Enhancing Trust in Trust Services: Towards an Intelligent Human-input-based Blockchain Oracle (IHiBO)

Liuwen Yu*
University of Luxembourg
liuwen.yu@uni.lu

Mirko Zichichi*
Universidad Politécnica de Madrid
mirko.zichichi@upm.es

Réka Markovich
University of Luxembourg
reka.markovich@uni.lu

Amro Najjar
University of Luxembourg
amro.najjar@uni.lu

Abstract

As their name suggests, trust is of crucial importance in "trust service". Nevertheless, in many cases, these services suffer from a lack transparency, documentation, traceability, and inclusive multi-lateral decision-making mechanisms. To overcome these challenges, in this paper we propose an integrated framework which incorporates formal argumentation and negotiation within a blockchain environment to make the decision-making processes of fund management transparent and traceable. We introduce three possible architectures and we evaluate and compare them considering different technical, financial, and legal aspects.

1. Introduction

It is a commonplace that trust has a special importance in entering into contractual relations. There is a domain in which the importance of trust is so crucial that the whole type of service is named after it, i.e. trust services, where the fund managers are in the position of a fiduciary acting on behalf of the principals. The whole service the fiduciary provides is subject to the overall duty to act in the best interest of the client. The legislator can (and does[1]) declare the principal's right to check the fiduciary's relevant activities in order to give some weight to this duty by its intended controlability. On one hand, though, most probably there is a difference between the principal's and the fund managers' expertise and overview giving the very reason to enter in such a relationship, and on the other hand, the lack of the decision making processes being documented might limit the transparency one can gain by practicing this right.

While trust companies might rely on smart contracts when engaging in their core activities in the securities market—as suggested by scholars [1, 2] and proven by the surge of Decentralized Finance (DeFi) [3]—the involvement of Distributed Ledger Technologies (DLTs) for the securities transactions does not address the possible trust issues between the principal and the fiduciary: the former does not have access to the reason why the transaction took place and whether it was really in his interest. To this regard, trust can be understood as a relational attribute between a social actor and other actor and/or institutions, as in [4], but also as a technique for dealing with uncertainty about the actions and communications of other parties, as in [5].

Since the process to make decisions from incomplete and inconsistent information—both in general and in the fund management use case—is a complicated process which may involve different parties with their own interests, we argue that a reasoning system can be combined with DLTs, for making these decisions featured with auditability, transparency, traceability and explainability. In this paper, we address such a gap taking into consideration several aspects influencing such a situation. Regarding the fund managers' decision-making process about investing the principal's money, we argue that formal argumentation can help explain why a claim or a decision is made. In fact, argumentation and trust share a common function: they both deal with change and uncertainty in complex social environment [6]. Then, for enabling conflict-resolution between parties, negotiation can be used to determine the quantities, investment timing or other activities. Information incorporating the different fund managers' opinions is provided by: argumentation, e.g. to decide

*This work has received funding from the EU H2020 research and innovation programme under the MSCA ITN European Joint Doctorate grant agreement No 814177 LAST-JD-RIoE.

[1]For instance, the 6:315. § of the Hungarian Civil Code (Act V of 2013) says: *The principal and the beneficiary shall have the right to check the fiduciary's activities relating to asset management.*

whether to buy, sell or hold securities, and negotiation, e.g. to determine the quantities and investment timing.

The second set of aspects we take in account consists of the implication of the use of oracles and smart contracts, the verifyability and costs of the used technologies, and the extent and limits of the transparency to gain. Integrating formal argumentation and multi-agent negotiation for creating the proper external input triggering the transaction's smart contract leads us to framework we call Intelligent Human-input-based Blockchain Oracle (IHiBO).

The remainder of this paper is organized as follows. In Section 2 we provide the motivation and a use case for our work, while in Section 3 the background concepts. Section 4 has the purpose of providing an overview of formal argumentation and negotiation, while in Section 5 we specify the possible architectures needed for our solution. Discussions and conclusions are provided in Section 6.

2. Motivation

In this section, we describe the ecosystem that we intend to take as an example as a use case for the entire remainder of the paper, namely portfolio management (for the securities market), and delve into the roles of the parties and their relationship, especially that of the fund manager(s) and the fund management process. Fund managers play an important role in the investment and financial world, as they provide investors with the peace of mind that their money is in the hands of an expert [7]. However, reality is not always as hoped and investors tend to know but do not actually know where their money is going, why, and how much is the true profit. A possible simplified process of fund investment management includes the following activities, as Figure 1 shows.

In portfolio management, the core duties of fund managers under AIFMD[2] and UCITSD[3] is to perform portfolio and risk management on behalf of their investors The process of portfolio management on the manager side is formally defined as follows [8]: *a dynamic decision process, whereby a business's list of active new product (and development) projects is constantly up-dated and revised. In this process, new projects are evaluated, selected and prioritized; existing projects may be accelerated, killed or de-prioritized;*

[2]Directive 2011/61/EU of the European Parliament and of the Council of 8 June 2011 on Alternative Investment Fund Managers (AIFMD.http://data.europa.eu/eli/dir/2011/61/oj)

[3]Directive 2009/65/EC of the European Parliament and of the Council of 13 July 2009 on the coordination of laws, regulations and administrative provisions relating to undertakings for collective investment in transferable securities (UCITS). http://data.europa.eu/eli/dir/2009/65/oj

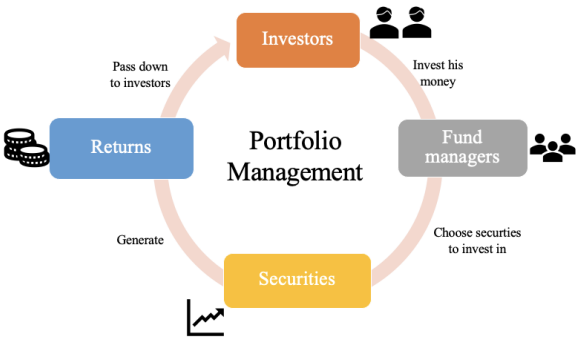

Figure 1. Fund Investment Process

and resources are allocated and re-allocated to active projects. The portfolio decision process is characterized by uncertain and changing information, dynamic opportunities, multiple goals and strategic considerations, interdependence among projects, and multiple decision-makers and locations.

