A Blockchain-Based IoT Framework for Oil Field Remote Monitoring and Control

YANJUN ZUO AND ZHENYU QI
Department of Accountancy and Information Systems, University of North Dakota, Grand Forks, ND 58202, USA
Corresponding author: Yanjun Zuo (yanjun.zuo@und.edu)

This work was supported by the North Dakota Established Program to Stimulate Competitive Research (ND EPSCoR).

ABSTRACT

The oil and gas industry involves a high level of operational expenditure and often faces high risks of asset safety and operational failures. It has never been so important to monitor and control the oil field operations remotely in real-time to ensure safety and efficiency. The traditional monitoring and control systems for oil field operations are typically centralized, prone to failure, and lack efficiency. Blockchain technology mitigates the centralization problem by creating a decentralized, immutable and transparent control environment for automatic monitoring and control of industrial operations. In this study, we propose a blockchain-based IoT framework for real-time monitoring and control to increase oil field operation and asset efficiency and safety. We present the key components of the framework, including the system architecture, operation flows, algorithms, and smart contracts. As a proof-of-concept modeling, a smart contract is developed and validated on a blockchain test platform. A comparative analysis shows the advantages of using blockchain technology and smart contract to provide trustworthy and automatic monitoring and control for oil field operations.

INDEX TERMS

Internet of Things, blockchain, smart contract, remote monitoring, control, industrial operations.

I. INTRODUCTION

For the oil and gas industry, the real-time monitoring of its assets is important for safety and efficiency. Oil and gas is a highly regulated and capital-intensive industry. Many operations are conducted in remote and harsh environments. While the global demand for crude oil still remains high, low oil prices for a longer period and increasing regulatory constraints challenge the oil and gas operators to be more innovative to reduce the costs and increase efficiency [1]. Advanced techniques such as high-speed networks, mobile devices, data analytics, and artificial intelligence have never been so important for the oil and gas industry to increase productivity and revenue, reduce safety and environmental risks, and increase operational reliability. It is essential for oil and gas companies to operate assets optimally while ensuring safety.

In the past few decades, the oil and gas industry has mainly used supervisory control and data acquisition (SCADA) system and wireless sensor networks for remote monitoring and control. SCADA uses industry-standard or manufacturer proprietary protocols to collect and transmit data. Typical legacy SCADA protocols include Modbus RTU, RP-570, Profinet and ComNet. These communication protocols, with the exception of Modbus, are all SCADA-vendor specific but are widely adopted and used. Standard protocols are IEC 60870-5-101 or 104, IEC 61850 and DNP3. Many of these protocols now contain extensions to operate over TCP/IP [2]. However, there are some drawbacks of SCADA, such as non-interoperability between objects, high maintenance costs, inability to expand equipment, and high latency [1].

As an improvement, the Internet of Things (IoT) and the related technologies have been proposed to ease these problems. As a ubiquitous computing technique, the IoT enables various smart devices to communicate using the Internet, therefore achieving automatic machine-to-machine communications. Advance in networking and data analysis has increased the ability and applications of IoT in various industries. IoT has dramatically increased the ability of organizations to monitor industrial equipment and environmental conditions. Equipped with smart devices, data can be collected, and useful information can be retrieved from the data to remotely monitor and control the operations of the
equipment. Essentially, IoT enables seamless real-time data collection, remote monitoring, and control of the equipment and processes in oil field operations. This can not only reduce the human interventions of the oil field operations but also increase automatic diagnosis and control of the essential equipment and operations more efficiently. Together with the advanced data analysis techniques, the data collected through IoT systems provide actionable knowledge to ensure that the oil field operations can run smoothly and safely. IoT has been used in the oil and gas industry for remote monitoring [3], [4], asset optimization [5], security and predicative maintenance [6], safety and compliance [7], and situational awareness [8]. Deploying an IoT-based solution as part of the digital transformation is one of the areas that many oil and gas companies are investing. Examples of IoT-based remote monitoring and control include pressure relief valve monitoring, tank-level monitoring, infrastructure monitoring, sucker-pump monitoring, corrosion monitoring, production and well monitoring/optimization, pipeline monitoring, the performance of rotating equipment monitoring, and leak detection [1].

Traditionally, the majority of the IoT systems are centralized, i.e., the IoT devices connect to a centralized server to communicate with each other. This raises several issues, including security and data privacy and trust among the IoT users and the centralized center [9]. The single point of failure is also a concern as the centralized server could be the vulnerable point of the entire system [10]. If the server is compromised or malfunctions, the operations of the entire system are jeopardized. From an operation and security perspective, a decentralized system using IoT is highly desirable to achieve a higher level of reliable, secure, and automatic monitoring and control of the oil field operations. Any disruptions of the oil field operations could cause a significant loss of energy production and business revenues.

Given the problems of traditional remote monitoring and control systems, Blockchain provides a promising solution to achieve a trusted and secure decentralized IoT monitoring and control system for the oil and gas industry. As a distributed digital ledger technology, Blockchain provides transparent and immutable data storage to record any transactions. It is based on a decentralized architecture that allows multiple independent entities to maintain a local copy of the data and transactions through a mining process and a consensus protocol to ensure consistency among various local copies. Therefore, a true, global view of the data and transactions is available to everybody and can be verified reliably. Blockchain was first widely used in Bitcoin [11] to support cryptocurrency transactions. Blockchain uses a peer-to-peer network with a consensus mechanism to allow users to reliably conduct transactions. The recorded transactions are validated, agreed-up, and replicated by all the entities in the blockchain network. Blockchain enhances fault tolerance and removes the single point of failure of a traditional IoT system. Using a fully distributed peer to peer network, all the IoT devices can communicate with each other without going through a centralized server. Since no single entity controls the blockchain system, IoT data and transactions are immutable [12]. Since Blockchain enhances trust, security, reliability, and decentralization, a trustless platform can be established among the participants of the IoT system.

Furthermore, Blockchain provides the functionality of programmable logics through smart contracts and treat IoT interactions as transactions. A smart contract is a computer code deployed in a blockchain network, which automatically executes based on the rules defined in the software. A smart contract is duplicated in each node of the blockchain system and resists to any tamper or modification. It eliminates the involvement of any third party, thus eradicating any unnecessary costs and possible processing delays [13]. Importantly, a smart contract can provide a degree of automatic in identifying problems in oil field operations and takes the prompt actions without human intervention. In our research, we use smart contracts programmed with a set of rules which issue a command of actions based on real-time IoT data.

The characteristics of the decentralization, security, and transparency of the blockchain technology have made it well applied in many areas. For example, in the healthcare industry, BrustIQ,¹ as a blockchain healthcare data company, has developed a blockchain platform to record personal health data, which enables sensitive medical information to be safely transmitted between patients and doctors. Patientory² is a blockchain-supported medical record system, through which doctors can access the patient’s medical history, medical records, and a series of information that doctors need to know. MedShare [14] is a decentralized framework to facilitate secure electronic health record sharing. It uses smart contracts help build a trusted environment where for the healthcare centers to share their encrypted health records. In the media industry, the Open Music Initiative³ has created an open-source agreement to put music copyright data on the Blockchain so that music creators can be recognized for their work and get the right remuneration. Blockchain can also improve government. At present, some state governments are using voting platforms built on Blockchain to ensure the authenticity of votes. In the financial industry, blockchain-based transfers can make the transfer process more efficient and reduce third-party fees. In the logistics industry, Blockchain can build greater trust in the industry and make the logistics process more streamlined and automated [15]. Even the mobile gaming can benefit from the technological developments in Blockchain and location-based IoT systems [16].

