Can Blockchains Serve an Accounting Purpose?

ABSTRACT

The blockchain has enabled the successful creation of decentralized digital currency networks. This success has prompted further investigation into the usefulness of blockchains in other business settings. Because of the blockchain's use as a ledger, the question arises whether the blockchain could become a more secure alternative to current accounting ledgers. We show that this is infeasible. By casting this question in the context of the Byzantine Generals Problem that the blockchain was designed to solve, we identify multiple flaws hindering implementation of the blockchain as a financial reporting tool. Whereas digital currencies only exist within the blockchain, economic transactions exist outside of accounting records. This distinction prevents an acceptable level of transaction verification using the blockchain model. Additionally, the security benefits of the blockchain that render it ostensibly immutable are not fully available or reliable in an accounting setting.

Keywords: financial accounting, blockchain, ledger, database

Can Blockchains Serve an Accounting Purpose?

[Neil] Postman argued that our society was sliding into a troubling relationship with technology. We were, he noted, no longer discussing the trade-offs surrounding new technologies, balancing the new efficiencies against the new problems introduced. If it's high-tech, we began to instead assume, then it's good. Case closed. [...] We're instead quick to idolize these digital doodads as a signifier of progress and a harbinger of a (dare I say, brave) new world. (Newport, 2016).

1. Introduction

This article reports on the applicability of private and public blockchains as accounting ledgers. Specifically, we explain the conceptual problem of decentralized collaboration, known as the Byzantine Generals Problem, that the blockchain was designed to solve (Lamport et al., 1982). Then, we highlight two current blockchain use cases (i.e., cryptocurrency and smart contracts) to demonstrate how that blockchain has provided practical, real-world solutions to the Byzantine Generals Problem. Finally, we highlight multiple hurdles that restrict the applicability of blockchain technology to accounting. The three hurdles we identify are (1) the desire for confidentiality that renders public blockchains undesirable; (2) the ability for firms retroactively to manipulate private blockchains; and (3) the limited transaction verification that the blockchain provides.

For as long as the accounting profession has existed, ledgers have existed to track economic transactions. Even though new technologies, such as Big Data and machine learning, have begun to change the landscape of accounting, transaction records continue to be necessary. In the current digital age, these ledgers take the form of databases, and several proprietary and open source database solutions exist with similar fundamental characteristics. The blockchain is also a database that has recently emerged in the cryptocurrency and smart contract arenas as an alternative ledger technology.[1] Because of the potential application of a ledger technology to accounting, many have expressed the desire to cast the blockchain as the future of accounting record-keeping.

The primary and most interesting difference between traditional databases and the blockchain is the blockchain's novel solution to control. Each new block of transactions added to the end of the chain is cryptographically linked to the prior block. Any attempt to manipulate a prior transaction necessitates a re-processing of all subsequent blocks in the chain. This activity would need to outpace the rate at which new blocks are added to the chain. As a result, many view the blockchain as immutable or immune to manipulation, which is the main draw of the attempt to adapt it to accounting as a transaction ledger (White, 2016). However, this assertion of immunity is not, strictly speaking, correct (Goodin, 2014).

Blockchains can be public or private. Public blockchains are permissionless, meaning that anyone can add new blocks to the chain. This is not necessarily a security risk because public blockchains are also distributed, and every member of the network has the opportunity to vote on the "true" set of transactions. This voting system replaces the role of a central authority, and the set of transactions accepted by the majority of the network is treated, going forward, as the accurate blockchain. Private blockchains add two layers of control that may render them more desirable to firms as accounting ledgers than their public counterparts. The first layer of control is privacy. Firms may not wish to publicize their transaction ledgers, and private

[1] Other uses for blockchains as decentralized databases have also emerged, such as database backends for websites. In this article we focus specifically on current and potential financial implementations, as these are the most applicable, as well as the most prevalent.

blockchains are not distributed. Second, private blockchains are permissioned, and only authorized maintainers may add transaction blocks to the chain.

We evaluate the potential for either public or private blockchains to serve as corporate transaction ledgers. This is not the first attempt to consider the implications of the blockchain for accounting and auditing. SEC Chair Mary Jo White has announced intentions to study blockchain technology for securities regulation (White, 2016). We contribute to this investigation by casting the potential for an accounting blockchain squarely in the context of the problem the blockchain was designed to address. Because of the fervor that arises at the advent of new technologies, such as Big Data, data analytics, Internet of Things, and the blockchain, the desire to adapt the blockchain to accounting will likely impair judgment regarding its feasibility. We, on the other hand, attempt to rein in this effort by highlighting the intractability of an attempt to replace current transaction ledgers with blockchains.

Our analysis proceeds as follows. In the next section, we explain the conceptual problem that the blockchain was designed to address and the minimum requirements to find a solution. In section three, we provide use cases for distributed blockchains and explain how these solve the problem presented in section two. In section four, we explain three hurdles preventing the application of the blockchain to accounting. In section five, we conclude.

2. Byzantine Generals

The Byzantine Generals Problem explains how corrupt communication threatens successful coordination across a decentralized network (Lamport et al., 1982). In this problem, a commanding general must communicate an order to multiple lieutenants. Loyal lieutenants must obey the order of a loyal general, and all loyal lieutenants must obey the same order. However, the general and any of the lieutenants might be traitorous. As a result, the general may send a different order to each lieutenant or may send no order to one or more lieutenants. Assuming that the lieutenants cannot communicate with each other, they cannot guarantee that they will all obey the same order.

The Byzantine Generals Problem was an attempt to personify machine communication. For example, the commanding general might be a database server, and the lieutenants might be clients connected to the server. The clients wish to calculate total sales revenue for the current period. A reliable database would report the same input data to all clients, but a corrupted database might report different input values to each client causing the clients to disagree about the true value of total revenue. Interestingly, this problem also applies to human failure. For example, the general might represent a customer, and the vendor and the bank might represent two lieutenants. Figure 1 demonstrates this scenario. In panel A, a loyal customer orders goods from the vendor and orders the bank to pay the vendor. (Since for the context of this problem it is necessary that the orders from a loyal general be identical across lieutenants, the uniform order could be to "exchange assets.")[2] In panel B, a traitorous customer orders goods from the vendor and then orders the bank not to release payment. Since the vendor could not distinguish between a loyal and a traitorous customer without knowing what the customer ordered the bank to do, commerce within this network could not exist.[3] Otherwise, the vendor would always ship goods

[2] These examples use dichotomous orders: exchange/not exchange. However, the results also apply to continuous orders, such as how much to exchange (Lamport et al., 1982).

