2(K)$	$b_3(K)$
A	X_{a1}	X_{a2}	X_{a3}
B	X_{b1}	X_{b2}	X_{b3}
C	X_{c1}	X_{c2}	X_{c3}
	1	0	0
	0	1	0
	0	0	1
	1	0	0
	0	0	1
	0	1	0
	0	1	0
	1	0	0
	0	0	1
	0	0	1
	1	0	0
	0	1	0
	0	1	0
	0	0	1
	1	0	0
	0	0	1
	0	1	0
	1	0	0

$X_{i,j} \sim [0,1]$

1: winning bid

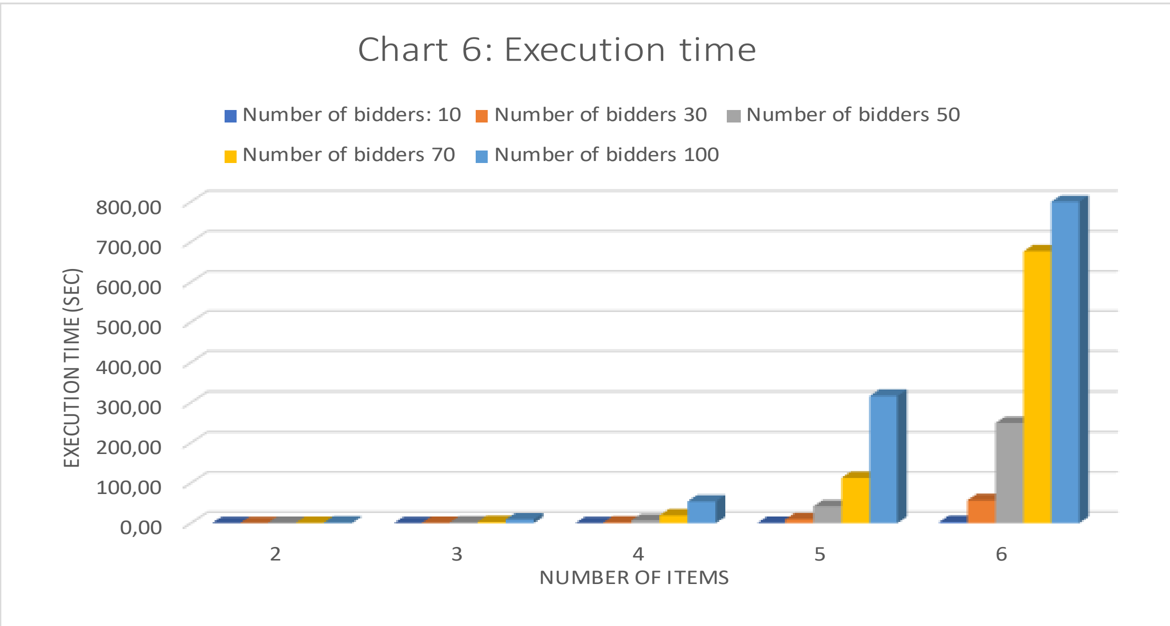
0: lost bid

Table 17: Final local WDPs and the best WDP.

Local Packages	Local Winners	Seller's Revenue
[[A, B, C]]	b_3[A, B, C]	800
[[A], [B, C]]	b_1[A]+b_2[B,C]	750
[[A, B], [C]]	b_3[C]+b_1[A,B]	850
[[A, C], [B]]	b_2[B]+b_3[A,C]	975
[[A], [B], [C]]	**b_1[A]+b_2[B]+b_3[C]**	**1350**

Simulation and Efficiency

Having the integrated GCAS, we perform several simulations to record the efficiency of the algorithm and the time it takes to perform the calculations. The machine in which the simulations have been done runs Windows 10 - 64-b, with CPU: AMD Ryzen 5 5600X 3.7 GHz and RAM: 32G, while the algorithm is written in the programming language Python v3.9. Below we have the results from the simulations that were performed (Chart 6).