The fund can be managed by one person, by two people as co-managers, or by a team of three or more people. Fund managers primarily research and determine the best stocks, bonds, or other securities to fit the strategy of the fund, then buy and sell them. Since the fund managers are responsible for the success of the fund, they must also research companies, and study the financial industry and the economy. Keeping up to date on trends in the industry helps the fund managers make key decisions that are consistent with the fund's goals. Typically, analysts assist fund managers with individual research on investment ideas and subsequent buy, sell, or hold opinions (or several managers work together). The main feature of investing in a fund is entrusting investment management decisions to the professionals who maintain it.

3. Blockchain and Financial Agreements

In this section, we outline the potential of DLTs to revolutionize financial agreements and a particular instance of how fund managers trade securities on behalf of their clients on blockchain platform. We firstly introduce the backgound of DLTs, blockchain, smart contracts and oracles which are indispensable components in the paradigm of the decentralized financial market infrastructures and also in our following sections where we propose the framework for developing fund management. The overall discussion of blockchain-based securities market is out of the scope of this work.

3.1. Blockchain, Smart contracts and Oracles

The blockchain is part of the realm of Distributed Ledger Technologies (DLTs) which consists of a network of nodes that maintain a ledger by following the same protocol. In the case of the blockchain, the ledger is organized into chronologically ordered blocks where each block is sequentially linked to the previous one [9]. When the majority of network nodes execute the exact same protocol, such as in the Bitcoin network [9], the blockchain is cryptographically guaranteed to be tamper-proof and unforgeable, and this allows to create a trust mechanisms for multiple users in a distributed environment, without the need for third party intermediaries [4].

A feature that some DLTs enable is the possibility to execute smart contracts, firstly introduced by the Ethereum blockchain [10]. Smart contracts consist of instructions that, once deployed on the ledger, cannot be altered and thus allowing the outcome of their execution to be always the same for anyone who runs it (i.e. the DLT network nodes). This enable the realization of a wide range of applications far beyond cryptocurrency transactions [11, 12, 13]. Usually, the possible instructions of a smart contract are embedded in the DLT protocol and their execution can only involve data coming from other smart contracts or from the user's inputs, e.g. smart contracts cannot fetch a webpage on the Internet. This "closure" ensures the execution of smart contracts to be more resistant to attacks with higher degree of certainty, thus making the whole system more secure [12]. This obviously limits the possibility usage of these technologies, since the vast majority of the possible smart contract applications would require real time information from the network external world.

In order for smart contracts to operate in the real world, data must flow in both directions and thus the high demand of applications gave birth to blockcahin oracles. These third-party systems act as a bridge that connects the DLT network and the "outside" world, providing the ability to retrieve, verify and digest the data into smart contracts. Oracles can be implemented as [14]: (i) software, that interact with the information needed from online sources; (ii) hardware, retrieve data from physical world directly through scanners sensors; (iii) human, interacting with individuals. In all cases their off-chain execution is either centralized, i.e. coming from a single source, or decentralized, consensus-based multitude of sources.

3.2. Decentralized Financial Market Infrastructures

Often, when talking about financial markets, one immediately visualizes a centrally organized institution being part of a monolithic paradigm that has been established over the years. It is therefore difficult to deviate from a logic so rooted in the social structure as the market centralization. The advent of DLT, however, seems to be able to restructure this paradigm by breaking the stigma, only apparently immutable, of centrality and of central counterparties (CCPs) [15, 1]. Decentralized Financial Market Infrastructures (dFMI) [1]: *are consortium entities whose members are comprised of the main participants in a market, organized in a peer-to-peer model, which is governed by dFMI participants themselves rather than a central intermediary. [...] dFMIs can fuse together the advantages of a decentralised market structure with the functions of CCPs, such that the public confidence in CCP capabilities can be met with the right alignment of interests. Investors dependent on the proper discharge of CCP functions would ultimately assume a level of risk that more accurately accords to their level of risk aversion.*

For instance, in some applications smart contracts can take on a role similar to that previously played CCPs, e.g. acting as a margin calculating agent and taking on the task of transferring collateral. Although in a different way, the smart contract can be used to resolve disputes in the event of non-compliance with payment [16]. Alternatively, smart contracts can support the central counterparty, which can maintain the business model by leveraging the blockchain to calculate and update collateral as well as manage funds, thus relying on financial cryptography. A concrete application of DLTs for the trading of securities by fund managers is Lianjiaorong, a blockchain AssetBacked securitization platform, built by the Bank of Communications in China [17]. The blockchain is maintained by original stock holders, trust companies, investors, rating agencies, accountants, lawyers, regulators and it links funds and assets on the ledger, realizing the credit penetration of the securities business system.

