IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 9, 2021, accepted March 17, 2021, date of publication March 23, 2021, date of current version March 30, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3068239

Robonomics in the 6G Era: Playing the Trust Game With On-Chaining Oracles and Persuasive Robots

ABDELJALIL BENIICHE, SAJJAD ROSTAMI, AND MARTIN MAIER
Optical Zeitgeist Laboratory, Institut National de la Recherche Scientifique (INRS), Montréal, QC H5A 1K6, Canada
Corresponding author: Martin Maier (maier@ieee.org)

This work was supported by NSERC Discovery under Grant 2016-04521.

ABSTRACT 6G is anticipated to become more human-centered than 5G and should not only explore more spectrum at high-frequency bands but, more importantly, converge driving technological trends such as blockchain technologies and connected robotics. This paper focuses on the emerging field of *robonomics*, which studies the sociotechnical impact of blockchain technologies on social human-robot interaction and behavioral economics for the social integration of robots into human society. Advanced blockchain technologies such as oracles enable the on-chaining of blockchain-external off-chain information stemming from human users. In doing so, they leverage on human intelligence rather than machine learning only. In this paper, we investigate the widely studied trust game of behavioral economics in a blockchain context, paying close attention to the importance of developing efficient cooperation and coordination technologies. After identifying open research challenges of blockchain-enabled implementations of the trust game, we first develop a smart contract that replaces the experimenter in the middle between trustor and trustee and demonstrate experimentally that a social efficiency of up to 100% can be achieved by using deposits to enhance both trust and trustworthiness. We then present an on-chaining oracle architecture for a networked N-player trust game that involves a third type of human agents called observers, who track the players' investment and reciprocity. The presence of third-party reward and penalty decisions helps raise the average normalized reciprocity above 80%, even without requiring any deposit. Finally, we experimentally demonstrate that mixed logical-affective persuasive strategies for social robots improve the trustees' trustworthiness and reciprocity significantly.

INDEX TERMS 6G, behavioral economics, blockchain on-chaining, networked N-player trust game, oracles, smart contracts, social robots.

I. INTRODUCTION

A major limitation of the conventional blockchain is its inability to interact with the ''outside world'' since smart contracts can only operate on data that is on the blockchain. In the emerging blockchain Internet of Things (B-IoT), sensors are typically deployed to bring sensor measurement data onto the blockchain [1]. Advanced blockchain technologies enable the *on-chaining* of blockchain-external off-chain information stemming also from real users, apart from sensors and other data sources only, thus leveraging also on human intelligence rather than machine learning only. To overcome this limitation, smart contracts may make use of so-called *oracles*, which are trusted decentralized blockchain entities whose primary task is to collect off-chain information and bring it onto the blockchain as trustworthy input data to smart contracts. Several decentralized oracle systems exist that rely on *voting-based games*, e.g., ASTRAEA [2].

Blockchain-external data sources imply the risk that the on-chained data may be unreliable, maliciously modified, or untruthfully reported. Typically, various game-theoretical mechanisms are used to incentivize truthful provisioining of data. According to [3], however, those approaches address only partial aspects of the larger challenge of assuring

The associate editor coordinating the review of this manuscript and approving it for publication was Pedro Neto.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

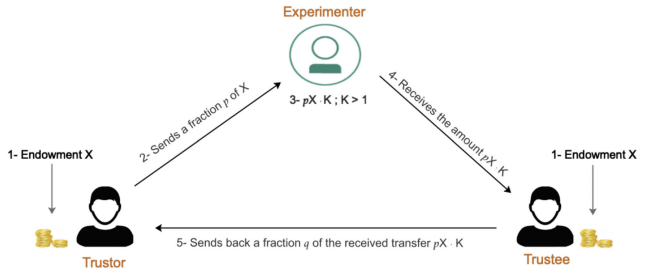

FIGURE 1. Classical trust game involving two human players (trustor and trustee) and one experimenter in the middle.

trustworthiness in data on-chaining systems. A key property of trustworthy data on-chaining systems is truthfulness, which means that no execution of blockchain state transition is caused by untruthful data provisioning, but instead data is always provisioned in a well intended way. The challenge that derives from truthfulness is the building of incentive compatible systems, where participants are assumed to act as rational self-interest driven *homini oeconomici*, whose primary goal is to maximize their individual utility via monetary rewards and penalties for their actions and behavior.

In this paper, we focus on the *trust game* widely studied in behavioral economics. The trust game hasn't been investigated in a blockchain context yet, though it allows for a more systematic study of not only trust and trustworthiness but also reciprocity between human actors [4]. Next, we present a networked version of the trust game leveraging the beneficial characteristics of the social robots in changing players' behavior. Toward this end, we elaborate on the emerging field of *robonomics*, which studies the sociotechnical impact of blockchain technologies on social human-robot interaction. The classical trust game involves only two human players referred to as trustor and trustee, who are paired anonymously and are both endowed with a certain amount X of monetary units. Fig. 1 illustrates the sequential exchange between trustor and trustee. The trustor can transfer a fraction $0 \leq p \leq 1$ of her endowment to the trustee. The experimenter then multiplies this amount by a factor $K > 1$, e.g., doubled or tripled. The trustee can transfer a fraction $0 \leq q \leq 1$ of the received amount directly back to the trustor without going through the experimenter. Note that the trust game captures any generic economic exchange between two actors. According to [8], the trust game will remain an important instrument for the study of social capital and its relation to economic growth for many years to come, whereby research on efficient cooperation and coordination technologies will be of particular interest.

The remainder of the paper is structured as follows. In Section II, we first briefly review the 6G vision of future mobile networks, followed by a discussion of the challenges and benefits of blockchain in 6G networks paying close attention to the anticipated role of blockchain oracles and persuasive robots. In Section III, we identify open research challenges of realizing a blockchain-enabled trust game, including its social efficiency performance and the design of suitable reward and penalty mechanisms. We then delve into the technical issues of implementing a smart-contract based decentralized version of the classical trust game by applying basic blockchain technologies and validating them experimentally. Section IV explores advanced blockchain technologies, most notably on-chaining oracles, to facilitate equitable social efficiency in a networked N-player (i.e., multiplayer) trust game. In Section V, we put the N-player trust game in the context of robonomics leveraging the beneficial characteristics of robot persuasive strategies to foster prosocial human behavior. Finally, Section VI concludes the paper.

