COOPERATION AMONG STRANGERS: ALGORITHMIC ENFORCEMENT OF RECIPROCAL EXCHANGE WITH BLOCKCHAIN-BASED SMART CONTRACTS

(pre-print version, forthcoming in *Academy of Management Review*)

Robert Wayne Gregory, University of Miami, rwgregory@miami.edu

Roman Beck, IT University of Copenhagen and Halmstad University, romb@itu.dk

Ola Henfridsson, University of Miami, ohenfridsson@miami.edu

Niam Yaraghi, University of Miami, niamyaraghi@miami.edu

Acknowledgements:

We sincerely thank associate editor Ruth Aguilera and three anonymous reviewers for their constructive guidance on developing the paper. Ruth provided invaluable editorial guidance throughout the process. We also thank our friendly reviewers for the valuable feedback on the causal arguments and concepts of earlier versions of the manuscript. The friendly reviewers were: Eleunthia Wong Ellinger, Thomas Widjaja, Sophia Mannina, Shamel Addas, Abayomi Baiyere, and Kalle Lyytinen.

COOPERATION AMONG STRANGERS: ALGORITHMIC ENFORCEMENT OF RECIPROCAL EXCHANGE WITH BLOCKCHAIN-BASED SMART CONTRACTS

Enhancing cooperation among strangers is challenging. Strangers, who lack previous interactions and trust, cannot rely on human reciprocity as they engage in social and economic exchange. They have instead a tendency to defect for maximizing individual interests rather than to cooperate for benefitting each party in the exchange. Blockchain-based smart contracts comes with the promise of solving this dilemma of cooperation. In this paper, we trace this promise to a new mechanism of cooperation, programmed reciprocity, defined as coded instructions for automatically returning good for good (positive reciprocity) and ill for ill (negative reciprocity). Programmed reciprocity is rooted in the algorithmic enforcement capability of blockchain networks, defined as the ability to guarantee the execution of the rules of an exchange agreement without a central authority and the possibility of human interference by either of the involved parties. We propose that algorithmic enforcement capability positively affects the viability of cooperation among strangers on the blockchain through programmed reciprocity. This is contingent on the level of contract complexity and blockchain confidence. Our proposed framework extends the nascent literature on blockchain governance with a novel explanation of how programmed reciprocity can enhance cooperation among strangers. In doing this, it also addresses a significant yet unresolved problem in the literature on cooperation in social and economic exchanges.

INTRODUCTION

The question under what conditions cooperation without a central authority will emerge in a world of rational, self-interested parties has long intrigued scientists studying social and economic exchanges (Axelrod & Hamilton, 1981; Maitland, Bryson, & Van de Ven, 1985; Van de Ven, 1976). Cooperation is defined as an act by one party that benefits the interests of another party in an exchange relationship (Sachs, Mueller, Wilcox, & Bull, 2004). Cooperation involves a dilemma: each party, if and when engaging in the exchange, must decide whether to maximize individual interests (defection) or maximize the collective interests of all parties (cooperation) (Axelrod, 2006; Komorita & Parks, 1995; Orbell & Dawes, 1993). As research has pointed out, it is generally more profitable to defect, but if all parties act rationally and do so, all are worse off than if everyone had cooperated to maximize collective interests (Messick et al., 1983). Because of the immense collective payoffs of mutual cooperation in the long term, scholars theorize how the dilemma can be resolved and how cooperation is enhanced (Chen, Chen, & Meindl, 1998), across contexts such as alliances and networks (Bruyaka, Philippe, & Castañer, 2018; Das & Teng, 1998; Gnyawali & Madhavan, 2001; Zeng & Chen, 2003), entrepreneur-venture capitalist relationships (Cable & Shane, 1997), firm-stakeholder relationships (Bridoux & Stoelhorst, 2022), and other economic or social exchange settings (Griesinger, 1990).

Prior research highlights that cooperation is enhanced when exchange parties discover relevant information generated through human reciprocity, either directly through repeated interactions (direct reciprocity) or indirectly through reputation scores (indirect reciprocity), helping them to predict the cooperative strategy of the other party and deal with the dilemma (Falk & Fischbacher, 2006; Zaggl, 2014). For example, research on enhancing cooperation in alliances highlights the role of human reciprocity in building trust by encouraging risk taking

behavior when positive information about the reputation of the exchange party becomes available, helping to establish confidence in partner cooperation (Das & Teng, 1998).

However, as widely documented in the literature (Bolton, Katok, & Ockenfels, 2005; Macy & Skvoretz, 1998; Ockenfels, 1993), human reciprocity falls short in enhancing cooperation among strangers because of the indefinite number of interactions required to discover reliable information that is actionable for enhancing cooperation. For example, Axelrod coined the notion of "evolution of cooperation" to reflect the waxing and waning of cooperation over prolonged time periods (Axelrod, 2006). Strangers, who lack previous interactions, can therefore not rely on human reciprocity mechanisms of cooperation (direct and indirect reciprocity) (Bolton et al., 2005; Macy & Skvoretz, 1998).

In this paper, we present a framework for exploring the role of smart contracts[1] deployed on blockchain[2] networks for enhancing cooperation in economic and social exchanges involving strangers. The diffusion of blockchain-based smart contracts gives rise to algorithmic enforcement capability, defined as the ability to guarantee the execution of the rules of an exchange agreement without a central authority and the possibility of human interference by either of the involved parties (Murray, Kuban, Josefy, & Anderson, 2021; Szabo, 1994; Szabo, 1996; Szabo, 1997; Werbach & Cornell, 2017). We theorize that the algorithmic enforcement capability of blockchain networks positively affects cooperation among strangers in social and economic exchanges (Bolton et al., 2005; Macy & Skvoretz, 1998; Ockenfels, 1993) indirectly

[1] Smart contracts are defined as a set of rules codified in software, which self-executes autonomously when event data confirms that the pre-specified conditions are met (Szabo, 1996).
[2] Blockchain is defined as a distributed ledger that transparently verifies and stores transaction data in sequential order within a peer-to-peer network of participants who all have a unified view of the data and abide by the same rules and processes (Nakamoto, 2008).

through programmed reciprocity, defined as coded instructions for automatically returning good for good (positive reciprocity) and ill for ill (negative reciprocity) (Falk & Fischbacher, 2006).

We introduce programmed reciprocity as a new mechanism of cooperation (Zaggl, 2014), contributing to literatures on blockchain governance (e.g., Beck, Müller-Bloch, & King, 2018; Lumineau, Wang, & Schilke, 2021; Murray et al., 2021) and cooperation among strangers in social and economic exchanges (e.g., Bolton et al., 2005; Butler & Fehr, 2024; Macy & Skvoretz, 1998; Ockenfels, 1993). As our theoretical framework explains, programmed reciprocity works as a mechanism linking algorithmic enforcement capability to enhanced cooperation among strangers on the blockchain. We theorize this relationship as a conditional indirect effect[3] as its strength varies based on the value of a first stage moderated mediation[4], i.e., contract complexity (complex contracts are more difficult to develop for algorithmic enforcement of reciprocal exchange), as well as a second stage moderated mediation, i.e., blockchain confidence (the exchange parties must have confidence in the technology to use smart contracts intuitively and reap the collective benefits of cooperation on the blockchain).

CONCEPTUAL BACKGROUND

The Puzzle: Enhancing Cooperation among Strangers

Cooperation plays a fundamental role for parties engaging in social and economic exchange to get more collective benefits. Cooperation binds individuals together, allowing for the emergence of groups, organizations, communities, institutions, and social structures in general (Sachs et al., 2004). Cooperation plays an indispensable role in the social sciences (e.g.,

[3] A conditional indirect effect is defined as "the magnitude of an indirect effect at a particular value of a moderator (or at particular values of more than one moderator)" (Preacher, Rucker, & Hayes, 2007:186)

[4] Moderated mediation is defined as mediation relations that are contingent on the level of a moderator (James & Brett, 1984). When the moderator affects the strength of the relationship between the predictor and the mediator, the term first stage moderated mediation is used; when it affects the relationship between the mediator and the outcome, the term second stage moderated mediation is used (Hayes, 2017). For an example of a moderated mediation model including both first stage and second stage moderators, see Figure 1 in Li, Shaffer, & Bagger (2015).