Currently, a new infrastructure is taking hold that is not completely tied to traditional finance, but is entirely built using DLTs. Decentralized finance (DeFi) is a novel P2P financial infrastructure, based on smart contracts, that provides non-custodial, permissionless, openly verifiable and composable operations [3]. With DeFi protocols such as Decentralized Exchanges (DEX), anyone can engage in non-custodial exchange

of on-chain digital assets, e.g. tokens. It is worth noting, in this case, that there are already some DeFi oracles are used in practice. MakerDAO is the first and the most popular blockchain-based protocol to launch a major automated cryptocurrency-lending platform. In the Maker Protocol, each collateral type has a corresponding oracle that feeds real-time price to the aggregator, such data is from a number of independent feeds that consist of individuals and organizations, and the aggregator then to calculate the median price as a reference price that the system uses [18]. Compound is a blockchain-based platform where the users are able to earn interest by lending their cryptoassets, it also gathers price from multiple sources for aggregating the median one.

4. Formal Argumentation and Negotiation

In previous sections, we discussed how fund managers can directly execute securities transactions without providing clients with information about how and why they make investment plans and actions. Moreover, we also know that various managers may have different investment plans based on their research that may conflict with each other. In this section, we present the background knowledge that can lead to enable fund managers to be more transparent in their decisions and to be able to resolve conflicts through formal argumentation and negotiation.

Formal argumentation or computational argumentation in Artificial Intelligence (AI) is a formalism for representing and reasoning with incomplete and inconsistent information. A wide variety of reasoning and dialogical activities can be captured by argumentation models in a formal and still quite intuitive way, allowing the integration of different concrete techniques and the development of applications that humans can trust. Dung's work in 1995 illustrates an argumentation system consisting of a set of arguments and the relation (attacks) between them [19]. Formal argumentation also can be used for modeling the dynamic interactions among agents which is particularly at stake in a multi-agent context: the system evolves as the agents put forward new arguments or retract arguments and relations [20]. There are variants of Dung's original framework, extending the theory with preference [21], support [22, 23], etc. In this section we use agent abstract argumentation which is introduced in one of the authors' latest work [24], and autonomous negotiation for dealing with conflicting information raised by agents.

4.1. Agent Argumentation

We first generalize argumentation frameworks studied by Dung [25].

Definition 1 (Argumentation framework) *An* argumentation framework *(AF) is a pair* $\langle \mathscr{A}, \rightarrow \rangle$ *where* $\mathscr{A}$ *is a set called arguments, and* $\rightarrow \subseteq \mathscr{A} \times \mathscr{A}$ *is a binary relation over* $\mathscr{A}$ *called attack.*

Dung's admissibility-based semantics is based on the concept of defense. A set of arguments defends another argument if they attack all its attackers.

Definition 2 (Admissibility) *Let* $\langle \mathscr{A}, \rightarrow \rangle$ *be an AF.* $E \subseteq \mathscr{A}$ *is* conflict-free *iff there are no arguments a and b in E such that a attacks b.* $E \subseteq \mathscr{A}$ defends *c iff for all arguments b attacking c, there is an argument a in E such that a attacks b.* $E \subseteq \mathscr{A}$ *is* admissible *iff it is conflict-free and defends all its elements.*

Baroni and Giacomin then define semantics as a function from argumentation frameworks to sets of subsets of arguments [26].

Definition 3 (Dung semantics) *A Dung semantics is a function* σ *that associates with an argumentation framework* $AF = \langle \mathscr{A}, \rightarrow \rangle$*, a set of subsets of* $\mathscr{A}$*, the elements of* $\sigma(AF)$ *are called extensions.*

Dung distinguishes several definitions of extension.

Definition 4 (Extensions) *Let* $\langle \mathscr{A}, \rightarrow \rangle$ *be an AF.* $E \subseteq \mathscr{A}$ *is a* complete extension *iff it is admissible and it contains all arguments it defends, i.e.,* $E = \{a | E \text{ defends } a\}$. $E \subseteq \mathscr{A}$ *is a* grounded extension *iff it is the smallest (for set inclusion) complete extension.* $E \subseteq \mathscr{A}$ *is a* preferred extension *iff it is a largest (for set inclusion) complete extension.* $E \subseteq \mathscr{A}$ *is a* stable extension *iff it is conflict-free and it attacks each argument which does not belong to E.*

An agent argumentation framework extends an argumentation framework with a set of agents and a relation associating arguments with agents [23]. Note that an argument can belong to one agent or multiple agents.

Definition 5 (Agent argumentation framework) *An* agent argumentation framework *(AAF) is a 4-tuple* $\langle \mathscr{A}, \rightarrow, \mathscr{S}, \sqsubset \rangle$ *where* $\mathscr{A}$ *is a set of arguments,* $\rightarrow \subseteq \mathscr{A} \times \mathscr{A}$ *is a binary relation over* $\mathscr{A}$ *called attack,* $\mathscr{S}$ *is a set of agents or sources,* $\sqsubset \subseteq \mathscr{A} \times \mathscr{S}$ *is a binary relation associating arguments with agents.* $\mathscr{A}_\alpha = \{a \in \mathscr{A} | a \sqsubset \alpha\}$ *for all arguments that belong to agent* α*,* $\mathscr{S}_a = \{\alpha | a \sqsubset \alpha\}$ *for all agents that have argument a.*

For fund management, we use social semantics, which is based on a reduction to preference-based

argumentation by for each argument counting the number of agents that have the argument [21]. It thus interprets agent argumentation as a kind of voting, as studied in social choice theory or judgment aggregation, this is also the most closed to fund management.