In this paper, we present a blockchain-based IoT framework for the oil field equipment and operation monitoring and control. Our framework facilitates collecting timely data for remote conditional monitoring, performing the most appropriate control actions to minimize facility downtime, and

¹ https://www.burstiq.com/
² https://patientory.com/
³ https://open-music.org/
conducting machine-to-machine communications for greater automation and control. Although considerable work has been proposed in the areas of Blockchain and IoT, the study and use of an integrated blockchain and IoT system in the context of the oil and gas field has not been explored much. To our best knowledge, our research is among the very few early works to apply IoT and Blockchain in the oil and gas industry to achieve (1) reliable, secure, and effective remote monitoring, and (2) automatic and efficient control of the oil field operations. Specifically, we make the following contributions in this work:

- We propose a blockchain-based IoT framework for remote monitoring and control in the oil and gas industry with system architecture and operation flows
- We develop algorithms and models for secure authentications of IoT components and Blockchain smart contract-based oil field monitoring and control
- We simulate and verify the smart contract with three oil field use cases
- We discuss the advantages of using blockchain and smart contracts for oil field remote monitoring and control and analyze the security properties of the system

The rest of the paper is organized as follows. Section 2 discusses the background and related work. Section 3 describes our proposed framework, including the system architecture, operation flows, the main components of the framework, and the algorithms. Section 4 presents our simulation of the smart contract as the core part of our framework. Section 5 provides additional discussions about the proposed framework. Finally, we conclude our paper in Section 6.

II. BACKGROUND AND RELATED WORK

In this section, we first discuss the related background knowledge on IoT and Blockchain. Then we discuss the related work to this research.

A. BACKGROUND

1) INTERNET OF THINGS (IoT)

The Internet of Things (IoT) refers to a set of technologies to link physical objects – “things” or smart devices - to exchange data over the Internet without requiring human-to-human or human-to-machine interactions. Those physical objects are equipped with sensing, identification, processing, communication, and actuation capacities and programmed with software and other technologies to collect data and execute certain commands correspondingly.

A basic IoT system architecture can be described with three tiers: devices, edge gateway and the cloud [18]. The devices include physical objects embedded with sensors and actuators which can collect data and communicate with an edge gateway. The sensors of IoT can be selected according to the nature of the objects they need to monitor. There are different types of sensors, including temperature sensors, humidity sensors, pressure sensors, gyroscopes, infrared sensors, and so on. The actuator is a device that is embedded in the object to make it respond to the commands of IoT. The commonly used communication protocols in an IoT system include Modbus, Bluetooth, Zigbee or other proprietary protocols. The IoT gateway is a more computationally powerful device that connects to multiple IoT devices and provides communications and data aggregation functions such as preprocessing data, securing connectivity to the cloud, and even edge analytics or fog computing. In our study, an IoT gateway may participate a blockchain network as a member node to carry out the functions of issuing transactions, mining blocks, and synchronizing the blockchain data. The cloud if an IoT system includes various database systems that store sensor data such as time series databases or asset stores using back-end data storage systems (e.g., Cassandra, PostgreSQL) [17].

Several technologies enable an IoT system. Smart devices are assigned unique identifiers such as Electronic Product Code for RFID tags and IP addresses (IPv4 or IPv6) for various communicating devices. The communication systems include short-range wireless (e.g., Bluetooth meshing networking, Near-field communication, RFID, Wi-Fi, Zigbee), medium-range wireless (e.g., LTE-Advanced and 5G), long-range wireless (e.g., Low-power wide-area networking and Very small aperture terminal) and wired (e.g., Ethernet and Power-line communication) [18].

2) BLOCKCHAIN TECHNOLOGY

Blockchain is a distributed digital ledger technology, providing transparent and immutable data storage to record any transactions. Blockchain uses a peer-to-peer network with a consensus mechanism to allow a group of users to conduct transactions reliably. The recorded transactions are validated, agreed-up, and replicated by all the entities in the blockchain network.

Mathematically, a blockchain $BC$ can be formulated as a tuple:

$BC \equiv (PAT, CoB, P2P, SC, CO)$

where $PAT$ represents a set of blockchain participants, either in a permissioned or permissionless blockchain system, $CoB$ represents chain of blocks, $P2P$ represents blockchain peer-to-peer network, $SC$ represents smart contracts, and $CO$ represents consensus protocols used by the blockchain participants to validate transactions, and achieve a global truth view of the blockchain transactions and data.

We next describe each of the components with a focus on the chain of blocks which records the Blockchain transactions and data.

a: $PAT$: BLOCKCHAIN PARTICIPANTS

A participant $p \in PAT$ in a blockchain $BC$ represents a subject participating the blockchain activities, including submitting transactions, validating transactions, and recording the blockchain data. Each participant $p$ represents a node in the blockchain peer-to-peer network. The content of a blockchain is duplicated in every node of the system. A node is a full client if it has a full copy of the blockchain data starting from
the genesis block. The blockchain participants update their versions of the blockchain when a new block is validated and added to the system. Eventually, all the participants in the network have the same view of the transactions recorded in the blockchain.

b: CoB: CHAIN OF BLOCKS
The blockchain organizes data in ordered blocks. Each block contains a set of validated transactions. All the blocks are chained together (hence the name “blockchain”) to prevent any modification or deletion of blockchain data. The first block is called the genesis block. Each following block contains the hash value of the previous block’s information. Therefore, any change of block data will result in an invalidation of the following blocks. The basic structure of a block (e.g., the $n^{th}$ block) includes the following components: (1) a block size describing the number of bytes of data in the block; (2) a block header containing metadata about the block; (3) a transaction counter indicating the number of transactions in the block; and (4) a block body containing all the transactions included in the block. As shown in Figure 1 (a), the block header contains the following components: a version number to track software/protocol update, the SHA256 cryptographic hash of the previous block (i.e., the $n-1^{st}$ block header), the timestamp of the current block (the block header), the nonce and a difficulty level used for the proof-of-work algorithm, and the hash of the root of the Merkle tree of the current block. The block body contains all the transactions included in the block. As shown in Figure 1 (b), a block consists of a collection of transactions. A transaction is a block of self-executable code deployed on the rules defined in the software. Mathematically, a smart contract can be described as a deterministic finite-state machine.

c: P2P: PEER-TO-PEER NETWORK
Blockchain uses a decentralized network communication model for the participants to conduct transactions and update the blockchain data. Each participant acts as a peer to others, and there is no centralized server or administration. The operations of the blockchain rely on the underlying communication protocols and the built-in consensus mechanism. The block propagation and synchronization can be conducted in several ways, including advertisement-based propagation, send headers propagation, unsolicited push propagation, relay network propagation, and push/advertisement hybrid propagation.

d: SC: SMART CONTRACTS
A smart contract is a block of self-executable code deployed in a blockchain network, which automatically executes based on the rules defined in the software. Mathematically, a smart contract $SC$ can be described as a deterministic finite-state machine.
The state transition function $\delta$ in $SC_M$ is represented by a set of smart contract functions $FCT$ defined in $SC$. Each smart contract function $f \in FCT$ defines the actions that the smart contract $SC$ can take. $f$ may take zero or multiple input values and generate zero or a set of output values. The smart contract function $f$ may change the variables of $SC$, therefore changing the state of $SC_M$. A smart contract function $f$ can be called from an external blockchain address (e.g., called from a user blockchain account), by another function in the same $SC$, or by a function from a different smart contract. Similarly, $f$ can call an external program, called an oracle, to get the external or environmental information about a subject topic of interest (e.g., the real-time temperature, the current stock price).

The input alphabet $\Sigma$ in $SC_M$ specifies the valid values when calling the functions of $SC$. Those input values are taken by the smart contract functions. The smart contract variables of $SC$ may be updated based on the function logic given those input values. The updated values of the variables of $SC$ represent the final state $F$ of $SC_M$. Since a deterministic finite-state machine is used to represent a smart contract, there is only one final state $F$. The final state must be a valid state, i.e., $F \in S$. That means that the smart contract has transited from one valid state to another one consistently.

**e: CO: CONSENSUS PROTOCOL**
A consensus protocol used in a blockchain $BC$ is to achieve a global truth about the blockchain state among all the participants of $BC$. For any transaction to be valid, the majority of the participants must agree to it. There are various consensus protocols in the current implementations of blockchains, including proof of work, proof of stack, and practical Byzantine fault tolerance (to name a few). Validating and recording transactions is called mining. The entity which conducts mining is called a miner. A consensus protocol used in a blockchain system can determine the speed of the transactions added to the blockchain, e.g., the throughput of the blockchain. The following steps illustrate how a blockchain transaction is submitted, validated, and completed.