II. 6G VISION: BLOCKCHAINS AND ROBOTS

A. BLOCKCHAIN BENEFITS FOR 6G

Blockchain is used to generate the large-scale index as a security measure for all network communication. It serves as a mutual, collective, and common ledger. Blockchain performs the transition from client-server to a trusted peer-to-peer network. According to [5], blockchain and distributed ledger technologies (DLT) may be viewed as the next generation of distributed sensing services, whose need for connectivity will require a synergistic mix of ultra-reliable and low latency communications (URLLC) and massive machine-type communications (mMTC) to guarantee low-latency, reliable connectivity, and scalability [6]. A combination of blockchain technologies and 6G communications network yield the following benefits:

- **Intelligent Resource Management:** According to [7], network resource management and sharing play a significant role in 6G. Resource management operations such as spectrum sharing, orchestration, and decentralized computation have to be compatible with massive infrastructure volumes. Toward this end, blockchain and smart contracts are anticipated to play a major role for self-organizing network resource management. Further, smart contracts help handle and automate the relationship between operators and end-users.
- **Security and Privacy Features:** Another important benefit is the sophisticated use of all 6G network resources, services, and user data without compromising user security and privacy [7]. In this regard, security and privacy-preserving solutions based on blockchain such as decentralized authentication and access control, data ownership, integrity, traceability, and monitoring as well as the self-sovereign identity (SSI) paradigm, have been emerging to provide users with mechanisms that enable them to become anonymous, secure, and take control of their personal data during digital transactions.
- **Trustworthy 6G Communications:** 6G will fuse the digital and physical worlds for the purpose of sensing the real world and integrate far-reaching applications, ranging from autonomous systems to extended reality [7]. The opportunities for exploiting blockchain in 6G network infrastructures enhance the

trustworthiness and performance gains of new services. For instance, blockchain can enable a trusted charging and billing without centralized intermediaries. In addition, blockchain helps establish trusted and decentralized service level agreement (SLA) management given that, similar to 5G, 6G builds on virtualized and sliced network architectures. However, these solutions still need to be implemented at an extremely large scale. As a result, 6G is expected to support a very wide range of use cases with diverse SLA guarantees that need to be managed it in a trusted manner.

B. BLOCKCHAINS AND ROBOTS

The authors of [5] observed that the ongoing deployment of 5G cellular systems is exposing their inherent limitations compared to the original premise of 5G as an enabler for the Internet of Everything (IoE). They argue that 6G should not only explore more spectrum at high-frequency bands but, more importantly, converge driving technological trends. Among others, they claim that there will be the following three driving applications behind 6G: (*i*) blockchain and distributed ledger technologies, (*ii*) connected robotics and autonomous systems, and (*iii*) wireless brain-computer interaction (a subclass of human-machine interaction). In fact, in 6G, there is a strong notion that the nature of mobile terminals will change, whereby intelligent mobile robots are anticipated to play a more important role [6]. More specifically, in [7], the authors argue that 6G services that could provide human users with good advice would certainly be appreciated. According to the world's first 6G white paper published by the 6Genesis Flagship Program (6GFP) in September 2019, 6G will become more human-centered than 5G, which primarily focused on industry verticals.

This brief review of the 6G vision shows that blockchain technologies and robots are anticipated to play a central role in future mobile networks, which will become more human-centered than previous generations of cellular networks. Advanced blockchain technologies such as oracles that enable the on-chaining of blockchain-external off-chain information stemming from human users hold promise to leverage also on human intelligence rather than machine learning only. Similarly, intelligent mobile robots interacting with human users appear a promising solution to not only give physical and/or emotional assistance, but also to nudge human behaviour by benefitting from persuasive robots.

III. BLOCKCHAIN-ENABLED TRUST GAME

In this section, we first identify open research challenges, followed by a blockchain-enabled baseline implementation of the trust game for benchmark comparison via experiments.

A. OPEN RESEARCH CHALLENGES

The use of decentralized blockchain technologies for the trust game should tackle the following research challenges:

- **Social Efficiency:** Recall from above that the trust game allows the study of social capital for achieving economic growth. Towards this end, the closely related term *social efficiency* plays an important role. Social efficiency is defined as the optimal distribution of resources in society, taking into account so-called externalities as well. In general, an externality is the cost or benefit that affects third parties other than the voluntary exchange between a pair of producer and consumer. We will study the impact of externalities below, when we extend the classical trust game to multiplayer games.
  We measure social efficiency as the ratio of the achieved total payoff of both trustor and trustee and the maximum achievable total payoff, which is equal to $X(K+1)$. A social efficiency of 100% is achieved if the trustor sends her full endowment X (i.e., $p=1$), which is then multiplied by K, and the trustee reciprocates by sending back the received amount XK fully or in part, translating into a total payoff of $q\cdot XK+(1-q)\cdot XK+X=X(K+1)$. Note that maximizing the total payoffs requires to set $p=1$ for a given value of K, though q may be set to any arbitrary value. The parameter q, however, plays an important role in controlling the (equal or unequal) distribution of the total payoffs between trustor and trustee, as discussed in more detail shortly. Conversely, if the trustor decides to send nothing (i.e., $p=0$) due to the lack of trust (on the trustor's side) and/or lack of trustworthiness (on the trustee's side), both are left with their endowment X and the social efficiency equals $2X/X(K+1)=2/(K+1)$. How to improve social efficiency in an equitable fashion in a blockchain-enabled trust game is an important research challenge.
- **Trust and Trustworthiness in N-Player Trust Game:** In the past, games of trust have been limited to two players. In [9], the authors introduced a new N-player trust game that generalizes the concept of trust, which is normally modeled as a sequential two-player game to a population of multiple players that can play the game concurrently. According to [9], evolutionary game theory shows that a society with no untrustworthy individuals would yield maximum wealth to both the society as a whole and the individual in the long run. However, when the initial population consists of even the slightest number of untrustworthy individuals, the society converges to zero trustors. The proposed N-player trust game shows that the promotion of trust is an uneasy task, despite the fact that a combination of trustors and trustworthy trustees is the most rational and optimal social state.
  It's important to note that the N-player trust game in [9] was played in an unstructured environment, i.e., the population was not structured in any specific spatial topology or social network. In [10], the authors investigated whether a *networked* version of the N-player trust game would promote higher levels of trust and global net wealth (i.e., total payoffs) in the

population than that of an unstructured population. To do so, players were mapped to a spatial network structure, which restricts their interactions and cooperation to local neighborhoods. Unlike [9], where the existence of a single untrustworthy individual would eliminate trust completely and lead to zero global net wealth, the authors of [10] discovered the importance of establishing network structures for promoting trust and global net wealth in the N-player trust game in that trust can be promoted despite a substantial number of untrustworthy individuals in the initial population. Clearly, the development of appropriate communication network solutions for achieving efficient cooperation and coordination among players with different strategies in a networked N-player trust game represents an interesting research challenge.