We first give the definition of a preference-based argumentation framework [21].

Definition 6 (Preference-based AF) *A preference-based argumentation framework (PAF) is a 3-tuple* $\langle \mathscr{A}, \rightarrow, \succ \rangle$ *where* $\mathscr{A}$ *is a set of arguments,* $\rightarrow \subseteq \mathscr{A} \times \mathscr{A}$ *is a binary attack relation,* $\succ$ *is a partial order (irreflexive and transitive) over* $\mathscr{A}$ *, called preference relation.*

There are different reductions of preference have been introduced [27, 28]. We refer to those papers for an explanation and motivation, and we choose one of the reductions in our use case below which satisfies the essential conflict-free principle analyzed in [23].

Definition 7 (Reductions of PAF to AF (PR)) *Given an* $PAF = \langle \mathscr{A}, \rightarrow, \succ \rangle$*:* $PR(PAF) = \langle \mathscr{A}, \rightarrow' \rangle$*, where* $\rightarrow' = \{a \rightarrow' b | a \rightarrow b, b \not\succ a,$ *or* $b \rightarrow a,$ *not* $a \rightarrow b, a \succ b,$ *or* $a \rightarrow b,$ *not* $b \rightarrow a\}$.

In social agent semantics, an argument is preferred to another argument if it belongs to more agents. The reduction from AAF to PAF is used as an intermediary step for social agent semantics.

Definition 8 (Social Reductions of AAF to PAF (SAP)) *Given an* $AAF = \langle \mathscr{A}, \rightarrow, \mathscr{S}, \sqsubset \rangle$*,* $SAP(AAF) = \langle \mathscr{A}, \rightarrow, \succ \rangle$ *with* $\succ = \{a \succ b | |\mathscr{S}_a| > |\mathscr{S}_b|\}$.

Definition 9 (Social Reductions of AAF to AF (SR)) *Given an* $AAF = \langle \mathscr{A}, \rightarrow, \mathscr{S}, \sqsubset \rangle$*,* $SR_i(AAF) = PR_i(SAP(AAF))$*,* PR_i *is one of the four reductions of PAF to AF, where the semantics* $\delta(AAF) = \sigma(SR_i(AAF)) = \sigma(PR_i(SAP(AAF)))$ *for* $i \in \{1,2,3,4\}$.

4.2. Autonomous Agents and Negotiation

A software agent is a software that acts on behalf of another actor (often a human user) to perform a task or achieve a given goal [29]. Agents are designed to be bound to individual perspectives [30]. This makes agents good candidates to represent the subjectivity and nuances of different expert opinions. Multi-agent systems [31] provide a distributed platform capable of implementing intelligence in decentralized ecosystems such as DLT-based systems where agents are capable, using well-established conflict-resolution mechanisms (e.g. negotiation), of helping the different stakeholders finding agreements that satisfy their often conflicting interests.

In his influential book, Dean Pruitt provides one of negotiation's most widely accepted definitions [32]: *the process by which a joint decision is made by two or more parties. The parties first verbalize contradictory demands and then move towards agreement by a process of concession making or search for new alternatives.* The problem being negotiated, or the topic under discussion (e.g. car purchase) can be usually divided into issues (also called attributes). Some negotiations involve only single issue (e.g. car price) whereas others involve multiple issues (e.g. price and delivery time). Negotiators may not only disagree on the value assigned to each issue, the priority given to each issue can differ from one negotiator to another and hence this can be a source of both divergence and convergence.

Automated negotiation is one taking place among autonomous agents [33]. Autonomous negotiation has a protocol. The latter is the set of rules that governs the interactions during a negotiation session (also called a thread). Whereas the negotiation protocol defines what is the set of possible actions that can be taken during a negotiation session, an agent has a decision model [34, 35] that allows the agent to (i) evaluate the value of an offer received from the opponent (e.g., using a utility function), (ii) decide whether it is acceptable (also called acceptance condition [36]), and (iii) determine what to do next (known as the negotiation strategy [34]). Automated negotiation has been applied to solve conflicts and reach agreements in several domains including cloud and service provisioning [37], smart grid and power distribution [38], and trading and stock market [39].

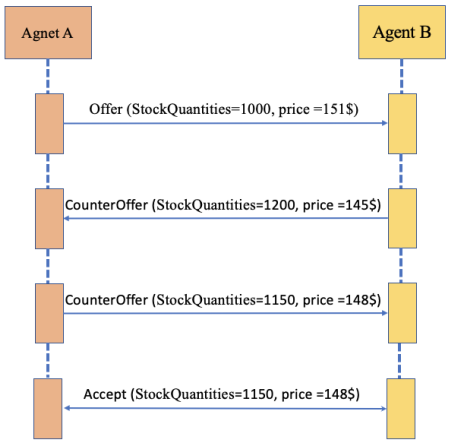

Figure 2. Negotiation Sequence to Decide The Quantities and The Price

5. Intelligent Human-input-based Blockchain Oracle (IHiBO)

In this section, we present how our solution, IHiBO, a framework based on the use of the blockchain and smart contracts can be leveraged to provide the features of treceability and transparency. IHiBO can deal with the potentially inconsistent information input by human experts: we explain how the oracle can manage the information by argumentation and negotiation considering three different architectures.

5.1. Conflict Resolution

As described in Section 2, the process of portfolio management fits well with argumentation theory in AI. The decision can be seen as being based on arguments and counter-arguments. Argumentation, as the result, can be useful for deriving decisions and explaining a choice already made. Managers provide their arguments from their own research to identify promising stocks with different level of accuracy and thereby make different portfolio choices which are likely to be incomplete and inconsistent.

