Hindawi
Scientific Programming
Volume 2021, Article ID 3988070, 12 pages
https://doi.org/10.1155/2021/3988070

Research Article

Smart Payment Contract Mechanism Based on Blockchain Smart Contract Mechanism

Xianyun Ge

International Law School, Southwest University of Political Science & Law, Chongqing 401120, China

Correspondence should be addressed to Xianyun Ge; 2004760@shcc.edu.cn

Received 16 September 2021; Accepted 5 November 2021; Published 2 December 2021

Academic Editor: Shujaat Hussain Kausar

In recent years, blockchain technology has become a hot topic in various industries. With the development and maturity of blockchain technology, it has been applied to finance, law, etc., with its advantages of decentralization, openness, information security, and concealment. The application scenarios of industry are becoming more and more abundant. Compared with the traditional TPA payment contract form, the smart contract mechanism based on blockchain technology is obviously more efficient, convenient, and safe. Against this background, we design a smart payment contract suitable for cloud storage by studying Ethereum. The relationship clause in the smart payment contract should be regulated around the contract law. The smart contract payment linkage clause can be classified into three forms, including conditional effective type, contract joint type, and contract link type, which correspond to the contract law. Therefore, the contract legal system for smart contract payment linkage clauses should follow typified thinking. Based on blockchain technology, smart contracts not only reduce the number of interactions in contract execution but also allow users to stop paying for cloud services when data is lost or damaged. The precise method is to generate each node with a private chain and place the smart contract on the private chain. With the decentralization of the blockchain private chain, the advantages of read-only data, and traceability of information, the storage of smart payment contract data is more secure. Both parties to the transaction are more trustworthy. Therefore, the proposed system has a safe and efficient smart contract payment mechanism, which brings a good user experience to users, which proves the significance and value of this research.

1. Introduction

With the popularity of the Bitcoin project, the blockchain technology behind it has gradually attracted the attention of scholars, and people have begun to realize that blockchain technology is a kind of scenario that can be applied in situations where users do not trust each other and there is no centralized organization. At present, blockchain technology is not only a Bitcoin project but has begun to be applied in many fields including medical care, credit investigation, finance, education, and the Internet of Things [1–4]. With the rapid development of the blockchain open-source community, its blockchain technology is continuously expanding to all walks of life. For example, in the power industry, blockchain technology is applied to jointly maintain the same set of power accounts of the power company under the premise that each meter does not trust each other to ensure the traceability and nontampering of power data [5]. Among various open-source blockchain communities, the application of smart contracts in Ethereum [6] makes it a leader in the blockchain community. Ethereum is a combination of blockchain technology and smart contract technology. Virtual cryptocurrency is only an application of blockchain technology, and Ethereum has expanded it and added smart contract technology, enabling blockchain technology to be used in more and more industries. "Smart contract" is a concept first proposed by cryptographer Szabo [7] in the 1990s, almost the same age as the Internet. However, due to the lack of a credible execution environment, smart contract technology has not been practically applied. But with the birth of Bitcoin, it is natural to provide a credible execution environment for smart

contracts. Szabo pointed out that "smart contracts promote the execution of contracts through the use of protocols and user interfaces." Essentially, a smart contract is a code that sets rules and processes information according to the rules. It accepts information and data from the outside world and then outputs corresponding information and operations through the internal settings of the smart contract. The set conditions are triggered, and the program will execute the corresponding processing operations in accordance with the rules.

Recently, with the popularity of the Ethereum open-source community, some industries at home and abroad have successfully applied blockchain technology and smart contract technology to develop decentralized APP (DAPP) applications, but some industries are still at a stage of confusion and continuous experimentation. The most important thing in the development of DAPP applications is actually to find the most relevant application scenarios in your industry and the characteristics of blockchain technology. For example, the use of blockchain can decentralize [8, 9] and ensure data security and nontamperability to develop DAPP applications without the need for a central organization. Therefore, blockchain technology can be applied to all scenarios that require a central organization, as shown in Figure 1.