- **Reward & Penalty Mechanism Design:** For the implementation of desirable social goals, the theory of *mechanism design* plays an important role. According to [11], the theory of mechanism design can be thought of as the ''engineering'' side of economic theory. While the economic theorist wants to explain or forecast the social outcomes of mechanisms, the mechanism design theory reverses the direction of inquiry by identifying first the social goal and then asking whether or not an appropriate mechanism could be designed to attain that goal. And if the answer is yes, what form that mechanism might take, whereby a mechanism may be an institution, procedure, or game for determining desirable outcomes.

  An interesting example of mechanism design is the so-called *altruistic punishment* to ensure human cooperation in multiplayer public goods games [12]. Altruistic punishment means that individuals punish others, even though the punishment is costly and yields no material gain. It was experimentally shown that altruistic punishment of defectors (i.e., untrustworthy participants) is a key motive for cooperation in that cooperation flourishes if altruistic punishment is possible, and breaks down if it is ruled out. The design of externalities such as third-party punishment and alternative reward mechanisms for incentivizing human cooperation in multiplayer public goods games in general and N-player trust game in particular is of great importance.
- **Decentralized Implementation of Economic Experiments:** A widely used experimental software for developing and conducting almost any kind of economic experiments, including the aforementioned public goods games and our considered trust game, is the *Zurich Toolbox for Ready-made Economics (z-Tree)* [13]. The z-Tree software is implemented as a client-server application with a central server application for the experimenter, called z-Tree, and a remote client application for the game participants, called z-Leaf. It is available free of charge and allows economic experiments to be conducted via the Internet. At the downside, however, z-Tree does not support peer-to-peer (P2P) communications between players, as opposed to a decentralized blockchain-enabled implementation.

B. BLOCKCHAIN-ENABLED IMPLEMENTATION

In this section, we develop a blockchain-enabled implementation of the classical trust game using Ethereum and experimentally investigate the beneficial impact of a simple yet effective blockchain mechanism known as *deposit* on enhancing both trust and trustworthiness as well as increasing social efficiency.

1) EXPERIMENTER SMART CONTRACT

First, we develop a smart contract that replaces the experimenter in the middle between trustor and trustee (see Fig. 1). The development process makes use of the Truffle framework, a decentralized application development framework. The resultant experimenter smart contract is written in the programming language Solidity. We then compile the experimenter smart contract into Ethereum Virtual Machine (EVM) byte code. Once the experimenter smart contract is compiled, it generates the EVM byte code and Application Binary Interface (ABI). Next, we deploy the experimenter smart contract on Ethereum's official test network Ropsten. It can be invoked by using its address and ABI. More specifically, in our experimenter contract, we use the following global variables: (*i*) *msg.value*, which represents the transaction that is sent, and (*ii*) *msg.sender*, which represents the address of the player who has sent the transaction to the experimenter smart contract, i.e., trustor or trustee. Both trustor and trustee use their Ethereum Externally Owned Account (EOA), which uses public and private keys to interact and invoke each function of our experimenter smart contract. In the following, we provide a brief overview of the core functions and parameters of our experimenter smart contract:

- **Function investFraction():** This function allows the trustor to invest a portion p of her endowment X. Once called, it takes the received msg.value p from the trustor, multiplies it by factor K using the contract balance, and transfers it directly to the trustee's account. The trustee receives $msg.value \cdot K$.
- **Function splitFraction():** This function allows the trustee to split a portion q of the received investment from the trustor. Once called, it takes the set split amount from the trustee's account and sends it to the trustor's account.
- **Parameter Onlytrustor (modifier type):** This modifier is applied to the *investFraction()* function. Thus, only the trustor can invoke this function of the experimenter smart contract.
- **Parameter Onlytrustee (modifier type):** This modifier is applied to the *splitFraction()* function. Thus, only the trustee can invoke this function of the experimenter smart contract.

We note that after the execution of each function of the experimenter smart contract, an event is used to create notifications and saved logs. Events help trace and notify both players about the current state of the contract and activities.

2) BLOCKCHAIN MECHANISM DEPOSIT

The use of one-way security deposits to provide trust for one party with respect to the other is quite common, particularly for the exchange of goods and services via e-commerce and crowdsourcing platforms. In the context of blockchains, a deposit is an agreement smart contract that defines the arrangement between parties, where one party deposits an asset with a third party. An interesting use case of the blockchain mechanism deposit can be found in [14]. In this paper, the authors propose a new protocol that achieves the fulfillment of all the desired properties of a registered e-Deliveries service using blockchain. In this protocol, the authors included a deposit mechanism with the aim to encourage the sender to avoid dishonest behavior and fraud attempts, and also to conclude the exchange in a predefined way following the phases of the protocol. The deposit will be returned to the sender if he finishes the exchange according to the protocol.

In our work, we propose to add an optional function *deposit*() to our experimenter smart contract to improve trust and trustworthiness between both players. Towards this end, we make the following two modifications:

- **Function deposit():** This function allows the trustee to submit an amount of $2 \leq D \leq X$ monetary units (i.e., Ether in our considered case of Ethereum) as a deposit to the experimenter smart contract. The deposit is returned to the trustee only if a transaction with $q > 0$ is completed. Otherwise, with $q = 0$, the trustee loses the deposit. It should be noted that the aforementioned *Onlytrustee()* modifier is also applied to this function.
- **Function splitFraction():** We make a modification to this function to allow the trustee to split the received amount (i.e., $q > 0$). Otherwise, the transaction is rejected until the trustee splits the received amount. Once this happens, the function transfers the amount to the trustor's account and returns the deposit D to the trustee's account.

3) EXPERIMENTAL VALIDATION

Next, we investigate the impact of the deposit as an effective pre-commitment mechanism on the trust game performance via Ethereum-based blockchain experiments. We set $K = 2$ in our experimenter smart contract and consider different deposit values of $D = \{0, 2, 5, X\}$ Ether, whereby $D = 0$ denotes the classical trust game without any deposit. The experiment was conducted with two graduate students from different universities. The rationale behind the selection of only two students is to first focus on the conventional trust game that by definition involves only two players. This allows us to be more certain that the effects of the deposit mechanism are real. In addition, conducting our experiment with the same two participating students allows us to better observe the behavior change during the rounds of the game. As for our inclusion criteria, we note that the students didn't know each other's identity, which was important to ensure anonymity between them. Further, the students hadn't conducted any behavioral research experiment before. Nor did either participant had any prior knowledge or experience with the trust game or any other investment game experiments. The two participating students were male and their age was 23 and 25 years, respectively.