The fictitious simple example (the real life cases would be much more complex) is as follows. Manager α holds the arguments a: *To buy the stocks, since the company just donated to charities that is beneficial to good commercial reputation*, while another two managers β and γ at the same time are against buy the stocks, they hold the same arguments b and c, b is *To sell the stocks, since there is evidence that the leader is under accusations of charity fraud*, and c is *To sell the stocks, since the company has poor sales performance.*

Based on the above, we can build an agent argumentation framework on the left up side of Fig.2, $AAF = \langle \mathscr{A}, \rightarrow, \mathscr{S}, \sqsubset \rangle$ where $\mathscr{A} = \{a,b,c\}$, $\rightarrow = \{(a,b),(b,a),(a,c),(c,a)\}$, $\mathscr{S} = \{\alpha,\beta,\gamma\}$, $\sqsubset = \{(a,\alpha),(b,\beta),(b,\gamma),(c,\beta),(c,\gamma)\}$. Since $|\mathscr{S}_b| > |\mathscr{S}_a|$ and $|\mathscr{S}_c| > |\mathscr{S}_a|$, $b \succ a$ and $c \succ a$, we get the corresponding PAF, and giving the reduction from PAF to AF, we have the only AF on the downside of Fig.2. Then we can calculate the only acceptable set $\{b,c\}$, which is grounded, preferred and stable extension. The set tells the final decision is to sell the stocks. One thing needs to be noticed: argumentation does not always provide a definite outcome. Depending on the decision making process, different protocols can be specified in advance for such cases: e.g. to roll back or to assign weights to the arguments and the relation among them (so that these cannot be always equal).

After deciding to sell the stocks, the next problem is the numbers of stocks to sell and the sell timing. Here

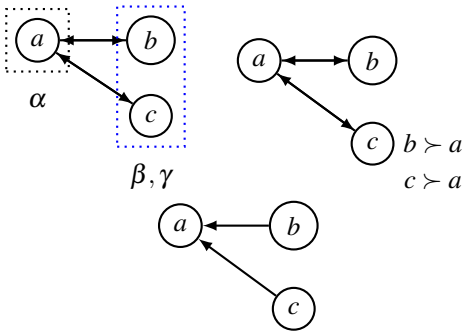

Figure 3. Social reduction

the computational automated negotiation comes into play. To illustrate how it works, we give an example of the negotiation sequence based on the on the quantities of stocks to sell. The negotiation process is based on the alternating offer protocol [40]. Agents can bid new offers to the opponent ($Offer()$ function). When receiving an offer, and agent can accept it using $accept()$ function or reject it and propose a counter-offer (with the $CounterOffer()$ function). In the example, we have a manager A, i.e., agent A, and manager B, i.e., agent B. Agent A proposes to sell 1000 stocks at the price of 151\$, while agent B counteroffers to sell 1200 stocks at the price of 145\$, then agent A proposes to sell 1150 stocks at the price of 148\$. The final offer given by A is accepted by both parties which means they come to an agreement.

5.2. Blockchain Framework

Although formal argumentation has the ability to provide various ways for explaining why a claim or a decision is made, it lacks of the features of auditability, traceability and transparency. In order to gain these features, we propose to integrate formal argumentation with a blockchain and smart contracts framework, which provides a favourable environment with its salient properties. In recent years, the interest in the blockchain and in smart contracts has been growing, as more and more different kind of systems rely on their use to transfer digital assets in a "trustless" way [41].

The main principle here is the fact that the immutability property of DLTs enables a favourable environment for storing information that can be later audited. We assume that the outcome of the negotiation, i.e. the decision, is given in input to a smart contract that will enact an action, e.g. security transaction, buy a stock in the fund management use case. We refer to this smart contract as the TransactionSC, and can be the implementation of any of the use cases shown in Section 3.

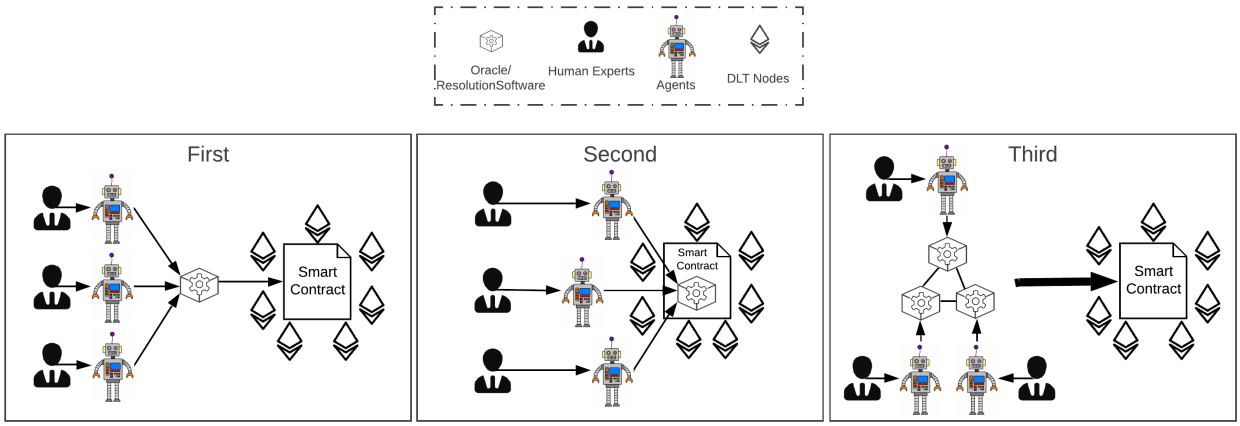

Figure 4. IHiBO with three architectures

Based on the above consideration, we compare three different architectures that can take form in our blockchain framework (Figure 4):

1. **Centralized Oracle** - The first architecture we consider is the simplest one, where argumentation and negotiation phases do not involve any blockchain process, neither a smart contract execution. These are executed in a "centralized" environment, e.g. a web platform or a company application, and then the decision will be given in input to the TransactionSC by a specific service in this environment, providing the role of an oracle.

