Guest Editorial Preface

Blockchain and Smart Contract:
A Review

Hemang Subramanian, Florida International University, USA
Rong Liu, Stevens Institute of Technology, USA

INTRODUCTION

Every 15 years, computing has undergone a paradigm shift. Today, close to the twelfth anniversary of the public launch of Bitcoin, we have the "trust" based computing era, wherein computing systems and applications can replace a centralized authority and execute programs on a censor-proof decentralized and Turing-complete, distributed global computing system (Nakomoto, 2008). Such "trust" based computing has implications for many industries such as information systems, financial and banking systems, healthcare systems, supply-chain systems, accounting methods, and business contracts. Beyond improving efficiencies and lowering costs in matching transacting entities, and facilitating faster, more-secure, soft-real time transactions, Blockchain also enables a completely new set of automated rule-based functions for smart contracts (Subramanian, 2018). In addition to the spawning of a completely different asset class of cryptocurrencies (Liu & Tsyvinski, 2018), the blockchain technology facilitates many different functions such as immutability, security, byzantine fault-tolerance and distributed transaction validation.

The special issue in Blockchain and Smart Contract for the Journal of Database Management invited paper submissions that studied interesting questions pertaining to the Blockchain and Smart Contract in several research areas. First, Blockchain, as a class of distributed-transaction systems, encompasses several key technologies, including distributed ledger, cryptography, consensus protocols, and smart contracts. These components are leveraged together to achieve desirable properties such as disintermediation, immutability, transparency, and automation. It would be essential for researchers and practitioners to understand these underlying components, their variations, and the pros and cons of each variation so that the community can appreciate Blockchain better. Second, the research community would also be interested in exploring innovative enterprise applications using private or public blockchain architectures. Although Blockchain is gaining more momentum in industries, very few blockchain applications are operational. Moreover, several challenges have been revealed through early experiments, including efficiency, scalability, interoperability, complexity, and regulation. Therefore, systematic approaches are needed to help firms make various technological or economic decisions in designing and operationalizing blockchain based systems. Finally, it would be very helpful to provide thorough analyses on existing challenges on blockchain systems (e.g., Bitcoin, Ethereum) and insightful discussion of algorithms and mechanisms to address these challenges.

In this paper, we will first provide a review of the foundational technologies of Blockchain. In particular, we emphasize one of the key components, smart contracts, which automate business agreements as enforceable program code without a central authority, because we envision that smart contracts would be a trending research area for scholars in Information Systems (IS). Then we

introduce several innovative architectures and applications, including the four papers published in this special issue. Finally, we discuss challenges with Blockchain and Smart contract, referred to as the Blockchain Trilemma, and identify research opportunities in each of these areas.

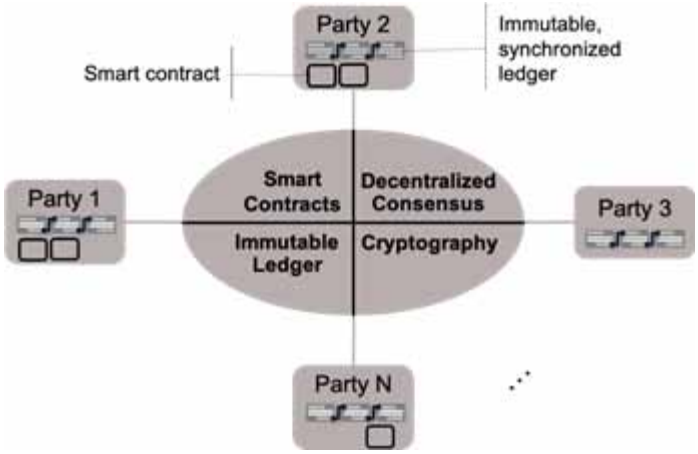
BLOCKCHAIN FUNDAMENTALS

Blockchain has been considered as a promising technology to enable decentralize E-commerce marketplace (Subramanian 2017). As a network model, a blockchain system connects participants to make transactions without the need for preestablished trust as usually required in traditional marketplaces. Instead, it provides "meta trust" (Babich and Gilles, 2020) where a participant can rely on consensus mechanisms, immutable ledgers, smart contract algorithms, and underlying infrastructure of protocols to substantiate trades with even unknown or anonymous others. Figure 1 shows an overall architecture of a blockchain system. This network contains several parties each of which keeps an identical copy of the ledger. Parties engage in business transactions through smart contracts. There are four core components involved to make transactions happen in this distributed environment: smart contracts, decentralized consensus mechanisms, immutable ledger, and cryptography protocols. Next, we describe each component with details.

First, on a blockchain network, parties make transactions via smart contracts, i.e. self-executing code installed on each node of the blockchain network that implements the terms of agreements between parties. When deployed, a smart contract automatically executes when specific logical conditions are met. All read and write operations on the blockchain can only be performed by invoking smart contracts. Smart contracts may also connect to external data sources, referred to as "oracles," which could either be database instances or could be crowdsourced from other smart contracts. For example, in the case of a "flight delay" insurance, the blockchain can validate an event of interest i.e., a delayed flight from flight monitoring systems, and process insured customer claims automatically (Sheth and Subramanian, 2019). Smart contracts can be written in a language such as Solidity, Java or Go. We will describe smart contracts with more details in Section 3 and their impact on business.

A new transaction created out of smart contract execution is assembled into a block. Then the block is validated through consensus, a mechanism by which all parties in a blockchain network reach agreement about each new block to be added to the ledger and maintain an identical copy of the ledger on each node. The most famous consensus is the proof-of-work algorithm featured in the Bitcoin and Ethereum networks. By this algorithm, nodes in the network, called miners, race to find a nonce, a

Figure 1. Overall Architecture of Blockchain Systems



random number satisfying a criterion predefined in a new block, through brute-force search. This search process is considered as work and the nonce is the proof of this work. Only winner of the race can append the block into the ledger. Since a miner's odds of winning the race is tightly associated with its computing power, this consensus provides the blockchain network resilience to Sybil attacks (Kolb et al. 2020). Although a strong consensus mechanism, proof-of-work is widely criticized because tremendous energy is consumed during the search for nonces without serving other useful purpose. In addition, the race to find a nonce is nondeterministic and may involve considerable delay, making this consensus infeasible for high frequency transactions. Therefore, alternative consensus mechanisms, for example, proof of stake, proof of elapsed time, and proof of authority, were invented to overcome these disadvantages (Kolb et al. 2020). For example, proof-of-authority is a consensus mechanism often used by permissioned or consortium blockchain systems, where parties must obtain permission through some membership protocols in order to join. With appropriate authorization, a subset of nodes on a network are designated as validators to approve new blocks. This simple consensus enables fast processing of transactions and avoids the computational effort of proof of work, under the assumption that validator nodes can be fully trusted in this permissioned network.

The ledger records history of all transactions and is a write-once only log produced after transactions ratified by parties. In the ledger, each successive block contains a signature of the previous block, making any change to a block will essentially invalidate the block signature stored in the successive blocks. Such a chain structure along with cryptography can ensure the immutability property of the ledger. Each party maintains a copy of the ledger and these copies are synchronized constantly to ensure there is only one "ground truth" of the transaction history. Besides block signatures, cryptography has been extensively used to identify parties, authenticate transactions, and maintain transaction privacy and secrecy.

In general, blockchain systems can be divided into two categories, permissionless and permissioned. In a Permissionless blockchain system, such as Ethereum, everyone can join the network. A strong consensus, i.e. proof of work or similar ones, is required since parties are anonymous without enough trust among them. As a result, the scalability of the system may be limited due to the latency incurred in reaching consensus. In addition, privacy is also raised as a concern in Permissionless blockchain (Kolb et al. 2020). For example, when two people make an exchange of Bitcoins, everyone on the network can see their public keys and the number of tokens transferred (Kolb et al. 2020). Moreover, although a public key used by a transaction does not explicitly identify its owner, there is a high risk that transactions can be deanonymized by analyzing the patterns between the keys used in a series of transactions.

On the other hand, permissioned Blockchain systems can better address enterprises' concerns about transaction security, privacy and scalability (Kumar et al. 2020). Only parties with proper authorization can join a permissioned network and they have more creditability. As a result, weaker consensus mechanisms, such as proof-by-stake or proof-by-authority, may be sufficient to maintain transaction integrity. These weak censuses mechanisms also lead to better scalability and transaction efficiency. Also, transactions can only be viewed by authorized users to ensure privacy.