Axelrod, 2006; Axelrod & Hamilton, 1981), especially for economics (e.g., Fehr, Gächter, & Kirchsteiger, 1997; Ockenfels, 1993) and management (e.g., Bridoux & Stoelhorst, 2022; Chen et al., 1998; Maitland et al., 1985; Van de Ven, 1976).

There are several reasons why the question how to foster cooperation intrigues management scholars. First, cooperation has been shown to generate long-term payoffs for exchange parties. When everyone is better off collectively through cooperation, exchange parties will compound their payoffs and get more benefits through repeated reciprocal exchange in the future (Axelrod & Dion, 1988). In research on alliances, Zeng & Chen (2003) observe the disadvantage to defect in self-interest as long-term cooperation can generate more collective benefits for alliance partners. Second, cooperation can emerge without a central authority (Axelrod, 2006). This has stimulated scholars to explore new decentralized ways of organizing (e.g., Ellinger, Gregory, Mini, Widjaja, & Henfridsson, 2024; Lumineau et al., 2021).

Reciprocity (direct and indirect) is a key mechanism of cooperation (Zaggl, 2014). At the basic level, reciprocity involves a strategy of returning good for good (mutual cooperation) and ill for ill (mutual defection) (Falk & Fischbacher, 2006; Kahan, 2003). Direct reciprocity (which works on the basis of direct discovery of exchange party information through repeated interactions) and indirect reciprocity (which works on the basis of indirect discovery of reputational exchange party information) are linked to the evolution of cooperation by revealing cooperative strategies of one another (Axelrod & Hamilton, 1981). For instance, Axelrod demonstrates that cooperation can emerge when there is an indefinite number of interactions based on the policy of cooperating first and then reciprocating in the further evolution of the exchange (Axelrod, 2006).

However, as explained by Axelrod and Hamilton (1981), the strategy of unconditional defection is evolutionarily stable too, which means that cooperation is never guaranteed. This had led scholars to explore alternatives to human reciprocity-based mechanisms of cooperation (Axelrod & Dion, 1988), even though none of the documented strategies in the literature offer satisficing answers to the problem of enhancing the initial viability of cooperation among strangers. Despite the vast amount of research on this topic, this represents an enduring puzzle of perennial interest (Butler & Fehr, 2024). Axelrod and Hamilton (1981) originally described this puzzle in terms of the problem of how an evolutionary trend toward cooperative behavior could be started in the first place—the initial viability of cooperation. A decade later, Ockenfels (1993) suggested that cooperation is unlikely in exchange settings involving anonymous parties who do not know each other, something that Macy and Skvoretz (1998) refer to as the problem of transient and anonymous exchange. More recently, Bolton et al. (2005) portrayed the decision of strangers to cooperate as a puzzle that remains unresolved in the literature.

Duffy and Xie's (2016) study provides a good point of departure for exploring the role of blockchain-based smart contracts to increase the initial viability of cooperation among strangers. It highlights that cooperation among strangers becomes less likely when a larger number of human actors are involved, suggesting that behavioral uncertainty stands in the way of cooperation. The authors suggest to solve the problem by introducing robot players, defined as actors that are programmed to participate in a multi-party exchange according to an a priori defined strategy and set of rules. The benefit of placing a robot player in the middle of an exchange is that it makes exchange behavior more predictable. Smart contracts can potentially play that role. As we explain in the following section, the use of blockchain-based smart contracts offers the potential to reduce behavioral uncertainty by eliminating human factors from

the execution of pre-specified rules of the exchange, making cooperation among strangers more likely in the absence of human reciprocity.

Blockchain-Based Smart Contracts

Blockchains with capabilities to run self-executing software-based rules, so-called smart contracts (Buterin, 2014), provide an infrastructure for organizing transactions (Lumineau et al., 2021) and facilitating cooperation in exchanges. A blockchain is a distributed ledger technology that transparently verifies and stores transaction data in sequential order within a peer-to-peer network of participants who all have a unified view of the data and abide by the same rules and processes (Beck et al., 2018; Nakamoto, 2008). There are three key characteristics of blockchain technology: (1) *transparency* (all permissioned parties have a unified and always accessible view of the data stored in the blockchain ledger), (2) *immutability* (the blockchain ledger maintains a permanent, irreversible, and unalterable history of transactions), and (3) *decentralized consensus* (the specific state and set of information stored in the ledger is agreed upon by all agents via protocols, without the need to trust and rely upon a centralized authority).

Following the first wave of blockchain technology innovation (2009-2013), emphasizing the currency use case of blockchain technology (Nakamoto, 2008), the second wave (2014-today) introduced the ability to upload, store, and run smart contracts directly on a blockchain. Smart contracts can be defined as computer programs that automatically execute the codified terms of an exchange agreement upon data-driven confirmation that pre-specified conditions are fulfilled without the need for a central authority (Szabo, 1994; Szabo, 1996).

There are three key characteristics of smart contracts (Cong & He, 2019). First, smart contracts aim at reducing ambiguity through *codified rules and conditions* (Murray et al., 2021). Whereas text-based contracts use human language, which may be subject to diverse

interpretations, smart contracts use formal code and algorithms to make specific components and rules of an exchange agreement (e.g., payments, transfer of asset ownership) machine-readable and self-executable (Szabo, 1996). For example, farmers and seed companies who cooperate for the first time can embed contractual provisions and potential penalties for the violation of intellectual property contained within the plant varieties (created by the seed companies) into software code and deploy this code on the blockchain where it leaves an immutable and transparent transaction record (Rapela & Lehtinen, 2023).

Second, smart contracts follow the principle of *data-driven automation* (Lindebaum, Vesa, & den Hond, 2020; Raisch & Krakowski, 2021). Once pre-specified conditions are met, the software automatically executes and enforces the codified terms of the agreement (Szabo, 1994; Szabo, 1996). In this regard, the underlying blockchain infrastructure guarantees the execution of smart contracts as coded and triggered by event data fed into the smart contracts (Lumineau et al., 2021; Werbach & Cornell, 2017). For example, even small farmers in remote villages who may be a stranger to exchange parties can expect to get paid reliably as enabled by smart contracts that guarantee payments. Before executing the payment, the smart contract will verify that the pre-specified conditions are met including the successfully completed shipment and the promised quality and quality conditions of the traded goods (Law, 2017).

Lastly, smart contracts deployed on a blockchain are designed for *direct exchange*, facilitating instantaneous, rapid transactions on a global, peer-to-peer basis (Catalini & Gans, 2020) enabled by decentralized consensus rather than a centralized authority (Cong & He, 2019; Drummer & Neumann, 2020). Direct exchange assumes that exchangeable assets exist in a state of readiness for immediate transfer over a blockchain, and that an asset's ownership and source of origin can be easily validated (Swan, 2019). For example, smart contracts can be used to

instantaneously determine the provenance of goods, giving actors greater flexibility to act upon timely information and transfer value directly. This can be beneficial to foster cooperation among stakeholders with interests related to health and food safety, environmental sustainability, and ethicality (Marchese & Tomarchio, 2022).

FRAMEWORK FOR EXPLAINING COOPERATION AMONG STRANGERS

We develop a framework (see Figure 1) for explaining the role of blockchain-based smart contracts for increasing the viability of cooperation among strangers, where *cooperation* is defined as an act by one exchange party that benefits also the interests of another party or collective (Sachs et al., 2004).

The framework is based on the following assumptions. First, we assume that strangers are rational and self-interested actors (Bolton et al., 2005; Macy & Skvoretz, 1998; Ockenfels, 1993). Second, strangers have the means and know-how to interact with a machine (blockchain-based smart contract), i.e., a robot player (Duffy & Xie, 2016), to facilitate exchange without relying on a central authority.