1. A participant issues a transaction to the blockchain network. The transaction is signed by the sender using its private key and addressed to the receiver’s blockchain address.
2. The transaction is validated.
3. The miners compete to create a block that contains a set of validated transactions.
4. The first successfully created block is broadcast to the blockchain network.
5. Every node in the network validates each transaction in the block.
6. If all the transactions are validated and a consensus is achieved, the block is appended to the Blockchain.
7. The miner who successfully created the block receives rewards.
8. The transaction is complete.
B. RELATED WORK

We discuss some related work to our research. The focus is on the IoT and Blockchain technologies and their applications in various fields, mainly in the oil and gas and closely related energy industry. In particular, we review that the combination of IoT and Blockchain could vastly improve the security, efficiency, and reliability of many applications.

1) IoT AND BLOCKCHAIN

He et al. (2018) [21] proposed a privacy-preserving IoT devices management scheme based on Blockchain. The scheme achieves the sharing of device information across organizations and systems using Blockchain to make the stored data temper-free and verifiable. The scheme also uses time-bound key management mechanisms to achieve automatic key revocation. The authors indicated that their scheme is practical and deployable for sharing over encrypted data in IoT. Christidis et al. (2016) [22] described how the combination of IoT and Blockchain could facilitate the sharing of services and resources into a marketplace between the IoT devices and how it allows the users to automate several existing, time-consuming workflows. The authors concluded that the blockchain and IoT combination could cause significant transformations across several industries. Similarly, Samaniego et al. (2016) [23] discussed using Blockchain as a service for IoT. They also presented ideas to evaluate the performance of a cloud and edge hosted blockchain implementation. Zhou et al. (2018) [24] proposed a blockchain-based IoT system called Beekeeper. The IoT devices send transaction information to the blockchain network for storage in this system. The user can then send a request to the Blockchain as needed to query certain results. The server reads the relevant data from the Blockchain to analyze it and sends the results back to the blockchain network. Finally, the user obtains the results from the blockchain network. The system has the characteristics of the Blockchain, such as decentralization and verifiability, to ensure that data is authentic and cannot be modified. The main drawback of this system is the limitation of the storage capacity of the Blockchain. The Blockchain can only be used to record transaction records and cannot store large amounts of data. Therefore, it is necessary to consider using the cloud or physical storage to store the data.

Although the integration of IoT and Blockchain provides various benefits, there are multiple challenges that must be overcome to make effective integration of these two technologies. IoT devices are typically small and resource-constrained. Consequently, blockchain protocols should be light-weighted to be applied to IoT devices. Bandara et al. (2021) [25] proposed a lightweight blockchain platform called Tikiri, using Apache Kafka for the consensus mechanism. The platform uses a new blockchain architecture to handle concurrent transactions. As a lightweight and scalable Blockchain, Tikiri is particularly suitable for resource-constrained IoT devices. Similarly, Biswas et al. (2019) [26] developed a lightweight proof of block and trade (PoBT) consensus algorithm for IoT blockchain. The goal is to reduce the computations to validate block and trade. Furthermore, a ledger distribution mechanism was proposed to decrease the memory requirements for IoT nodes. Analysis and evaluation showed significant improvement in terms of the performance of the system.

To address the challenges in applying Blockchain to IoT, such as security, comparability, energy consumption, and heterogeneity of devices, Yazdinejad et al. (2020) [27] proposed a secure and energy-efficient blockchain-enabled architecture of software-defined networking (SDN) controllers for IoT networks using a cluster structure with a new routing protocol. Both public and private blockchains can be used for peer-to-peer communications between the IoT devices and the SDN controllers. The computationally expensive Proof-of-Work consensus mechanism in the blockchain systems is avoided, making the Blockchain suitable for resource-constrained IoT devices. Huang et al. (2019) [8] proposed a blockchain system with a credit-based consensus mechanism for industrial IoT. A credit-based proof-of-work (PoW) mechanism was proposed for resource-constrained IoT devices. To protect data confidentiality, a data authority management method is used for securely accessing the IoT data. Since a directed acyclic graph structure is used, the performance is better than the Satoshi-style blockchains. The system was implemented on Raspberry Pi, which shows that system is secure and efficient in industrial IoT.

Lei et al. (2020) address the scalability issue of integrating Blockchain with fog computing. The authors proposed Groupchain, a novel scalable public blockchain of a two-chain structure suitable for fog computing of IoT services computing. Groupchain employs a leader group to collectively commit blocks for higher transaction efficiency and introduces bonuses and deposits into the incentive mechanism to supervise behaviors of members in the leader group. Groupchain was prototyped and the experimental results demonstrate performance efficiency in terms of transaction throughput. Similarly, Qiu et al. (2020) [29] tackled the scalability issue of blockchain-based IoT systems. The authors used a different approach with a permissioned blockchain for IoT architecture. More specifically, they proposed a service-oriented permissioned blockchain with different consensus protocols based on the users’ requirements for quality of service (QoS). Those include a few popular consensus protocols. Their proposal formulates consensus protocols selection, block producers selection, and network bandwidth allocation as a joint optimization problem.

Due to a lack of standards in IoT and Blockchain, it could be challenging to enable the interaction of IoT platforms with Blockchain. Mazzei et al. (2020) [30] presented an implementation of a portable, platform-agnostic and secure Blockchain Tokenizer for Industrial IoT trustless applications. An industrial data acquisition unit gathers data from the machines while also interfacing directly with the sensors. The acquired data is processed locally and then sent to a blockchain platform. Tests show that the system is
 capable of acting as a bridge between industrial assets and blockchain platforms, enabling the generation of immutable and trust-less “digital twins” for industrial IoT applications. Mhaisen et al. (2020) [31] described a smart contract-based IoT monitoring framework using public blockchains with a minimum monetary cost. A reinforcement learning agent is included that adapts to users’ needs and acts in real-time to set the data submission rate of IoT sensors. Results based on the Ethereum protocol show significant potential cost savings depending on users’ preferences.

2) APPLICATIONS OF IoT AND BLOCKCHAIN

IoT has been applied in various applications to improve remote sensing and data collection. The integration of Blockchain and IoT goes a step further and creates a trusted, transparent, decentralized environment for IoT operations. We summarized the related work in those fields.

Baudoin (2016) [32] introduced several IoT applications across different industries. IoT in the smart grid can dynamically adjust the supply and demand of the entire grid. In case of peak demands, the grid operators do not have to open expensive power generation facilities. In the manufacturing industry, the IoT can track the use of equipment to avoid misuse and perform predictive and just-in-time maintenance and enable remote operation of equipment. Lv et al. (2016) [33] proposed an IoT-based energy-saving system for a beam pumping unit (ESS-BPU) and have applied it in oil field operations in western China. The system can automatically adjust the working status of the pumping unit through the IoT sensors to improve its energy efficiency. The system includes four kinds of IoT sensors, including load sensors, angle sensors, voltage sensors, and current sensors. It acts on two aspects. The first is the automatic adjustment of the stroke speed of the pumping unit for oil wells with low production in the oil field. The other is the balance deviation rectification of the beam pumping unit. Baiji & Sundaravadivel (2019) [34] proposed an IoT-based oil leak detection system for smart cities. In the proposed system, sensors measure raw data and send it to the cloud for analysis. In the experiment, the authors used the water flow and volume in the pipeline to analyze whether there was a leak in the pipeline. These indicators will be severely affected by the pressure of the medium and the external environment in practical applications, resulting in inaccurate results. Wheelus & Zhu (2019) [35] discussed the security issues of the IoT. They first analyzed the threats and risks of the IoT based on the Cisco Seven-Layer IoT Reference Model and the Information Assurance and Security (IAS) Octave. Cyber-attacks can be classified according to the seven layers.

Lu (2019) [36] described how the Blockchain is applied to the oil and gas industry based on four aspects: trading, management and decision making, supervision, and cyber security. The authors discussed several benefits, including cost reductions and improved transparency and efficiency. The challenges of using this technology lie in technological, regulatory, and system transformation. The authors pointed out that the development of blockchains in the oil and gas industry will move towards hybrid blockchain architecture, multi-technology combination, cross-chain, hybrid consensus mechanisms, and more interdisciplinary professionals. Lakhanpal & Samuel (2018) [37] presented four areas where Blockchain can be applied to the oil and gas industry: supply chain and trading, regulatory, database management, and cybersecurity. While blockchain technology is expected to bring transparency, security, and convenience, its full potential has not been utilized. The authors provided a frame of reference for the oil and gas industry to realize the potential of blockchain technology and access its benefits.