[3] It is important to note that this problem is an *a priori* problem. Because of the existence of this threat to decentralized coordination (i.e., traitors *may* exist), a customer-vendor market would never form in the first place. The goal then is to find a solution that will result in the safe creation of the market.

even when a traitorous customer ordered them without the intent to pay. Furthermore, the problem requires that all loyal lieutenants follow the same order to exchange assets, but the loyal bank would follow the traitorous order not to exchange assets because the bank could not identify the customer as a traitor.

2.1 Distributed Transaction Verification

The first requirement to solve this problem is to require the lieutenants to communicate with one another and relay what orders they received.[4] A loyal lieutenant will report honestly what the general ordered, but a traitorous lieutenant might change the order when relaying it to the other lieutenants or may not relay an order at all. If the relayed order differs from the order received directly from the general, then a traitor exists, but the traitor might be the general or a lieutenant. Figure 2 displays this message relay system in the context of the customer-vendor relationship. In panel A, the traitorous customer orders goods from the vendor and orders the bank to withhold payment. The bank relays to the vendor that the customer has not authorized payment. In panel B, the loyal customer orders the bank to pay, but the traitorous bank relays to the vendor that the customer has withheld payment. Either way, the vendor cannot know for certain whether the customer has withheld payment, but the vendor lieutenant would again be compelled to ship the goods under both scenarios. The vendor-customer relationship remains threatened.

The reason for this inactivity is the inability to obtain a majority "vote" for the outcome to exchange assets. This is resolved by adding one additional loyal agent so that the number of

[4] The need for additional communication (i.e., verification) makes coordination between only two parties, one general and one lieutenant, impossible because the lieutenant has no ability to determine whether the general is a traitor. This follows from the unsolvable Two Generals Problem (Akkoyunlu et al., 1975).

loyal agents is at least three more than the number of traitors (Lamport et al., 1982). Figure 3 depicts customer-vendor scenarios with four agents and one traitor. In panel A, the loyal customer has a traitorous bank and a loyal credit card company and orders both to release payment (i.e., exchange assets), then although the traitorous bank would relay the order to the vendor to withhold payment (i.e., not exchange assets), the loyal credit card company would relay the order to the vendor to release payment (i.e., exchange assets). In a distributed network such as this no central authority exists to resolve disputes, and the majority vote would win and asset exchange would occur.

In panel B, a traitorous customer orders a loyal bank to exchange assets and a loyal credit card company not to exchange assets, then the bank and credit card company would report these disparate orders to the vendor. Because the vendor has also received an order to exchange assets, exchange would occur by majority vote (i.e., the traitorous customer would pay and receive goods regardless of the customer's true preference). Similarly, if a traitorous customer ordered a loyal bank and a loyal credit card company not to exchange assets, then the bank and credit card company would report this order to the vendor, and assets would not exchange. This latter outcome differs from the breakdown described earlier. In this case, the network by majority vote agrees not to exchange assets, whereas in the earlier case with only three agents, because of a lack of majority vote, the network could not agree to operate at all.

The scenario changes slightly if a traitor chooses not to send or relay an order. For example, if a traitorous customer ordered goods from the vendor and communicated no order to the bank or credit card company, these latter two would not communicate with the vendor. The transaction would never complete because all votes could never be tallied. As a result, it is necessary in this system to determine a default vote (e.g., not to exchange assets) for each order not received or relayed within a given time period (Lamport et al., 1982). Once the time has passed, the majority vote in this example would be not to exchange assets.

2.2 Identity Verification

Despite the benefit of distribution verification of orders received from the general, the problem remains that if more than one third of the agents in the network are traitorous, the network cannot function (Pease et al., 1980; Lamport et al., 1982).[5] An additional control is necessary to overcome this hurdle so that a decentralized network of three agents can function, even in the presence of one traitor.[6] The solution requires that any modification to the order of a loyal general by a traitorous lieutenant can be detected by allowing loyal generals to sign their orders with an unforgeable signature (Lamport et al., 1982). If the general is a traitor, then the lieutenants receive different orders. When they forward them to each other, the lieutenants recognize that the general is the traitor. Since a traitorous lieutenant cannot modify an order from a loyal general, the only valid order that a loyal lieutenant would receive would be from the loyal general, and the traitorous lieutenant would be discovered. Figure 4 displays this solution to the customer-vendor problem.

The addition of this control allows a network of three agents to function despite the presence of a traitor. Interestingly, the control of signed orders even allows two loyal agents to operate in a network of an arbitrarily large number of traitors (Lamport et al., 1982).

[5] Although this depiction of the Byzantine Generals Problem only includes one general and multiple lieutenants, the problem and solution also apply to a series of general-lieutenant combinations (Lamport et al., 1982).

[6] No solution exists if only one agent is loyal (Lamport et al., 1982).

3. Blockchain use cases

The blockchain is a database that satisfies both parts of the solution to the Byzantine Generals Problem: (1) distributed transaction verification and (2) identity verification. Two popular blockchain implementations are for cryptocurrency and smart contracts.

3.1 Cryptocurrency

The intent of cryptocurrency is to replace centralized, cash-based currency with fully decentralized, digital currency. The lack of a central authority prevents arbitration between parties in an asset exchange, so the Byzantine Generals Problem noted earlier arises. The specific issues with a decentralized, digital currency are (1) unauthorized spending and (2) double spending. Unauthorized spending is endemic to any payment system. In a cash-based system, possession of the cash is the vehicle for determining authorization, but cash can be stolen or even counterfeited. In a digital system, authorization is even harder to determine because it is more easily forged. Double spending arises from the nature of digital data. Although physical assets can only exist in a single place at one time, transmission of digital data duplicates the data, and the sender can retain a copy. In the context of digital currency, this would manifest as the ability to spend the same amount multiple times and continue to retain the same amount. The digital currency itself is not able to resolve these issues, but the ledger (i.e., blockchain) can by virtue of its solution to the Byzantine Generals Problem.