At the beginning of the experiment, both trustor and trustee were given an endowment of $X = 10$ Ether. We ran the experiment four times, each time for a different value of D. Each of the four experiments took five rounds. We note that for the experiment with $D = 10$ Ether, the trustee put her full endowment X into the deposit, thus $D = X$ Ether. All experiments were run across the Internet. Both participants interacted with our experimenter smart contract using their Ethereum accounts. We note that both the trustor and the trustee need to pay a gas fee. Gas refers to the pricing value, required to successfully conduct a transaction or execute a function in a smart contract on the Ethereum blockchain platform. Priced in small fractions of the cryptocurrency Ether, commonly referred to as Gwei. Each Gwei is equal to 0.000000001 ETH (10^{-9} Ether). Given its lowest cost, we considered transaction fees associated with deploying the smart contract and sending transactions negligible compared to the amounts invested and split.

Fig. 2 depicts the average social efficiency and normalized reciprocity (both given in percent) vs. deposit $D = \{0, 2, 5, X\}$ (given in Ether). We define normalized reciprocity as the ratio of q/p as a measure of the trustee's reciprocity, q, in response to the trustor's generosity, p. Note that the normalized reciprocity is useful to gauge the fair distribution of total payoffs from trustee to trustor, and vice versa, for a given achievable social efficiency. Note that Fig. 2 also shows the interval between minimum and maximum measured score for each value of D.

We make the following interesting observations from Fig. 2. First, the social efficiency continually grows for an increasing deposit D until it reaches the maximum of 100% for $D = X$. Thus, the social efficiency performance of the classical trust game can be maximized by applying the blockchain chain mechanism of deposit properly with $D = X$. This is due to the fact that the trustor sends her full endowment (i.e., $p = 1$) after the trustee has put in her maximum deposit. In doing so, a maximum total payoff of 30 Ether is achieved, translating into a social efficiency of 100%. It is worthwhile to mention that this was the case in all five rounds of the experiment. Second, the average normalized reciprocity improves significantly for increasing deposit D compared to the classical trust game without any deposit ($D = 0$). Specifically, in the classical trust game, the average normalized reciprocity is as low as 18%. By contrast, for a deposit of as little as $D = 2$ Ether, the average normalized

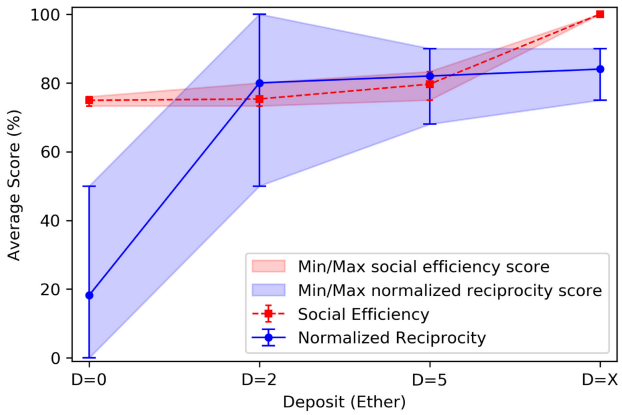

FIGURE 2. Average social efficiency and normalized reciprocity q/p vs. deposit $D = \{0, 2, 5, X\}$ Ether using experimenter smart contract with $K = 2$ and $X = 10$ (shown with minimum-to-maximum measured score intervals).

reciprocity rises to 80%. Interestingly, further increasing D does not lead to sizeable additional increases, e.g., average normalized reciprocity equals 83% for $D = X$. Hence, the amount of the deposit does not change the normalized reciprocity significantly with $q/p \approx 80\%$ for $D > 0$. Finally, Fig. 2 illustrates that for an increasing deposit D, the behaviour of the two players become more consistent, as indicated by the decreasing intervals of minimum to maximum measured scores.

In the subsequent section, we extend the classical two-player trust game to a networked N-player trust game and study how advanced blockchain technologies, most notably on-chaining oracles, drive the behaviour of players by means of different reward and penalty mechanisms. Among others, we seek to understand whether an increased normalized reciprocity is achievable without sacrificing social efficiency.

IV. ON-CHAINING ORACLE FOR NETWORKED N-PLAYER TRUST GAME

A. ORACLE TAXONOMY

Smart contracts need to acquire data about real-world states and events from outside the blockchain network, which cannot be achieved by smart contracts themselves because the blockchain environment is isolated from the external world. This is where oracles come into play to overcome the limitation by providing a link between off-chain and on-chain data. In general, oracles may get data from various sources. Depending on the respective source, oracles can be classified as follows:

- Software Oracles: Interact with online data sources and send information to the blockchain. The transmitted information may come from online databases, servers, websites, or, essentially, any data source on the web.
- Hardware Oracles: Designed to get information from the physical world and make it available to smart contracts. Such information may be relayed from sensors, IoT devices, RFID tags, barcode/QR scanners, robots, or other information reading devices.
- Human Oracles: Individuals with specialized knowledge and skills in a particular field may also serve as oracles. They can research and verify the authenticity of information from various sources and transfer the information to smart contracts. Further, human oracles may provide smart contracts with answers to questions, report specified actions, or vote on the truth of a given event.

Oracles use a number of nodes to on-chain data onto a smart contract. These nodes define the trust model used by oracles [15]. Accordingly, oracles are classified into the following two models:

- Centralized Oracles: Controlled by a single source. The efficiency of centralized oracle is high, but their main problem is the existence of a single point of failure, where availability, accessibility, and level of certainty about the validity of the data depends on only one node.
- Decentralized Oracles: Resolve the singular point of failure problem of centralized oracles and thus increase the reliability of the information provided to smart contracts. A smart contract is able to query multiple oracles to determine the validity and accuracy of the data. This is why decentralized oracles are also referred to as consensus oracles.

Various oracle systems have been proposed, which aim at providing data to a blockchain reliably by incorporating techniques such as advanced cryptography, trusted execution environments, reporting, and voting [3].

B. ARCHITECTURE OF ORACLE

Fig. 3 depicts the architecture of our proposed on-chaining oracle for the networked N-player trust game. The proposed architecture comprises a set of clusters or pools. Each cluster contains three types of agent: (*i*) trustors, (*ii*) trustees, and (*iii*) observers. The difference between observers and players (trustors/trustees) is that observers don't play, but track and evaluate trust and trustworthiness criteria such as investment (p) and split (q). Players interact with the experimenter smart contract using their public-private keys through a decentralized application (DApp). The different rounds of the game are monitored remotely by the observers using *Etherscan.io*, an Ethereum blockchain explorer that uses the experimenter contract address and shows the different transactions between each pair of trustor and trustee in real-time. We note that alternatively one may use *Alethio.io*, a monitoring tool that allows observers to send and receive alerts to and from any on-chain address, activity, or function.