Musleh et al. (2019) [38] discussed the application of a smart grid based on renewable energy in Blockchain. A blockchain-based smart grid can promote the local production and consumption of electricity. Electricity transactions can be carried out point-to-point between producers and consumers to reduce transmission losses. Also, the decentralization of the Blockchain can avoid the high infrastructure costs of centralized power management. The authors discussed the limitations and shortcomings of the Blockchain, including slower transactions, inability to meet the computing power and response speed of high-frequency transactions and scheduling, and high block transaction fees. Aitzhan & Svetinovic (2018) [39] proposed a proof-of-concept system for providing transaction security in decentralized smart grid energy trading without reliance on trusted third parties. The proposed system uses blockchain technology, multi-signatures, and anonymous encrypted messaging streams, enabling peers to anonymously negotiate energy prices and securely perform trading transactions. The authors conducted case studies for security analysis and performance evaluation. An important component of a smart grid is smart metering. One of the key information technology solutions applied to smart meters is ensuring data protection and cybersecurity. Ensuring data integrity is essential for the utilities and end-users. Blockchains can provide a means for cybersecurity and user trust. Olivares-Rojas et al. (2019) [40] proposed a blockchain-based architecture for smart metering systems to identify and enhance data security. The proposed multi-tiers architecture improves performance based on a scheme of edge, fog, and cloud computing. Furthermore, an optimized consensus algorithm was proposed for smart meters. The results show that the proposed architecture can effectively improve cybersecurity and data protection for smart metering applications.

Smart grids can manage the load, regulate frequency, and maintain grid stability. To provide a lightweight solution in view of the heavy cryptographic primitives’ execution on the smart devices, Jindal et al. (2021) [41] proposed a blockchain-based secure demand response management scheme called GUARDIAN. The proposal is to take energy trading decisions securely for managing the overall load of residential, commercial, and industrial sectors. In the GUARDIAN scheme, the chosen miner nodes authenticate the energy transactions initialized by end-users in a
smart grid. A user creates the block of the transaction to trade the energy. The miner nodes validate these blocks and add these in the Blockchain in order for the energy trading to be considered successful. Said (2021) [42] proposed a decentralized electricity trading framework between connected electric vehicles (CEVs) based on consortium blockchain, machine learning, and Game-theoretic model. A distributed smart contract solution was proposed to help CEVs to sell and buy electricity with their maximum profitability. The author simulated the model with MATLAB and Solidity to prove its effectiveness in the electricity markets. Based on the contract theory, incentive mechanism, and reputation system, a blockchain-based Energy Trading model was proposed in [43]. The model provides means to motivate the energy trading entities to trade energy locally and electric vehicles to participate in smart energy management. A consensus algorithm called Proof of Work based on Reputation is proposed to reduce transaction confirmation latency and block creation time. The performance of the proposed model is evaluated using peak to average ratio, social welfare, the utility of local aggregator, etc., as performance metrics.

Based on the currently reviewed literature, we believe that the applications of the Blockchain and IoT technology can provide numerous benefits for various applications, including oil and gas operations. In particular, the use of Blockchain on IoT can not only improve data security and transaction trust, but the deployment of smart contracts can also help the IoT system automatically execute commands to achieve remote monitoring and control. Given the typical challenging environment where the oil and gas operations are conducted, automatic and reliable monitoring and control is critical to minimize any unexpected disruptions, increase reliability and safety, and improve productivity. We present a blockchain-based IoT framework for oil field monitoring and control.

III. THE FRAMEWORK

In this section, we discuss our proposed framework. We first present the overall system architecture (Section III-A). Then we describe the sequence of actions for oil field monitoring and control (Section III-B). Next, we discuss the main components of the framework (Section III-C). The focus is the IoT subsystem and the blockchain subsystem, as they are the core parts of the proposed framework. We report our programming and verification of the smart contract as the main control unit of the framework using the Remix platform (Section III-D).

A. HIGH-LEVEL SYSTEM ARCHITECTURE

The major components of the framework include the IoT subsystem (IoT smart devices and gateways), the blockchain subsystem (the blockchain network, nodes and smart contracts), the cloud servers, and the enterprise system. Any data about the oil field operations, such as the working conditions of the equipment and the operating parameters of the oil well can be collected. The data provides the real-time input for the blockchain smart contracts. Based on the programmed monitoring and control rules, the smart contracts may determine the most appropriate actions to take. The system architecture of the framework is illustrated in Figure 2.

![FIGURE 2. The blockchain-based IoT framework architecture.](image)

There are several options to integrate IoT into Blockchain [44]. The most common one is to connect the IoT devices directly to Blockchain, where IoT devices act as blockchain nodes. Based on the unique characteristics of the oil field operations, we propose the IoT to the gateway to blockchain configuration, which is most suitable for a medium or large scope of oil field operations where IoT devices have relatively weak computational capacities, and the IoT gateways are more capable of pre-processing IoT data and participating in the blockchain network as full nodes. Furthermore, the smart devices used in oil operations are usually deployed in remote oil fields subject to various environmental challenges. They may have limited capacities to directly connect the blockchain network.

B. PROCEDURE FOR OPERATIONS

As shown in the sequence diagram in Figure 3, the Blockchain provides a reliable decentralized platform for data exchange between the IoT gateways, the blockchain smart contracts, the cloud servers, and the enterprise system. The sequence of actions for the oil field IoT data collection, smart contract data processing, cloud server data storage and enterprise system control are described in the following steps.

Step 1. IoT sensors $A, B, \ldots, N$ periodicallysend measurement data about the oil field operations to their designated IoT gateways.

Step 2. The IoT gateway pre-processes the data, including data grouping, formatting, timestamping, and even simple trend analysis. Then the gateway packs the relevant data in an appropriate format in a blockchain transaction.

Step 3. The IoT gateway sends the IoT data to the cloud servers for storage (Step 3.1). Since a blockchain has limited storage, it is practical to store a potentially large amount of IoT data off-chain. We propose to use a cloud server to store the IoT data itself. The blockchain stores the immutable identifiers of the data (e.g., the hash values of the data) in-chain. Since data in Blockchain cannot be modified, data integrity in
the cloud servers is guaranteed. Any modification of the data would be detected as their identifiers/hashes do not match those stored in the Blockchain.

In the meantime, the IoT gateway sends the blockchain transactions (created in step 2) to the blockchain network (Step 3.2).

**Step 4.** The blockchain participants perform transaction validation and block mining (Step 4). If a transaction is addressed to a smart contract, the smart contract is executed, taking the IoT data as input, processing the data following the programmed rules, and outputting the corresponding actionable commands.

**Step 5.** The action commands generated in the previous step are sent to the IoT gateway for the corresponding IoT actuators to carry out. A command is packed in a blockchain transaction addressed to that IoT gateway (Step 5.1). Any other alert/warning messages may be sent to the enterprise control system for further action or analysis (5.2).

**Step 6.** The IoT gateway retrieves the command action from the received transaction and repacks it into a format that the IoT actuator can interpret and implement.

**Step 7.** The IoT gateway sends the command to the corresponding IoT actuator for implementation.

**Step 8.** The enterprise control system periodically retrieves IoT data from the cloud servers (Step 8.1) and also from the blockchain subsystem (Step 8.2) for advanced data analysis to identify any potential issues.

**Step 9.** Based on the advanced analysis of the IoT data in step 8, some actionable commands for oil field equipment maintenance and repair are sent to the corresponding IoT gateway, which forwards the commands to the corresponding actuators to carry out the actions.

It should be noted that the process shown in the dotted line in Figure 3 represents periodic actions or system management work. For instance, steps 8 and 9 show that the enterprise system provides access control and management work to keep the blockchain secure and healthy.

**C. MAIN COMPONENTS OF THE FRAMEWORK**

In this section, we discuss the main components of the framework and their specifications to perform the functions as specified in the sequence diagram in Figure 3. We follow a bottom-up approach starting from the IoT subsystems at the lowest level to the enterprise systems at the highest level.