The blockchain solves the issue of unauthorized spending by requiring cryptographic identity verification for each transaction. The blockchain is a list of transactions (i.e., payments). Each payment has a sender and a recipient. In order to prove authorization, the sender must digitally sign the transaction. This signature irrefutably verifies the sender's identity. (The general's order is confirmed, and a traitor cannot modify the order without detection.)

Digital signatures are sufficient to prevent unauthorized spending, but another tool is necessary to prevent double spending: distributed transaction verification. A central authority would be able to prevent double spending by ruling duplicate payments invalid, but cryptocurrency has no central authority to determine which transaction in a double spending scheme is first and therefore valid. Instead, the blockchain relies on a voting system to determine the next set, or block, of transactions in the chain. A blockchain maintainer's decision to verify a particular set of transactions constitutes a vote for a blockchain with the newest block comprising that set of transactions. Different blockchains can have different methods for verifying transactions.

The earliest blockchain verification method is proof-of-work, in which the maintainer's computer, solves a math problem involving a hash of the set of transactions to verify and a hash of the prior block in the chain. No inferrable answer to the problem exists, so the computer must make recursive guesses. As a result, transaction verification must take time; this is the premise of proof-of-work. Proof-of-work prevents revisionist history. Because blocks are added to the blockchain sequentially and because each verification process involves a hash of the prior transaction in the chain, any attempt to manipulate an earlier transaction in the blockchain would require re-verification of all subsequent blocks. In order to complete the proof-of-work necessary to re-verify all subsequent blocks, a manipulator would need to outpace the combined computing power of all other maintainers, who continue to work on previously verified chains. Proof-of-work also creates a convenient method for tallying votes and determining which new block has received the most votes. The set of transactions on which maintainers spend the most computing power has the highest likelihood of having the proof-of-work problem solved first. As a result, a verified set of transactions represents the majority vote for the next block in the chain.[7]

Another popular verification method is proof-of-stake. Proof-of-stake is a less resource-intensive alternative to transaction verification. The holders of the cryptocurrency verify transactions based on their respective account balances. The more coins a participant holds, the more likely that participant is to receive the task of transaction verification. This method encourages holders of the currency to self-police. The majority vote in this case is not captured by a majority of effort invested to solve the mathematical problem, but rather a majority of vested ownership.

By virtue of these verification systems, the network participants can reach consensus, but only if they receive evidence of the verification. Distributed transaction verification requires distribution, which follows verification. Once a set of transactions is verified, the maintainer broadcasts the newly modified blockchain to all other members of the network. (The lieutenant has relayed the general's order to all other lieutenants.)

The Byzantine Generals Problem is solved. To use the example of the customer and vendor, the customer orders goods from the vendor and sends an unforgeable, digitally signed payment to the bank, who in this case is the entire network of ledger maintainers. The maintainers validate the transaction and add it to the blockchain and relay the blockchain to the

[7] This statement is also not, strictly speaking, accurate. The set that consumes the most processing power will most likely be the next block in the chain. However, an alternative block processed by a minority has a strictly positive probability of becoming the next block because any maintainer may be able to solve the proof-of-work problem first. The verification processes will not always represent the majority vote for a specific block, but over time, the blockchain will represent the majority vote as the majority chooses to continue to add blocks to a chain that includes one or more blocks verified by a minority or processing power.

vendor as confirmation that the customer has paid for the goods. The vendor can ship the goods with confidence that the customer is not a traitor.

3.2 Smart contracts

In the customer-vendor examples, the supposed victim of a failure in the network is the vendor who might ship goods without payment. However, it is equally likely that a loyal customer can fall victim to a traitorous vendor. Although a cryptocurrency blockchain can protect against traitorous customers who withhold payment from vendors, in the absence of additional contractual agreements, payments recorded in a blockchain are non-recourse (Brakeville & Perepa, 2016). In other words, once the customer's payment has been verified, it is irreversible should the vendor ship either incorrect or defective goods or no goods at all.

Smart contracts extend the blockchain functionality to provide an additional layer of protection for both parties in a transaction. Many refer to this functionality as part of Blockchain 2.0, but an early contributor to the source code of both cryptocurrencies and blockchains wrote a treatise on smart contracts in 1994, long before the conception of cryptocurrency (Szabo, 1994). Smart contracts are contracts whose terms are programmed into a blockchain. They are smart because the contract terms execute automatically when certain conditions are met. Because the history of the blockchain is unchangeable, neither party can manipulate the terms of the contract. Furthermore, because the blockchain is a database, it can not only trigger digital asset exchange, it can store digital assets prior to exchange in the same sense as an escrow account (Szabo, 1994). Using the customer-vendor example, a smart contract could hold the customer payment until the customer has received and verified the goods. Following this event, the smart contract could release payment to the vendor.

Recent investments in this technology have sought to apply the concepts of smart contracts to the financial, legal, and even music industries in which conditional digital payments often occur (e.g., derivatives, escrow, royalties, etc.) (Morrison, 2016). Many believe that blockchains are the future of digital asset management and digital contracts (Bakey, 2016).

4. Blockchain in accounting

In applying the Byzantine Generals Problem to accounting, the firm itself would play the role of the general, and the auditor and investors would be two lieutenants. Currently, firms can retroactively manipulate their ledgers to manipulate earnings, and the hope is that the adoption of a blockchain ledger would prevent this behavior (Morrison, 2016). A popular belief is that blockchains will be a more, and perhaps even perfectly, reliable alternative to traditional ledgers because no one can retroactively alter them. However, multiple hurdles exist preventing the adoption of this alternative as both an effective and efficient solution. Some of the hurdles directly affect the veracity of the fundamental belief that a blockchain is immutable.