For example, in e-commerce transactions, Alipay uses the self-developed universal relational database OceanBase to support a huge transaction volume. Its transactions are based on the centralized e-commerce centralized platform. The transactions running on the centralized platform have many drawbacks. For example, the user's transaction funds will be stored on the platform for a period of time. The operation of the business is based on the user's trust in the centralized center. If the third party is not credible, then the user's funds are at risk of loss. The decentralized e-commerce system has the advantages of low cost and high speed. In recent years, with the development of distributed public ledger blockchain technology, its storage layer distributed structure, transaction consensus mechanism, encryption algorithm, and other characteristics have achieved reliable mutual trust in transactions without the supervision of a third party. Blockchain technology replaces the TPA mechanism and promotes the development of the decentralized e-commerce trading market. This paper designs a blockchain-based smart contract mechanism. The main tasks are as follows:

(i) Security design is used to improve the system security from three levels and design additional verification mechanisms when logging in at the application layer to effectively prevent DDoS attacks; at the data storage level, the design is based on the storage mechanism of the combination of the blockchain and the off-chain database, the user's private information is stored off-chain, and the password information is Hash-encrypted to prevent violent database attacks and account information leakage; at the transaction processing level, commodity transaction operations are completed by smart contracts. The transaction information is stored in the blockchain to realize safe and reliable transactions.

Figure 1: Smart contract based on blockchain.

(ii) Query optimization design is used to index the user information and product information stored in the off-chain database, design the product information and transaction storage structure in the smart contract, establish the mapping relationship between users and product transactions, and improve the query efficiency of product information and transaction status. This article completes blockchain-based transaction processing based on the Ethereum platform.

(iii) Due to the automatic execution, decentralized supervision, and irrevocability of the smart contract itself, coupled with the complex diversity of the smart contract payment linkage clauses, the smart contract payment linkage clauses are in terms of validity determination, postrelief, and prereglulation. Both have brought huge challenges to the regulation of contract law, and this article also proposes a solution to this problem.

2. Related Work

In 2008, blockchain technology was born. It was proposed by a person named Satoshi Nakamoto in a research report entitled "Bitcoin: A Peer-to-Peer Electronic Cash System" published by him. The blockchain described in this report is a decentralized shared ledger that combines data blocks in a chain into a specific data structure in chronological order and is cryptographically guaranteed [10]. After he submitted this report, he uploaded the earliest blockchain project: the Bitcoin project. In essence, it is a digital currency project based on cryptography and blockchain technology in the form of a P2P network. This currency is different from ordinary currencies in that it relies on a large amount of data to calculate and generate. The emergence of blockchain technology has successfully solved the two major problems that have been faced in the digital currency field before: the Byzantine Generals problem [11] and the double payment problem. The Byzantine Generals problem refers to the problem of how to reach a consensus in order to achieve a

goal in a distributed network that lacks a central node and is composed of nodes that do not trust each other [12]. The problem of double payment refers to the fact that, in the field of digital currency, money is just a string of numbers and it is easy to copy, which may lead to a situation where "the same amount of money" is spent twice or more. The blockchain system is a decentralized system that uses cryptographic encryption technology, distributed storage databases, and consensus mechanisms to solve the double payment problem in the digital currency field without the need for a centralized organization and uses a consensus mechanism to solve the Byzantine problem. At present, the application of blockchain in the field of digital currency, especially the virtual currency such as Bitcoin, has been going on for a long time, while the project of combining blockchain and smart contract like Ethereum is still in its preliminary stage. For example, many research studies on the application of blockchain in medical insurance are still in the demonstration stage or just put forward concepts. Some demonstrative projects abroad are as follows: Pokit Dok [13], a company that provides medical API services, announced cooperation with Intel to launch the "Dokchain" medical blockchain technology solution. The Dokchain medical blockchain project can provide identity management services for doctors and patients. It can verify and record the identity information of both doctors and patients. After the verification is successful, the edited smart contract will be executed immediately, which will greatly improve the efficiency of medical claims and be used for medical supply. The verification of the chain can record the prescription drugs on the blockchain when the doctor writes a prescription for the patient, and consumers can view their prescriptions and open and transparent drug prices in real time. Change Healthcare is a medical IT company in the United States. Its main business is to recommend appropriate medical services to users based on historical data, including cost information and service quality, to improve transparency in the medical and health field. Change Healthcare joins the Hashed Health blockchain alliance and uses the open-source blockchain architecture Hyperledger Fabric 1.0 [14,15] initiated by the Linux Foundation to create a distributed ledger. Companies can track submissions throughout the entire claim cycle in real time and accurately and remittance status. At the same time, in addition to improving transparency and efficiency, the introduction of blockchain technology in smart payment networks will also help improve auditability, traceability, and credibility, thereby improving the company's revenue cycle management.