The design of a third-party punishment and reward mechanism for incentivizing player cooperation in our networked N-player trust game is based on crowdsourcing. Specifically, observers provide their collective human intelligence to the nudge contract in order to punish a cluster or an individual player, who demonstrates inappropriate behaviour, or provide a positive reward for good behaviour. The nudge contract

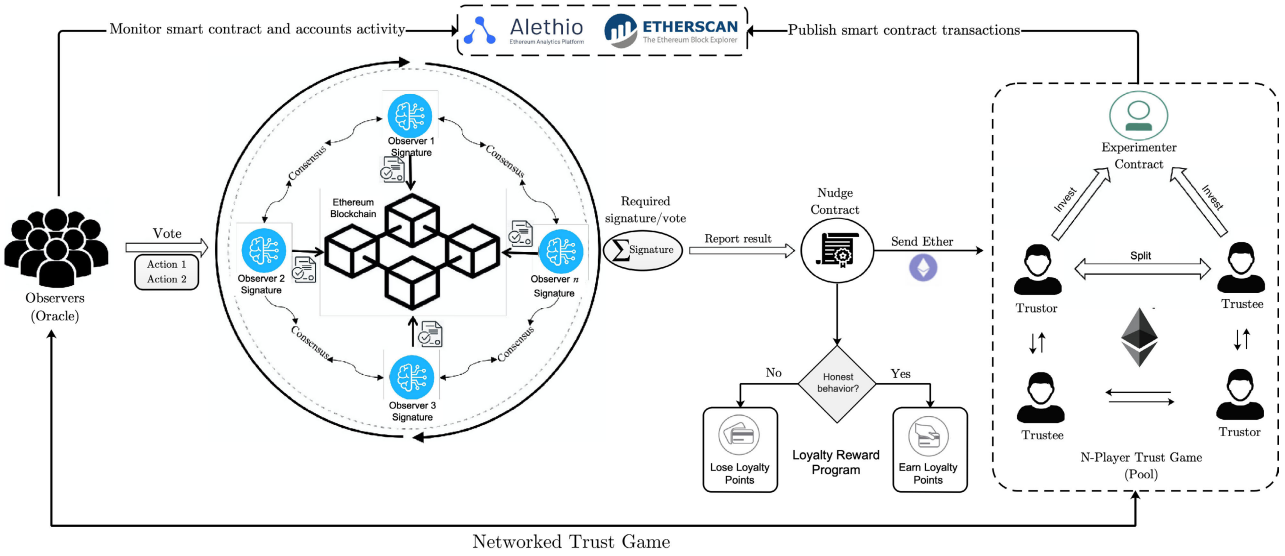

FIGURE 3. Architecture of on-chaining oracle for networked *N*-player trust game.

manages the reward-penalty mechanism in the form of loyalty points. A trustor can earn loyalty points for a honest transaction, investment, and engagement in the game and redeem earned points for rewards. Similarly, the trustee is rewarded for generous reciprocity. Loyalty points keep the players engaged and aware of the overall goals, i.e., increase of total payoff, social efficiency, and normalized reciprocity. In addition, the players have a score profile associated with their public key, whereby players earn 1 point for every honest action and loose 1 point if their action is dishonest. The scoring profile is managed by the nudge contract. Trustor and trustee can check the status of their loyalty reward points by calling the function *getTrustorLoyalty*() and *getTrusteeLoyalty*(), respectively. Furthermore, an incentive strategy was designed to incorporate principles of behavioural psychology using economic outcomes to render the system more effective in changing the players' behaviour. Players earn a monetary reward in the form of Ether after reaching a certain number of loyalty reward points in the game, e.g., 10 points $=$ 1 Ether. The Ethers earned are added to the player's endowment X, which will be used for the investment and payoff in future rounds of the game.

C. ON-CHAINING OF VOTING-BASED DECISIONS

In our oracle implementation, we assigned predetermined public keys to both players and observers. The creation of each key pair can be accomplished by using several options, including Ethereum wallets and online/offline Ethereum address generators, e.g., Vanity-ETH. All public keys are declared in the nudge contract, whose purpose is to allow only registered observers to vote while automatically rejecting malicious voters. To facilitate the formation of a majority, the number of possible voting options is restricted to the four following functions on the nudge contract: *VOTE_RewardTrustor*, *VOTE_RewardTrustee*, *VOTE_PunishTrustor*, and *VOTE_PunishTrustee*. Recall that a function is a code that resides at a specific smart contract address on the Ethereum blockchain. Further, to ensure a trustworthy on-chaining decision, a k-out-of-M threshold signature is used to reach a consensus on the function to be executed. A k-out-of-M threshold signature scheme is a protocol that allows any subset of k players out of M players to generate a signature, and disallows the creation of a valid signature if fewer than k players should participate. The right decision is determined as the one that has received the desired number of votes. Once the function is executed, the nudge contract allocates the reward or punishment loyalty points to each player who behaved in a trusted or untrusted way, respectively.

D. EXPERIMENTAL VALIDATION

We compare the performance of our proposed on-chaining oracle for the multiplayer N-player trust game with the conventional two-player baseline experiment. Towards this end, we invited the same two students, who have played the classical two-player trust game before, and asked them to play the game again, i.e., without any observers. Next, we invited them to play the game in the presence of two observers. The two players were informed that their account is associated with loyalty reward points, which will be increased if they act honestly. Otherwise, they will be punished and loose 1 loyalty point. Both players were aware that they will be rewarded with 1 Ether for each 10 accumulated loyalty reward points. In addition, they are notified that the decision will be made by two observers, who will monitor their online transactions in order to make their independent reward/penalty decisions. All four participants interact anonymously via the Internet.

Fig. 4 compares the average social efficiency of the two-player trust game without observers with that of the four-player trust game with observers. The figure clearly

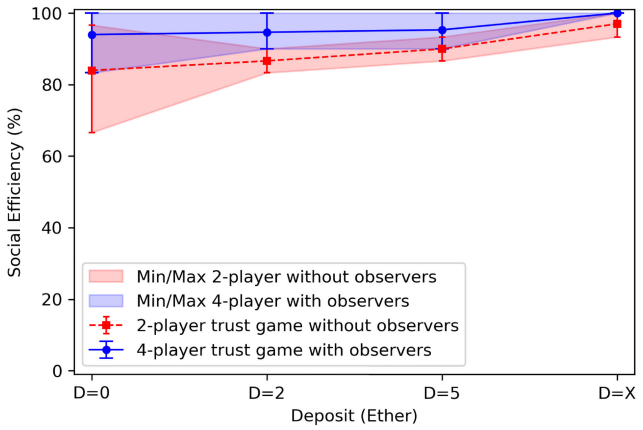

FIGURE 4. Average social efficiency vs. deposit $D = \{0, 2, 5, X\}$ Ether for 2-player trust game without observers and 4-player trust game with observers (shown with minimum-to-maximum measured score intervals).