From the simulation data we observe that the GCAS algorithm is very efficient for cases up to 6 or 7 items, while for a larger number it consumes more time to do all the calculations. This is something we expected as we knew that the complexity of the WDP solution increases exponentially as the items for auction increase. However, there is room for improvement in the efficiency of our algorithm in various ways. Initially a more powerful machine would improve the computation time of the data, secondly a more efficient syntax of the algorithm could achieve better results and thirdly a conversion of the application into a multiprocessing algorithm would significantly improve its efficiency. We are looking forward to improving it in future papers.

Conclusions

In this dissertation we first presented the theory of auctions and then described the revolutionary blockchain technology in order to examine their coexistence. We have studied through other relevant works the benefits that this technology can offer in the auction process and how its basic features can provide solutions to the problems that emerge from the traditional centralized form of conducting auctions. Finally, we presented a complete double auction process applied on a Blockchain network and based on the pillars of this application we built a new complex combinatorial auction model where we proposed our own effective algorithm the General Combinatorial Auction Solver (G.C.A.S). This study reveals the fact that a cleverly designed blockchain is the technology that can solve many of the problems of auctions and contribute to all active participants. In future works we intend to complete this project with a fully-featured blockchain code in which the existing GCAS will have an important role.

Appendix 1 ICOs [27]

ICO (initial coin offering) is the corresponding method of IPOs (initial public offering) in the cryptocurrencies sector. The developers/companies of a project, in order to have access to financing in order to carry out a project, address the interested investors. They usually publish a white paper revealing the plan and course of their project and announce how much money is required to achieve it along with a roadmap. Although ICOs are very similar to the known IPOs, they differ quite a bit in some key points. Initially ICOs are observably very unregulated in contrast to IPOs located within the securities market. Although the SEC (securities exchanges commission) has suggested that all ICOs are considered securities and should be accounted for as such and therefore subject to the same regulations. In fact, we recently heard the lawsuit filed by the SEC against the company Ripple, which is the creator of XRP cryptocurrency [25]. ICOs still have more freedom in terms of their structure due to the lack of institutional framework but also due to the decentralized character that most of them have. An additional feature that ICOs have and differs from IPOs is the fact that they are usually offered for an already widely accepted cryptocurrency (Bitcoin, Ethereum, etc.) and not for fiat currency like IPOs. At the same time, payouts are usually made in a new cryptocurrency which is usually the token with which one will be able to access their project. When the project is ready to operate the token appears in various exchanges and the initial investors have the opportunity to sell it with potential profits.

Appendix 2 CCPs [4],[8]

There have been several demands for securities trading to take place through a central counterparty, such as the European central counterparty, since the global recession. The CCP fulfills the role of central clearing for securities, but also serves as a trustee for transactions (clearinghouse). After a deal has been completed, the counterparty sets itself up between sellers and buyers. As a consequence, the central counterparty, clears his transaction with the seller, and at the same moment does the same thing with the purchaser, therefore there is no actual contractual relation involving sellers and buyers. A central counterparty also covers against the danger of default one of the trading parties in its capacity as a trustee and then meets the responsibilities of the trade settlement (clearing open positions). One of the activities of the CCP is to replace the position of a party in case it goes bankrupt by selling its positions through an auction after first hedging them. With the introduction of CCPs, buyers and sellers of securities are free from the risk of bankruptcy by their counterparty. Nevertheless, the risk is transferred to the CCP where risk management is crucial as the consequences of inefficient management and the consequent collapse of such a large organization can be catastrophic and affect society as a whole.