In terms of the specific application of blockchain technology, most of the existing applications are relatively clear in structure and practical in function, but the degree of standardization is not high, each company will apply different technologies, and the application scale is still small [16]. In terms of privacy protection, Zyskind and Nathan [17] proposed a personal mobile phone privacy protection method, which transmits information through a third-party client. The system is divided into the mobile phone owner, the third-party client provider, and the continuous operation mechanism of the blockchain. The third party is to encrypt and store the user's private data through blockchain technology. Akinyele et al. [18] proposed to use encryption algorithms to encrypt digital information on mobile devices and use blockchain technology to store information to ensure the security of digital information. In the field of education, the first school to use blockchain to record academic information is Hoboton College in San Francisco, USA. The information in the academic certificate is stored on the blockchain through blockchain technology to ensure the authenticity of the information. It can also be faster and safer when verifying academic qualifications [19]. The MIT Media Lab has also developed a degree certificate detection system based on blockchain technology [20]. Similarly, Sony announced that it is cooperating with IBM to develop a set of education services, using blockchain technology to track students' learning progress, and to ensure that student file records cannot be tampered with [21]. Watanabe and others [22] proposed a digital information distribution system based on blockchain, which applies the blockchain to copyright protection and changes the traditional digital TV conditional access system CAS and digital rights management DRM. Domestic experts and scholars also proposed specific scenarios for the combination of blockchain and various industries. Li [23] proposed a research on the construction of a blockchain-based electronic invoice cloud platform, using the framework of the blockchain and cloud computing to build a distributed electronic invoice cloud platform. Yang et al. [24] and Liu [25] proposed a method for designing and implementing a blockchain-based supply chain information platform; Ye et al. [26] designed a supply chain system that combines blockchain technology. The smart contract is combined with the Internet of Things to save transaction costs and ensure system security, and the feasibility of the system is proved through experiments. However, the application of blockchain technology and smart contracts in our country is still in the initial stage of exploration. Therefore, domestic research on blockchain smart contracts is still mainly at the basic introduction and technical application level of blockchain smart contracts, as well as research on the legal level. Through the search of authoritative papers database, as of early September 2017, there is currently no legal Chinese literature on smart contracts. Although the legal attributes of smart contracts are controversial, most scholars believe that "code is law" is just a good idea of computer engineers, and smart contracts should still be in the scope of contract laws and regulations. As for the specific terms of the smart contract, relevant legal research has not yet been involved. According to estimates by the World Economic Forum, by 2025, assets equivalent to 10% of global GDP will be placed in the blockchain. At that time, smart contracts will be widely popularized and used, and the research on the legal issues of smart contract terms brooks no delay.

3. Method

In this section, the proposed methodology is explained in detail. The subsequent sections enlighten each step of the study.

3.1. Mainstream Consensus Mechanism. This section explains POW workload proof mechanism and POS equity proof mechanism.

3.1.1. POW Workload Proof Mechanism. The POW workload proof mechanism was first used to solve denial-of-service attacks. e-mail is taken as an example. Before the client initiates a request, it needs to perform a hash calculation. After the requirements are met, the mail server can be requested to ensure that the mail server provides services at the same time and can filter spam. This algorithm requires all nodes to continue to perform calculations. Relying on the node's own computing power, its algorithm needs to satisfy the following equation:

$$\text{Hash}(\text{param} \parallel \text{nonce}) < \text{target}, \tag{1}$$

where param is the block-related information, nonce is a random number, which meets the SHA256 value and consists of N leading zeros, and target is the target value determined by the current difficulty value. Finding a qualified nonce can only be achieved by the exhaustive method; in theory, the better the performance is and the higher the computing power is, the earlier the value can be found. The node that first calculates the nonce value obtains the right to bookkeeping, and other nodes only need to verify whether the result is correct, and all nodes in the network recognize the generation of the new block and reach an agreement.