Despite of their differences, both types of blockchain systems enjoy a number of intrinsic properties which are attractive to new business applications. Table 1 summarizes some of these features. For example, due to its immutability, the ledger can serve as a ground truth for transaction history, which provides transaction traceability or provenance for little extra cost. This property makes blockchains highly attractive for supply chains, in particular, in the areas of global trade, food supply chain, and high-value goods. Another interesting feature comes from "meta trust" where parties trust the underlying Blockchain infrastructure without the need for preestablished trust as seen in traditional transactions. Moreover, smart contracts implement business logic transparently and their execution is fully automated. In a broad sense, Blockchain and Smart Contract can be considered as a kind of automation technologies (Wang and Siau 2019) that automates business transactions,

Table 1. Features of Blockchain and Smart Contracts

Features	Description
Decentralized execution	Smart contract code executes on a global decentralized network of computer nodes and virtual machines which validate conditions of execution through consensus mechanisms
Immutability	The ledger stores transaction blocks in a chain structure to ensure that transactions are immutable once committed. Similarly, smart contracts, once deployed, are immutable. If contracts have to be changed, then new smart contracts have to be written and redeployed.
Provenance	An immutable ledger acts as the unique source of ground truth and then offers traceability of transactions.
Transparency	Smart contracts are not only published through white papers, but are also visible in blockchain code, and can be executed, verified and analyzed by third parties.
Automated verification	Smart contract logic and execution are automated, ie., the programming logic automatically runs within a virtual machine-like environment.
Power and meta trust (Babich and Gilles, 2020, Kumar et al. 2020)	Power is disseminated among parties; parties trust the underlying Blockchain infrastructure without the need for preestablished trust

which otherwise would have to undergo complicate verification processes if transacting parties lacks sufficient trust between each other.

SMART CONTRACTS: REVOLUTIONIZING CONTRACTING MECHANISM

Nick Szabo, defined smart contracts as a secure, machine-readable and executable program that can automate specified procedures, including those used in legal contexts. The basic premise behind smart contracts is that different types of contractual clauses (e.g. rules pertaining to insurance settlements, collateral, bonding, property rights, etc.) can be embedded in computer programs using cryptographic validation mechanisms to make breach of contracts expensive (Szabo, 1997). It is "smart" because it contains programmable logic which can take many forms, and it can automatically execute the terms of the contract when specific logical conditions are met without the risks of censorship, downtime, fraud or third-party interference.[1]

Smart Contract Workflow

Smart contracts support decentralized and automated transaction consummation and validation. Figure 2 describes the various stages in the life cycle of a smart contract (Sheth and Subramanian 2019). The lifecycle of a smart contract starts with a pre-defined business contract, as it implements the terms and conditions of the contract agreed by transacting parties. The details of this stage can be expanded into a process flow shown in Figure 3, which describes how the contract progresses

Figure 2. Smart contract lifecycle (Sheth and Subramanian 2019)

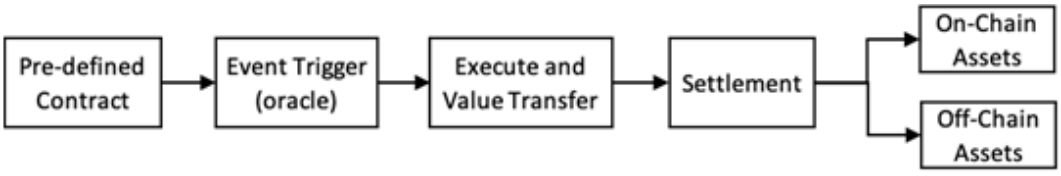
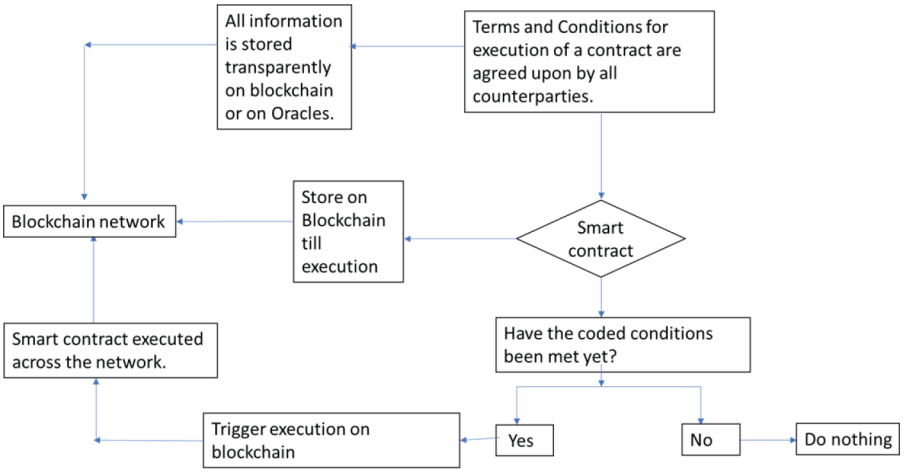


Figure 3. Workflow of Smart Contract Creation



from the setup stage to a state of dormancy, i.e. deployment on the blockchain. Once deployed to a blockchain, a new smart contract can be invoked by any node in the network. Often, it is triggered by external events or oracles (e.g. flight delay) and then a new transaction is created.

This new transaction is validated through a consensus mechanism as we described before. For example, a Ethereum smart contract is encapsulated as bytecode of Ethereum Virtual Machine (EVM). Miners who assemble a block that includes the transaction execute the bytecode to validate the transaction. Later, when the new block is mined and broadcast to the network, every node executes the bytecode again to update their local copies of the ledger. Thus, for each transaction, a smart contract's code is executed several times and every node verifies the execution outcome independently to reach global consensus. Since each transaction consumes considerable computation resources during this consensus process, to encourage efficient contract implementation and also to avoid malicious attacks, often the execution of every instruction in a smart contract involves a cost. For example, the execution of EVM bytecode is associated with a transaction fee, known as gas, to incentivize mining.

After these four stages, a business transaction progresses from "contract" definition to settlement with two different types of assets i.e., on-chain assets (e.g., crypto-tokens, or security tokens) or off-chain assets (i.e., physical assets that involve external agencies). On the overall, contracts – through automation – and operating as a software program with a lifecycle, with definite times of completion, and steps that are encoded in programmatic logic (which reduces ambiguity) can make business processes simpler. Moreover, in traditional contracts, complex transactional rules are often subject to different interpretations by contracting parties, but smart contracts reduce the ambiguity by explicitly programming the rules and specifying outputs for each rule in a programming language. Next, we provide detailed comparison between traditional contracts and smart contracts.

Traditional Contracts vs. Smart Contracts

Having introduced the basics of smart contracts, next, we compare smart contracts with tradition contracts to understand its potentials in revolutionizing business contracting. Hart and Holmstrom, winners of the Nobel Prize in Economics of 2017, sought to model firms and businesses as incomplete contracts, with contracts being written with the sole aim of smoothening frictions in transactions. In their seminal work on "Theory of Contracts"[2], Hart and Holmstrom modeled different forms of contracts that are pareto-optimal, such as the principal agent contract, the agency model, hidden action model, and hidden information model. They argued that contracts can reduce frictions in transactions that result from uncertain transaction costs, irrational behaviors of actors, incentive incompatibility,

and other factors. Hart and Holmstrom (1986) further contended that the economic credibility of the contractual approach to organizations can often become complex, and contracts themselves can become difficult to enforce and depend on the legal ability of the firm and its enforcement jurisdiction e.g., the country to enforce them. Therefore, frictions still exist when a legal recourse for enforcement is expensive and is often higher than the rents sought by transacting parties. Since the cost to arrive at an agreement is often high and contracting parties are better off to recover these costs over the longer term, traditional contracts are often longer-term contract, with limited flexibility. Accordingly, templatization (and repetition) is rare with respect to traditional business contracts.

For example, in the insurance sector, a customer who purchases an insurance policy – mandated by law – often encounters problems of enforcement when she must submit a claim. Increased layers of "insurance validators", "insurance agents" and other intermediaries between the customer and the insurance firm, making such transactions inefficient. In addition, insurers and intermediaries have different incentives than the insured with regards to the payment of claims. As a result, an insurance claim of $500 for a delayed flight can cost the customer many times more were s/he to claim the same through due legal process, if the claim were denied. Not to mention the asymmetric information, legal and market power of large insurance companies versus a lone individual customer. Such difficulties due to differences in market power, access to legal resources and domain knowledge by the parties involved in a contract often increase frictions in markets, thereby reducing demand from customers who are wary of the hassles of such claims. As another example, purchasing a life insurance, which is a contract, has a sales cycle that takes several weeks based on the insurance premium and customer criterion.