1) **THE IoT SUBSYSTEM**

The IoT subsystem is composed of the IoT sensors and actuators attached to the oil field equipment and well monitoring and control units. While the IoT sensors and actuators have limited storage and computational capacities, an IoT gateway is a more powerful device. The gateway can connect with the IoT’s smart devices, the blockchain network, and the enterprise control system.

The IoT gateways can be organized in a hierarchy structure, which covers different oil fields with different managing...
authorities. For the sake of simplicity, we assume a flat structure of IoT gateway, i.e., all the IoT gateways directly connect to the blockchain network and the enterprise system. A common structure of the IoT subsystem is a star topology, as shown in Figure 4. In this configuration, the IoT devices connect to their designated IoT gateway and the IoT gateway serves as the central point to link all the IoT devices. There is no direct communication link between any IoT devices.

Secure communications between the IoT devices and the gateway must be ensured. An IoT device needs to authenticate itself to the gateway and the IoT gateway must also authenticate itself to the IoT device. We design an enrollment protocol and a secure communication protocol for IoT devices and an IoT gateway to authenticate each other and consequently securely communicate.

The following describes those two protocols executed between an IoT smart device \( S \) and an IoT gateway \( G \). Both \( S \) and \( G \) have their unique identifiers and each has a pair of asymmetric cryptographic keys. The public key of \( S \) or \( G \) is available for any other entities, through either an online channel (e.g., PKI) or an offline approach (e.g., manually delivery). The symbols used in the two protocols are summarized in Table 1.

The enrollment protocol is for a smart device \( S \) to register with the IoT gateway \( G \). A registration means that the smart device is an authorized entity, i.e., it is authorized to send IoT data and/or receive commands from the blockchain smart contracts. The enrollment protocol is described in Figure 5. As we can see, there are five steps. First, \( S \) delivers its ID to \( G \), indicating its intention to join the authorization zone of \( G \). Using a challenge-response approach, \( G \) generates a random value \( n \) and sends it to \( S \). In the third step, \( S \) encrypts \( n + 1 \) using its own private key. The encrypted message \( E(n+1)_{PRIS} \) is the response to the challenge. In the fourth step, \( G \) verifies that the response is correct as only the legitimate smart device can correctly generate the response. If verified, \( G \) generates a session key \( sk \). In the final step, \( G \) sends the encrypted \( sk \) using \( S \)'s public key to \( S \). Since only \( S \) can correctly decrypt the message, \( sk \) can be securely delivered to \( S \). Then \( sk \) can be used as a cryptographic secret to encrypt and decrypt messages in the following secure communications between \( G \) and \( S \) as shown in the secure communication protocol.

After a smart device \( S \) enrolls with a gateway \( G \), the secure communications between \( S \) and \( G \) can be described using the secure communication protocol as shown in Figure 6. In the first step, \( S \) periodically collects a set of sensor data with the corresponding timestamp. In the second step, \( S \) sends a message with its identity and the sensor data along with its

### Table 1. Symbols of the IoT smart device \( S \) and IoT gateway \( G \) enrollment and communication protocols.

| Symbol  | Representation                                      |
|---------|-----------------------------------------------------|
| PUK_{S} | The public key of \( S \)                           |
| PRK_{S} | The private key of \( S \)                          |
| ID(S)   | The identifier of \( S \)                          |
| PUK_{G} | The public key of \( G \)                          |
| PRK_{G} | The private key of \( G \)                          |
| n       | A random nonce value                                |
| h(.)    | A cryptographic hashing function, e.g., SHA-1, SHA-256 |
| d_{i}   | A data item \( i \) sent by \( S \)                 |
| t_{i}   | The timestamp of \( d_{i} \)                        |
| E(M)_{k} | The encrypted message \( M \) using key \( k \)     |
| D(M')_{k'} | The decrypted message \( M' \) using key \( k' \)   |
| sk      | A session key between \( S \) and \( G \)           |
| A->B: M  | Entity A sends a message \( M \) to entity B        |
| M || M'  | The concatenation of messages \( M \) and \( M' \)     |
| V ← v    | Value \( v \) is assigned to variable \( V \)       |
| V_{1} = V_{2} | Variable \( V_{1} \) has the same value as variable \( V_{2} \) |

### Figure 4. The star topology of the IoT subsystem.

### Figure 5. IoT smart device enrollment protocol.

1. \( S \) sends its ID to \( G \).
2. \( G \) sends a random value \( n \) to \( S \).
3. \( S \) encrypts \( n+1 \) using its own private key and sends the encrypted message \( E(n+1)_{PRIS} \) to \( G \).
4. \( G \) verifies if \( (E(n+1)_{PRIS} = n+1) \) and \( (S \) is a legitimate sensor in the gateway zone of \( G \). If both (1) and (2) are true, \( S \) is authenticated. Then \( G \) generates a session key \( sk \).
5. \( G \) sends the encrypted session key to \( S \).
6. \( S \) can use the session key to encrypt and decrypt messages in the following secure communications.

### Figure 6. IoT smart device and gateway secure communication protocol.

1. \( S \) organizes a set of \( m \) IoT data items \( D = \{ (d_{1}, t_{1}), (d_{2}, t_{2}), \ldots, (d_{m}, t_{m}) \} \) to send to \( G \).
2. \( S \) encrypts \( D \) using the encryption key \( (E((H(D))_{PRIS}))_{PRIS} \) generated by \( S \).
3. \( G \) verifies if the data set \( D \) has not been tampered with and is sent by \( S \) by performing the following: \( D = (E((H(D))_{PRIS}))_{PRIS} \neq (E((H(D))_{PRIS}))_{PRIS} \).
4. \( G \) can use the session key to securely deliver the message to \( S \).

After a smart device \( S \) enrolls with a gateway \( G \), the secure communications between \( S \) and \( G \) can be described using the secure communication protocol as shown in Figure 6. In the first step, \( S \) periodically collects a set of sensor data with the corresponding timestamp. In the second step, \( S \) sends a message with its identity and the sensor data along with its
digital signature encrypted using the secret key $sk$ to $G$. Then, in the third step, $G$ verifies the data has not been modified. $G$ first decrypts the message using $sk$. Then, it verifies the digital signature associated with the data is correct. $G$ can do so as it knows $S$’s identity (and thus its public key). If verified, $G$ knows that message has not been modified and the data is indeed sent by $S$.

2) THE BLOCKCHAIN SUBSYSTEM
The blockchain network is a distributed peer-to-peer network. The participants of the blockchain subsystem in our framework include the IoT gateways and the enterprise control system. Each of those participants acts as a full node, which conducts block mining and transaction validation. Each participant is associated with a pair of public/private keys. Each has a digital wallet to manage the cryptocurrency and issue/receive transactions. A participant also has a unique blockchain address and maintains a copy of the blockchain data.

We use Ethereum as Blockchain in our proposal. An operating system called Ethereum Virtual Machine (EVM) is used to execute smart contracts in a fully decentralized fashion. As compared with the Bitcoin blockchain, Ethereum has a shorter block size. Its validation time is about 14 seconds per block as compared with 10 minutes in Bitcoin. Ethereum uses The Ethereum Greedy Heaviest Observed Subtree (GHOST) protocol for consensus and rewards to miners [45]. The Proof of Work (PoW) protocol is used for the nodes to achieve a consensus in Ethereum. Since Ethereum can be used as a private blockchain, the participating nodes are chosen and the PoW protocol is no longer required.

In our proposal, a blockchain transaction includes a set of IoT oil field operations measurement data. Like an external blockchain account, a smart contract has a unique blockchain address. A transaction can call the smart contract with a “message” field as its data value. A smart contract accepts inputs from one or more transactions. Based on the internal programmed rules, an output is produced in the form of a blockchain transaction. The transaction is addressed to a particular IoT gateway or a smart device to carry out the actions.