2. **Smart Contract Argumentation and Negotiation** - In the second architecture, argumentation and negotiation are directly implemented as smart contracts, and thus are executed following the blockchain protocol. It means that the human experts, through their agent software, directly interact with the blockchain for giving in input the data for constructing the argumentation graph and then for enacting the negotiation functions that are expressed as smart contract instructions.

3. **Decentralized Oracle** - Finally, the third architecture we consider consists of a network of agents that execute a distributed software independently of the blockchain protocol and that limit the execution of the smart contract instructions to only a few steps, necessary to be trustworthy. The implementation of such network consists in the so called "layer two" solution [42], where the same principle of decentralization of DLTs is applied. Indeed, an instance of such layer two solution would be another DLTs with different features in respect to the "main" one [43] where to write the negotiation outcome, e.g. consensus mechanism, or faster operations execution.

We take as reference Table 1 for comparing the three architectures. The first discriminator for choosing one architecture over another consists in where the argumentation graph (and all the data needed to the execution) is stored. Such data is needed for the execution of the argumentation and negotiation process, hence is constantly updated. This information is needed to be stored on the ledger only in the case of the second architecture. The drawbacks of storing large quantities of data on-chain are many, above all, the elevated transaction fee cost [11] and the almost impractical upload latency [44]. Furthermore, the fact that instruction execution would be unfavourable compared to the other two, again in terms of fees and latency between operations, should also be taken into account for Architecture 2. However, the advantage of this architecture is that the negotiation execution would be completely traced and verifiable, since the execution would be completely carried out through the smart contracts in the blockchain. On the other hand, the complete execution of the conflict resolution would be poorly verifiable using a classic centralized oracle (Architecture 1), because on the blockchain only the results would be stored. It would also be highly susceptible to single point of failures. A good compromise between the two architectures would be the use of a decentralized oracle, Architecture 3, for executing argumentation, negotiation and TransactionSC interaction. The data needed for carrying out these processes, such as the argumentation graph, would be stored in a secondary DLT with cheaper costs or in another layer two technology that preserves data immutability. The negotiation execution

Table 1. Comparison between the three architectures considered.

	Architecture 1 **Centralized**	Architecture 2 **Smart Contract**	Architecture 3 **Decentralized**
Argumentation Graph On-chain	No	Yes	No
Negotiation Execution Tracing	Poor	Very Good	Good
Verifiability	Poor	Very Good	Very Good
Single Point of Failure	Yes	No	No
Execution Overhead Costs (Latency, Fees)	Very Good	Poor	Good

can happen on off-chain and then be "committed" [42] on the main chain using an hash function in order to be verifiable. It would not be susceptible to single point of failure and the execution overhead cost would be favourable in respect to the second architecture.

Next to the technical and financial aspects, legal considerations should also be taken into account when comparing the different architectures. While our motivation is to provide transparency regarding the decision-making process to the principal to gain some insights whether the work of the fiduciary indeed happens according to his best interest, the transparency one should gain with using DLTs is subject to serious limitations. On one hand, the the principal's right to check is not limitless, it concerns strictly the processes of managing his assets, but more importantly, given the characteristics of DLTs, a(n unwantedly) broader audience would be involved in the disclosure of information if one chose not the appropriate architecture, threatening trade secrets and involving privacy problems. Architecture 3 seems to be the best option from this point of view too: in contrast to the public, permissionless verification that DLTs usually employ while smart contracts are executed, layer two solutions usually move this process off-chain. This definitely poses security issues compared to a protocol executed completely on-chain, however there are currently some viable solutions proposed that address this issue [1]. For instance, an application might be the use of a permissioned sidechain. In this case, information that would clash with trade secrets and privacy would be stored on that permissioned chain and maintained by the participants who have been nominated for this, e.g. joint data controllers as permissioned blockchain operators [45]. Through the use of commitments on the main chain [42], i.e., the permissionless one, the necessary steps for verification are implemented, and once the fiduciaries operating the sidechain reveal part of the information to the principals, the latter can verify its validity on-chain [43]. On the other hand, once the application of DLTs become widespread in the securities market, mandatory disclosure rules motivated by anti-tax avoidance should be aligned with the new technology [2].

5.3. Trust and Transparency

The IHiBO we proposed might have particular relevance in cases where the decision making process about what data should be fed in the smart contract needs to be transparent: for fund management, the investors don't know what exactly happens to their money, and especially why, so the question whether the fund managers do fulfill their legal and ethical commitment of acting in the best interest of the investor might remain unanswered.