Different from traditional contracts, smart contracts facilitate an equitable, automated, verifiable transaction mechanism among those who participate in a contract but with different incentives in a business transaction. Smart contracts facilitate distributed applications (Dapps) on top of one or more blockchain systems, and enable Dapps to store, execute and automate programmatic logic without a central authority. While it is commonly believed that smart contracts are just an artefact of a blockchain based system, it is not necessarily so. Smart contracts often interoperate between different stateful and stateless systems, including multiple blockchain systems and databases, with the blockchain remaining as the core operating mechanism for rule-based evaluation.

For example, in the case of a "flight delay" insurance, the blockchain can validate an event of interest i.e., a delayed flight from flight monitoring systems, and process insured customer claims automatically (Sheth and Subramanian, 2019). Smart contracts often connect to external data sources, referred to as "oracles," which could either be database instances or could be crowdsourced from smart contracts. As an example, an insurance firm can operate as a platform that creates the smart contract infrastructure atop the blockchain system(s), integrating with multiple external systems (e.g., the flight database, weather database, national KYC/AML registry, etc.). Independent developers, resellers, insurance agencies or crowdsourcing platforms can also create and encode unique risk models which operate using the smart contract infrastructure. Additionally, insurance agents (such as travel sites, etc.) and resellers can customize insurance policies to suite their customer pool. This is a shift from a "one size fits all" model used in the traditional "flight" insurance marketplace.

For instance, one could charge different insurance premiums for the "flight delay" insurance product based on the time of delay i.e., a flight delay by 30 minutes priced at $10 insurance could pay the customer about 25% of the ticket fare. Similarly, a flight delay by 1 hour, priced at a $20 premium could pay the customer 40% of the ticket fare in case of a 1-hour delay. While such customizations can be done with traditional paper-based contracts, it is time consuming to dynamically edit or change parameters in traditional contracts as changes have to often pass through multiple legal gates and risk models based on varieties of parameters are difficult to implement on the fly. With smart contracts, the time needed to effect a change is greatly shortened, since all components of these markets are software rule based. Software automation of these smart contracts makes micro-changes possible at a higher rate of efficiency.

In addition, with smart contracts, e.g., on Ethereum, the insurance policy itself is configurable since templates are readily available and risk models associated with each template is already known to the insurance firm. Moreover, sales, operationalization and settlement cycles for insurance are instantaneous with a Dapp. Due to its multi-faceted functionality, smart contracts can make traditional insurance products reach a larger customer base through templatization, ease of reproduction, crowdsourced risk models, faster sales cycles, and easier settlement.

Recent research has shown that when blockchain based smart contract functionality replaces existing contractual mechanisms, both the supply and demand sides of a market benefit from the change and the overall social welfare improves (Sheth and Subramanian, 2019). Similarly, other recent research has demonstrated the efficacy of smart contracts when multiple parties are involved in a single transaction (Kumar et. al, 2019).

Impact of Blockchain-Based Smart Contracts

Recent research has demonstrated the importance of deploying a variety of blockchain architectures. However, smart contracts facilitate three important functionalities. Firstly, smart contracts facilitate Turing-complete transactions to be executed atop blockchain(s). The powerful functionality bought forth by smart contracts enable many business functions to be delivered directly to users and facilitate efficiencies by removing intermediaries. These features are unique and facilitate new classes of business applications such as in decentralized finance, smart electronic art, etc.

Secondly, the adoption of smart contracts has seen a rapid increase over the past five years with transactions on the Ethereum blockchain increasing from a few hundred transactions per day – each of which validates some logic within a smart contract - in 2014 to more than 750,000[3] transactions per day. Demand for transactional support on the Ethereum smart contract platform is in excess of 1 million transactions per second[4]. Similar statistics are reported from other smart contract platforms such as the Lightning Network which supports instantaneous Bitcoin transactions and the EOS platform, among others.

Thirdly, scholars in Computer Science, Information Systems, Economics and Law are increasingly publishing research analyzing technical, legal and economic implications of smart contracts. Research in Information Systems (IS) focusing exclusively on smart contracts is still at early stages, though the focus has primarily been on Blockchain. Next, we summarize important papers in Blockchain and Smart contract research below and hope that the IS community will benefit from our summary and analysis.

RECENT LITERATURE ON BLOCKCHAIN AND SMART CONTRACT

Blockchain and Smart contract has been a trending research area for scholars in IS, law, economics and computer science (Ballis, 2017). Several scholarly journals have published very detailed and systematic surveys of literature on Blockchain. We summarize those surveys here, and later discuss literature on Smart contracts. Detailed surveys published in premier journals have included the following topics - Blockchain Governance Issues and Challenges (Beck et. al 2019), consensus and security mechanisms (Wang et. al 2019, Xiao et. al 2020), Blockchain Security and Privacy (Zhang et. al, 2019, Khalilov et. al 2018, Conti et. al 2018), Blockchain applications for the Internet of Things (Dai et. al, 2019, Ferrag et. al 2018, Ali et. al 2018, Lao et. al 2020, Wang et. al 2020), Cloud computing (Gai, et. al 2020,), Edge computing systems (Yang et. al, 2019), healthcare data privacy (Espocito et. al, 2018), Game theoretical perspective of blockchain (Liu, et. al 2019), Cryptocurrencies and decentralization (Tschorsch & Scheuermann, 2016), blockchain industry applications (Al-Jaroodi, 2019).

Kolb et al. (2020) provided a comprehensive tutorial of blockchain core concepts using Ethereum as a case study and discussed the challenges and future directions in Blockchain. Cryptocurrency-oriented blockchains such as Bitcoin, Ethereum, EOS, and Cardano support the creation of smart contracts. Similarly, enterprise grade blockchain frameworks such as Hyperledger support the creation

of business applications using the smart contract framework. Because a smart contract can encode any set of rules represented in its programming language, this programmatic codification of laws poses its own opportunities and challenges (Mik 2017).

Recent research in IS - mostly design science and socio-technical - has discussed both the designs of IoT based sensor networks and their applications in healthcare monitoring (Griggs et al. 2018). Liu and Subramanian (2019) propose the use of smart contracts to improve software outsourcing. Sheth and Subramanian (2019) describe how smart contracts facilitate a more efficient insurance marketplace. Hjálmarsson et al (2019) and Tso, Liu and Hsiao (2019) discuss applications of smart contracts in e-voting and e-bidding businesses. Similarly, Shahzad and Crowcroft (2019) discuss trustworthy electronic voting using Ethereum based smart contracts. Raskin (2017) gives an overview of different types of contracts, namely strong and weak contracts, and the legal forms of enforcement of contracts through traditional and non-traditional means. Raskin (2017) discusses the challenges with smart contracts at each of the three stages in contract law i.e., (1) formation of the contract, (2) performance and modification of the contract, and (3) enforcement, breach and remedies. Raskin highlights challenges in administering complex smart contracts with multiple stakeholders within the framework of contract law. An interdisciplinary team, consisting of IS scholars, computer scientists, and experts in contract law can analyze the full spectrum of legal, technical and economic implications of smart contracts.

There is an ever-growing stream of literature in computer science, IS and economics that focuses on smart contract security, testing, formal verification, implementation and design. Magazinni et. al (2017) explore issues and research challenges involved in the validation and verification of blockchain based smart contracts. Li et. al (2017) present a detailed survey of research in smart contract security. Luu et al. (2016) discuss three types of security flaws found in Ethereum smart contracts – by analyzing several thousand smart contracts, and discuss the design of a Python based tool, Oyente, which detects design flaws.

Consensys, a private organization that draws on open-source developer contributions has created a public repository of smart contract practices[5]. These best practices prevent common types of attacks on smart contracts. Atzei, Bartoletti and Cimoli (2017) survey major attacks on Ethereum based smart contracts and provide a detailed taxonomy of programming pitfalls which lead to vulnerabilities. Nikolic et. al (2017) present a systematic characterization of trace vulnerabilities that result from lifelong execution of smart contracts after analyzing 0.97 million Ethereum smart contracts. These authors document three types of trace vulnerabilities in smart contracts - ones that lock up funds, others that leak funds carelessly to users and the third type that can be terminated randomly. Wang and Malluhi (2019) describe the limitations of Turing complete smart contracts using hypothetical examples. Wang and Malluhi (2019) classify contracts into four types and several sub-types and contend that smart contract validity for four types of contracts are difficult to accomplish, owing to non-decidability of the universal Turing machine halting problem. For example, on Ethereum if the transaction fee (i.e., gas) exhausts from the sender's wallet, a transaction cannot be validated.