The algorithm for the smart contract representing the remote monitoring and control of an oil field equipment/well $EQ$ is shown in Figure 7. The input of the algorithm is a set of IoT measurement values ($ME$) for the oil field equipment/well $EQ$. $ME$ is included in a blockchain transaction submitted by an IoT gateway $G$ to the smart contract. The transaction involves the execution of the smart contract function $IoT_MC$. Variable $EQ_Rules$ represents a set of $m$ ($m \geq 1$) monitoring and control rules defined for $EQ$. Each rule $r$ has a format $(C_1, C_2, \ldots, C_k) \Rightarrow T$. We use $r$.Conditions to represent the
We describe our simulation and verification of a smart contract in an Ethereum testing environment. The smart contract is programmed with a set of oil field control and monitoring rules, which determine the corrective or preventative action for the IoT control units to take given a set of measurement values.

A. USE CASE SCENARIOS
We use the case studies presented in [50] to illustrate the application of our proposed framework to oil field monitoring and control. Three use case scenarios were discussed in [50] for oil field operation safety, reliability, revenue increase and cost reduction. We briefly summarize the use case scenarios below.

1) USE CASE SCENARIO ONE - REDUCED COSTS
The case is about monitoring a dual completion oil well. Well parameters such as Flow Tubing Pressure (FTP), Casing Pressure (CP) and Flowline Pressure (FLP) data were collected, logged and trended every 6 hours. At a certain time in 2019, the control center received a low FTP monitoring value at 250 psig. Furthermore, there were a reduction in FLP and an increased CP value. Operators suspected the downhole safety valve must have closed and later confirmed that was the case. The issue was rectified and the well was opened to production. Given this early detection of the issue, a timely response led to avoiding a potentially significant loss.

2) USE CASE SCENARIO TWO - RELIABILITY
This case is about a single completion, horizontal, high producer well. Once again, well parameters such as FTP, CP and FLP data were collected, logged and trended. At a certain time in 2019, the FTP of the well has been detected to decrease at a drastic rate. At a later time, the FTP of the well was noticed to have dropped further and the well equalized on FTP/FLP = 112/112psi. One solution was to do a rigless gaslift optimization on the well. Since a barge was already strategically in the area, it was mobilized the next day to the well-site and a rigless gaslift optimization was done on the well. Due to a timely response, the well operation was brought back online quickly.

3) USE CASE SCENARIO THREE - INCREASED REVENUE
This case is about an offshore producer well. FTP, CP and FLP data about the well were collected, logged and trended. At a certain time in 2013, the gas-lift system started showing fluctuations in casing pressure. It was discovered that hydrate formation downstream of the adjustable gas-lift choke supplying gas to the well equally caused the issue. The choke valve was stroked to dislodge the hydrates. The well test results before the identification of the hydrates and after gas lift optimization were evaluated with a gain of about 190BOPD seen and sustained over a 3-month period.

### III. SIMULATION AND VERIFICATION OF THE SMART CONTRACT

The core component of the oil field monitoring and control system is the blockchain smart contracts. In this section, we discuss the simulation and verification of a smart contract in an Ethereum testing environment. The smart contract is programmed with a set of oil field control and monitoring rules, which determine the corrective or preventative action for the IoT control units to take given a set of measurement values.

#### A. USE CASE SCENARIOS

We use the case studies presented in [50] to illustrate the application of our proposed framework to oil field monitoring and control. Three use case scenarios were discussed in [50] for oil field operation safety, reliability, revenue increase and cost reduction. We briefly summarize the use case scenarios below.

1) USE CASE SCENARIO ONE - REDUCED COSTS

The case is about monitoring a dual completion oil well. Well parameters such as Flow Tubing Pressure (FTP), Casing Pressure (CP) and Flowline Pressure (FLP) data were collected, logged and trended every 6 hours. At a certain time in 2019, the control center received a low FTP monitoring value at 250 psig. Furthermore, there were a reduction in FLP and an increased CP value. Operators suspected the downhole safety valve must have closed and later confirmed that was the case. The issue was rectified and the well was opened to production. Given this early detection of the issue, a timely response led to avoiding a potentially significant loss.

2) USE CASE SCENARIO TWO - RELIABILITY

This case is about a single completion, horizontal, high producer well. Once again, well parameters such as FTP, CP and FLP data were collected, logged and trended. At a certain time in 2019, the FTP of the well has been detected to decrease at a drastic rate. At a later time, the FTP of the well was noticed to have dropped further and the well equalized on FTP/FLP = 112/112psi. One solution was to do a rigless gaslift optimization on the well. Since a barge was already strategically in the area, it was mobilized the next day to the well-site and a rigless gaslift optimization was done on the well. Due to a timely response, the well operation was brought back online quickly.

3) USE CASE SCENARIO THREE - INCREASED REVENUE

This case is about an offshore producer well. FTP, CP and FLP data about the well were collected, logged and trended. At a certain time in 2013, the gas-lift system started showing fluctuations in casing pressure. It was discovered that hydrate formation downstream of the adjustable gas-lift choke supplying gas to the well caused the issue. The choke valve was stroked to dislodge the hydrates. The well test results before the identification of the hydrates and after gas lift optimization were evaluated with a gain of about 190BOPD seen and sustained over a 3-month period.

### III. SIMULATION AND VERIFICATION OF THE SMART CONTRACT

The core component of the oil field monitoring and control system is the blockchain smart contracts. In this section, we discuss our simulation and verification of a smart contract in an Ethereum testing environment. The smart contract is programmed with a set of oil field control and monitoring rules, which determine the corrective or preventative action for the IoT control units to take given a set of measurement values.

#### A. USE CASE SCENARIOS

We use the case studies presented in [50] to illustrate the application of our proposed framework to oil field monitoring and control. Three use case scenarios were discussed in [50] for oil field operation safety, reliability, revenue increase and cost reduction. We briefly summarize the use case scenarios below.

1) USE CASE SCENARIO ONE - REDUCED COSTS

The case is about monitoring a dual completion oil well. Well parameters such as Flow Tubing Pressure (FTP), Casing Pressure (CP) and Flowline Pressure (FLP) data were collected, logged and trended every 6 hours. At a certain time in 2019, the control center received a low FTP monitoring value at 250 psig. Furthermore, there were a reduction in FLP and an increased CP value. Operators suspected the downhole safety valve must have closed and later confirmed that was the case. The issue was rectified and the well was opened to production. Given this early detection of the issue, a timely response led to avoiding a potentially significant loss.

2) USE CASE SCENARIO TWO - RELIABILITY

This case is about a single completion, horizontal, high producer well. Once again, well parameters such as FTP, CP and FLP data were collected, logged and trended. At a certain time in 2019, the FTP of the well has been detected to decrease at a drastic rate. At a later time, the FTP of the well was noticed to have dropped further and the well equalized on FTP/FLP = 112/112psi. One solution was to do a rigless gaslift optimization on the well. Since a barge was already strategically in the area, it was mobilized the next day to the well-site and a rigless gaslift optimization was done on the well. Due to a timely response, the well operation was brought back online quickly.

3) USE CASE SCENARIO THREE - INCREASED REVENUE

This case is about an offshore producer well. FTP, CP and FLP data about the well were collected, logged and trended. At a certain time in 2013, the gas-lift system started showing fluctuations in casing pressure. It was discovered that hydrate formation downstream of the adjustable gas-lift choke supplying gas to the well caused the issue. The choke valve was stroked to dislodge the hydrates. The well test results before the identification of the hydrates and after gas lift optimization were evaluated with a gain of about 190BOPD seen and sustained over a 3-month period.
B. Testing and Verification

We developed the smart contract program using the Solidity language. Solidity is statically typed and supports inheritance, libraries and complex user-friendly types. It is designed for developing smart contracts that run on Ethereum. We programmed a smart contract based on the use case scenarios and tested it on the Remix IDE, which is an open-source web and desktop application. Remix provides a fast development environment of smart contracts with a set of plugins with intuitive GUIs. Since each transaction has to pay a fee proportional to its complexity to compensate the miners for the effort of maintaining the Ethereum Virtual Machine, Remix IDE provides multiple Ethereum accounts with a certain amount of Ether (default 100). We chose to use the JavaScript VM environment for our testing. A transaction on Ethereum is charged for the computation costs that it is involved. gas is the unit of measurement for the computational effort it takes to execute a transaction. gas is a very small unit. Each Ethereum transaction requires at least 21,000 gas [51].