The Indirect Effect of Algorithmic Enforcement Capability on Cooperation Among Strangers

Algorithmic enforcement capability. We propose an indirect positive relationship between algorithmic enforcement capability and the viability of cooperation among strangers. We define *algorithmic enforcement capability* as the ability to guarantee the execution of the rules of an exchange agreement without a central authority and the possibility of human interference by either of the involved parties (Murray et al., 2021; Szabo, 1994; Szabo, 1996; Szabo, 1997; Werbach & Cornell, 2017).

The algorithmic enforcement capability of blockchain networks (e.g., Ethereum, Solana, Cardano) varies with the characteristics of smart contracts (codified rules and conditions, data-driven operation, direct exchange) and blockchain infrastructure (transparency, immutability, decentralized consensus), influencing its costs (i.e., the expenditure of resources associated with smart contract development and execution) and reliability (i.e., the degree to which the smart contract development and execution is performed accurately and consistently).

First, algorithmic enforcement capability varies with the characteristics of smart contracts (Szabo, 1996; Werbach & Cornell, 2017). In particular, to enforce an exchange agreement algorithmically, the rules of the agreement (e.g., the terms of payment in exchange for receiving a good or service) as well as the conditions for when these rules are executed (e.g., receiving the good or service within the agreed timeframe) must be formally specified in algorithmic if-then-else language and codified in software using a programming language for smart contract development (e.g., Solidity for smart contracts deployed on the Ethereum blockchain) (*codified rules and conditions*), which can vary in costs and reliability from blockchain to blockchain. For example, Ethereum offers the ERC-20 standard which makes it comparatively more cost-efficient and reliable to develop and execute smart contracts on the Ethereum blockchain compared to other blockchains. Furthermore, the smart contracts must receive data feeds from so-called 'oracles' (e.g., price and event data feeds) to determine when pre-specified conditions are met, autonomously triggering the self-execution of the codified rules (*data-driven operation*). The costs and reliability of the data feeds that are necessary for smart contracts to function varies across blockchains. In addition, to guarantee the data-driven execution of rules, instantaneous execution of transactions to finality, without human interference by any of the involved parties and without a central authority, is critical to complete the exchange according to the rules (e.g.,

making a rapid payment of digital fiat currency or transfer of another type of value stored on the blockchain) (*direct exchange*). The costs and reliability of facilitating direct exchange through smart contracts varies across blockchains, for example in terms of the speed and fees incurred in every transaction over the network (e.g., Solana provides high throughput and low latency, making this blockchain suitable for direct exchange scenarios requiring rapid execution).

Second, as smart contracts are deployed and executed on a blockchain, algorithmic enforcement capability also varies with the blockchain characteristics. Blockchain technology makes transactions resulting from the execution of smart contracts visible and verifiable for all parties (*transparency*) (Werbach, 2018b). Blockchains that offer high transparency (e.g., public, permissionless blockchains such as Solana) makes smart contract monitoring and improvement easier, potentially causing variation in the costs and reliability of algorithmic enforcement capability. In addition, algorithmic enforcement capability involves the idea that the rules of an exchange agreement are executed without possibility of human interference by either of the involved parties, extending beyond the execution phase into the future. However, such execution depends on whether the trail of transactions left behind by smart contracts is permanent, irreversible, and unalterable, helping to meet the expectation of exchange parties that the execution of rules is guaranteed (*immutability*) (Nakamoto, 2008; Szabo, 1997). The greater the immutability of a blockchain, the more reliable it's algorithmic enforcement capability as human interference is prevented from happening. Furthermore, the execution of exchange rules cannot be guaranteed by algorithmic enforcement capability if a central authority controls large parts of the blockchain, as this would introduce risks for human or organizational interference in the execution of rules. Therefore, algorithmic enforcement capability also depends on a so-called consensus mechanism where the state and information stored on the blockchain is agreed upon

by all agents via protocols as opposed to authority (*decentralized consensus*) (Cong & He, 2019). Consensus mechanisms vary across blockchains, influencing both the cost and reliability of their algorithmic enforcement capability.

Robot players. The rise of algorithmic enforcement capability invites the exploration of new mechanisms of cooperation (Axelrod, 2006; Zaggl, 2014) with the potential to inform research on cooperation among strangers in social and economic exchanges (Butler & Fehr, 2024; Macy & Skvoretz, 1998). In what follows, we explore the indirect effect of algorithmic enforcement capability on cooperation among strangers (Bolton et al., 2005; Ockenfels, 1993).

Algorithmic enforcement capability enhances the viability of cooperation among strangers when a strong expectation can be created that good will be returned for good and ill will be returned for ill (Falk & Fischbacher, 2006). This can be achieved by introducing smart contracts as robot players (Duffy & Xie, 2016) into the middle of exchange relationships between unrelated parties to algorithmically enforce reciprocal exchange (Kranton, 1996a, b). As discussed above, smart contracts run on blockchains, and both the characteristics of smart contracts and blockchains can be a source of variation in the costs and reliability of a blockchain network's algorithmic enforcement capability. To the extent that the costs of algorithmic enforcement are low and the reliability is high, we propose that algorithmic enforcement capability positively affects the viability of cooperation among strangers indirectly through programmed reciprocity.

***Programmed reciprocity*.** When the algorithmic enforcement capability of blockchain networks is leveraged to enforce reciprocal exchange based on the agreed-upon and pre-specified rules of the exchange, behavioral uncertainty about the return of the favor following a cooperative act by one exchange party is reduced. Reducing behavioral uncertainty regarding the

cooperative strategy of the other exchange party has a positive effect on the decision of an exchange party to cooperate based on the strong expectation that the other party will cooperate too (Axelrod, 2006). More specifically, strangers are more likely to resolve the social dilemma (Komorita & Parks, 1995; Orbell & Dawes, 1993) in favor of cooperation by acting in a way that benefits not only themselves but also other involved parties (Sachs et al., 2004) if behavioral uncertainty can be reduced, creating a strong expectation based on the work of the machine in the middle (i.e., blockchain-based smart contract) that a cooperative act of one party will follow a cooperative act of another (Falk & Fischbacher, 2006).

This gives rise a new mechanism of enhancing cooperation among strangers based on the application of a blockchain network's algorithmic enforcement capability. We refer to this mechanism as *programmed reciprocity*, defined as coded instructions for automatically returning good for good (positive reciprocity) and ill for ill (negative reciprocity) (Falk & Fischbacher, 2006). Programmed reciprocity, as a function of the cost and reliability of the algorithmic enforcement capability of blockchain networks, provides a solution to the problem of enhancing cooperation among strangers. It is a novel mechanism of cooperation in that it does not require exchange party information discovery through the evolution of cooperation (Axelrod, 2006). Instead, it relies on the a-priori negotiation and specification of rules of the exchange, codification in software, as well as their automatic execution under pre-specified conditions through algorithmic enforcement (Werbach & Cornell, 2017). Programmed reciprocity thus comes with the promise of enhancing cooperation among strangers (Macy & Skvoretz, 1998; Ockenfels, 1993) by guaranteeing algorithmically that reciprocal exchange will be enforced (Kranton, 1996a, b). Rather than trusting each other, exchange parties rely on the machines (i.e., blockchain-based smart contracts) as robot players in the middle of their exchange relationship to

make cooperation viable (Werbach, 2018a). The reason why programmed reciprocity can work as an alternative is because if one exchange party cooperates (e.g., delivers the promised good to the destination address), this party perceives less behavioral uncertainty that the favor will be returned (e.g., making the payment for the good), resulting in mutual cooperation and higher collective benefits that resolves the dilemma of cooperation (Komorita & Parks, 1995). Overall, programmed reciprocity creates incentives to cooperate to maximize the collective interests of exchange parties (Sachs et al., 2004).