4.1 Confidentiality

The blockchain solution to the Byzantine Generals Problem relies on the distributed nature of the blockchain. In the absence of a central authority, the network must agree on current account balances (i.e., the true list of past transactions). This requires open publication of the entire blockchain. However, the mere fact that firms have not chosen to publish their current ledgers provides evidence that they prefer the confidentiality of private ledgers, notwithstanding recent calls to release this data to the public (Krahel & Titera, 2015). This preference for confidentiality is unsurprising as vendor and customer lists, unit costs, and strategic transactions stored in ledgers can constitute competitive trade secrets.

A public, permissionless blockchain is available to anyone who wishes to view it. Additionally, anyone can become a maintainer and verify the next block in the chain. No access controls or authorization protect reading or writing to this blockchain. A private, permissioned blockchain, on the other hand, is more similar to a traditional transaction ledger. The owner of the blockchain keeps its contents confidential, and only those users with read and write permissions have access to it (Buterin, 2015). Although the security and confidentiality of a private blockchain may appeal to a firm, without the public distribution of the blockchain, investors cannot directly participate in this network, and the network reverts to only two participants (firm and auditor), for which no solution to the Byzantine Generals Problem exists (Akkoyunlu et al., 1975). That is not to say that no network could exist, for example the two lieutenants could be the auditor and a private lender, but a blockchain that is private fails on its face to create a solution to the Byzantine Generals Problem that comprises the firm and its current and potential investors.[8]

However, when compared to other limitations of the application of the blockchain to accounting, the problem of privacy likely has multiple solutions. One of the basic tenets of cryptocurrency is to promote privacy, and researchers have investigated extensions to current

[8] Another network could involve the firm, its vendors or customers, and the auditor, but this would require that the supply chain network adopt the same blockchain technology. Even if all members of a single supply chain adopt the same blockchain technology, different supply chains might adopt different technologies which would force a single firm to have multiple duplicate blockchains depending on the counterparty. This is already a problem in the cryptocurrency arena because different currencies have different, incompatible blockchains (Chester, 2016), and it would present a significant hurdle to adoption in this setting, as well.

privacy-promoting practices (Kosba et al., 2016).[9] As a result, confidentiality may not pose permanent risks to the adoption of public or private blockchains for accounting.

4.2 51% attack

Even if firms elect to distribute their blockchains to the public, the blockchains may not serve their intended purpose if they are not protected against revisionist history. Proof-of-work prevents retroactive modification of the blockchain by requiring a manipulator to out-compute all other network participants. However, if the manipulator were a group with 51% of the computing power, revisions to the blockchain could arise. A firm's blockchain would constantly be at risk. If the firm kept the blockchain private, the firm would automatically have 100% control over transaction validation, and it would be able to rewrite any portion of the blockchain, as necessary. Even if the firm was unwilling to manipulate its own blockchain in this manner, a security breach could place this same 100% control in the hands of an unauthorized individual, who could proceed to rewrite the blockchain (Hampton, 2016).

If the firm used a distributed blockchain, the firm would likely retain well over 50% control over the blockchain because the role of primary maintainer would fall to the firm itself. Under either a proof-of-work model or a proof-of-stake model, current and potential investors would have little incentive to verify transactions because of the computing cost and/or the accounting knowledge and insider access necessary to verify an accounting transaction. This would return the majority of the control to the firm resulting in the same problem as with a private blockchain.

[9] We thank an anonymous reviewer for pointing this out to us.

One method for restricting the amount of computing power left with the firm could be to require the auditor to participate actively in the transaction verification process. This could shift as much as 50% or more of the control away from the firm. Furthermore, even if the firm insisted on a private blockchain, it could not restrict access to the blockchain from the external auditor, so this option would remain available in situations with both public and private chains.

4.3 Incomplete transaction verification

Even if the auditor and the firm shared transaction verification responsibility such that neither possessed a sufficient amount to rewrite the blockchain, transaction verification would remain ineffective. Blockchain verification methods are not sufficient for transaction validity from an accounting perspective. These methods serve primarily to prevent double spending.[10] The maintainers of these blockchains know nothing about the true validity of the transaction. In the blockchain application of the Byzantine Generals Problem, the general's order is to transfer assets. The blockchain maintainers know nothing of the agreement between the two parties that resulted in the asset transfer. They only know whether the transaction uses unspent inputs and is digitally signed. However, the economic events that financial, management, or tax accounting record occur independent of their measurement. In this case, the general's order would be, "Record *this* economic event in *that* manner." A maintainer would need to confirm that the transaction captures a verifiable economic event, that the recorded value satisfies accounting principles, that no economic events have been overlooked (i.e., completeness), etc. Following the Byzantine Generals setup, if the network of lieutenants, or maintainers, concluded that the

[10] This statement was confirmed by the chief cryptographer for the blockchain company Ripple. We thank an anonymous reviewer for making us aware of this blockchain solution.

general was a traitor, then they would refrain from recording the economic event. Unfortunately, failure to report is not an acceptable outcome.

Furthermore, the ability to determine the correct ledger entry requires extensive accounting knowledge. Many investors (i.e., potential maintainers) would not have the necessary knowledge. The auditor as one lieutenant has the ability to verify accounting transactions, but the historical exercise of this ability has not required a blockchain. Accountants already know that controls are necessary to prevent fraud, earnings management, and accounting error, and the adoption of a blockchain will not dispense with the need for these. For example, a blockchain will not prevent asset misappropriation, nor will it prevent erroneous measurement or estimation of valid transactions.

This difficulty arises, in large part, because of the difference between asset transfer (i.e., a transaction) and the recording of asset transfer (i.e., financial reporting). The blockchain as a database is not merely a record of transactions; it actually handles digital asset transfer. This is one reason why it is useful for smart contracts. In the case of cryptocurrency, all currency exists exclusively within the blockchain, so the blocks in the chain are asset transfers, not simply records of asset transfers. Financial accounting, on the other hand, is transaction measurement.

Given these limitations, it is not clear that a blockchain, which was designed to solve another specific problem, would be any better than a traditional ERP system, which was designed to solve specific accounting problems, in tracking accounting transactions. Firms already have methods for verifying transactions. They also already restrict access and establish audit trails to prevent retroactive manipulation of traditional ledgers. Proof-of-work is a much less efficient alternative to current verification and security measures because of high amount of processing power and electricity needed to process each set of transactions.