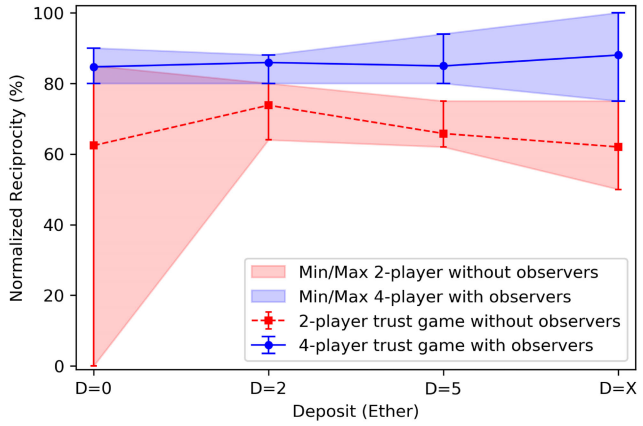

FIGURE 5. Average normalized reciprocity q/p vs. deposit $D = \{0, 2, 5, X\}$ Ether for 2-player trust game without observers and 4-player trust game with observers (shown with minimum-to-maximum measured score intervals).

demonstrates the beneficial impact of the presence of observers on social efficiency for all values of D. Note that with observers the instantaneous social efficiency reaches the maximum of 100% for all values of D, as opposed to the two-player trust game where this occurs requiring the full deposit of $D = X$ Ether. As for the normalized reciprocity achievable with and without observers things are similar, as shown in Fig. 5. However, while the presence of observers helps raise the average (and instantaneous) normalized reciprocity consistently above 80% (compared to below 80% in Fig. 2), there still remains room for further improvement, especially for $0 \leq D < X$.

V. ROBONOMICS: PLAYING THE *N*-PLAYER TRUST GAME WITH PERSUASIVE ROBOTS

Many studies have shown that the physical presence of robots benefits a variety of social interaction elements such as persuasion, likeability, and trustworthiness. Thus, leveraging these beneficial characteristics of social robots represents a promising solution towards enhancing the performance of the trust game. Social robots connected with human operators form a physical embodiment that creates the new paradigm of an immersive coexistence between humans and robots, whereby persuasive robots aim at changing the behaviour of users through social influence. Importantly, these robots are less like tools and more like partners, whose persuasive role in a social environment is mainly human-centric [16].

Recently, in [17], an experimental pilot study with 5 participants adapted the trust game from its original human-human context to a social human-robot interaction (sHRI) setting using a humanoid robot operated in a *Wizard-of-Oz (WoZ)* manner, where a person controls the robot remotely. The obtained findings suggest that people playing the sHRI trust game follow a human-robot trust model that is quite similar to the human-human trust model. However, due to the lack of common *social cues* present in humans (e.g., facial expressions or gestures) that generally influence the initial assessment of trustworthiness, almost all participants started investing a lower amount and increased it after actively exploring the robot's behaviour and trustworthiness through social experience.

In the following, we focus on the emerging field of *robonomics*, which studies the sociotechnical impact of blockchain technologies on sHRI, behavioral economics, behavioral game theory, and cryptocurrencies (both coins and tokens) for the social integration of robots into human society [18]. Robonomics involves persuasive robotics, whereby a physical or virtual robotic agent is used as enforcer or supervisor of human behavior modification via psychological rewards in addition to tangible rewards. In a recent exploratory sHRI study [19], ten multimodal *persuasive strategies* were compared with regard to their effectiveness of social robots attempting to influence human behavior. It was experimentally shown that two particular persuasive strategies—affective and logical strategies—achieved the highest persuasiveness and trustworthiness.

Similar to [20], we developed a *Crowd-of-Oz (CoZ)* platform for letting observers remotely control the gestures of Softbank's social robot Pepper placed in front of the trustee and have a real-time dialogue via web-based text-to-speech translation. The CoZ user interface is built using a Django web server. The trustee can communicate with Pepper through voice and Pepper's tactile tablet. To support voice communication, we implemented a web-based speech-to-text tool. When text is extracted from voice, the trustee can see his/her message on Pepper's tablet in order to verify it. Next, the speech-to-text function calls another function to add additional fields to the main message (extracted text), including sequence ID, sender ID, message type, and time to make the message distinguishable on the Django server. The called function executes a marshaling process and sends the message to the Django server through the OOCSI middleware. The OOSCI middleware is a message-based connectivity layer and is platform-independent inspired by the concept of RPC (Remote Procedure Call) for connecting web clients.

In our developed CoZ system, there are two types of message: information and control. The information messages are

created by the observers. This type of message is multicast to all observers and the trustee through Pepper for update them, but not the trustor. The trustee can see all the information messages on Pepper's tablet. Moreover, Pepper uses a text-to-speech function to transfer the observers' messages to the trustee. The control messages are used for important functionalities of the CoZ architecture, e.g., performing a gesture on Pepper. When an observer presses a social cue button, the CoZ web-interface invokes a JavaScript method to call a new event on the Django server. The invoked method sets all the related joints' angles plus the LED colors of Pepper's eyes. Given that two or more observers may press the same or different social cue buttons simultaneously, the Django server implemented a queue to synchronize all issued commands. While Pepper is performing a gesture, the Django server puts the next gesture in the queue and sends it to Pepper back-to-back.

Further, our CoZ user interface provides a section, where an observer can watch the trustee's environment through Pepper's eyes. To implement this part, we used OpenCV, Flask, and CV2 tools. The Django server invokes a method on Pepper called ''ALVideoDevice'' to start recording videos. Next, the Flask server stores the sequence of produced videos with a valid URL. To make live video streams accessible over the Internet we used VPN. Moreover, in our CoZ interface, we used an IFrame (Inline Frame) tag to demonstrate live video streaming using the valid URL. An IFrame is an HTML document embedded inside another HTML document on a website. The IFrame HTML element is often used to insert content from another source, such as a camera, into a Web page. In our CoZ user interface, we also realized four buttons to turn Pepper's head to left, right, up, and down. When an observer presses one of these buttons, the CoZ interface invoke a method to create a control message, marshaling process, and send it to the Django server. Upon reception, the Django server performs unmarshaling to extract the main message and then invokes the ''ALMotion'' along with initializing some parameters like speed, angle, and joint name. For each invocation, Pepper turns her head by ten degrees.