Programming positive and negative reciprocity. The use of algorithmic enforcement capability in an exchange setting can help to program both positive and negative reciprocity. It can be used to both (a) transfer value to guarantee payoffs of cooperation, and (b) administer punishments for noncooperative and defective behavior. Algorithmic enforcement capability thus simultaneously increases the promise of reward and the threat of punishment through programmed reciprocity (Andreoni, Harbaugh, & Vesterlund, 2003; Bruni, Panebianco, & Smerilli, 2014; Chen, Sasaki, Brännström, & Dieckmann, 2015; Herold, 2012), facilitating cooperation among strangers within a system of enforceable reciprocal exchange. Overall, algorithmic enforcement capability is positively associated with a higher viability of cooperation among strangers (Bolton et al., 2005; Macy & Skvoretz, 1998) by making cooperative behavior more attractive through guaranteed payoffs and punishments achieved through programmed reciprocity that helps to reduce behavioral uncertainty and increase mutual expectations of cooperative behavior. As stated by Kranton (1996b: 839), "a reciprocal-exchange relationship is "enforceable," that is, constitutes a perfect equilibrium, if and only if each partner has the incentive to produce and receive goods according to the rule and each partner has the incentive to carry out the punishment if any partner reneges."

Illustration. Consider a situation where an importer, based in Country A, wishes to purchase a large shipment of goods from a manufacturer located in Country B. Both parties are unfamiliar with each other and have never engaged in any business transactions before. In traditional trade setups, the lack of trust and familiarity between the importer and the manufacturer can create significant hurdles in establishing a cooperative exchange relationship. The importer may be concerned about the reliability of the manufacturer in delivering the goods as agreed, while the manufacturer may worry about the importer's commitment to making the payment promptly upon receipt of the goods. However, by leveraging blockchain technology and smart contracts, the two parties can establish a system of algorithmic enforcement that significantly reduces these concerns and fosters cooperation. Firstly, they agree on the terms of the trade agreement, including the quantity, quality, and price of the goods, as well as the payment terms and delivery schedule. These terms are then encoded into a smart contract deployed on a blockchain. The smart contract is programmed to execute the exchange automatically once the predefined conditions are met. For instance, upon the manufacturer's confirmation of the shipment, the smart contract releases the payment to the manufacturer. Conversely, if the manufacturer fails to deliver the goods within the agreed timeframe, the smart contract triggers penalties or refunds to the importer.

Viability of cooperation among strangers. This system of algorithmic enforcement provides both parties with a high expectation that the terms of the agreement will be upheld without the need for intermediaries or human intervention. The blockchain ensures that all transactions and contractual obligations are visible and irreversible, further enhancing programmed reciprocity as a mechanism to enhance the viability of cooperation among strangers. Moreover, the threat of penalties or refunds programmed into the smart contract acts as

a powerful incentive for both parties to fulfill their obligations promptly and honestly according to the principles of programmed reciprocity. The importer is assured that they will receive the goods as specified, while the manufacturer knows that they will be compensated for their efforts. As a result, the viability of cooperation between the importer and the manufacturer increases significantly. Both parties are more willing to engage in the transaction knowing that their interests are protected by the algorithmic enforcement capability of the blockchain network using smart contracts. This reduction in behavioral uncertainty and increase in mutual expectations of cooperative behavior ultimately leads to a more cooperative and thus successful exchange, benefiting both parties involved. In sum, we propose:

Proposition 1: Algorithmic enforcement capability positively affects cooperation among strangers indirectly through programmed reciprocity.

Conditional indirect effect. The relationship theorized above is a conditional indirect effect as its strength depends on the values of two moderators (see Figure 1) (Preacher, Rucker, & Hayes, 2007). In particular, the relationships between (a) algorithmic enforcement capability and programmed reciprocity, and (b) programmed reciprocity and cooperation among strangers are each contingent on the level of a moderator, also referred to as first stage (a) and second stage (b) moderated mediation (Hayes, 2017; James & Brett, 1984). In the first stage moderated mediation (a), the strength of the relationship is contingent on the level of contract complexity. The rationale for including this moderator in our theoretical framework is that programmed reciprocity results from developing smart contract code on the decentralized application layer of the blockchain network and instantiating its algorithmic enforcement capability to codify and implement the context-specific reciprocal exchange rules based on reciprocity norms. The ability of performing this programming task is contingent on contract complexity—complex contracts

are more difficult to program for algorithmic enforcement of reciprocal exchange. In the second stage moderated mediation (b), the strength of the relationship is contingent on the level of blockchain confidence. Here, the rationale is that strangers will more likely use smart contracts and cooperate with each other on the blockchain if they have positive expectations about the technology that it will reliably function and perform as programmed. A well-designed smart contract may afford the potential for enhancing cooperation among strangers through programmed reciprocity. However, if nobody is willing to use it due to a lack of confidence in the blockchain technology, cooperation behavior is unlikely.

----- insert Figure 1 about here -----

Contract Complexity

To algorithmically enforce reciprocal exchange and enhance cooperation among strangers on a blockchain, smart contracts must first be developed to instantiate its algorithmic enforcement capability and translate the context-specific reciprocal exchange rules into software code. The success of performing this development activity is contingent upon contract complexity, which determines the degree to which it is feasible and effective to inscribe reciprocity norms into code that executes on the blockchain.

Contract complexity is defined here in terms of the number of contingencies and possible outcomes that have to be accounted for in the reciprocal exchange agreement via computational mapping from the possible states of the world to the specific outcomes in a finite number of steps that can be delegated to a machine (Anderlini & Felli, 1994; Babaioff & Winter, 2014; Melumad, Mookherjee, & Reichelstein, 1997). The degree of contract complexity is important because it affects smart contract development through which the algorithmic enforcement capability of a blockchain network (e.g., Ethereum) is instantiated to achieve programmed reciprocity.

During smart contract development, performed by interdisciplinary teams of smart contract engineers, blockchain professionals, lawyers, domain experts, data scientists, product managers, and masters of a software development methodology, reciprocity norms are inscribed into the technology to increase the potential for cooperation among strangers on the blockchain. Inscription is a socio-technical process described by Latour in terms of the inextricable intertwinement of social practices and technological artifacts (Latour, 1992), and more specifically refers to here as a process whereby norms of reciprocity that can enhance cooperation become inscribed into smart contracts (cf. De Souza, Froehlich, & Dourish, 2005).

The effectiveness of inscription is contingent upon the degree of contract complexity. High contract complexity makes it cognitively difficult to decompose the complexity (Simon, 1962) and map all possible contingencies and outcomes of the agreement to programmable instructions to put the blockchain network's algorithmic enforcement capability into action. High contract complexity can therefore result in human errors and software flaws, weakening the relationship between algorithmic enforcement capability and programmed reciprocity. Low contract complexity, on the other hand, reduces cognitive effort and increases the likelihood that the development team will be able to account for all possible contingencies and outcomes during the design of smart contracts. This, in turn, will likely strengthen the relationship between algorithmic enforcement capability and programmed reciprocity.

For example, consider a scenario in the context of real estate rental agreements to illustrate how a low degree of contract complexity strengthens the relationship between algorithmic enforcement capability and programmed reciprocity by improving inscription. Imagine a platform that facilitates rental agreements between landlords and tenants through smart contracts deployed on a blockchain. These smart contracts are designed to automatically

enforce the terms of the rental agreement, including rent payments, maintenance responsibilities, and lease duration, without the need for intermediaries. In a rental agreement with low contract complexity, the terms of the agreement are straightforward and standardized. For example, the agreement might stipulate a fixed monthly rent amount, a predefined lease duration, and basic maintenance responsibilities for the tenant. With fewer contingencies and possible outcomes to account for, reciprocal interactions can be programmed relatively easily into smart contract code. Landlords and tenants can trust that the terms of the agreement will be enforced accurately, leading to a high viability of cooperation between the two parties.