Furthermore, auditors can receive distributed, or shared, copies of transaction ledgers and independently verify the transactions without the use of a blockchain. Investors can receive relevant information without access to transaction lists, and even if they desired and received transaction lists, it is not clear that a blockchain would increase the reliability of the numbers.[11] Investors with public blockchains as transaction lists could be candidates for proof-of-stake verification (e.g., based on number and age of shares held), but the ability and time to verify transactions consistent with accounting regulations, as well as the necessary insider access to supporting documentation, is unavailable to many investors, rendering this verification method, as yet, ineffective.[12] In a network including the firm, the auditor, and the investor, if the firm is the traitor, the firm can manipulate its own message. The role of the auditor is to prevent that, and it can serve that role without a blockchain. If the auditor is also a traitor, then only one loyal agent remains, in which case the problem has no solution, anyway.

5. Conclusion

Accounting ledgers are a central part of a firm's accounting records, and in today's digital environment, the ledger resides in a database. The blockchain is a new database solution that has received considerable attention in the business community. It has already enabled the

[11] Following the model of smart contracts, it would be possible to program a blockchain to record a transaction according to predetermined accounting rules, but this possibility would not be unique to the blockchain. Rather, it would likely extend to many database configurations, including ERP systems. The feasibility of condensing accounting regulations down into a series of programmable, computer-interpretable commands is beyond the scope of this paper.

[12] Below, we briefly discuss conditions for potential future implementation of this consensus method.

creation of multiple decentralized digital currency networks (e.g., Bitcoin, Litecoin, Ethereum, etc.), and firms, most notably in the financial industry, have begun to invest in the development of this technology for other purposes (Morrison, 2016). Perhaps because of the intuitive link between the concept of the blockchain ledger and accounting ledgers, some have begun to wonder whether this blockchain database technology could become a more secure, immutable alternative to existing ledger database solutions.

Blockchains have many useful business implications. The mere fact that they facilitate decentralized asset exchange is valuable. Additionally, smart contracts reduce the need for intermediaries in financial transactions, such as futures contracts, escrow payments, royalties, insurance payments, and many other contracts that involve conditional payment streams. However, we conclude that the difference between digital asset management, including the negotiation of digital contracts, and financial reporting represents a chasm that the blockchain is not as well suited to bridge.

Changes to the business world necessitate changes, including technological changes, to the accounting profession, but not every new business technology will facilitate advancements in accounting. We attempt to discourage some of the fervor common to the advent of new technologies that has arisen with respect to the blockchain. Our purpose is to make clear what problem the blockchain solves (i.e., the Byzantine Generals Problem) and to explain how this problem and its solution fit poorly in the context of financial reporting. Although some of the characteristics of the blockchain, such as encrypted accounts to promote counterparty confidentiality in financial transactions, may represent valuable innovations for the future of accounting records, issues currently remain preventing effective application of this ledger technology to accounting.

One characteristic of the blockchain that may play an interesting future role in accounting is the mechanism for establishing consensus. Transaction verification is designed to obtain consensus, but this is not unique to the blockchain. In the current accounting world, auditors provide a form of verification, and investors conclude that audited financial statements satisfy the minimum requirements for consensus. Proof-of-work is not an efficient solution, but proof-of-stake as a verification method provides an intriguing future opportunity for investors, as well as creditors, to play a more active role in the verification and consensus process. In the current regulatory environment, investors have limited capacity to verify accounting records personally because many have insufficient accounting knowledge and because their access to insider information is restricted. However, if firms were willing to be completely open with their internal accounting documentation—not only ledgers, but also all supporting documentation—then institutional investors, as well as some high net-worth retail investors, and creditors could be asked to verify transactions recorded by the firm based on their relative debt or equity stake in the firm. Even though this could be a cumbersome task, because of the monetary value of their stake, these stakeholders would derive benefit from this effort, even if it involved employing their own auditors to perform the verification process on their behalf.

However, even in an open world that would facilitate this opportunity, the verification model would need to expand to address unrecorded (i.e., missing) transactions, and not simply a verification of recorded transactions. Additionally, blockchain networks choose not to record a transaction if consensus fails, but failure to record in an accounting setting is not an acceptable solution. Future research can further investigate the application of blockchain methods of transaction verification and consensus to the problem of unrecorded transactions or disagreement of accounting treatment.

REFERENCES

Akkoyunlu, E. A., Ekanadham, K. & Huber, R. V. (1975). Some constraints and tradeoffs in the design of network communications. *SIGOPS Oper. Syst. Rev. 9*(5), 67-74.

Bakey, P. (2016). How blockchain is reshaping business. http://www.forbes.com/sites/sap/2016/11/03/how-blockchain-is-reshaping-business/

Brakeville, S. and Perepa, B. (2016). Blockchain basics: Introduction to business ledgers. https://www.ibm.com/developerworks/cloud/library/cl-blockchain-basics-intro-bluemix-trs/

Buterin, V. (2015). On public and private blockchains. https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/

Chester, J. (2016). The blockchain wars: How startups and enterprises are competing to create the web 2.0. http://www.forbes.com/sites/jonathanchester/2016/04/14/the-blockchain-wars-how-startups-and-enterprises-are-competing-to-create-the-web-2-0/

Goodin, D. (2014). Bitcoin security guarantee shattered by anonymous miner with 51% network power. http://arstechnica.com/security/2014/06/bitcoin-security-guarantee-shattered-by-anonymous-miner-with-51-network-power/

Hampton, N. (2016). Understanding the blockchain hype: Why much of it is nothing more than snake oil and spin. http://www.computerworld.com.au/article/606253/understanding-blockchain-hype-why-much-it-nothing-more-than-snake-oil-spin/

Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016). "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. *2016 IEEE Symposium on Security and Privacy*, 839-858.

Krahel, J. P., & Titera, W. R. (2015). Consequences of Big Data and formalization on accounting and auditing standards. *Accounting Horizons 29*(2), 409-422.