Another recent stream of research focuses on smart contract vulnerabilities. Operating in open (or permission-less) networks where participation is unrestricted, smart contract platforms have to manage a wider spectrum of risks and attempted manipulation by adversaries, including miners and contract users (Luu et al., 2016). Several recent Blockchain protocols such as coinbase.com, kraken.com, etc. have started deploying Bug Bounty Programs to detect vulnerabilities faster, and fix them (Subramanian and Malladi, 2020). An area of research that would be exciting is how to incorporate Bug Bounty Programs into software development processes for detecting critical vulnerabilities. For example, Ethereum and Bitcoin allow miners to decide which transactions to accept, how to order transactions, and how to timestamp a block, making these transactions open to manipulation. Research has previously examined vulnerabilities in smart contracts (Luu et al., 2016). Smart contracts can implement a wide range of applications (e.g., financial instruments, money transfers, savings wallets, wills, outsourced computations, and decentralized gambling) that provide multi-disciplinary research

opportunities. Cousins et al. (2019) argue that scholars in IS can measure and monitor mechanisms which enforce and improve trust among the different stakeholders in smart contract systems.

Recent Development in Blockchain and Smart Contracts

In this section, we first introduce the innovative blockchain-based architectures or applications brought by the four papers published in this special issue. These papers propose new mechanisms for blockchain fundamentals such as consensus or oracles, and innovatively apply blockchain and smart contracts to solve real business problems. Then we continue to introduce other important blockchain applications in different business areas.

Introduction to Papers in the Special Issue

The current special issue has accepted four papers. The first paper titled *"On Elastic Incentives for Blockchain Oracles"* studies a fundamental open question for oracles in blockchain environments about the determination of the amount of trust to be placed in oracles. Oracles serve as intermediaries between a trusted blockchain environment and the untrusted external environment where the oracles fetch data. This paper develops a model for commoditization of trust and provides a dynamic trust environment that incorporates oracle selfishness. The paper considers the equilibrium behavior for the demand and supply for trust and introduces elastic incentives for increasing the trust. These results are used to determine optimum size of the network that can be served by an oracle with varying degrees of selfishness. Meanwhile, important consequences and challenges of incorporating oracles in trusted distributed ledger environments are presented.

The next paper titled *"Usurping Double-Ending Fraud in Real Estate Transactions via Blockchain Technology: Blockchain Technology Usurping Double-Ending Fraud"* discusses the problem of double-ending fraud in real estate transactions – a type of transactional fraud wherein agents handling real estate transactions unfairly benefit, e.g., by simultaneously representing both the buy and sell side of a real estate transaction in a manner that unfairly boosts the commission they receive, or colluding to increase their commission in a real estate transaction at the expense of the buyer and/or seller of the real property. The paper proposes a unique blockchain solution design that leverages blockchain's properties of transparency and creates tamper-resistant audit trails to reduce opportunities for double-ending fraud and increase real estate market participants' trust in the handling of their transactions. The paper details a prototype solution based on Hyperledger Fabric and Sails and demonstrates that the inherent transparency of the proposed design offers optimal allocation for both sellers and buyers through agent-based modeling simulation.

The next paper titled *"Enhancing the retailer gift card via blockchain: Trusted resale and more"* discusses how blockchains can be used to implement the retailer gift card. The paper discusses the contradiction in gift cards that – though it encourages consumer spend, it also places a great number of customers in troublesome situations due to its current limitations. First, dealing with unwanted gift cards is often time-consuming, costly or even risky due to the frequent occurrences of gift card resale frauds. Worse still, the issuance and redemption of gift cards happen inside the retailer as in a "black-box", indicating that a compromised retailer can cheat customers (or even third-party auditors) to deny the issuances of some unredeemed gift cards. This paper proposes a practical middle-layer solution based on blockchain to address the fundamental issues of the existing gift card system, with incurring minimal changes to the current infrastructure.

Our final paper in this special edition titled "*Proof-of-Useful-Work as Dual-Purpose Mechanism for Blockchain and AI: How its Use for Blockchain Consensus Enables Privacy Preserving Data Mining*" presents the benefits and challenges of a consensus mechanism called proof-of-useful-work among blockchains and its practical applications for optimizing and executing machine learning models in blockchains. This new consensus mechanism is proposed to address the well-known concern of proof-of-work on energy consumption (see Section 2). Blockchains rely on a consensus among participants to achieve decentralization and security. However, reaching consensus in an

online, digital world where identities are not tied to physical users is a challenging problem. Proof-of-work provides a solution by linking representation to a valuable, physical resource. While this has worked well, it uses a tremendous amount of specialized hardware and energy, with no utility beyond blockchain security. This paper proposes an alternative consensus scheme that directs the computational resources to the optimization of machine learning (ML) models – a task with more general utility. This is achieved by a hybrid consensus scheme relying on three parties: data providers, miners, and a committee. The data provider makes data available and provides payment in return for the best model, miners compete about the payment and access to the committee by producing ML optimized models, and the committee controls the ML competition.

Initial Coin Offerings (ICO) and Decentralized Applications

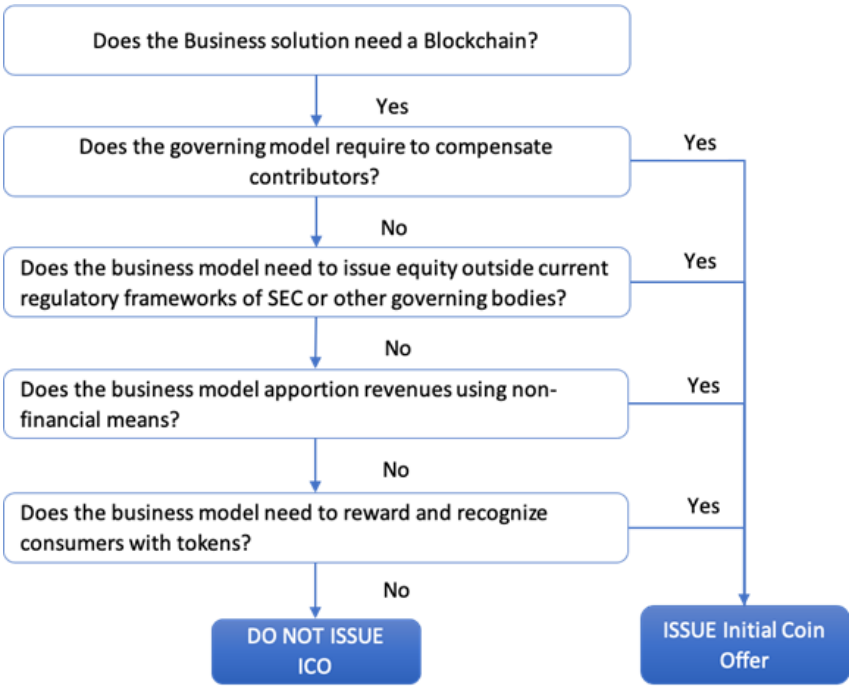
ICOs are an emerging method of entrepreneurial financing and a new mechanism used by blockchain-based ventures to raise capital by selling tokens to a crowd of investors without surrendering control rights (Fisch 2019). Since the first ICO dated back to 2013, as of now more than 5,600 projects have sourced their funding through ICO campaigns and the total amount has reached 27 billion. ICOs provide an illustrative example of smart contracts operating on permission-less networks as well as some interesting business implications and considerations for IS and economics scholars. ICOs often lead to a much more faster capital acquisition route. As one example, the Brave ICO was able to purportedly raise 30 million USD in 30 seconds for its Basic Attention Token (BAT).

To better understand ICOs, we will explain the smart contracts architecture and then the business logic and implications of this method of funding. ICOs comprise of two initial design steps: 1) issuing digital tokens ("coins") and 2) specifying the transfer parameters. In particular, the ERC20 standard (Ethereum Request for Comment 20 standard) is the standard software specification for actually issuing new tokens, where a creator specifies the token's name, symbol, total supply, price in terms of Bitcoin or ETH, fundraising period, among others[6]. These parameters are incorporated into a smart contract that also establishes the transfer parameters, and procedure for securely validating and then triggering, withdrawal, access, transfer, and payment of issued tokens.