The enterprise control system deploys the smart contract through a blockchain transaction. An IoT gateway collects data about the IoT measurement values of a particular equipment/well and submits the data through a blockchain transaction to the smart contract. The smart contract processes the data and determines if any action needs to be taken given the current oil field operating situation. The smart contract loads the monitoring and control rules through its constructor function when it is first deployed. For illustration purposes and for reducing the transaction costs on Ethereum, we used a simple numerical encoding scheme (as seen below) to represent those rules. We also used events that allow logging to the Ethereum blockchain.

1) Simulation Preparation

The following Ethereum addresses, each of which is a 20-byte value, are assigned to the owner (the enterprise control system) and an IoT gateway (as a blockchain participant). After a smart contract is deployed, it also assigns a blockchain address so that other entities can call a function of the smart contract by sending a transaction to that address.

- Owner (enterprise control system) address: 0x5.38Da6a701c568545dCfeB03FcB875f56beddC4
- IoT gateway blockchain address: 0xAb8483F64d9C6d1EcF9b849Ae677d3315835cb2

2) Testing Cases

Based on the three use case scenarios, we developed 9 testing cases. Each use case scenario has its own set of rules. We tested three cases for each use scenario: (1) an action needs to take given the IoT measurement values; (2) no immediate action is required, but a warning is generated due to certain operating concerns; and (3) normal operation and no action needs to take. Those 9 testing cases are summarized in Table 2. For each rule set, we specified the number of rules (num_rules) and the number of measurement parameters (num_para) for the type of oil field operation to be monitored. An encoding scheme is used to transform the rule sets into a series of numerical sequences to be input into the smart contract. The output actions include “Open downhole safety valve,” “Start rigless gaslift optimization,” and “Open choke valve.” Our simulation generates the action command that the smart contract outputs for the IoT actuator to carry out. This will result in automatic control of remote oil field operations with a minimum of human intervention. Since blockchain transactions are used, a permanent, immutable trace of actions is maintained to provide a complete, true picture of the oil field operations. As an example, Case 1 of the Use Scenario One indicates that if the Flow Tubing Pressure (FTP) level in a monitored well is 180 psig, the Flowline Pressure (FP) level is decreasing (as encoded by the FP_Trend value 4321), and the Casing Pressure (CP) level is increasing (as encoded by the CP_Trend value 1234), then the smart contract produces a command message to open the downhole safety valve. As another example, for the same use case scenario, Case 2 indicates that if the Flow Tubing Pressure (FTP) level in a monitored well is 220 psig, but the FP and CP do not show decreasing or increasing pattern (as encoded by the symbol 0), then the smart contract produces a warning message “FTP low” as this FTP level is below the threshold 250 psig.

Snapshots of the smart contract simulation with the rule set coding [2, 250, 3, 4321, 3, 1234, 1001, 2, 250, 3, 0, 3, 0, 1004] and measurement values $m_1 = 180$, $m_2 = 4321$ and $m_3 = 1234$ are shown in Figure 8. As we can see, the smart contract outputs the target action “Open downhole safety valve,” which can be verified based on the Use Case Scenario One. To further illustrate the operations of the smart contract,
TABLE 2. IoT smart contract testing cases.

| Monitoring and Control Rule Sets | Encoded Rules | Measurement Parameters | Encoded Measurement Parameters | Smart Contract Decision |
|----------------------------------|---------------|------------------------|--------------------------------|-------------------------|
| Use Scenario One: Reduced Costs | [2, 250, 3, 4321, 3, 1234, 1001, 2, 250, 3, 0, 3, 0, 1004] | $m_1 = 180$
$ m_2 = 4321$
$ m_3 = 1234$ | [180, 4321, 1234] | “Open downhole safety valve” |
| Rules:                          |               | (num_rules = 2; num_para = 3) |                              |                         |
| IF FTP < 250 AND FLP_Trend = 4321 AND CP_Trend = 1234 THEN “Open downhole safety valve” |               |                                  |                              |                         |
| IF FTP < 250 AND FLP_Trend = 4321 AND CP_Trend = 0 THEN SCSSV = “Warning: FTP low” |               |                                  |                              |                         |
| Symbols:                        |               | SCSSV: downhole safety valve | FTP: Flow Tubing Pressure | FLP: Flowline Pressure CP: Casing Pressure FLP_Trend: FLP value trend CP_Trend: CP value trend 4321: code for decreasing value 1234: code for increasing value 0: code for any value |
| Use Scenario Two: Reliability   | [3, 4321, 2, 112, 1002, 3, 4321, 1, 112, 1005] | $m_1 = 4321$
$ m_2 = 108$ | [4321, 108] | “Start rigless gaslift optimization” |
| Rules:                          |               | (num_rules = 2; num_para = 2) |                              |                         |
| IF FTP_Trend = 4321 AND FTP <= 112 THEN “Start rigless gaslift optimization” |               |                                  |                              |                         |
| IF FTP_Trend = 4321 AND FTP >112 THEN “Warning: FTP decreasing” |               |                                  |                              |                         |
| Symbols:                        |               | FTP_Trend: FTP value trend | 4321: code for decreasing value | RGO: Rigless Gaslift Optimization |
| Use Scenario Three: Increased revenue | [3, 12321, 3, 4321, 1003, 3, 12321, 3, 0, 1006] | $m_1 = 12321$
$ m_2 = 4321$ | [12321, 4321] | “Open choke valve” |
| Rules:                          |               | (num_rules = 2; num_para = 2) |                              |                         |
| IF CP_Trend = 12321 AND GLC = 4321 THEN “Open choke valve” |               |                                  |                              |                         |
| IF CP_Trend = 12321 and GLC = 0 THEN “Warning: Casing pressure is fluctuating” |               |                                  |                              |                         |
| Symbols:                        |               | GLC: Hydrate formation downstream of gas-lift choke 4321: code for decreasing value 12321: code for fluctuating value 0: code for any value |

we also provide Figures 9 and 10, which show the execution logs of two blockchain transactions. The first one is initialized by the enterprise control system to deploy the smart contract and the second is initialized by the IoT gateway to call the smart contract function $IoT_{MC}(..)$ with measurement value $[180, 4321, 1234]$. Both figures show that the transactions have been successfully executed to deploy a smart contract and to execute the smart contract with a set of oil field measurement values.

V. DISCUSSIONS
A. ADVANTAGES OF USING BLOCKCHAIN AND SMART CONTRACTS
The SCADA and IoT have been used in many industrial fields, including the oil and gas industry. They have brought considerable benefits to the industry. However, most of the SCADA and IoT adopt a centralized server model. As the number of data increases, the scale and cost of the server will increase, and the failure of the central server will also...
cause loss to the entire system. Considering these issues, the deployment of the blockchain will become a great solution.

The proposed framework aims to provide automatic and trustworthy monitoring and control of oil field operations. Followings are the main advantages that the industry can get from the architecture:

- **Real-time data processing and automatic control** As an extended application of the blockchain, a smart contract can automatically execute the oil field control rules by checking the IoT measurement values and making the corresponding decisions. The compatibility between blockchain and smart contracts enables the system to take control of actions in real-time and realize timely monitoring of oil field equipment and operations, including hydraulic valve monitoring, pressure level control, and well equipment issues.

- **Elimination of central server and administration** The use of blockchain and smart contracts removes the need for any central server or administration in order to control the oil field operations and to conduct remote monitoring. A blockchain system is fully decentralized. The point-to-point transactions without any centralized server would improve efficiency and avoid single point of failure of any central server in the traditional IoT and SCADA systems.

- **Decentralization** Blockchain uses a fully decentralized peer-to-peer network to communicate, store data, and perform computation. It is more robust and fault-tolerant as compared with the traditional remote monitoring and control systems. The disruption of any single node does not affect the operation of the entire blockchain system. Since Blockchain data is fully distributed in the peer-to-peer network, the operations of the system are resistant to functional failures and malicious attacks.

- **Data and transaction transparency and traceability** Blockchain provides transparent transactions. Every oil field monitoring and control transaction is logged, stored, and bound to the individual entity which submitted the transaction. The logged transactions and events can be used to reconstruct the oil field operations for further investigation and analysis in case of incidents or performance evaluation.