Lamport, L., Shostak, R., & Pease, M. (1982). The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems 4*(3), 382-401.

Morrison, A. (2016). Blockchain and smart contract automation: How smart contracts automate digital business. http://www.pwc.com/us/en/technology-forecast/blockchain/digital-business.html

Newport, C. (2016). *Deep work: Rules for focused success in a distracted world*. New York: Grand Central Publishing.

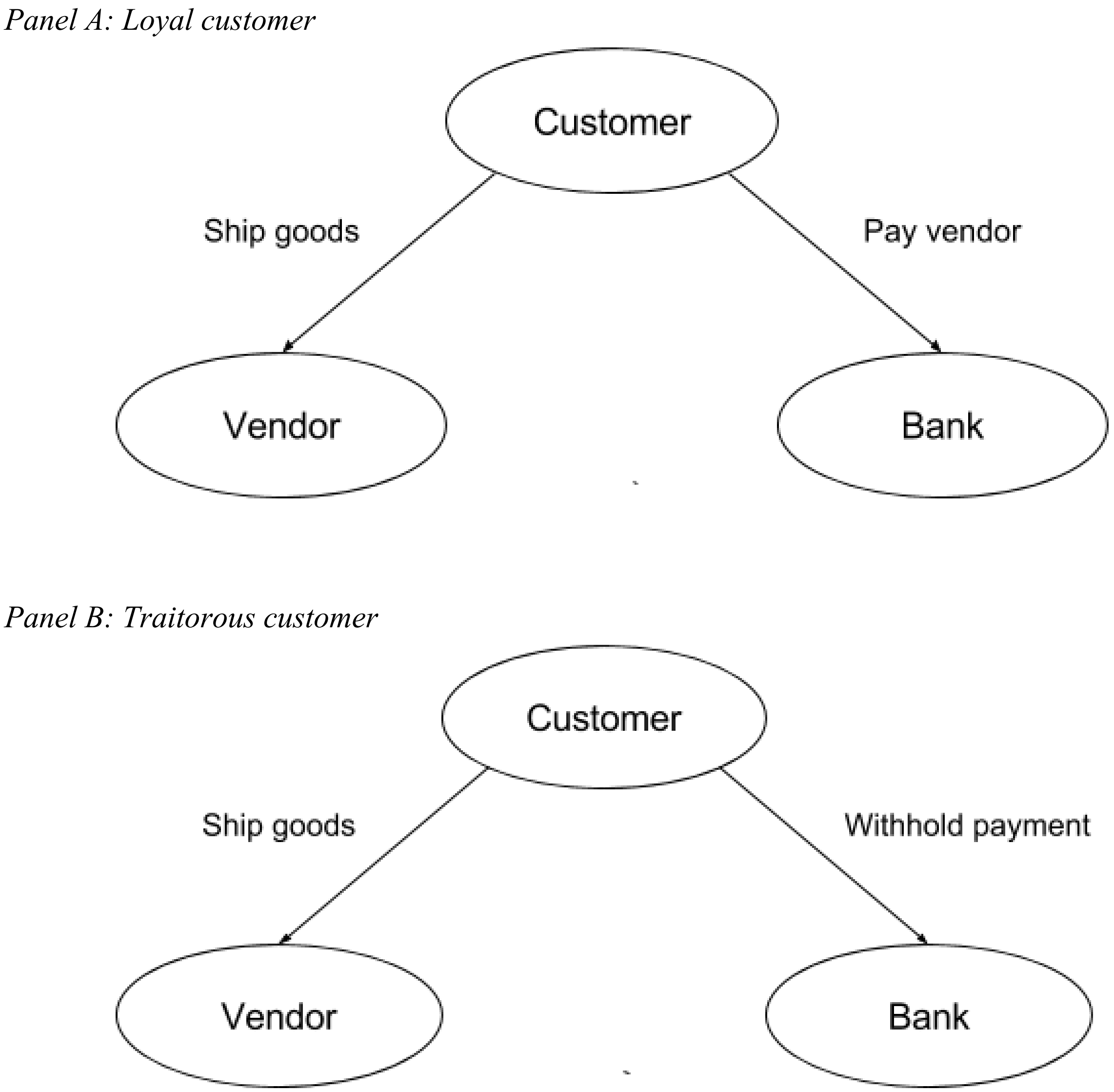
Pease, M., Shostak, R., & Lamport, L. (1980). Reaching agreement in the presence of faults. *J. ACM 27*(2), 228-234.

Szabo, N. (1994). Smart contracts.
http://www.virtualschool.edu/mon/Economics/SmartContracts.html

White, M. J. (2016). Keynote Address at the SEC-Rock Center on Corporate Governance Silicon Valley Initiative. https://www.sec.gov/news/speech/chair-white-silicon-valley-initiative-3-31-16.html