Many studies have investigated a variety of factors that affect both ICO success and ex-ante failures of entrepreneurial ventures that raise capital (Howell, et. al 2018, Fisch 2019). Scholars contend that crypto-tokens facilitate early stage equity-investment functions and that tokens facilitate awareness among investors about the following:1) Governance models – the model which governs how tokens are issued, how tokens are capitalized by the entrepreneurs and various decision-making rules for businesses. Innovations such as DAICOs have given credibility to successful business models wherein token owners are presented with proportional voting rights and can influence disbursement of capital and future business directions of the firm.2) Equity Issuance models – the issuance of securities (or shares) of the underlying businesses based on rules for holding, selling and divesting tokens. Equity issuance policies are similar to monetary policies of central banks and govern the mechanisms by which equity is issued.3) Revenue models– the underlying business model and mechanism by which the firm issuing the token earns revenues. This provides the much-needed ability for firms to apportion revenues.4) Reward(s) and Recognition models – wherein participants on the network are rewarded commensurate to their participation on the network. Subramanian (2019) discusses the role of crypto-tokens in creating, facilitating security token marketplaces for a wide variety of business needs. ICOs are subject to regulatory oversight in many countries. A detailed discussion of ICOs and their security and legal implications are discussed in WA Kall (2018).

Ultimately, the business applications of ICOs happen beyond just initial creation of the token and facilitation of transfers to users through wallets, crypto-exchanges, and other similar mechanisms. Among others, Howell, Niessner & Yermack (2018) and Rhue (2018) discuss how ICOs meet their financial goals via token sales. Similarly, Subramanian, et. al (2020) apply the Pederson (2018) et. al framework to analyze four different applications and recommend how cryptocurrencies or the blockchain(s) are applicable to each such case. The four main questions business have to ponder about

Figure 4. Justification flowchart of Issuance of Initial Coin Offerings by Blockchain business



whether to use ICOs or not are relevant to our discussion on the need for governance, equity, revenue or rewards and recognition. Once a business has decided that it is Blockchain ready (Subramanian, 2020), additional steps should be used for deciding if the firm needs to do an ICO as shown in Figure 4.

Each of these steps leads to a question that the corresponding blockchain deploying firm has to answer i.e., does the governing model require to compensate governors? If the answer is yes, then an ICO will be used to acquire the necessary capital for compensation. Similarly, other questions that are pertinent and core to the decision of any Business planning to do an ICO are: Does the business model need to issue equity outside the existing legal frameworks permitted by legal bodies such as the Securities and Exchange Commission? Does the business model provision for revenue shares using non-financial compensation? Does the business model need to reward and recognize customers or participants on the blockchain network? If justification for these core business questions are not positive, then ICOs do not necessarily provide the best mechanism to raise capital and can often lead to fraudulent or suspicious business practices. Even if these mechanisms are followed and appropriate justification for ICOs is justifiable, appropriate legal mechanisms based on jurisdiction have to be implemented (Subramanian et. al. 2020).

Security Tokens

Another large area that is being impacted is security tokens which create unique financial instruments that have had no parallel so far in today's markets, lending themselves to cross-national/cross-border trade (Subramanian, 2019). For example, an entity could securitize a portion of his mortgage and sell it to the highest bidder on the market thus enabling current market demand to determine his property price. Similarly, with Decentralized Finance (DeFi) applications, retail owners of stable coins (i.e., cryptocurrencies whose value is fixed) can engage in peer to peer lending (e.g., Compound.Finance) or other smart contract based financial applications (e.g., Celsius.Network, nexo.io) virtually bypassing bank networks and national/state boundaries. Similarly, real estate owners could securitize their rental

properties and trade them on open markets, as dividend earning properties and sell such security tokens to interested parties. Both ownership, dividend transfers and value shared are automated using smart contracts thus reducing the risks of defaults, and other issues with regular contracts. Such securitization, are nevertheless, subject to laws of governing entities like the SEC, FinCen, CFDC, etc. and provide a mechanism of trust between parties involved in such transactions. The nature of the smart contracts brings the necessary trust, security, privacy and universality to such transactions across borders.

Artificial Intelligence Applications

This automation of business functions at a very high level of trust lends itself to advancements in areas such as automated reasoning in Artificial Intelligence (Subramanian, 2018). An area of application is the formal verification of logic and software development that has been plagued by several inadequacies, owing to different levels of automation of test-suites, and dependency on human-labor. With automated reasoning, formal logic could become validated through consensus algorithms. For example, firms, such as Synthetic Minds, formally verify the logic behind programs, enable automatic generation of contracts from existing code bases and simulate behaviors of contracting parties with generated contracts. Similarly, Blockchains enable new types of marketplaces that enable hedge fund marketplaces such as numer.ai which abstracts financial management of hedge funds and crowdsources models that improve over existing known models, in exchange of rewards in cryptocurrencies. Similarly, with platforms such as open-mined, users can improve their ability to model data using federated learning and homomorphic encryption – where without sharing the actual data, firms can crowdsource improvements to existing data and machine learning models in exchange of rewards in cryptocurrencies.

Supply Chain Applications: Everledger[7] and IBM Food Trust[8]

Everledger is a company which provides a platform to track the provenance of diamonds and other high-value products based on Hyperledger blockchain technology. This platform takes a forensic approach to identifying and tracking asset provenance to provide confidence in the transparency of global supply chains. In particular, this platform creates a digital fingerprint for each piece of high-value products, for instance, diamonds, based on the product's physical unclonable features (PUF). Smart contracts are used to store this figure print along with certificates, for example digital diamond grading reports issued by leading independent diamond grading authority, on the ledger of a global blockchain. Similarly, this platform uses smart contracts to provide real-time provenance related services, for example, validation of diamond certificates, to customers and business partners in the supply chains.

IBM Food Trust is a solution-as-a-service cloud platform built on Hyperledger Fabric. It is considered as one of the few enterprise blockchain systems that have been fully deployed at scale. This platform connects participants in a food supply chain and allows them to share food system data securely. The main purpose of this platform is to provide food traceability, the ability to specifically identify and locate products that may be subject to food recalls, which are extremely expensive, wasteful and challenging under current processes. Moreover, this platform is designed to be compatible with GS1, a standards framework used by the food industry, to ensure interoperability and standardization so that it can be integrated with other enterprise systems used by participants. In addition, IBM Food Trust tries to build an ecosystem for all players that touch the food supply, including not just retailers and suppliers but also other third parties, for example, auditors who ensure process compliance, and service providers who track temperatures throughout the food transport process.

Smart contracts were implemented provide services in three modules. The tracing module facilitates the tracking of food products throughout the ecosystem, the certification module verifies the provenance of products that have been digitally certified as organic or fair trade, and the data

entry and access module gives participants the ability to upload, manage and access their data within the system.

CHALLENGES WITH BLOCKCHAIN AND SMART CONTRACTS

There are three types challenges facing any blockchain based systems. This is often referred to as the Blockchain Trilemma and represented as follows: security, scalability and decentralization, as shown in Figure 5. The Trilemma states that "no ledger can satisfy security, scalability and decentralization at the same time" (Abadi, 2018). While, firms such as Algorand claim to have solved all three problems simultaneously (Conti et. al, 2019); other popular blockchain systems have seen significant challenges with at least one of the three properties. Below we discuss the recent research about Security.