Conversely, in a rental agreement with high contract complexity, the terms of the agreement are more intricate. For instance, the agreement might include clauses for variable rent payments based on usage, flexible lease durations with options for renewal, and detailed maintenance protocols tailored to the property's specific features. With numerous contingencies and potential outcomes to consider, programming reciprocity into the smart contract becomes significantly more challenging and it will be more incomplete as the likelihood of errors increases. As a result, there is a greater risk of misinterpretation and erroneous execution of the agreement terms, diverging from reciprocity norms. We thus propose that:

Proposition 2: Contract complexity moderates the relationship between algorithmic enforcement capability and programmed reciprocity, such that the relationship is stronger (weaker) when contract complexity is low (high).

Blockchain Confidence

To algorithmically enforce reciprocal exchange and enhance cooperation among strangers on a blockchain, smart contracts must also be used by strangers engaging in economic or social exchange. Thus, the second conditional factor affecting the main relationship in our

model is related to the need for strangers (exchange parties) to actively use smart contracts as "robot players" (Duffy & Xie, 2016) to enact the inscribed reciprocity norms and thus engage in cooperative exchange on the blockchain. Strangers will not cooperate, however, if they do not have a high level of confidence that the blockchain technology, which provides the execution environment for smart contracts (Werbach & Cornell, 2017).

Blockchain confidence is defined as the expectation of proper operations of blockchain systems based on an understanding of their "procedural and rule-based functioning" (De Filippi, Mannan, & Reijers, 2020). Insofar as confidence in blockchain's predictable functioning and performance is high (Lankton, McKnight, & Tripp, 2015; Lippert & Michael Swiercz, 2005; Mcknight, Carter, Thatcher, & Clay, 2011), the need for trust in any central authority, as well as the requirement to trust any of the exchange parties who interact over a blockchain network, is reduced as exchange parties can cooperate with each other through the use of smart contracts (De Filippi et al., 2020). The higher the level of blockchain confidence, the more likely it is that cooperation among strangers is enhanced through programmed reciprocity, because strangers will more likely engage in active usage of smart contracts to cooperatively exchange with each other on the blockchain. In the presence of high levels of blockchain confidence, we should expect more cooperation among strangers to occur as smart contract adoption and use increases, which makes it more likely that strangers reap the collective benefits of cooperation on the blockchain (Sachs et al., 2004). However, as explained in what follows, this logic only holds up if blockchain confidence is not negatively affected by interruptions in the data-driven and automatic operation of smart contracts, which are deployed and run on a blockchain.

Blockchain technology has proven itself over more than a decade to perform and function predictably, helping to establish a high level of blockchain confidence (De Filippi et al., 2020).

From a human reasoning and decision-making standpoint, the positive performance track record of blockchains such as Ethereum induces Type 1 processes that are intuitive, fast, and autonomous (Kahneman & Tversky, 1973). The outcome of Type 1 processes is a reinforcement of enhancing cooperation through programmed reciprocity, because as long as strangers experience fast technology responses to a cooperative act according to the logic of positive reciprocity, they will themselves be more inclined to engage in Type-1 thinking-fast mode and continue using smart contracts intuitively to enable cooperative exchange.

There is always the possibility of a service interruption of a blockchain. This could be caused by a variety of different factors such as the discovery of software vulnerabilities that lead to an exploit or a spike in demand for the blockchain's resources that results in congestion and longer execution times due to delays in validating transactions. A blockchain service interruption is likely to trigger cognitive gear-switching from Type 1 to Type 2 processes, which are reflective, slow, and resource demanding (Pennycook, Fugelsang, & Koehler, 2015). In the first stage of this gear-switching process, the interruption generates an initial response through intuitive Type 1 reasoning that in the second stage of reasoning brings to light a conflict rooted in the mismatch of expectations about the technology versus its actual performance. Finally, in the third stage, it is likely that the interruption, depending on its length and severity, leads to cognitive decoupling, which refers to additional cognitive processing to inhibit and override an intuitive response (i.e., continued use of the smart contract feature in an intuitive response to its usage by the other exchange party) (Pennycook et al., 2015).

Overall, this results in cognitive gear-switching into Type-2 thinking-slow mode, which is likely to reduce blockchain confidence, affecting their decision to cooperate with the robot player in the middle by using smart contracts. This cognitive gear-switching, triggered by an

interruption of blockchain service, reduces blockchain confidence, weakening the relationship between programmed reciprocity and the viability of cooperation among strangers. Conversely, in the absence of blockchain service interruptions, blockchain confidence is expected to go up over time due to positive feedback loops about blockchain's predictable functioning, strengthening the relationship between programmed reciprocity and the viability of cooperation among strangers.

As an example, consider the scenario where Alice and Bob, two strangers, are engaging in a cooperative exchange facilitated by a decentralized finance (DeFi) platform. Bob needs liquidity for a car purchase and wishes to borrow cash with his long-term Bitcoin investment as collateral. Alice wishes to lend the cash to Bob to earn interest. They decide to utilize the smart contract features on the DeFi platform to ensure a secure and transparent transaction without the need for intermediaries. Alice, having high blockchain confidence due to the platform's predictable functioning in the past, initiates the exchange by interacting with the smart contract. She deposits her cash into the smart contract, which locks it until Bob deposits his collateral. Alice trusts that the smart contract will execute the transaction as programmed, thanks to her confidence in the blockchain's ability to perform reliably. Bob, seeing Alice's interaction with the smart contract, decides to cooperate by fulfilling his end of the deal. He deposits his collateral into the smart contract, agrees to the terms stipulated by the smart contract, and receives access to the borrowed cash. As long as the blockchain continues to perform the execution of smart contracts as expected, Alice and Bob will continue to rely on Type 1 fast, intuitive thinking based on reinforced blockchain confidence.

However, an outage occurs in the blockchain infrastructure underlying the DeFi platform, meaning that Alice cannot verify the collateral and Bob cannot access the borrowed cash (or his

collateral). This outage could be caused by various factors, such as software vulnerabilities or network congestion, leading to longer transaction processing times. As a result of this disruption, both Alice and Bob experience cognitive gear-switching, as Alice gets worried about Bob's repayment and Bob gets worried about his collateral. Initially, they rely on Type 1 fast, intuitive thinking, assuming that the blockchain will quickly resolve the issue and execute the smart contract as expected. However, as the outage persists, they transition to Type 2 slow, reflective thinking. In the second stage of reasoning, they confront the mismatch between their expectations of seamless blockchain operation and the reality of the disruption. They begin to question the proper operation of the blockchain and the smart contracts running on it. Finally, in the third stage, they undergo cognitive decoupling as they grapple with the decision to continue trusting the blockchain technology despite the interruption. They weigh the risks and uncertainties associated with the disrupted blockchain against the benefits of cooperation.

Ultimately, low blockchain confidence weakens the relationship between programmed reciprocity and cooperation among strangers. Alice and Bob become less inclined to interact with the smart contracts on the DeFi platform, fearing potential errors or delays in transaction execution. This reluctance to engage undermines the effectiveness of programmed reciprocity in enhancing cooperation among strangers on the blockchain. If, on the other hand, the blockchain-based smart contracts on the DeFi platform have a long history of functioning reliably without major service interruptions, Type 1 processes and enhanced cooperation through programmed reciprocity is reinforced over time as blockchain confidence builds up. In sum, we propose that:

Proposition 3: Blockchain confidence moderates the relationship between programmed reciprocity and cooperation among strangers, such that the relationship is stronger (weaker) when blockchain confidence is high (low).

DISCUSSION

Our contribution is a framework for explaining how cooperation among strangers can be enhanced by leveraging the algorithmic enforcement capability of blockchain networks to program positive and negative reciprocity (cf. Falk & Fischbacher, 2006), enhancing cooperation in reciprocal-exchange relationships (cf. Kranton, 1996a; Kranton, 1996b). Our framework extends the nascent literature on blockchain governance (e.g., Beck et al., 2018; Lumineau et al., 2021; Murray et al., 2021) with a novel explanation of how programmed reciprocity can enhance cooperation among strangers, addressing an important yet unresolved puzzle in the literature on cooperation in social and economic exchanges (e.g., Axelrod & Dion, 1988; Axelrod & Hamilton, 1981; Bolton et al., 2005; Macy & Skvoretz, 1998; Ockenfels, 1993; Sachs et al., 2004). Several implications for research follow from our theory development.