FIGURE 1

Decentralized Network without Distributed Message Verification



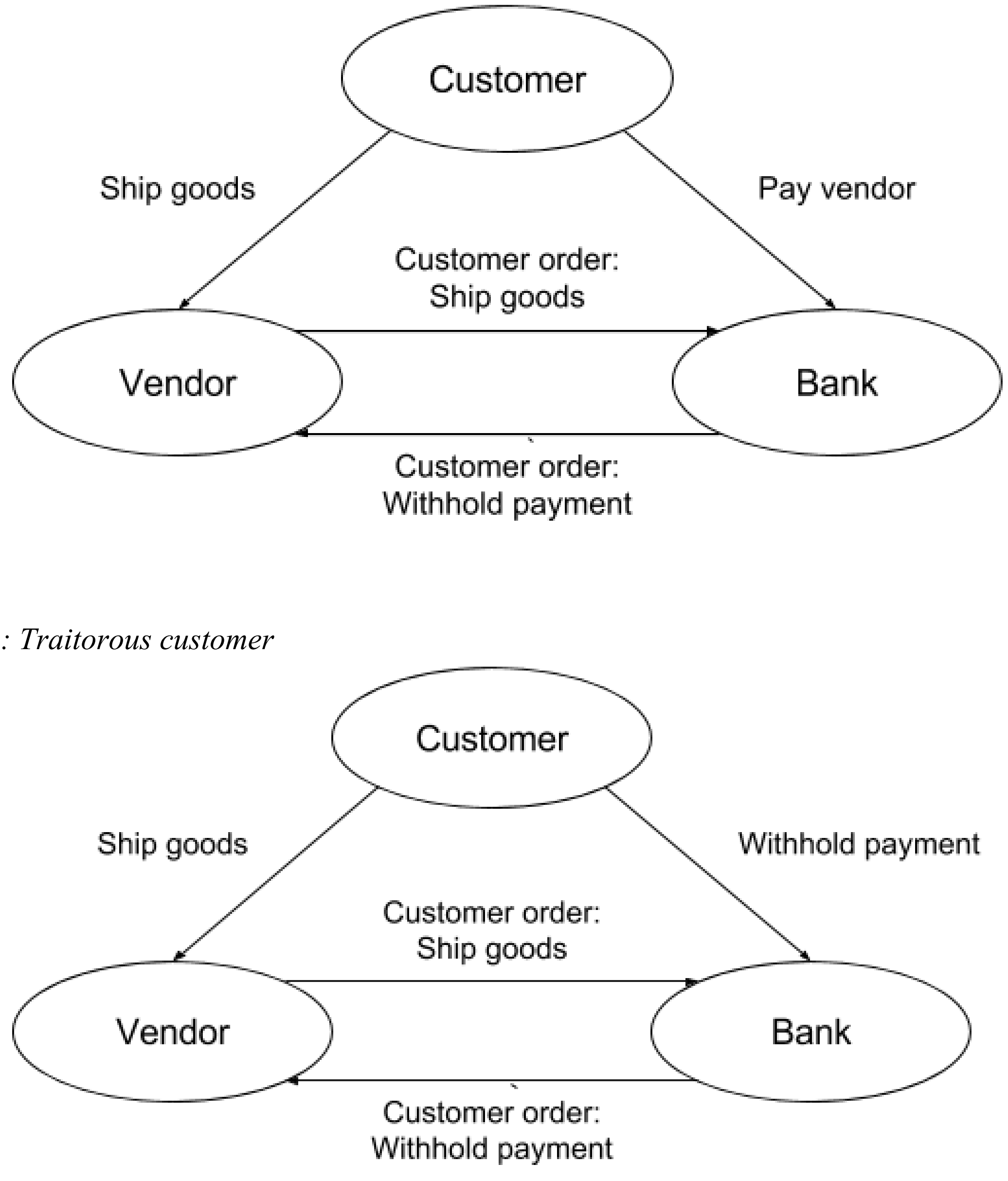
Panel A: Loyal customer

Panel B: Traitorous customer

This figure portrays the problems that arise in a decentralized network of agents attempting to exchange assets. In panel A, the loyal customer gives the same order to both the bank and the vendor. In panel B, the traitorous customer gives different orders to the bank and the vendor. Because the bank and vendor cannot communicate with each other, they cannot distinguish between a loyal and a traitorous customer.

FIGURE 2

Decentralized Network with Distributed Message Verification and Three Agents



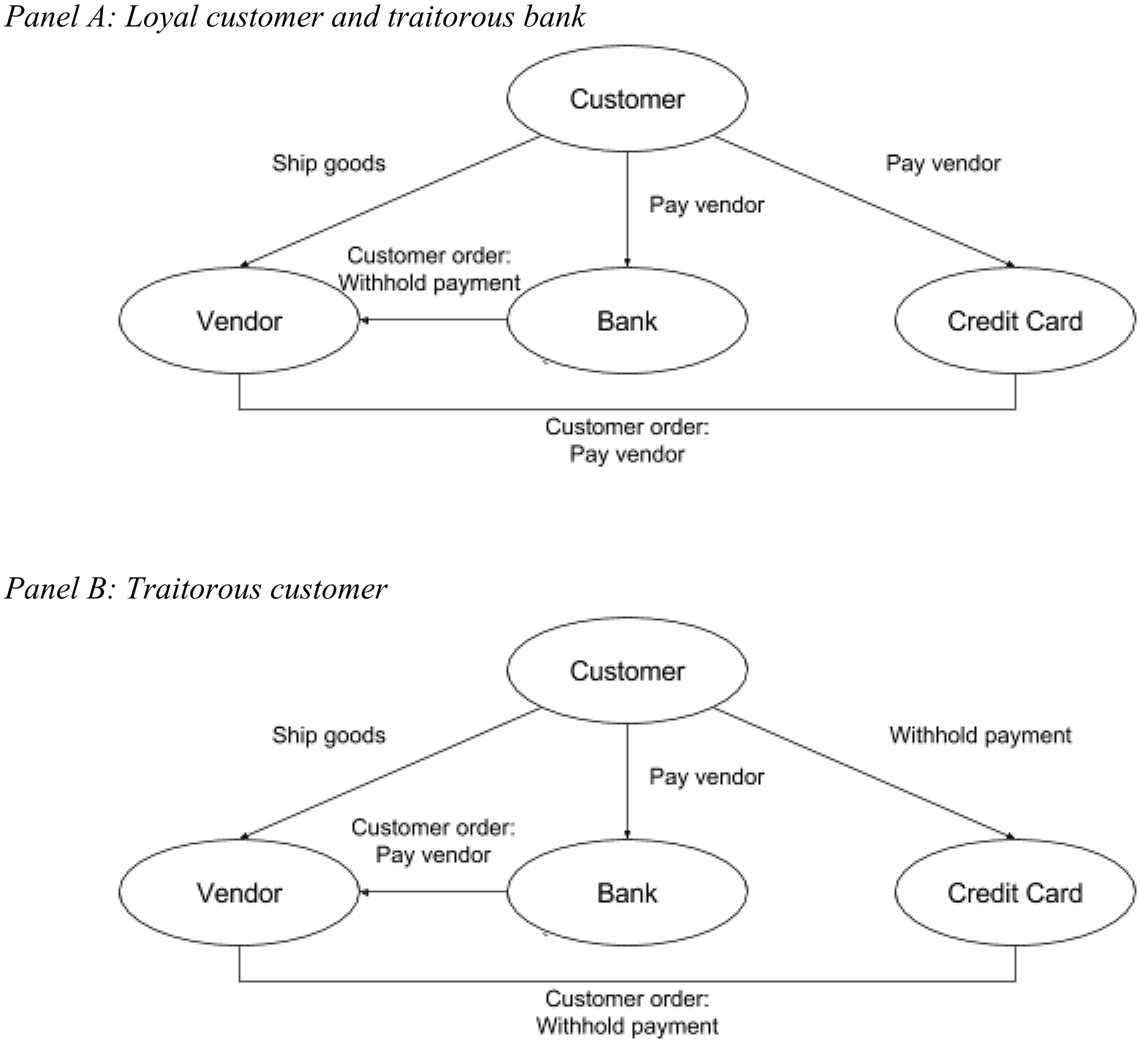
Panel A: Loyal customer and traitorous bank