Security of Blockchain and Smart Contracts

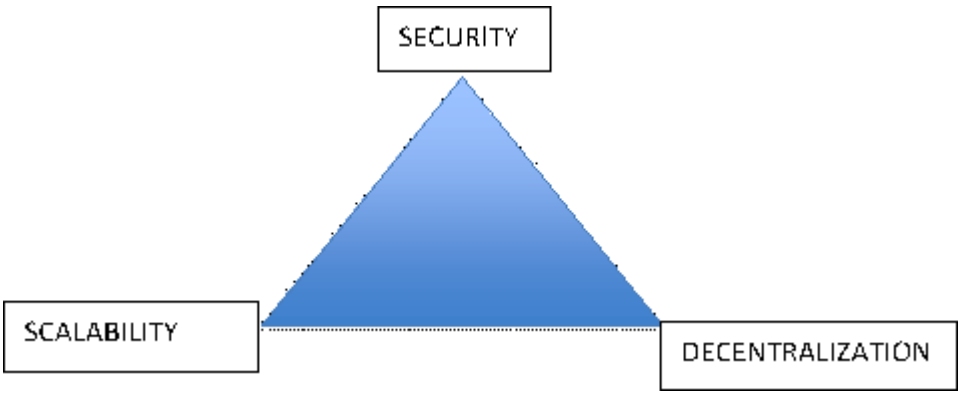
One of the main weaknesses of smart contracts are rooted in software bugs and insecure programming languages. A recent study of more than 19,000 smart contracts running on the Ethereum platform found that 44% of the contracts contained errors. Due to the complexity of the system, and the lack of qualified and experienced programming resources, contracts often contain errors (Ivica, et. al., 2018). Transferring Ether to a smart contract involves dynamic dispatch, which has led to a new class of bugs known as reentrancy vulnerabilities. Another major concern lies in the need to confirm that smart contract code reflects the intent of parties and contains no inadvertent coding errors (Mik, 2017). However, it is practically impossible to ensure that every piece of software code is bug-free.

Examples of recent loss events are the "Parity hack" and "The DAO" error which led to massive loss of funds which led to the "fork" of the Ethereum network, which split the network in two separate blockchain systems. Such challenging problems face many large software systems, and we expect improvements in existing systems (Luu et al., 2016) and the emergence of new auditing tools and services will help reduce risks in the underlying infrastructure. Further, new systems that may mitigate such risks through "formal methods" (Avizienis et al., 2004) that provide an alternative implementation for new applications developed in the years ahead. We next discuss issues with legality and disputes of adjudication of blockchain and smart contracts.

Legality and Disputes of Adjudication

Subramanian, et al. (2020) discuss blockchain regulations and disputes of adjudication of Bitcoin based crypto assets. For example, different legal institutions such as Securities and Exchange Commission, FinCen, Internal Revenue Services, etc. view cryptocurrency-based assets differently. An area

Figure 5. The blockchain trilemma



of ongoing development is how to address dispute adjudication and reconciliation for automated contracts. In practice, there will be events and outcomes that face dispute and will require satisfactory mechanisms for adjudicating decisions that may be disputed by a contract party. For example, Blockchain technology is believed to have a potential to create decentralized global platforms to support global supply chains (Babich and Hilary, 2018), where a smart contract may be executed by globally distributed vendors who are a part of the supply chain. However, in this case, adjudicating decisions can be very complicated, since smart contracts may not be enforceable in every country. Levi and Lipton (2018) suggested that an overarching governing law is essential to determine what specific law will apply for the interpretation of a smart contract and which jurisdiction will adjudicate disputes. Moreover, there are even more complicated issues related to the interpretation of legal prose in a smart contract as discussed by Mik (2017).

There are several approaches to formalize such systems for smart contract applications, based on both older and newer research. Miller proposes writing "split contracts" which distinguishes between machine-interpreted and human-interpreted sections to limit the scope of adjudicating conflicting interpretations and disagreements (Miller and Drexler, 1988; Miller et al., 1997; Stiegler and Miller, 2013). Another approach to this challenge includes implementing Ricardian contracts (Grigg, 2004) – these are smart contracts coded in software with the legal force of natural language contracts – in a way that integrates smoothly with existing arbitral and legal frameworks. Levi and Lipton (2018) suggest that smart contracts can be used to supplement traditional text-based contracts, given the early stage of smart contract adoption.

From a jurisdictional point of view, smart contract technologies enable the creation of generalized and templatized versions of software policies that are global in nature. Legal and jurisdictional implications for such products need further research to be fully understood. However, real-world use cases of such technologies are still very limited for the purposes of analysis. More recently, firms such as "Elliptic" and "Chainalytics" deal with three facets of smart contract legality namely cryptocurrency compliance, cryptocurrency forensics and investigation services. These firms provide a wide range of compliance services to financial institutions through support of KYC/AML procedures based on jurisdiction, blockchain analytics-based crime detection and blockchain forensics support to both the industry and government. There still needs to be more research to analyze implications of smart contracts with respect to compliance issues and the risks newer types of transactions will surface.

Business Process Transformation Using Blockchain and Smart Contracts

To create smart contract applications, we will need to transform business logic, including business processes, business rules, and regulatory policies, etc. into a set of smart contracts. However, as discussed by Kumar et al. (2019), the wording of rules and policies specified in plain text can be rather convoluted and they do not lend themselves easily to mapping into an executable language as used by smart contracts. To illustrate, consider a sample clause in a flight delay insurance policy: "When terms and conditions are met, a party is eligible for reimbursement of *reasonable* additional expenses incurred when a purchased trip is delayed for more than six (6) hours or requires an overnight stay." First, this clause contains "desired ambiguities" (Mik, 2017), such as "reasonable expense," which leave parties the flexibility to work out a resolution when an unanticipated expense occurs. However, such a clause can hardly be represented formally or measured objectively such that it can be translated accurately into smart contract code (Kumar et al. 2019). Second, executing contracts often requires information from third parties. For example, the insurance policy does not cover delay due to a public hazard which was made known to the party prior to the party's departure. An independent authority may be needed to validate such a hazard. However, to ensure smart contract determinism, a state where every party has to reach the same conclusion after execution, a smart contract is unable to retrieve off-chain resources but can request a trusted third party called an "oracle" to push necessary resources onto the blockchain system. The dependency on oracles for off-chain data raises risks of erroneous data and weakens the benefits of decentralization. Increasing the reliability of such systems

has been subject of recent research, and has included recommendations to crowdsource information, building and creating reputation models, and providing fault-tolerant sub systems that are variants of other blockchains.

When transforming a business process into smart contracts which are executed in a decentralized way, another concern is the compliance of the overall process to regulatory requirements (Cuomo et al. 2018a). A recent study gives an alarming estimate that 46% of bitcoin transactions involved illegal activities (Foley et al. 2019). Similarly, for Everledger, it is critical to ensure the diamond global supply chain is compliant with Kimberley Process Certification Scheme, which removes conflicting sub-standard diamonds from the supply chain[9]. Cognitive solutions have been proposed to analyze smart contract transactions and conduct process compliance checking (Cuomo et al. 2018b, Cuomo et al. 2018c), but it is not clear which parties can play such a role to oversee the entire process in a decentralized environment. In addition, privacy can be another concern because such parties need to access transactions in the entire supply chain. Another enduring issue related to process transformation is process verification. Although Blockchain and Smart Contract is promising to automate a business process, a real challenge lies in how to ensure it can be successfully executed to reach its business goals without deadlocks or livelocks (Soffer and Kaner 2011), and it is well aligned with business strategies or goal-oriented requirements (Poels et al. 2013).

In addition, researchers believed an efficient way to facilitate smart contracts is through standardized templates, called smart legal templates, which can be easily customized for various situations by providing specific parameter values, because many contracts in business domains, such as insurance and finance, are quite standard (Clack et al., 2016). Smart legal templates can facilitate smart contracts, connect legal agreements to automated business logic, simplify processes, drive standards adoption via reusable templates, and reduce costs through the use of common components (Clack et al., 2016). However, it is also challenging to design generic templates for businesses in each industry – though over a long period of time, we would expect society and businesses to build and share best practices to create a repository of most common business cases.

Pederson et. al, 2019 suggest a ten-step decision path for choosing a particular blockchain approach i.e., permission less, permissioned public blockchain or permissioned private blockchain to decentralized and distributed applications. While Pederson et. al (2019) deal with the choice of an overarching blockchain based solution, smart contracts can take this approach a step further. For example, smart contract technologies such as atomic swaps enable smart contract bytecode on one platform to be machine translated into another blockchain's code automatically providing much needed enterprise level interoperability to blockchains. Firms such as Komodo[10] and Wan-chain enable traditional centralized applications to work across blockchains. Similarly, GDPR laws and Homomorphic encryption supported by ZKSnarks technologies put limitations on data and are making progress with respect to blurring the differences between permission-less and permissioned blockchains.