Implications for Research

First, we propose programmed reciprocity as a new mechanism of cooperation among strangers. Programmed reciprocity is coded instructions for automatically returning good for good (positive reciprocity) and ill for ill (negative reciprocity). It complements human reciprocity mechanisms for cooperation such as direct and indirect reciprocity (Zaggl, 2014). In the context of cooperation among strangers, the use of human reciprocity, whether direct or indirect reciprocity, is limited (Macy & Skvoretz, 1998; Ockenfels, 1993), because trust and reputation of an exchange party cannot develop without significant exchange party information discovery (Axelrod 1984). Consider that direct reciprocity is based on the idea that repeated interactions between individuals who behave cooperatively creates the emotion of gratitude, which humans can memorize and consider in forming future predictions about the behavior of others (Axelrod, 2006; Axelrod & Hamilton, 1981). Direct reciprocity thus involves giving

favors to one another and can be simplified to 'I help you and you help me' (Nowak & Sigmund, 2005; Taylor & Nowak, 2009). Similarly, indirect reciprocity is based on reputation or social status determined by an actor's past behavior. Reputation may serve as an effective signal of future cooperative behavior depending on environmental conditions that humans with their high cognitive abilities can analyze and consider when judging the reliability of the signal (Carter, 2014; Zaggl, 2014). Indirect reciprocity is thus based on trust and involves the logic of 'I help you and somebody else helps me' (Nowak & Sigmund, 2005; Taylor & Nowak, 2009). However, in the context of strangers, who are unrelated exchange parties, favors and trust are ineffective as mechanisms for enhancing cooperation (Macy & Skvoretz, 1998). In this context, programmed reciprocity provides a more viable option than direct and indirect reciprocity to enhance cooperation. Overall, rather than 'I help you and you help me' or 'I help you and somebody else helps me' logic (Nowak & Sigmund, 2005; Taylor & Nowak, 2009), programmed reciprocity is based on 'we program a-priori and trust the machine to execute to finality' logic. This new programmed machine logic provides an alternative mechanism of cooperation that is particularly suitable for strangers.

Second, in relation to techno-leviathan visions about 'trustless' technology (Scott, 2015), our framework highlights that blockchain confidence is a significant condition for programmed reciprocity to work as a mechanism for enhancing cooperation among strangers (Werbach, 2018b). Cooperation among strangers without confidence that the blockchain network will reliably and predictably execute the rules of the reciprocal-exchange agreement is unlikely (Lumineau, Schilke, & Wang, 2022). Similar to other technologies, blockchain confidence is associated with expectations about reliable and predictable functioning as well as high performance (Lankton et al., 2015; Lippert & Michael Swiercz, 2005; Mcknight et al., 2011),

and it provides the key benefit that there is no need for trust in any central authority or any of the exchange parties directly (De Filippi et al., 2020). In this regard, if blockchain confidence is high, the techno-leviathan vision about blockchain as a trustless technology (Scott, 2015) could in fact come into fruition as the mechanism of programmed reciprocity is successfully activated under the condition of low contract complexity.

Third, our research highlights that low contract complexity strengthens the positive relationship between algorithmic enforcement capability and programmed reciprocity, which mediates its relationship to cooperation among strangers. As contract complexity increases, the risk of implementing incomplete contracts increases, which, in turn, will lead to lower degree of programmed reciprocity. There are at least two aspects that increase the likelihood of errors in the specification of a contractual agreement and address the common issue of ambiguity as discussed in incomplete contract theory (Hart, 1988). First, even though smart contracts can decompose the complexity of an exchange agreement into specific code-based, contract-like components (e.g., payments, delineation of property rights) (Szabo, 1996) and encourage parties to precisely specify the obligations of each party (Cong & He, 2019; Murray et al., 2021), it is virtually impossible to make a contract for a complex exchange agreement entirely complete. Given the algorithmic enforcement capability of a blockchain, the risk of introducing contract incompleteness when programming a complex smart contract will therefore hamper the positive effect of programmed reciprocity. Second, even in cases when the smart contract is complete, time might gradually introduce the risk of incompleteness as the environmental conditions assumed in the smart contract is changing while the contract cannot be adapted once committed to the blockchain.

Limitations of Algorithmic Enforcement Capability

There are at least three limitations of a blockchain network's algorithmic enforcement capability. First, algorithmic enforcement is realistically not a substitute for contract law (Werbach & Cornell, 2017), despite what the term smart *contracts* implies (Szabo, 1996). Smart contracting to leverage the algorithmic enforcement capability of blockchain networks and enhance cooperation among strangers should thus be viewed as a complement to contract law that has the potential to reduce the need for contract litigation due to the strong expectations and incentives created by programmed reciprocity for exchange parties to cooperate. Furthermore, despite the potential ability for governments to model laws and regulations with objectively verifiable parameters and incorporate them into code, the resulting "lex cryptographia" (De Filippi & Hassan, 2018) does not necessarily have legal legitimacy (Becker, 2022). In sum, establishing legal legitimacy of a blockchain network's algorithmic enforcement capability entails an element of uncertainty.

The second limitation of a blockchain network's algorithmic enforcement capability is related to its costs and reliability. In fact, proper cost-benefit and risk-reward analysis are advisable. For example, the data-driven operation of smart contracts presents potential software vulnerabilities that could lead to costly hacks and exploits that reduce the integrity of the blockchain. Examples include compromised data feeds (Zhang, Cecchetti, Croman, Juels, & Shi, 2016) (e.g., on October 12, 2023, decentralized finance protocol Platypus suffered a flash loan exploit based on a compromised price feed resulting in a loss of over $2 million); reentrancy attacks, which occur whenever a smart contract, interacting with another smart contract, calls back the original contract (re-entrant call) before the state in the original contract has been fully updated (Samreen & Alalfi, 2020) (e.g., the famous attack of 'The DAO' in 2016 involving an attacker draining funds from the target by recursively calling the target's withdraw function,

causing a loss of $60 million); and software code vulnerabilities involving faulty access control (Zamani, He, & Phillips, 2020) (e.g., in November 2017, an anonymous user was able to make itself the owner of a critical software component in the Parity Multisig Wallet, allowing the user to steal over $30 million worth of ETH).

The third limitation relates to artificial intelligence (AI). Smart contracts are not smart in the sense of having learning capability. Consider that 'data-driven learning,' as a characteristic of AI (Gregory, Henfridsson, Kaganer, & Kyriakou, 2021), is not the same as the 'data-driven operation' of smart contracts (Murray et al., 2021; Szabo, 1996). Smart contracts are oftentimes written in procedural code that is rigid and inflexible by design. On the one hand, codifying rules and conditions in this manner creates a sense of stability and reduces perceived behavioral uncertainty, as explained by our theoretical framework. On the other hand, AI-driven smart contracting is already emerging as a new phenomenon and the subject of recent research (Khan, Sholla, Assad, & Shafi, 2023), suggesting the convergence of different digital technologies, including AI providing learning functions, the Internet providing data, and blockchain networks providing algorithmic enforcement capabilities. For example, a relevant use case is to strengthen the security of smart contracts using AI (Krichen, 2023).

Future Research

Future research on the interplay between human and programmed reciprocity mechanisms would be worthwhile. In particular, in complex organizational settings without shared goals, the alignment of goals may have to be fostered through human reciprocity mechanisms (e.g., repeated interactions among members of different organizations to achieve cooperation in aligning goals and interests) and supporting relational governance, as to build the foundation for co-design and implementation of programmed reciprocity in social and economic

exchange (e.g., implementing automated origin verification and payment processes on the blockchain using smart contracts). Complex settings require examination of how human and programmed reciprocity mechanisms substitute or complement each other under varying conditions.