The user CoZ inteface also displays nine social cue buttons to prevent possible typos and save time for observers to fill communication gaps. The nine social cue buttons were as follows: ''Gain time'', ''Tell me about it'', ''Good job'', ''Hi'', ''Bye'', ''Open arms'', ''Taunting hands'', ''No'', and ''Ask for attention''. Observers may press to perform different gestures of Pepper during conversation and thereby influence the trustee's behavior. In addition, we drafted two scripts, one for a logical persuasive strategy appealing to the left side of the brain (i.e., logics) and another one for an affective persuasive strategy appealing to the right side of the brain (i.e., emotions) of the trustee. Each script contains pre-specified sentences stored in pull-down menus in the CoZ interface, from which observers may choose in order to nudge the trustee's behavior toward reciprocity via real-time text-to-speech messages. The different persuasive robot strategies operate as follows:

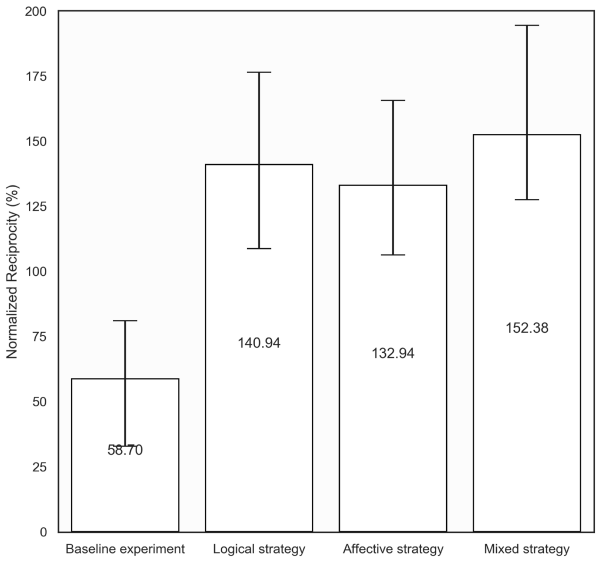

FIGURE 6. Average normalized reciprocity q/p without (baseline experiment) and with using logical, affective, and mixed logical-affective persuasive strategies for $D = 0$ (shown with minimum-to-maximum measured score intervals).

- **Logical Strategy**: Contains a set of reward and punishment mechanisms. In addition, Pepper performs some economical and technical advice via text-to-speech through the above described CoZ platform.
- **Affective Strategy**: Contains a set of reward/punishment mechanisms and Pepper uses text-to-speech encouragement messages through the CoZ platform. In addition, Pepper shows social cues by means of gestures and embodied communications toward the trustee.
- **Mixed Strategy**: Combines the above logical and affective strategies into one mixed strategy. It contains a set of reward/punishment mechanisms and Pepper provides not only economical and technical advice but also encouragement via text-to-speech messages through the CoZ platform. In addition, Pepper shows social cues by means of gestures and embodied communications toward the trustee.

For illustration, Table 1 lists the social cues used by Pepper in our proposed mixed logical-affective persuasive strategy. In this strategy, one observer plays the logical strategy and the other observer plays the affective strategy such that the trustee receives mixed messages and mixed embodied communications. Depending on the trustee's behavior, the observers carries out the ''Trusted behavior action'' or the ''Untrusted behavior action'' in each round of the experiment. The social cues in Table 1 enable the observers to control Pepper's text-to-speech and embodied communications using our developed CoZ platform.

We ran large-scale experiments involving 20 students to measure the effectiveness of our developed persuasive robotics strategies. Similar to our last experiment in the two players' trust game, the participating students didn't know each other's identity. Also, students hadn't conducted

TABLE 1. Social cues used by Pepper in mixed persuasive strategy.

Round number	Trusted behavior action	Untrusted behavior action
Round 1	*Text-to-speech:* Trust Game is a cooperative investment game. You all play together to get the best total payoff!	Untrusted behavior will be shown in Round 2
Round 2	*Text-to-speech:* Awesome! That's a split worth celebrating! *Embodied communication:* Open arm gesture.	*Text-to-speech:* If this behavior is repeated, you will receive a punishment from the observers. *Embodied communication:* Taunting hand gesture.
Round 3	*Text-to-speech:* If this good behavior is repeated, your partner will invest more in the next round. *Embodied communication:* Open arm gesture.	*Text-to-speech:* Weak reciprocity can cause costly punishment for you. *Embodied communication:* Taunting hand gesture.
Round 4	*Text-to-speech:* Incredible! Your partner must be impressed! *Embodied communication:* Open arm gesture.	*Text-to-speech:* With such a behavior, the punishment will be executed next round. *Embodied communication:* Taunting hand gesture.
Round 5	*Text-to-speech:* Congrats! Your good behavior toward your partner has provided you with an incremental total payoff over all rounds of the game. *Embodied communication:* Open arm gesture.	*Text-to-speech:* Your bad behavior translated into a very weak total payoff. *Embodied communication:* Taunting hand gesture.

any behavioral research experiment before. The age of the selected students was between 24 and 32 years. Three students were female and seventeen students were male. The experiment was divided into four trials: baseline, logical, affective, and mixed strategy. Each trial involved 5 rounds. We first conducted a baseline trust-game experiment, where trustees didn't interact with Pepper, as done previously, followed by experiments exposing trustees to Pepper's logical, affective, and mixed logical-affective persuasive strategies. Both trustor and trustee interacted via a blockchain account with the experimenter's smart contract. The trustor played the game from a separate room, while the trustee was in the lab alone with Pepper. Pepper was controlled via our CoZ platform remotely by the observer. We used the same parameter settings, i.e., endowment $X = 10$ Ether for the trustor and $K = 2$. Further, in all persuasive strategies, we didn't use any deposit mechanism (i.e., $D = 0$).

Fig. 6 demonstrates the superior effectiveness of our persuasive strategies, especially mixed ones appealing to both sides of the brain, resulting in average normalized reciprocity well above 100%. Further, to better reveal the differences among the persuasive strategies, we have calculated the measurement range for the four strategies. The measurement range for the baseline experiment is 48.2 (Max = 81, Min = 32.8), while for the logical strategy it is 67.8 (Max = 176.4, Min = 108.6), for the affective strategy it is 59.4 (Max = 165.6, Min = 106.2), and the mixed strategy it is 67 (Max = 194.4, Min = 127.4). As the results show, the baseline experiment has the smallest measurement range. Next, we computed the standard deviation for the baseline experiment as well as logical, affective, and mixed strategies, which is equal to 15.6, 21.75, 21.10, and 22.73, respectively. The results show that the baseline experiment has the smallest standard deviation among all considered strategies, while the mixed strategy has the largest one. Finally, we have computed the variance for the persuasive strategies under consideration. The calculated variance equals 245.83, 473.17, 445.25, and 517.03 for the baseline, logical, affective, and mixed strategy, respectively. Based on the gathered results, we observe that the baseline experiment has the smallest and the mixed strategy has the largest variance.