CONCLUSION

As discussed in the review above, blockchain and smart contract have seen increased relevance to researchers across disciplines. Scholars can study and research implications of smart contracts with respect to monetary policies and incentive structures with respect to different well proven business models. Scholars in IS, Computer Science, and allied fields can study the design and application of blockchain and smart contracts in current business settings, for example, applications of blockchain in enabling more efficient markets. Next, scholars in strategy and economics of information systems can study mechanism designs that support incentive compatibility, incentive alignment and other game theoretic problems associated with smart contract enabled business transactions. Scholars in Business Law and Contracts can study and understand how newer forms of blockchain applications and smart contracts can be written and applied to these markets. Finally, scholars in sociology,

psychology and human behavior can analyze how societal transformations where algorithmic forms of "trust" plays a key role in shaping business transactions. Overall, blockchain and smart contracts provide a powerful, yet multi-faceted mechanism to implement business processes.

Finally, we would like to thank the Senior Editors of the Journal of Database management for having placed trust in our abilities to execute and run this detailed scholarly addition to Blockchain and Smart Contract. The papers in this special edition use a variety of methods, from design science-based prototyping, to analytical modeling and simulations and algorithm design that will form a very strong basis for future work in this domain. We hope the readers and scholars will find great practical import in the papers of this special edition.

REFERENCES

Abadi, J., & Brunnermeier, M. (2018). Blockchain economics (No. w25407). National Bureau of Economic Research.

Al-Jaroodi, J., & Mohamed, N. (2019). Blockchain in industries: A survey. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 36500–36515. doi:10.1109/ACCESS.2019.2903554

Ali, M. S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., & Rehmani, M. H. (2018). Applications of blockchains in the Internet of Things: A comprehensive survey. *IEEE Communications Surveys and Tutorials*, *21*(2), 1676–1717. doi:10.1109/COMST.2018.2886932

Atzei, N., Bartoletti, M., & Cimoli, T. 2017, April. A survey of attacks on ethereum smart contracts (sok). In *International Conference on Principles of Security and Trust* (pp. 164-186). Springer. doi:10.1007/978-3-662-54455-6_8

Babich, V. R., & Hilary, G. (2018). (Forthcoming). Distributed ledgers and operations: What operations management researchers should know about blockchain technology. *Manufacturing & Service Operations Management*.

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*, *19*(10), 1. doi:10.17705/1jais.00518

Bhargavan, K., Delignat-Lavaud, A., Fournet, C., Gollamudi, A., Gonthier, G., Kobeissi, N., Kulatova, N., Rastogi, A., Sibut-Pinote, T., Swamy, N., & Zanella-Béguelin, S. (2016, October). Formal verification of smart contracts: Short paper. In *Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security* (pp. 91-96). ACM. doi:10.1145/2993600.2993611

Buterin, V. (2013). *Ethereum: a next generation smart contract and decentralized application platform*. Academic Press.

Casino, F., Dasaklis, T. K., & Patsakis, C. (2018). A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telematics and Informatics*.

Clack, C. D., Bakshi, V. A., & Braine, L. (2016). *Smart contract templates: Essential requirements and design options*. arXiv preprint, 1612.04496

Constantinides, P., Henfridsson, O., & Parker, G. G. (2018). Introduction—Platforms and infrastructures in the digital age. *Information Systems Research*, *29*(2), 381–400. doi:10.1287/isre.2018.0794

Conti, M., Gangwal, A., & Todero, M. 2019, August. Blockchain trilemma solver algorand has dilemma over undecidable messages. In *Proceedings of the 14th International Conference on Availability, Reliability and Security* (pp. 1-8). doi:10.1145/3339252.3339255

Conti, M., Kumar, E. S., Lal, C., & Ruj, S. (2018). A survey on security and privacy issues of bitcoin. *IEEE Communications Surveys and Tutorials*, *20*(4), 3416–3452. doi:10.1109/COMST.2018.2842460

Cousins, K., Subramanian, H., & Esmaeilzadeh, P. (2019). A Value-sensitive Design Perspective of Cryptocurrencies: A Research Agenda. *Communications of the Association for Information Systems*, *45*(1), 27. doi:10.17705/1CAIS.04527

Cuomo, G. A., Dillenberger, D. N., Heath, F. F., III, Liu, R., & Vaculin, R. (2018b). *International Business Machines Corp, 2018. Automatic generating analytics from blockchain data*. U.S. Patent Application 15/462,873.

Cuomo, G. A., Dillenberger, D. N., Liu, R., & Vaculin, R. (2018a). *International Business Machines Corp, 2018. Cognitive regulatory compliance automation of blockchain transactions*. U.S. Patent Application 15/462,875.

Cuomo, G. A., Dillenberger, D. N., Liu, R., & Vaculin, R. (2018c). *International Business Machines Corp, 2018. Cognitive blockchain automation and management*. U.S. Patent Application 15/462,877.

Dai, H. N., Zheng, Z., & Zhang, Y. (2019). Blockchain for Internet of Things: A survey. *IEEE Internet of Things Journal*, *6*(5), 8076–8094. doi:10.1109/JIOT.2019.2920987

Dinh, T. T. A., Wang, J., Chen, G., Liu, R., Ooi, B. C., & Tan, K. L. 2017, May. Blockbench: A framework for analyzing private blockchains. In *Proceedings of the 2017 ACM International Conference on Management of Data* (pp. 1085-1100). ACM. doi:10.1145/3035918.3064033

Esposito, C., De Santis, A., Tortora, G., Chang, H., & Choo, K. K. R. (2018). Blockchain: A panacea for healthcare cloud-based data security and privacy? *IEEE Cloud Computing*, *5*(1), 31–37. doi:10.1109/MCC.2018.011791712

FabricH. (n.d.). https://hyperledger-fabric.readthedocs.io/en/latest/index.html

Ferrag, M. A., Derdour, M., Mukherjee, M., Derhab, A., Maglaras, L., & Janicke, H. (2018). Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet of Things Journal*, *6*(2), 2188–2204. doi:10.1109/JIOT.2018.2882794

Fisch, C., & Offerings, I. C. (2019, January). (ICOs) to Finance New Ventures (September 29, 2018). *Journal of Business Venturing*, *34*(1), 1–22. doi:10.1016/j.jbusvent.2018.09.007

Foley, S., Karlsen, J. R., & Putniņš, T. J. (2019). Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies? *Review of Financial Studies*, *32*(5), 1798–1853. doi:10.1093/rfs/hhz015

Gai, K., Guo, J., Zhu, L., & Yu, S. (2020). Blockchain Meets Cloud Computing: A Survey. *IEEE Communications Surveys and Tutorials*, *22*(3), 2009–2030. doi:10.1109/COMST.2020.2989392

Griggs, K. N., Ossipova, O., Kohlios, C. P., Baccarini, A. N., Howson, E. A., & Hayajneh, T. (2018). Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. *Journal of Medical Systems*, *42*(7), 130. doi:10.1007/s10916-018-0982-x PMID:29876661

Hart, O. D., & Holmstrm, B. (1986). *The theory of contracts*. MIT Press.

Hjálmarsson, F. Þ., Hreiðarsson, G. K., Hamdaqa, M., & Hjálmtýsson, G. 2018, July. Blockchain-based e-voting system. In *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)* (pp. 983-986). IEEE. doi:10.1109/CLOUD.2018.00151

Howell, S. T., Niessner, M., & Yermack, D. (2018). Initial coin offerings: Financing growth with cryptocurrency token sales (No. w24774). National Bureau of Economic Research.

Kaal, W. A. (2018). Initial coin offerings: The top 25 jurisdictions and their comparative regulatory responses (as of May 2018). *Stan. J. Blockchain L. & Pol'y*, *1*, 41.

Kaal, W.A. (2018). Initial Coin Offerings: The top 25 jurisdictions and their comparative regulatory responses. *CodeX Stanford Journal of Blockchain Law & Policy*.

Khalilov, M. C. K., & Levi, A. (2018). A survey on anonymity and privacy in bitcoin-like digital cash systems. *IEEE Communications Surveys and Tutorials*, *20*(3), 2543–2585. doi:10.1109/COMST.2018.2818623

Kolb, AbdelBaky, Katz, & Culler. (2020). Core Concepts, Challenges, and Future Directions in Blockchain: A Centralized Tutorial. *ACM Comput. Surv., 53*(1). DOI:<ALIGNMENT.qj></ALIGNMENT>10.1145/3366370

Kolluri, A., Nikolic, I., Sergey, I., Hobor, A., & Saxena, P. (2018). *Exploiting the laws of order in smart contracts.* arXiv preprint arXiv:1810.11605

Lao, L., Li, Z., Hou, S., Xiao, B., Guo, S., & Yang, Y. (2020). A survey of IoT applications in blockchain systems: Architecture, consensus, and traffic modeling. *ACM Computing Surveys*, *53*(1), 1–32. doi:10.1145/3372136

Levi, S. D., & Lipton, A. B. (2018). *An introduction to smart contracts and their potential and inherent limitations*. Available at https://corpgov.law. harvard.edu/2018/05/26/an-introduction-to-smart-contracts-and-their- potential-and-inherent-limitations/

Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2017). A survey on the security of blockchain systems. *Future Generation Computer Systems*.