Another avenue for future work is to theorize and examine the convergence of blockchain-based smart contracts with other technologies, including sensor technologies that provide smart contracts with data, AI technologies that enable smart contracts to learn and improve over time, and decentralized platform architectures that enable unique models of value creation, appropriation, and redistribution across multiple stakeholder interests. For instance, scholars could explore when and how the convergence of these technologies impacts established platform models and its governance and centralized architectures and data assets. Even though it remains to be seen, this convergence could give rise to novel forms of organizing with digital ownership built into it by design.

REFERENCES

Anderlini, L., & Felli, L. 1994. Incomplete written contracts: Undescribable states of nature. ***The Quarterly Journal of Economics***, 109(4): 1085-1124.

Andreoni, J., Harbaugh, W., & Vesterlund, L. 2003. The carrot or the stick: Rewards, punishments, and cooperation. ***American Economic Review***, 93(3): 893-902.

Axelrod, R. 2006. ***The Evolution of Cooperation: Revised Edition***: Basic Books.

Axelrod, R., & Dion, D. 1988. The further evolution of cooperation. ***Science***, 242(4884): 1385-1390.

Axelrod, R., & Hamilton, W. D. 1981. The Evolution of Cooperation. ***Science***, 211(4489): 1390-1396.

Babaioff, M., & Winter, E. 2014. Contract complexity. ***EC 14***: 911.

Beck, R., Müller-Bloch, C., & King, J. L. 2018. Governance in the Blockchain Economy: A Framework and Research Agenda. ***Journal of the Association for Information Systems***: 1020-1034.

Becker, K. 2022. Blockchain matters—lex Cryptographia and the displacement of legal symbolics and imaginaries. ***Law and Critique***, 33(2): 113-130.

Bolton, G. E., Katok, E., & Ockenfels, A. 2005. Cooperation among strangers with limited information about reputation. ***Journal of Public Economics***, 89(8): 1457-1468.

Bridoux, F., & Stoelhorst, J. W. 2022. Stakeholder Governance: Solving the Collective Action Problems in Joint Value Creation. ***Academy of Management Review***, 47(2): 214-236.
Bruni, L., Panebianco, F., & Smerilli, A. 2014. Beyond carrots and sticks: How cooperation and its rewards evolve together. ***Review of Social Economy***, 72(1): 55-82.
Bruyaka, O., Philippe, D., & Castañer, X. 2018. Run away or stick together? The impact of organization-specific adverse events on alliance partner defection. ***Academy of Management Review***, 43(3): 445-469.
Buterin, V. 2014. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform.
Butler, J. V., & Fehr, D. 2024. The causal effect of cultural identity on cooperation. ***Journal of Economic Behavior & Organization***, 221: 134-147.
Cable, D. M., & Shane, S. 1997. A prisoner's dilemma approach to entrepreneur-venture capitalist relationships. ***Academy of Management review***, 22(1): 142-176.
Carter, G. 2014. The Reciprocity Controversy. ***Animal Behavior and Cognition***, 1(3).
Catalini, C., & Gans, J. S. 2020. Some simple economics of the blockchain. ***Communications of the ACM***, 63(7): 80-90.
Chen, C. C., Chen, X.-P., & Meindl, J. R. 1998. How can cooperation be fostered? The cultural effects of individualism-collectivism. ***Academy of management review***, 23(2): 285-304.
Chen, X., Sasaki, T., Brännström, Å., & Dieckmann, U. 2015. First carrot, then stick: how the adaptive hybridization of incentives promotes cooperation. ***Journal of the royal society interface***, 12(102): 20140935.
Cong, L. W., & He, Z. 2019. Blockchain Disruption and Smart Contracts. ***The Review of Financial Studies***, 32(5): 1754-1797.
Das, T. K., & Teng, B.-S. 1998. Between trust and control: Developing confidence in partner cooperation in alliances. ***Academy of management review***, 23(3): 491-512.
De Filippi, P., & Hassan, S. 2018. Blockchain technology as a regulatory technology: From code is law to law is code. ***arXiv preprint arXiv:1801.02507***.
De Filippi, P., Mannan, M., & Reijers, W. 2020. Blockchain as a confidence machine: The problem of trust & challenges of governance. ***Technology in Society***, 62: 101284.
De Souza, C., Froehlich, J., & Dourish, P. 2005. ***Seeking the source: software source code as a social and technical artifact***. Paper presented at the Proceedings of the 2005 ACM International Conference on Supporting Group Work.
Drummer, D., & Neumann, D. 2020. Is code law? Current legal and technical adoption issues and remedies for blockchain-enabled smart contracts. ***Journal of Information Technology***, 35(4): 337-360.
Duffy, J., & Xie, H. 2016. Group size and cooperation among strangers. ***Journal of Economic Behavior & Organization***, 126: 55-74.
Ellinger, E., Gregory, R. W., Mini, T., Widjaja, T., & Henfridsson, O. 2024. The Transformational Potential of Decentralized Autonomous Organizations. ***MIS Quarterly***, 48(1): 245-272.
Falk, A., & Fischbacher, U. 2006. A theory of reciprocity. ***Games and economic behavior***, 54(2): 293-315.
Fehr, E., Gächter, S., & Kirchsteiger, G. 1997. Reciprocity as a contract enforcement device: Experimental evidence. ***Econometrica: journal of the Econometric Society***: 833-860.
Gnyawali, D. R., & Madhavan, R. 2001. Cooperative networks and competitive dynamics: A structural embeddedness perspective. ***Academy of Management Review***, 26(3): 431-445.
Gregory, R. W., Henfridsson, O., Kaganer, E., & Kyriakou, H. 2021. The Role of Artificial Intelligence and Data Network Effects for Creating User Value. ***Academy of Management Review***, 46(3): 534–551.
Griesinger, D. W. 1990. The human side of economic organization. ***Academy of Management Review***, 15(3): 478-499.

Hart, O. D. 1988. Incomplete Contracts and the Theory of the Firm. ***Journal of Law, Economics, and Organization***, 4(1): 119-139.
Hayes, A. F. 2017. Partial, conditional, and moderated moderated mediation: Quantification, inference, and interpretation. ***Communication monographs***, 85(1): 4-40.
Herold, F. 2012. Carrot or stick? The evolution of reciprocal preferences in a haystack model. ***American Economic Review***, 102(2): 914-940.
James, L. R., & Brett, J. M. 1984. Mediators, moderators, and tests for mediation. ***Journal of applied psychology***, 69(2): 307.
Kahan, D. M. 2003. The Logic of Reciprocity: Trust, Collective Action, and Law. ***Michigan Law Review***, 102(1): 71-103.
Kahneman, D., & Tversky, A. 1973. On the psychology of prediction. ***Psychological review***, 80(4): 237.
Khan, M., Sholla, S., Assad, A., & Shafi, H. 2023. AI-Powered Smart Contracts: The Dawn of Web 4. ***Authorea Preprints***.
Komorita, S. S., & Parks, C. D. 1995. Interpersonal relations: Mixed-motive interaction. ***Annual review of psychology***, 46(1): 183-207.
Kranton, R. E. 1996a. The Formation of Cooperative Relationships. ***The Journal of Law, Economics, and Organization***, 12(1): 214-233.
Kranton, R. E. 1996b. Reciprocal Exchange: A Self-Sustaining System. ***The American Economic Review***, 86(4): 830-851.
Krichen, M. 2023. Strengthening the security of smart contracts through the power of artificial intelligence. ***Computers***, 12(5): 107.
Lankton, N. K., McKnight, D. H., & Tripp, J. 2015. Technology, humanness, and trust: Rethinking trust in technology. ***Journal of the Association for Information Systems***, 16(10): 1.
Latour, B. 1992. Where are the missing masses? The sociology of a few mundane artifacts. ***Shaping technology/building society: Studies in sociotechnical change***, 1: 225-258.
Law, A. 2017. ***Smart Contracts and their Application in Supply Chain Management***. MIT, MIT Geospatial Data Center.
Lindebaum, D., Vesa, M., & den Hond, F. 2020. Insights From "The Machine Stops" to Better Understand Rational Assumptions in Algorithmic Decision Making and Its Implications for Organizations. ***Academy of Management Review***, 45(1): 247-263.
Lippert, S. K., & Michael Swiercz, P. 2005. Human resource information systems (HRIS) and technology trust. ***Journal of information science***, 31(5): 340-353.
Lumineau, F., Schilke, O., & Wang, W. 2022. Organizational trust in the age of the fourth industrial revolution: Shifts in the nature, production, and targets of trust. ***Journal of Management Inquiry***, forthcoming.
Lumineau, F., Wang, W., & Schilke, O. 2021. Blockchain Governance—A New Way of Organizing Collaborations? ***Organization Science***, 32(2): 500-521.
Macy, M. W., & Skvoretz, J. 1998. The evolution of trust and cooperation between strangers: A computational model. ***American Sociological Review***: 638-660.
Maitland, I., Bryson, J., & Van de Ven, A. 1985. Sociologists, economists, and opportunism. ***Academy of Management Review***, 10(1): 59-65.
Marchese, A., & Tomarchio, O. 2022. A Blockchain-Based System for Agri-Food Supply Chain Traceability Management. ***SN Computer Science***, 3: 279.
Mcknight, D. H., Carter, M., Thatcher, J. B., & Clay, P. F. 2011. Trust in a specific technology: An investigation of its components and measures. ***ACM Transactions on management information systems (TMIS)***, 2(2): 1-25.
Melumad, N., Mookherjee, D., & Reichelstein, S. 1997. Contract complexity, incentives, and the value of delegation. ***Journal of Economics & Management Strategy***, 6(1): 257-289.