The algorithm flow is shown in Figure 2. When a malicious node in the blockchain network tries to tamper with data, it can be seen from the blockchain workflow that the longest chain is considered the correct chain, so a longer chain needs to be generated from the tampered block to replace the current one. The proof-of-work mechanism of chain requires a lot of computing power. The time to generate a new chain is the time when all blocks are currently generated. Assuming that the original chain also generates new blocks after generation, the data cannot be successfully tampered with, so from a statistical perspective in other words, after 6 new blocks are generated after one block, the block cannot be tampered with. The advantages of the proof-of-work mechanism are simple and direct, and the nodes can be expanded. As the number of nodes in the blockchain network continues to increase, the computational cost of tampering with data continues to increase, which improves the security of the system. At the same time, as long as 51% of the nodes of the entire blockchain network are honest, the entire system can operate normally and stably. However, because the generation of blocks requires a lot of calculations, which wastes computing resources, blocks cannot be quickly generated. Taking Bitcoin as an example, only about 6 blocks can be confirmed an hour, the transaction speed is slow, and the efficiency is low.

3.1.2. POS Equity Proof Mechanism. As a virtual resource, digital currency can be used for circulation or has storage value. The more nodes that hold tokens, the more stable the value of the currency is hoped, and the more inclined to maintain the stability of the system. Therefore, the use of the relevant information of the tokens held by the node to determine the accounting node is called the equity proof mechanism. Among them, there are different types of calculation methods according to different token holding information. According to the concept of coin age, coin age is the product of the number of tokens held and the time of holding the tokens, and the algorithm needs to satisfy the following equation:

$$\text{Hash}(t \parallel \text{param}) < \text{coinage} \times \text{target}, \tag{2}$$

where t is the relevant parameter of the transaction and coinage is the coin age. When the coin age is older, the easier it is for the Hash calculation to satisfy the inequality, and the easier it is to obtain the accounting right. However, the time of holding tokens will continue to increase, even if it is offline, the age of the token will continue to increase, so the number of tokens can be used to replace the age of the token, and the algorithm needs to satisfy the following equation :

$$\text{Hash}(t \parallel \text{param}) < \text{coins} \times \text{target}. \tag{3}$$

If the concept of currency age is not used, then equation (4) can be used:

$$F(\text{Hash}(t \parallel \text{param}) < \text{balance} \times \text{takttime} \times \text{target}, \tag{4}$$

where balance is the balance and takttime is the interval time between the block and the previous blocks. Since the equity proof mechanism is not completed by calculating the hash of the specified difficulty, it saves computing resources compared to the workload proof mechanism. However, as the system continues to develop and blocks increase, nodes with a large number of tokens have a greater probability of obtaining the right to record transaction data and generate blocks every time a transaction occurs. This will lead to the excessive concentration of accounting rights in the initial nodes, causing initial distribution problems, and users will hoard tokens because of their own verification probability, which reduces the liquidity of tokens and is not conducive to the sustainable development of the entire system.

3.2. The Realization of Smart Contracts and the Construction of Private Chains. This section describes the realization of smart contracts, the establishment of the Ethereum private chain, and the deployment of smart contracts on the Ethereum private chain. This section takes common medical contracts in life as an example to study the related mechanisms of smart payment contracts. These contents can realize the intelligence of medical insurance and ensure the security and transparency of medical insurance funds.