Liu, R. & Subramanian, H. (2019, June). Smart Contracts Based Agile Software Development. *IEEE Blockchain Technical Briefs*.

Liu, Z., Luong, N. C., Wang, W., Niyato, D., Wang, P., Liang, Y. C., & Kim, D. I. (2019). A survey on blockchain: A game theoretical perspective. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 47615–47643. doi:10.1109/ACCESS.2019.2909924

Mik, E. (2017). Smart contracts: Terminology, technical limitations and real world complexity. *Law, Innovation and Technology*, *9*(2), 269–300. doi:10.1080/17579961.2017.1378468

Nikolić, I., Kolluri, A., Sergey, I., Saxena, P., & Hobor, A. (2018). Finding the greedy, prodigal, and suicidal contracts at scale. In *Proceedings of the 34th Annual Computer Security Applications Conference*, (pp. 653-663). ACM. doi:10.1145/3274694.3274743

Nikolić, I., Kolluri, A., Sergey, I., Saxena, P., & Hobor, A. (2018). Finding the greedy, prodigal, and suicidal contracts at scale. In *Proceedings of the 34th Annual Computer Security Applications Conference*, (pp. 653-663). ACM. doi:10.1145/3274694.3274743

Orcutt, M. (n.d.). Ethereum's smart contracts are full of holes. *MIT Technology Review*. https://www.technologyreview.com/s/610392/ethereums-smart-contracts-are-full-of-holes/

Pedersen, A. B., Risius, M., & Beck, R. (2019). A Ten-Step Decision Path to DetermineWhen to Use Blockchain Technologies. *MIS Quarterly Executive*, *18*(2), 3.

Poels, G., Decreus, K., Roelens, B., & Snoeck, M. (2013). Investigating goal-oriented requirements engineering for business processes. *Journal of Database Management*, *24*(2), 35–71. doi:10.4018/jdm.2013040103

Raskin, M. (2017). The law and legality of smart contracts. *Geo. L. Tech. Rev., 305*. https://perma.cc/UC8L-KTW3

Sengputa, A., & Subramanian, H. (2020). *CHASM: A Blockchain Design Pattern*. https://www.researchgate.net/publication/342815273_CHASM_A_Blockchain_Design_Pattern

Shahzad, B., & Crowcroft, J. (2019). Trustworthy Electronic Voting Using Adjusted Blockchain Technology. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 24477–24488. doi:10.1109/ACCESS.2019.2895670

Sheth, A., & Subramanian, H. (2019). Blockchain and contract theory: Modeling smart contracts using insurance markets. *Managerial Finance*, *46*(6), 803–814. doi:10.1108/MF-10-2018-0510

Soffer, P., & Kaner, M. (2011). Complementing business process verification by validity analysis: A theoretical and empirical evaluation. *Journal of Database Management*, *22*(3), 1–23. doi:10.4018/jdm.2011070101

Subramanian, H. (2018). Decentralized blockchain-based electronic marketplaces. *Communications of the ACM*, *61*(1), 78–84. doi:10.1145/3158333

Subramanian, H. (2019). Security tokens: Architecture, smart contract applications and illustrations using SAFE. *Managerial Finance*, *46*(6), 735–748. doi:10.1108/MF-09-2018-0467

Subramanian, H. C., Cousins, K. C., Bouayad, L., Sheth, A., Conway, D., Salcedo, E., & Pineda, J. (2018). *Blockchain Regulations and Decentralized Applications*. Panel Report from AMCIS2018.

Subramanian, H. C., & Malladi, S. (2020). Bug Bounty Marketplaces and Enabling Responsible Vulnerability Disclosure: An Empirical Analysis. *Journal of Database Management*, *31*(1), 38–63. doi:10.4018/JDM.2020010103

Szabo, N. (1996). Smart contracts: building blocks for digital markets. *EXTROPY: The Journal of Transhumanist Thought, 18*, 2. https://hyperledger-fabric.readthedocs.io/en/release-1.4/smartcontract/smartcontract.html

Szabo, N. (1997). The idea of smart contracts. *Nick Szabo's Papers and Concise Tutorials,* 6.

Tschorsch, F., & Scheuermann, B. (2016). Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. *IEEE Communications Surveys and Tutorials*, *18*(3), 2084–2123. doi:10.1109/COMST.2016.2535718

Tso, R., Liu, Z. Y., & Hsiao, J. H. (2019). Distributed E-Voting and E-Bidding Systems Based on Smart Contract. *Electronics (Basel)*, *8*(4), 422. doi:10.3390/electronics8040422

Wang, W., Hoang, D. T., Hu, P., Xiong, Z., Niyato, D., Wang, P., Wen, Y., & Kim, D. I. (2019). A survey on consensus mechanisms and mining strategy management in blockchain networks. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 22328–22370. doi:10.1109/ACCESS.2019.2896108

Wang, W., & Siau, K. (2019). Artificial intelligence, machine learning, automation, robotics, future of work and future of humanity: A review and research agenda. *Journal of Database Management*, *30*(1), 61–79. doi:10.4018/JDM.2019010104

Wang, X., Zha, X., Ni, W., Liu, R. P., Guo, Y. J., Niu, X., & Zheng, K. (2019). Survey on blockchain for Internet of Things. *Computer Communications*, *136*, 10–29. doi:10.1016/j.comcom.2019.01.006

Wang, Y., & Malluhi, Q. M. (2019). The limit of blockchains: Infeasibility of a smart Obama-Trump contract. *Communications of the ACM*, *62*(5), 64–69. doi:10.1145/3274276

Wood, G. (2014). *Ethereum: a secure decentralised generalised transaction ledger*. gavwood.com/paper.pdf

Xiao, Y., Zhang, N., Lou, W., & Hou, Y. T. (2020). A survey of distributed consensus protocols for blockchain networks. *IEEE Communications Surveys and Tutorials*, *22*(2), 1432–1465. doi:10.1109/COMST.2020.2969706

ENDNOTES

1 "Smart Contracts," Extropy Magazine (Szabo, 1996).

2 The Theory of Contracts is a thesis published by MIT and authored by Hart and Holmstrom. The thesis extends accepted principals of Industrial Organization to demonstrate how frictions in are mitigated through Contracts.

3 Number of Ethereum transactions - https://bitinfocharts.com/comparison/ethereum-transactions.html

4 Number of transactions per second on the Ethereum blockchain should increase to more than 1 million transactions per second (https://www.ccn.com/vitalik-buterin-ethereum-will-eventually-achieve-1-million-transactions-per-second/).

5 Refer to https://consensys.github.io/smart-contract-best-practices/known_attacks/

6 Refer source code for the DappToken from https://www.dappuniversity.com/articles/code-your-own-cryptocurrency-on-ethereum

7 https://www.ibm.com/blogs/think/2018/05/everledger/

8 https://www.ibm.com/blockchain/solutions/food-trust

9 The Everledger use case is described in the following link https://www.altoros.com/blog/a-close-look-at-everledger-how-blockchain-secures-luxury-goods/

10 Cross blockchain platforms and smart contracts are an interesting research are and Komodo platform focuses on this niche in making smart contracts interoperable across chains. https://komodoplatform.com/interoperability-cross-chain-smart-contracts/

Hemang Subramanian is an assistant professor of information systems at Florida International University business school. His work on Blockchains, Spatial Arbitrage and IT entrepreneurship has been published at Information Systems Research, CAIS, CACM, Managerial Finance, IEEE Software, Journal of Database Management and IEEE Blockchains. He has presented his work on blockchains at various national and international conferences and is the author of 4 blockchain based books. He holds a Ph.D. in Information Technology Management from Georgia Institute of Technology.

Rong Liu is an associate professor of information systems at Stevens Institute of Technology, New Jersey USA and a researcher on Blockchains at IBM Watson Research Laboratory, USA. Rong Liu has widely published her research about Blockchains in Decision Sciences, IEEE, etc. and is one of the early proponents of Blockchain technologies. She holds several patents in the blockchain domain. Rong holds a Ph.D. from Penn State University.