Panel B: Traitorous customer

This figure portrays the problems that arise in a decentralized network of three agents attempting to exchange assets when the agents can communicate with each other. In panel A, the traitorous bank manipulates the customer's order. In panel B, the traitorous customer relays separate messages to the vendor and the bank. Even though the vendor and the bank can relay the customer orders, the vendor cannot determine whether the customer or the bank is the traitor.

FIGURE 3

Decentralized Network with Distributed Message Verification and Four Agents



Panel A: Loyal customer and traitorous bank

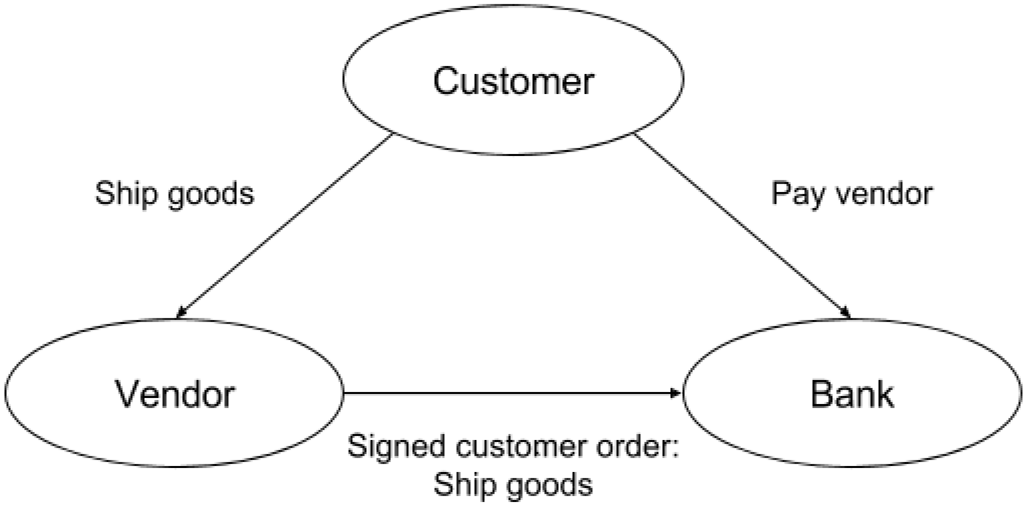
Panel B: Traitorous customer

This figure shows how adding an additional loyal agent to the network allows for consensus in a decentralized network. Because the agents can communicate with each other, they can determine by majority vote which order to follow. If the customer is loyal, the majority vote will be for the customer's order. For simplicity, the figure only displays messages relayed to the vendor, but in actuality every agent receives a copy of the order received by every other agent. In panel A, the traitorous bank manipulates the customer's order, but the credit card company relays the true order. The vendor acts based on the majority order to exchange assets. In panel B, the customer sends different messages to the bank and the credit card company, but they relay these to the vendor. The vendor again acts based on majority vote to exchange assets.

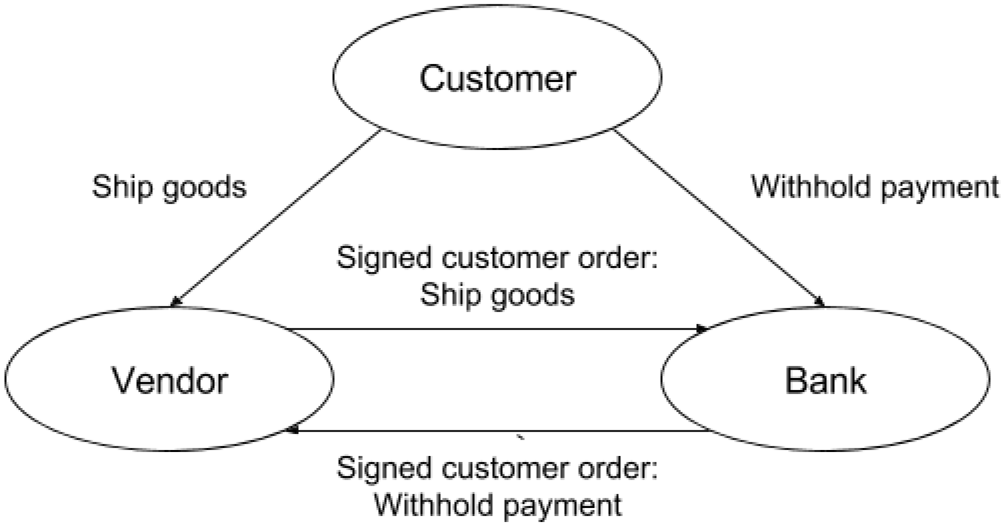
FIGURE 4

Decentralized Network with Distributed Message Verification and Identify Verification

Panel A: Loyal customer and traitorous bank



Panel B: Traitorous customer



This figure portrays how signed messages solve the problem of collaboration in decentralized network of three agents attempting to exchange assets when the agents can communicate with each other. In panel A, the traitorous bank cannot manipulate the customer's order and refrains from relaying anything to the vendor. The vendor follows the loyal customer's order. In panel B, the loyal bank forwards the customer's signed order to the vendor, and the vendor see that it is different from the order received directly from the customer. The vendor knows that the customer is a traitor.