In general, the transparency that can be gained due to the proposed intelligent oracle architecture could be highly valuable in any trust services. The concept of the fiduciary is based on—as the name of these services show—trust: it requires being bound both legally and ethically to operate and use its expertise in the investor's best interests on the fiduciary's side, and it requires trust on the investor's side to believe in that the fiduciary has done and will do so. This trust can be, to some extent, replaced by intelligent, decentralized solutions providing full transparency of, for instance, fund management: not only the transactions can be fully traced but the expert opinion input and the decision mechanism too. By implementing argumentation and negotiation phases through oracles into smart contract or make them on a side-chain can generate more transparency for investors: investors can know how the final decision is made at the end of reasoning. This could be highly relevant for the investor practicing his right to check the fiduciary's activities in the case of an asset management contract. From Explainable AI perspective, Architecture 2 and Architecture 3 offer an

explanation to how a specific decisions has been made.

We argue that a layer two solution, the decentralized oracle solution in Architecture 3, provides the proper mid ground in terms of cost of execution, for latency and fees, and verifiability of the complete process. Indeed, there might be use cases where some data should not be disclosed, and an argumentation and negotiation architecture based on a full execution on smart contracts would not allow it. In the other extreme case, for a centralized oracle, the entire process behind a decision made could be concealed or its log could be altered. In a decentralized oracle architecture the complete execution could be logged off-chain and then committed on-chain, making it impossible to alter the logs, while not disclosing these entirely [42].

Members of the management body[4] shall have adequate access to information and documents which are needed to oversee and monitor management decision-making[5]. In our second and third architectures, each execution of all the smart contracts can be audited, validated and maintained by every participants, thus reduce the time and fee of extra work of surveillance, which will in turn reduce potential corruption or conflicts of interests.

6. Conclusion and Future Perspectives

The main contribution of this paper is proposing an integrated framework which incorporates formal argumentation and negotiation within a blockchain environment for making the decision-making processes of fund management transparent and traceable. There is a very broad literature devoted on the notions of trust. In this paper, we do not engage with a unique trust because of its multifaceted and complex nature. However, from an interpersonal perspective, trust inevitably comes along with a certain degree of risk and vulnerability [46], the more familiar users are with the technology, the more willing they are to take the risks. Especially in the case of information and power asymmetries, the fund managers have the ability to act against the interests of the principals. What our proposal can enhance is the transparency and reliability of some business processes which is expected to raise the willingness of entering to these business relations on the principal side.

Our motivation came from trust services, so we explained our idea in a fund management scenario, but our proposal is not bound to this domain. Also, the research on oracles is still in its infant stage, there are multiple pressing questions and challenges for the future. A possible work could be to investigate on the integration of consensus mechanisms for a layer two solution to the dispute resolution phase, in order to narrow the gap between blockchain and argumentation as well as negotiation, since there is no specialized blockchain yet that has a protocol that integrates reasoning. For instance if there is a blockchain based on *Proof of Stake* (instead of *Proof of Work*), validators need to vote to validate a transaction based on a reasoning process where each validator has a different set of knowledge data. Lastly, another follow up work is to explore the more advanced argumentation theory. For instance, probabilistic argumentation captures the quantitative aspects of uncertainties by underlying probabilistic logic which makes it possibly more suitable for picture the financial reasoning.

[4]Art. 4(8) MiFID II: 'management body' means the body or bodies of an investment firm, market operator or data reporting services provider, which are appointed in accordance with national law, which oversee and monitor management decision-making and include persons who effectively direct the business of the entity.

[5]DIRECTIVE 2014/65/EU OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 15 May 2014 on markets in financial instruments and amending Directive 2002/92/EC and Directive 2011/61/EU

References

[1] S. Feenan, D. Heller, A. Lipton, M. Morini, R. Ram, R. Sams, T. Swanson, S. Yong, and D. B. Zalles, "Decentralized financial market infrastructures," *The Journal of FinTech, Forthcoming*, 2020.

[2] M. Thuvarakan, "Regulatory changes for redesigned securities markets with distributed ledger technology," *The Knowledge Engineering Review*, vol. 35, 2020.

[3] S. M. Werner, D. Perez, L. Gudgeon, A. Klages-Mundt, D. Harz, and W. J. Knottenbelt, "Sok: Decentralized finance (defi)," *arXiv preprint arXiv:2101.08778*, 2021.

[4] M. Becker and B. Bodó, "Trust in blockchain-based systems," *Internet Policy Review*, vol. 10, no. 2, 2021.

[5] A. Koster, J. Sabater-Mir, and M. Schorlemmer, "Argumentation and trust," in *Agreement Technologies*, pp. 441–451, Springer, 2013.

[6] F. Paglieri, "Trust, argumentation and technology," *Argument Comput.*, vol. 5, no. 2-3, pp. 119–122, 2014.

[7] P. M. Bosse, D. M. Grim, and C. Frank Chism, "Duty, opportunity, mastery: Investment committee best practices," 2017.

[8] R. G. Cooper, S. J. Edgett, and E. J. Kleinschmidt, "Portfolio management in new product development: Lessons from the leaders—i," *Research-Technology Management*, vol. 40, no. 5, pp. 16–28, 1997.

[9] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[10] V. Buterin *et al.*, "Ethereum white paper," 2013.

[11] Y. Kurt Peker, X. Rodriguez, J. Ericsson, S. J. Lee, and A. J. Perez, "A cost analysis of internet of things sensor data storage on blockchain via smart contracts," *Electronics*, vol. 9, no. 2, p. 244, 2020.

[12] Z. Zheng, S. Xie, H.-N. Dai, W. Chen, X. Chen, J. Weng, and M. Imran, "An overview on smart contracts: Challenges, advances and platforms," *Future Generation Computer Systems*, vol. 105, pp. 475–491, 2020.