VI. CONCLUSION

Robonomics is a recently emerging sociotechnical field of interdisciplinary research that integrates behavioral economics with advanced blockchain technologies and persuasive robotics. Given its prominent role in behavioral economics and the relevance of trust in blockchains, we focused on the trust game, including its networked multiplayer extension. We experimentally demonstrated the beneficial impact of the blockchain mechanisms deposit and on-chaining oracle on improving both social efficiency and reciprocity significantly. Our experimental results show that the presence of third parties such as human observers and in particular social robots play an important role in a blockchain-enabled trust game. While the trust game's central experimenter may be easily replaced with our presented experimenter smart contract, the peer pressure executed by on-chaining oracles and especially the embodied communications enabled by persuasive robots were shown to have a potentially greater social impact than monetary incentives such as deposit, opening up new research avenues for future work.

REFERENCES

[1] O. Novo, "Blockchain meets IoT: An architecture for scalable access management in IoT," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1184–1195, Apr. 2018.

[2] J. Adler, R. Berryhill, A. Veneris, Z. Poulos, N. Veira, and A. Kastania, "ASTRAEA: A decentralized blockchain oracle," in *Proc. IEEE Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom), IEEE Cyber, Phys. Social Comput. (CPSCom), IEEE Smart Data (SmartData)*, Halifax, NS, Canada, Jul. 2018, pp. 1145–1152.

[3] J. Heiss, J. Eberhardt, and S. Tai, "From oracles to trustworthy data on-chaining systems," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Atlanta, GA, USA, Jul. 2019, pp. 496–503.

[4] J. Berg, J. Dickhaut, and K. McCabe, "Trust, reciprocity, and social history," *Games Econ. Behav.*, vol. 10, no. 1, pp. 122–142, Jul. 1995.

[5] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May/Jun. 2020.

[6] B. Zong, C. Fan, X. Wang, X. Duan, B. Wang, and J. Wang, "6G technologies: Key drivers, core requirements, system architectures, and enabling technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 18–27, Sep. 2019.
[7] K. David and H. Berndt, "6G vision and requirements: Is there any need for beyond 5G?" *IEEE Veh. Technol. Mag.*, vol. 13, no. 3, pp. 72–80, Sep. 2018.
[8] C. Alós-Ferrer and F. Farolfi, "Trust games and beyond," *Frontiers Neurosci.*, vol. 13, pp. 1–14, Sep. 2019.
[9] H. Abbass, G. Greenwood, and E. Petraki, "The N-player trust game and its replicator dynamics," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 470–474, Jun. 2016.
[10] M. Chica, R. Chiong, M. Kirley, and H. Ishibuchi, "A networked N-player trust game and its evolutionary dynamics," *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 866–878, Dec. 2018.
[11] E. S. Maskin, "Mechanism design: How to implement social goals," *Amer. Econ. Rev.*, vol. 98, no. 3, pp. 567–576, May 2008.
[12] E. Fehr and S. Gächter, "Altruistic punishment in humans," *Nature*, vol. 415, no. 6868, pp. 137–140, Jan. 2002.
[13] U. Fischbacher, "Z-tree: Zurich toolbox for ready-made economic experiments," *Experim. Econ.*, vol. 10, no. 2, pp. 171–178, May 2007.
[14] M. Mut-Puigserver, M. A. Cabot-Nadal, and M. M. Payeras-Capella, "Removing the trusted third party in a confidential multiparty registered eDelivery protocol using blockchain," *IEEE Access*, vol. 8, pp. 106855–106871, 2020.
[15] H. Al-Breiki, M. H. U. Rehman, K. Salah, and D. Svetinovic, "Trustworthy blockchain oracles: Review, comparison, and open research challenges," *IEEE Access*, vol. 8, pp. 85675–85685, 2020.
[16] M. Siegel, C. Breazeal, and M. I. Norton, "Persuasive robotics: The influence of robot gender on human behavior," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 2563–2568.
[17] R. C. R. Mota, D. J. Rea, A. Le Tran, J. E. Young, E. Sharlin, and M. C. Sousa, "Playing the 'trust game' with robots: Social strategies and experiences," in *Proc. 25th IEEE Int. Symp. Robot Human Interact. Commun. (RO-MAN)*, New York City, NY, USA, Aug. 2016, pp. 519–524.
[18] I. S. Cardenas and J.-H. Kim, "Robonomics: The study of robot-human peer-to-peer financial transactions and agreements," in *Proc. Companion ACM/IEEE Int. Conf. Hum.-Robot Interact.*, Cambridge, U.K., Mar. 2020, pp. 8–15.
[19] S. Saunderson and G. Nejat, "It would make me happy if you used my guess: Comparing robot persuasive strategies in social human–robot interaction," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1707–1714, Apr. 2019.
[20] T. Abbas, V.-J. Khan, and P. Markopoulos, "CoZ: A crowd-powered system for social robotics," *Elsevier SoftwareX*, vol. 11, pp. 1–7, Jan./Jun. 2020, Art. no. 100421.

ABDELJALIL BENIICHE received the B.Sc. degree in telecommunications engineering and the M.Sc. degree in information systems from Hassan First University, Morocco, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree with the Optical Zeitgeist Laboratory, INRS. His research interests include blockchains, the tactile Internet, human-agent-robot networks, and behavioral economics.

SAJJAD ROSTAMI received the B.Sc. degree in information technology engineering from Sheikh Bahaei University, Iran, in 2013, and the M.Sc. degree in computer network engineering from Qazvin Islamic Azad University (QIAU), Qazvin, Iran, in 2018. He is currently pursuing the Ph.D. degree in telecommunications with INRS. His current research interests include human–robot interaction, social robots, and the Internet of No Things.

MARTIN MAIER received the M.Sc. and Ph.D. degrees *(summa cum laude)* from the Technical University of Berlin, Germany, in 1998 and 2003, respectively. He is currently a Full Professor with INRS, Montreal, Canada. He is also a coauthor of the book *Toward 6G: A New Era of Convergence* (Wiley-IEEE Press, January 2021). He was a Marie Curie IIF Fellow of the European Commission, from March 2014 to February 2015. He was a co-recipient of the 2009 IEEE Communications Society Best Tutorial Paper Award. In March 2017, he received the Friedrich Wilhelm Bessel Research Award from the Alexander von Humboldt (AvH) Foundation in recognition of his accomplishments in research on FiWi enhanced networks. In May 2017, he was named one of the three most promising scientists in the category "Contribution to a Better Society" of the Marie Skłodowska-Curie Actions (MSCA) 2017 Prize Award of the European Commission. In winter 2019/2020, he held the Chair of UC3M-Banco de Santander Excellence with Universidad Carlos III de Madrid (UC3M), Madrid, Spain.