Appendix 3 code

Below is the main algorithm written in Python.

```
import numpy as np
from functions import *
from time import process_time

if __name__ == '__main__':
    t = process_time()
    N = validate('int', 'Number of items: ', 'Please give an integer!')
    combinations = getCombinations(range(1, N + 1))
    bidders = validate('int', 'Number of bidders: ', 'Please give an inte
ger!')
    bids = np.zeros((len(combinations), bidders))
    for bIndex, column in enumerate(bids.T):
        print(f'Bidder {bIndex + 1}')
        for i in range(len(column)):
            bid = validate('float', f'Combination {combinations[i]}: ', '
Only numbers!')
```

```
            column[i] = bid

    WDPCombinationsMax = []
    subset = list(range(1, N + 1))
    for WDPCombination in getWDPCombinations(subset):
        combinationList = []
        for value in WDPCombination:
            for cIndex, cValue in enumerate(combinations):
                if value == cValue:
                    combinationList.append(bids[cIndex])

        combinationArray = np.asarray(combinationList)
        if combinationArray.shape[0] == combinationArray.shape[1]:
            permutations = hpArrays(combinationArray.shape[0])
            perms = {}
            for pIndex, permutation in enumerate(permutations):
                permutationSum = 0
                perms[pIndex] = {'details': []}
                for i in range(permutation.shape[0]):
                    for j in range(permutation.shape[1]):
                        if permutation[i][j] == 1:
                            perms[pIndex]['details'].append({
                                'bidder': j + 1,
                                'combination': WDPCombination[i]
                            })
                            permutationSum += combinationArray[i][j]

                    perms[pIndex]['sum'] = permutationSum

            # Find max
            maxSum = -1
            maxIndex = 0
            for index, value in enumerate(perms):
                if perms[value]['sum'] > maxSum:
                    maxSum = perms[value]['sum']
                    maxIndex = index

            WDPCombinationsMax.append(perms[maxIndex])
        elif len(combinationArray) != 1:
            permutations = customPermutation(combinationArray.shape[0], c
ombinationArray.shape[1])
            perms = {}
            for pIndex, permutation in enumerate(permutations):
                permutationSum = 0
                perms[pIndex] = {'details': []}
                for i in range(permutation.shape[0]):
                    for j in range(permutation.shape[1]):
                        if permutation[i][j] == 1:
```

```
                            perms[pIndex]['details'].append({
                                'bidder': j + 1,
                                'combination': WDPCombination[i]
                            })
                            permutationSum += combinationArray[i][j]

                perms[pIndex]['sum'] = permutationSum



            # Find max
            maxSum = -1
            maxIndex = 0
            for index, value in enumerate(perms):
                if perms[value]['sum'] > maxSum:
                    maxSum = perms[value]['sum']
                    maxIndex = index

            WDPCombinationsMax.append(perms[maxIndex])
        else:
            perms = {'details': [{
                'bidder': combinationArray.argmax() + 1,
                'combination': WDPCombination
            }], 'sum': max(combinationArray[0])}

            WDPCombinationsMax.append(perms)

# Find final max
    maxSum = -1
    maxIndex = 0
    for index, combination in enumerate(WDPCombinationsMax):
        if combination['sum'] > maxSum:
            maxSum = combination['sum']
            maxIndex = index

    finalMax = WDPCombinationsMax[maxIndex]
    print('\nResults:')
    for value in finalMax['details']:
        print(f"Bidder {value['bidder']} with combination: {value['combin
ation']}")
    print(f"Revenue: {finalMax['sum']}")
    print(f'Execution time: {process_time() - t}')
```