The first step is to build the overall structure of the medical insurance smart contract. Among the three parties of patients, medical institutions, and insurance providers, medical institutions and insurance providers act as administrators. Medical institutions are designated by the insurance provider. Only medical institutions approved by the insurance provider upload the patient's medical

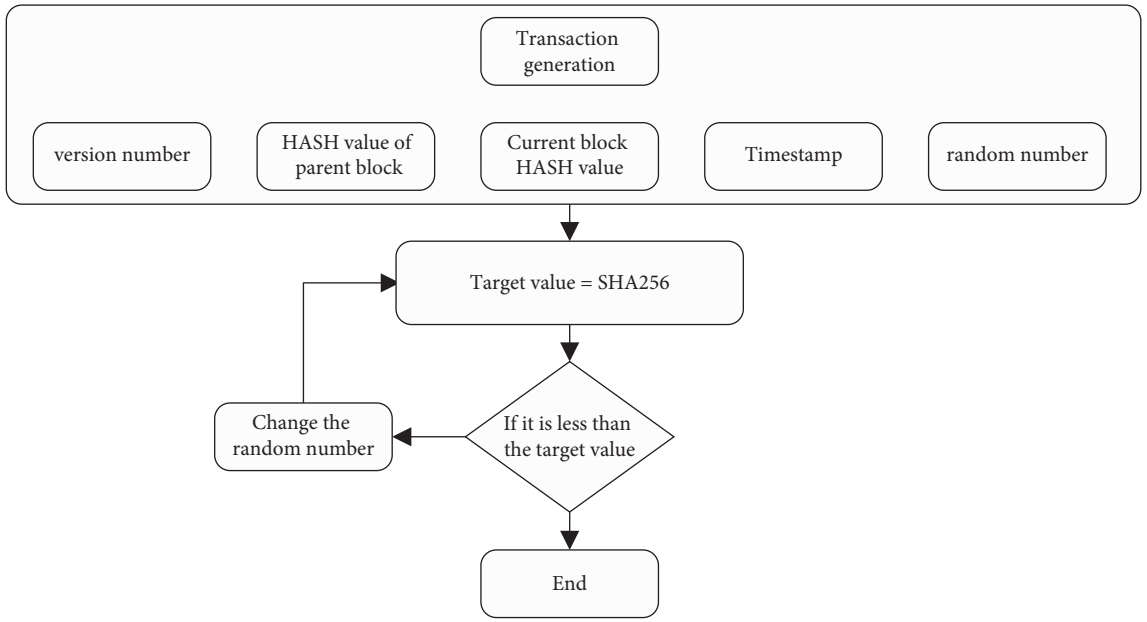

Figure 2: POW workload proved flowchart.

information. Only institutions can obtain insurance compensation.

(1) Patient purchase contract: the insurance provider provides an alternative insurance contract, which mainly contains various diseases that can be reimbursed. When a patient needs to purchase insurance, medical institutions can judge whether the patient has successfully purchased it based on the patient's health record before purchasing the insurance. If the patient has these diseases, the contract is not generated. If there are no such diseases, the contract is generated. The patient can obtain the purchase authorization and purchase the contract from the insurance provider through the purchase interface provided by the contract, the corresponding patient health record, the insurance provider, and the insurance contract. The payment amount will be deployed on the blockchain in the form of a smart contract.

(2) Medical payment and uploading medical information: medical institutions can modify the medical payment amount in the contract within the validity period of the patient contract and add the treatment amount of the insured disease in the patient insurance to the contract so that the insurance provider can pay what it believes insurance compensation. Medical institutions can also upload the patient's medical information to the blockchain for the insurer to determine whether the patient meets the conditions of the purchase contract. At the same time, it can also be used as the basis for payment. If a change of the patient's medical health information occurs, such as suffering from a certain disease, the insurer can pay for the disease in a timely manner with the medical and health information uploaded by the medical institution.

(3) When the insurer pays the compensation to the patient and needs to obtain compensation, the medical institution also judges whether the patient can obtain the compensation based on the latest health record of the patient. If possible, the patient can obtain the compensation authorization from the medical institution, and the insurance provider will compensate the patient through the compensation interface of the contract. The same data will also be permanently recorded in the blockchain.

(4) The insurer provides the insurance contract. The insurer has an interface for formulating the contract, which is the core interface of the system. The interface is mainly used to formulate a customized insurance contract for the patient. The insurance contract contains the details of the diseases that the policy can insure, the price of the insurance, the amount of insurance compensation, and so on. Patients can purchase insurance contracts through this interface, and only those who meet the conditions can purchase exclusive insurance contracts. After the patient purchases this contract, the insurer can provide follow-up services to the patient. The system function diagram is shown in Figure 3.