Messick, D. M., Wilke, H., Brewer, M. B., Kramer, R. M., Zemke, P. E., & Lui, L. 1983. Individual adaptations and structural change as solutions to social dilemmas. ***Journal of personality and social psychology***, 44(2): 294.
Murray, A., Kuban, S., Josefy, M., & Anderson, J. 2021. Contracting in the Smart Era: The Implications of Blockchain and Decentralized Autonomous Organizations for Contracting and Corporate Governance. ***Academy of Management Perspectives***, 35(4): 622-641.
Nakamoto, S. 2008. Bitcoin: A peer-to-peer electronic cash system. ***Decentralized Business Review***.
Nowak, M. A., & Sigmund, K. 2005. Evolution of indirect reciprocity. ***Nature***, 437(7063): 1291-1298.
Ockenfels, P. 1993. Cooperation in prisoners' dilemma: An evolutionary approach. ***European Journal of Political Economy***, 9(4): 567-579.
Orbell, J. M., & Dawes, R. M. 1993. Social welfare, cooperators' advantage, and the option of not playing the game. ***American sociological review***: 787-800.
Pennycook, G., Fugelsang, J. A., & Koehler, D. J. 2015. What makes us think? A three-stage dual-process model of analytic engagement. ***Cogn Psychol***, 80: 34-72.
Preacher, K. J., Rucker, D. D., & Hayes, A. F. 2007. Addressing Moderated Mediation Hypotheses: Theory, Methods, and Prescriptions. ***Multivariate Behavioral Research***, 42(1): 185-227.
Raisch, S., & Krakowski, S. 2021. Artificial Intelligence and Management: The Automation-Augmentation Paradox. ***Academy of Management Review***, 46(1): 192-210.
Rapela, M., & Lehtinen, L. 2023. Non-fungible Plant Variety (NFPV): A Proposal for an Innovative Way of Controlling Seed Trade of Protected Plant Varieties. ***International Journal of Innovative Science and Research Technology (IJISRT)***, 8(1): 1357-1367.
Sachs, J. L., Mueller, U. G., Wilcox, T. P., & Bull, J. J. 2004. The evolution of cooperation. ***The Quarterly review of biology***, 79(2): 135-160.
Samreen, N. F., & Alalfi, M. H. 2020. ***Reentrancy vulnerability identification in ethereum smart contracts***. Paper presented at the 2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE).
Scott, B. 2015. Visions of a techno-leviathan: The politics of the bitcoin blockchain.
Simon, H. A. 1962. The architecture of complexity. ***Proceedings of the American Philosophical Society***, 106(6): 467-482.
Swan, M. 2019. Blockchain Economic Networks: Economic Network Theory—Systemic Risk and Blockchain Technology. In H. Treiblmaier, & R. Beck (Eds.), ***Business Transformation through Blockchain: Volume I***: 3-45. Cham: Springer International Publishing.
Szabo, N. 1994. Smart Contracts, Vol. 2022. https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html.
Szabo, N. 1996. Smart contracts: building blocks for digital markets. ***EXTROPY: The Journal of Transhumanist Thought***, 18(2): 28.
Szabo, N. 1997. Formalizing and securing relationships on public networks. ***First monday***, 2(9).
Taylor, C., & Nowak, M. A. 2009. How to evolve cooperation, ***Games, groups, and the global good***: 41-56: Springer.
Van de Ven, A. H. 1976. On the nature, formation, and maintenance of relations among organizations. ***Academy of management review***, 1(4): 24-36.
Werbach, K. 2018a. ***The Blockchain and the New Architecture of Trust***. Cambridge, MA: MIT Press.
Werbach, K. 2018b. Trust, but Verify. ***Berkeley Technology Law Journal***, 33(2): 487-550.
Werbach, K., & Cornell, N. 2017. Contracts Ex Machina. ***Duke Law Journal***, 67: 313-382.

Zaggl, M. A. 2014. Eleven mechanisms for the evolution of cooperation. ***Journal of Institutional Economics***, 10(2): 197-230.

Zamani, E., He, Y., & Phillips, M. 2020. On the security risks of the blockchain. ***Journal of Computer Information Systems***, 60(6): 495-506.

Zeng, M., & Chen, X.-P. 2003. Achieving cooperation in multiparty alliances: A social dilemma approach to partnership management. ***Academy of Management Review***, 28(4): 587-605.

Zhang, F., Cecchetti, E., Croman, K., Juels, A., & Shi, E. 2016. ***Town crier: An authenticated data feed for smart contracts***. Paper presented at the Proceedings of the 2016 aCM sIGSAC conference on computer and communications security.

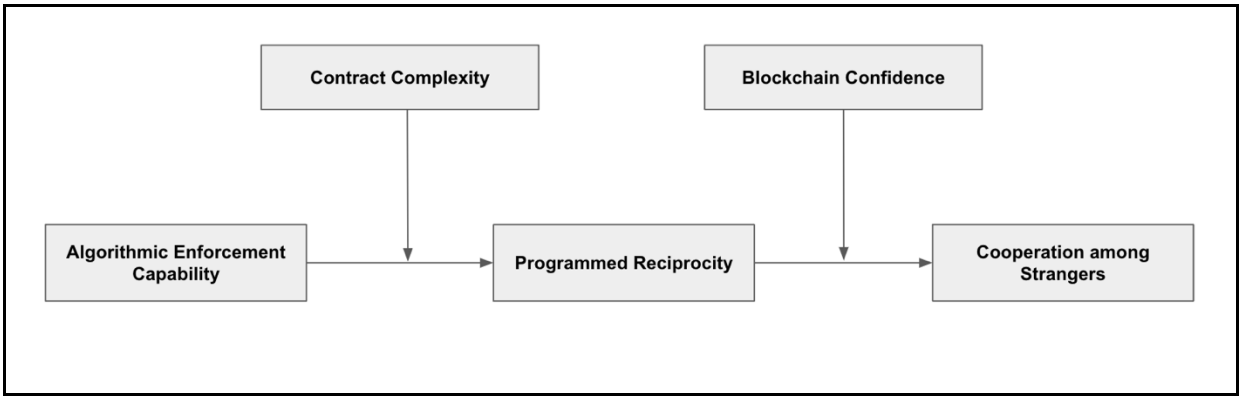

FIGURE 1: Algorithmic Enforcement of Reciprocal Exchange to Enhance Cooperation Among Strangers