The functions we used in the main part

```
import numpy as np
import itertools
```

```python
def getWDPCombinations(collection):
    if len(collection) == 1:
        yield [collection]
        return

    first = collection[0]
    for less in getWDPCombinations(collection[1:]):
        for k, subset in enumerate(less):
            yield less[:k] + [[first] + subset] + less[k + 1:]
        yield [[first]] + less


def getCombinations(items):
    result = [[]]
    for item in items:
        subsets = [subset + [item] for subset in result]
        result.extend(subsets)

    result.pop(0)
    return sorted(result, key=len)


def validate(inputType, inputMessage, errorMessage):
    if inputType == 'int':
        while True:
            try:
                return int(input(inputMessage))
            except ValueError as e:
                print(errorMessage)
    else:
        while True:
            try:
                return float(input(inputMessage))
            except ValueError as e:
                print(errorMessage)


def createArray(a, n):
    array = np.zeros((n, n))
    for i in range(n):
        for j in range(n):
            array[i][j] = a[i, j]

    return array


global heapList
```

```python
heapList = []


def heapPermutation(a, size, n, I):
    global heapList
    if size == 1:
        heapList.append(createArray(I[a], n))
        return

    for i in range(size):
        heapPermutation(a, size - 1, n, I)
        if size & 1:
            a[0], a[size - 1] = a[size - 1], a[0]
        else:
            a[i], a[size - 1] = a[size - 1], a[i]


# Based on Heap's algorithm -
> https://en.wikipedia.org/wiki/Heap%27s_algorithm
def hpArrays(n):
    I = np.identity(n, dtype=int)
    a = np.arange(n)
    heapPermutation(a, n, n, I)
    return np.asarray(heapList)

# Custom permutation function for nxm array
def customPermutation(n, m):
    results = []
    a = np.zeros((n, m))
    for k in range(a.shape[0]):
        for h in range(a.shape[1]):
            a[k][h] = 1
            for i in range(a.shape[0] - 1, -1, -1):
                for j in range(a.shape[1] - 1, -1, -1):
                    if np.sum(a[i]) < 1 and np.sum(a[:, j]) < 1:
                        a[i][j] = 1
                        results.append(a.tolist())
                        a = np.zeros((n, m))
                        a[k][h] = 1

            a = np.zeros((n, m))

    results.sort()
    arrays = []
    for result in list(k for k, _ in itertools.groupby(results)):
        arrays.append(np.asarray(result))

    return np.asarray(arrays)
```

REFERENCES

[1] Asunción Mochón, Yago Sáez (2015), “Understanding Auctions”, Springer Texts in Business and Economics.

[2] Amitai Porat, Avneesh Pratap, Parth Shah, Vinit Adkar (2018), “Blockchain Consensus: An analysis of Proof-of-Work and its applications.”, Stanford, http://www.scs.stanford.edu/17au-cs244b/labs/projects/porat_pratap_shah_adkar.pdf

[3] Andreas M. Antonopoulos, (2016), “The Internet of Money Volume One”, 29, 86 USA: Merkle Bloom LLC.

[4] Bank for International Settlements and International Organization of Securities Commissions (2020), “Central counterparty default management auctions – Issues for consideration”

[5] Chiara Braghin, Stelvio Cimato, Ernesto Damiani and Michael Baronchelli (2020) “Designing Smart-Contract Based Auctions”, In: Yang CN., Peng SL., Jain L. (eds) Security with Intelligent Computing and Big-data Services. SICBS 2018. Advances in Intelligent Systems and Computing, vol 895. Springer, Cham. https://doi.org/10.1007/978-3-030-16946-6_5

[6] Chin-Chen Chang, Ya-Fen Chang, (2003), "Efficient anonymous auction protocols with freewheeling bids", Computers & Security, 22(8), 728–734., https://doi.org/10.1016/s0167-4048(03)00013-0

[7] Data: “Bitcoin energy consumption relative to selected countries in 2020” “https://www.statista.com/statistics/881522/bitcoin-energy-consumption-relative-to-select-countries/”

[8] European Central Bank, (2007), “THE ROLE OF CENTRAL COUNTERPARTIES JULY 2007”

[9] Fahad Saleh, (2020) “Blockchain without Waste: Proof-of-Stake, The Review of Financial Studies”, hhaa075, https://doi.org/10.1093/rfs/hhaa075

[10] Han-Lim Choi, Luc Brunet, Jonathan P. How, (2009) “Consensus-Based Decentralized Auctions for Robust Task Allocation”, IEEE Transactions on Robotics, 25(4), 912–926. https://doi.org/10.1109/tro.2009.2022423

[11] Heap's algorithm “https://en.wikipedia.org/wiki/Heap%27s_algorithm”

[12] Hossein Kakavand, Nicolette Kost De Sevres, (2017), “The Blockchain Revolution: An Analysis of Regulation and Technology Related to Distributed Ledger Technologies”, Available at SSRN: https://ssrn.com/abstract=2849251 or http://dx.doi.org/10.2139/ssrn.2849251

[13] Imran Bashir, (2018), “Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition”, Packt Publishing.