[13] M. Zichichi, S. Ferretti, and G. D'angelo, "A framework based on distributed ledger technologies for data management and services in intelligent transportation systems," *IEEE Access*, vol. 8, pp. 100384–100402, 2020.

[14] A. Beniiche, "A study of blockchain oracles," *arXiv preprint arXiv:2004.07140*, 2020.

[15] R. Priem, "Distributed ledger technology for securities clearing and settlement: benefits, risks, and regulatory implications," *Financial Innovation*, vol. 6, no. 1, pp. 1–25, 2020.

[16] M. Morini, "Managing derivatives on a blockchain. a financial market professional implementation," *A Financial Market Professional Implementation (May 5, 2017)*, 2017.

[17] W. Pan and M. Qiu, "Application of blockchain in asset-backed securitization," in *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, pp. 71–76, IEEE, 2020.

[18] B. Liu, P. Szalachowski, and J. Zhou, "A first look into defi oracles," *arXiv preprint arXiv:2005.04377*, 2020.

[19] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," *Artificial intelligence*, vol. 77, no. 2, pp. 321–357, 1995.

[20] B. Liao, L. Jin, and R. C. Koons, "Dynamics of argumentation systems: A division-based method," *Artificial Intelligence*, vol. 175, no. 11, pp. 1790–1814, 2011.

[21] S. Kaci and L. van der Torre, "Preference-based argumentation: Arguments supporting multiple values," *International Journal of Approximate Reasoning*, vol. 48, no. 3, pp. 730–751, 2008.

[22] L. Yu, R. Markovich, and L. Van Der Torre, "Interpretations of support among arguments," in *Legal Knowledge and Information Systems*, pp. 194–203, IOS Press, 2020.

[23] L. Yu and L. van der Torre, "A principle-based approach to bipolar argumentation," in *NMR 2020 Workshop Notes*, p. 227, 2020.

[24] L. Yu, D. Chen, L. Qiao, Y. Shen, and L. van der Torre, "A principle-based analysis of abstract agent argumentation semantics," 2021. under review.

[25] P. M. Dung, "On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games," *Artificial Intelligence*, vol. 77, pp. 321–357, 1995.

[26] P. Baroni and M. Giacomin, "On principle-based evaluation of extension-based argumentation semantics," *Artificial Intelligence*, vol. 171, no. 10-15, pp. 675–700, 2007.

[27] L. Amgoud and S. Vesic, "Rich preference-based argumentation frameworks," *International Journal of Approximate Reasoning*, vol. 55, no. 2, pp. 585–606, 2014.

[28] L. van der Torre and S. Vesic, "The principle-based approach to abstract argumentation semantics," *FLAP*, vol. 4, no. 8, 2017.

[29] M. Wooldridge, *An introduction to multiagent systems*. John wiley & sons, 2009.

[30] K. P. Sycara, "Multiagent systems," *AI magazine*, vol. 19, no. 2, pp. 79–79, 1998.

[31] G. Weiss, *Multiagent Systems*. MIT Press, 2013.

[32] D. G. Pruitt, *Negotiation behavior*. Academic Press, 2013.

[33] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: prospects, methods and challenges," *International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.

[34] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Robotics and Autonomous Systems*, vol. 24, no. 3-4, pp. 159–182, 1998.

[35] A. Najjar, "Multi-agent negotiation for qoe-aware cloud elasticity management," 2015. PhD Thesis.

[36] T. Baarslag, K. Hindriks, and C. Jonker, "Acceptance conditions in automated negotiation," in *Complex Automated Negotiations: Theories, Models, and Software Competitions*, pp. 95–111, Springer, 2013.

[37] A. Najjar, X. Serpaggi, C. Gravier, and O. Boissier, "Multi-agent negotiation for user-centric elasticity management in the cloud," in *2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing*, pp. 357–362, IEEE, 2013.

[38] R. J. Tom, S. Sankaranarayanan, and J. J. Rodrigues, "Agent negotiation in an iot-fog based power distribution system for demand reduction," *Sustainable Energy Technologies and Assessments*, vol. 38, p. 100653, 2020.

[39] M. P. Wellman, A. Greenwald, and P. Stone, *Autonomous bidding agents: Strategies and lessons from the trading agent competition*. Mit Press, 2007.

[40] A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica: Journal of the Econometric Society*, pp. 97–109, 1982.

[41] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," *International Journal of Web and Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.

[42] L. Gudgeon, P. Moreno-Sanchez, S. Roos, P. McCorry, and A. Gervais, "Sok: Layer-two blockchain protocols," in *International Conference on Financial Cryptography and Data Security*, pp. 201–226, Springer, 2020.

[43] A. Singh, K. Click, R. M. Parizi, Q. Zhang, A. Dehghantanha, and K.-K. R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, p. 102471, 2020.

[44] M. Zichichi, S. Ferretti, and G. D'Angelo, "Are Distributed Ledger Technologies Ready for Intelligent Transportation Systems?," in *Proc. of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock 2020), co-located with the 26th Annual International Conference on Mobile Computing and Networking (MobiCom 2020), ACM*, pp. 1–6, ACM, 2020.

[45] T. Lyons, L. Courcelas, and K. Timsit, "Blockchain and the gdpr," in *The European Union Blockchain Observatory and Forum*, 2018.

[46] P. De Filippi, M. Mannan, and W. Reijers, "Blockchain as a confidence machine: The problem of trust & challenges of governance," *Technology in Society*, vol. 62, p. 101284, 2020.