Those blockchain data provides a trustworthy source for the oil field operators and managers to identify the problems and trace back to the causes and sources of problems.

- **Trustless platform** Blockchain provides a trust mechanism to allow the transactions to be conducted between the parties which may not fully trust each other. The records of transactions are duplicated across the nodes in the entire network and updating the blockchain data is conducted via consensus, thus ensuring the trustworthiness of the transactions.

- **Cost reduction** Since Blockchain eliminates the need for any third parties or middlemen, the operational costs of those infrastructures can be saved. Furthermore, the smart contract provides automatic data processing and decision-making based on the pre-defined rules. This eliminates much labor-intensive manual work, which further reduces the costs of operations. As compared with the traditional oil field monitoring and control systems, a blockchain-based system can significantly reduce the costs, which is vital for the capital-intensive oil and gas industry.

A comparative analysis summary of our proposed framework and the traditional remote monitoring and control systems is given in Table 3.

### B. SECURITY ANALYSIS OF THE PROPOSED FRAMEWORK

The main security requirements of our proposed framework can be analyzed from three aspects: entity authentication and authorization, message integrity, and nonrepudiation. IoT data integrity is much a concern as any modified data or injected false data could severely disrupt the oil field operations and cause significant losses to the oil well productions. As discussed in Section III-C-1 and presented in Figure 5, each IoT smart device must authenticate itself to the IoT gateway before it can send any data to the blockchain smart contract. In the meantime, the IoT gateway must also authenticate itself to the IoT device to make sure each smart

| Table 3. Comparative summary of our proposed framework with traditional remote monitoring and control systems for oil field operations. |
|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
|                                | SCADA-based remote monitoring and control systems | Traditional IoT based monitoring and control systems | Our proposed blockchain-based IoT framework |
| Elimination of central server and administration | ✗ | ✗ | ✓ |
| Decentralization | ✗ | ✗ | ✓ |
| Real-time data processing and automatic control | Somehow less | Somehow less | ✓ |
| Data and transaction transparency and traceability | Somehow less | Somehow less | ✓ |
| Operational costs | High | High | Low |
| Trustless platform | ✗ | ✗ | ✓ |

**Note:** ✓: yes

🔧: no or not necessarily

FIGURE 10. Execution log of smart contract function call.
device communicates with the right IoT gateway. This mutual authentication makes sure that any unauthorized entity cannot simply connect to an IoT gateway and send bogus messages to the blockchain system. Only the legitimate IoT gateways and the enterprise control system can participate the blockchain activities. This would eliminate any unauthorized users to submit random or fake transactions to the Blockchain. Furthermore, a smart contract can only be deployed by the enterprise control system. Once deployed, the smart contract can no longer be modified or deleted. This would eliminate any human intervention and potential human errors in data processing as the smart contracts are self-executed based on a set of rules. No unauthorized entity can deploy a smart contract with its own set of control rules.

The secure communication protocol, as presented in Section III-C-1 (Figure 6) ensures that data is securely transmitted between an IoT gateway and an IoT smart device. No message can be modified as otherwise the digital signature of the message cannot be verified. No false message can be injected into the IoT network as neither the sender nor the digital signature of the message can be verified. Once the IoT data is submitted to the Blockchain, it cannot be modified. This is determined by the fundamental characteristics of a blockchain system. Smart contracts are immutable and can’t be changed without detection. All those can ensure the IoT data integrity in both the IoT subsystem and the blockchain subsystem.

Similarly, the proposed IoT smart device and gateway enrollment and secure communication protocols and the characteristics of a blockchain system can guarantee message nonrepudiation. Any entity cannot deny that a message was ever sent if he indeed sent the message. The Blockchain maintains an immutable log of all the transactions and each message is associated with a unique digital signature. The originators of the operations are logged. The log cannot be changed or deleted as it is duplicated in every node of the blockchain system. All those features guarantee message nonrepudiation.

C. LIMITATIONS

Although our proposed approach can fulfill the responsibility of automatically monitoring and control of oil field operations, the performance and operating costs of a blockchain system must be considered. Blockchain mining and transaction validation introduce delays in IoT data processing and message communications. While the Bitcoin blockchain generates a block every 10 minutes, the Ethereum blockchain can generate a block in a faster speed, for example, in every few seconds. However, this transaction speed is still not fast enough for many real-world industrial applications. If a private or hybrid blockchain is used, the transaction speed can be further improved, which can avoid the costly Proof-of-Work consensus process. In general, blockchain performance represents a potential bottleneck of the entire system.

To minimize the performance delay, we propose to use IoT gateways to pre-process the IoT measurement data generated by the IoT smart devices. Some data can be aggregated, summarized, or combined and the data trend can be analyzed locally. This would also reduce the amount of data to submit to the blockchain network and decrease the frequency of calling a smart contract. In case of an oil field emergency detected based on the IoT data, immediate actions must be taken by the IoT gateways without necessarily going through the blockchain network and waiting for a response from the smart contract. This requires a pre-processing data capacity of the IoT gateways. From a performance perspective, our framework is most suitable for the situation where the reliability, traceability and security of IoT data overweight the urgency of the oil field operation monitoring and control.

Blockchain transactions require certain fees (e.g., cryptocurrency) paid by the sender to execute the transactions. Deploying a smart contract is via a transaction and therefore also requires a fee. For each transaction in an Ethereum blockchain, the caller may declare a gas price, namely how much Ether the caller is willing to pay. Usually, the higher the proposed gas price, the more likely the transaction will be executed quicker. Using a private or hybrid blockchain can eliminate or reduce transaction costs. However, the deployment of a private blockchain means reduced decentralization. A tradeoff must be made in terms of the costs to execute blockchain transactions and the degree of decentralization to fully unite the merits of a blockchain. In our application scenarios, automatic monitoring and control is the ultimate goal of the oil field operations and a private or hybrid blockchain is appropriate to minimize or eliminate the blockchain transaction costs.

VI. CONCLUSION

The highly regulated and capital-intensive oil and gas industry demands a high level of digital transformation to facilitate data-driven decisions and automatic monitoring and control of oil field operations. This paper presents a blockchain-based IoT framework for remote monitoring and control of oil field operations. The primary functions of the framework include:

- Blockchain-based IoT system for real-time oil field data collection and monitoring.
- A fully decentralized and trustworthy platform for data storage and further analysis.
- A mechanism provided by the smart contracts to automatically execute a set of rules on the collected data to determine the actions for the safety and efficiency of the oil field operations.

This kind of automatic monitoring and control is carried out in a peer-to-peer environment without human intervention, thereby preventing human error and the risk of accidents and losses. We introduced the main components of the framework, and provided simulation and verification of smart contracts in the blockchain network.

ACKNOWLEDGMENT

The authors would like to thank Dr. Hui Pu and Joseph Lindemann for their discussions about this research.
YANJUN ZUO received the Ph.D. degree from the University of Arkansas, Fayetteville, AR, USA. He is currently a Professor at the University of North Dakota, Grand Forks, ND, USA. He has published numerous articles in refereed journals and conference proceedings, including IEEE TRANSACTIONS ON SECURE AND DEPENDABLE COMPUTING, IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS, ENTERPRISE INFORMATION SYSTEMS, Decision Support Systems, Journal of Information Management and Computer Security, Information Systems Frontiers, International Conference on Information Systems, and Hawaii International Conference on Systems Science. His research interests include information security, Blockchain, the Internet of Things, and trustworthy systems.

ZHENYU QI was born in Baotou, Inner Mongolia, China, in 1996. He received the B.S. degree in accountancy from the University of North Dakota, Grand Forks, ND, USA, in 2020, where he is currently pursuing the M.S. degree in accountancy.

From 2020 to 2021, he was a Research Assistant with the Department of Accountancy, University of North Dakota. His research interests include Blockchain, the Internet of Things, and corporate reporting of environment, social, and governance.

Mr. Qi is a member of the North Dakota Certified Public Accountants Society.

YANJUN ZUO, Z. Qi: Blockchain-Based IoT Framework for Oil Field Remote Monitoring and Control