[14] Joseph Poon, Thaddeus Dryja, (2016), “The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments”, http://lightning.network/docs/

[15] Jung-San Lee, Chit-Jie Chew, Ying-Chin Chen, Kuo-Jui Wei, (2020), “Preserving Liberty and Fairness in Combinatorial Double Auction Games Based on Blockchain”, IEEE Systems Journal, 1–11. https://doi.org/10.1109/jsyst.2020.3027948

[16] Kazue Sako, (2000), "An Auction Protocol Which Hides Bids of Losers." In: Imai H., Zheng Y. (eds) Public Key Cryptography. PKC 2000. Lecture Notes in Computer Science, vol 1751. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-46588-1_28
[17] Leslie Lamport Robert Shostak Marshall Pease, (1982), "The Byzantine Generals Problem", ACM Transactions on Programming Languages and Systems July 1982, pp. 382-401
[18] LI-HSING YEN AND GUANG-HONG SUN (2019), "Decentralized Auctioneerless Combinatorial Auctions for Multi-Unit Resource Allocation", in IEEE Access, vol. 7, pp. 78625-78639, 2019, doi: 10.1109/ACCESS.2019.2922688.
[19] Malamas, Vangelis and Dasaklis, Thomas and Arakelian, Veni and Chondrokoukis, Gregory, (2020) "A Block-Chain Framework for Increased Trust in Green Bonds Issuance"., SSRN Electronic Journal, 1–20. https://doi.org/10.2139/ssrn.3693638
[20] Paul Klemperer, (2004), "Auctions: Theory and Practice (The Toulouse Lectures in Economics)", Princeton University Press.
[21] Peter Cramton, Yoav Shoham, and Richard Steinberg, (2006), "Combinatorial Auctions", MIT Press, Cambridge, Massachusetts, London, England, 2006, ISBN 0-262-03342-9.
[22] Satoshi Nakamoto, (2008), "Bitcoin: A Peer-to-Peer Electronic Cash System", https://bitcoin.org/bitcoin.pdf
[23] Soohyeong Kim, Yongseok Kwon, Sunghyun Cho, (2018), "A Survey of Scalability Solutions on Blockchain", International Conference on Information and Communication Technology Convergence (ICTC), Jeju, 2018, pp. 1204-1207, doi: 10.1109/ICTC.2018.8539529.
[24] Trevathan J., Read W. and Ghodosi H. (2005). "DESIGN ISSUES FOR ELECTRONIC AUCTIONS" In Proceedings of the Second International Conference on e-Business and Telecommunication Networks, pages 340-347
[25] U.S. SECURITIES AND EXCHANGE COMMISSION, (2020) "SEC Charges Ripple and Two Executives with Conducting $1.3 Billion Unregistered Securities Offering" https://www.sec.gov/news/press-release/2020-338
[26] WISE Token Teal Paper, (2020), https://wisetoken.net/teal/index.html
[27] Yan Chen, (2018), "Blockchain tokens and the potential democratization of entrepreneurship and innovation", Business Horizons, Volume 61, Issue 4, Pages 567 575, ISSN 0007-6813, https://doi.org/10.1016/j.bushor.2018.03.006.
[28] Yi-Hui Chen, Shih-Hsin Chen, Iuon-Chang Lin, (2018), "Blockchain based Smart Contract for Bidding System", IEEE International Conference on Applied System Invention (ICASI), Chiba, pp. 208-211, doi: 10.1109/ICASI.2018.8394569.
[29] Yli-Huumo J, Ko D, Choi S, Park S, Smolander K (2016) "Where Is Current Research on Blockchain Technology? — A Systematic Review", PLoS ONE 11(10): e0163477. https://doi.org/10.1371/journal.pone.0163477.