3.3. Smart Contract Implementation Code. In the subsequent sections, member creation and initialization, realization of patient purchase insurance contract, and payment compensation provided by the insurer have been explained.

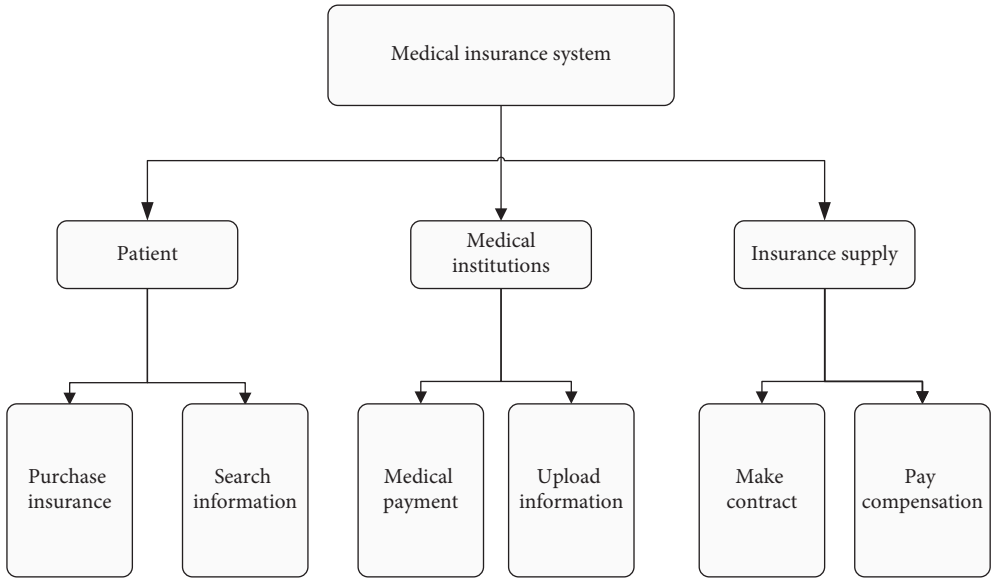

Figure 3: Proposed system flowchart.

3.3.1. Member Creation and Initialization. Smart contracts can be stored forever as long as they are deployed in the blockchain. As long as the smart contract exists, the contract will be executed in accordance with the internal agreement. The contract is similar to an insurance contract, which is provided by an insurance provider, so there should be an insurance provider as an administrator to deploy smart contracts and pay corresponding measures in accordance with the contract. The initial definition of the insurance provider as an administrator is shown in Figure 4. Here, this article defines the address of the insurance provider as an "address" variable, where "constructor" is a constructor function. The concept of a constructor is the same as the definition in C++, JAVA, and other high-level languages, and it is a function with the same name as the contract class. The function is to initialize the address of the insurance provider to "insurance" when the contract is first deployed, and it cannot be modified afterwards. The "setHospital" method, by verifying the Ethereum address of the insurance provider, ensures that the incoming Ethereum address is the Ethereum address of a medical institution recognized by the insurance provider, and then initializes this address to the hospital, which has been passed through afterwards.

The "address" type is a special type provided by the Solidity language, which actually refers to a 20-byte Ethereum address. And "msg" is the global variable "msg.sender" provided by the Solidity language. The function of "msg.sender" is to call its member value "sender" through "msg," where "sender" refers to the address of the external caller of the current function. Next, the disease structure and the patient structure are defined, where the disease structure contains the name of the disease and whether the patient has the disease, "true" means suffering, and "false" means not. The patient structure contains the patient's information, such as the patient's name, disease array, insurance contract amount, patient payment

```
address private insurance;
constructor () public{
    insurance=msg.sender;
}
uint index=0;
address private hospital;
function setHospital (address addr)public{
    require (msg.sender== insurance);
    hospital= addr;
}
```

Figure 4: Administrator and medical institution initialization.

insurance contract amount, patient treatment disease payment amount, obtained insurance compensation amount, and the effective deadline of the contract, add one at the end. The insurance contract has an invalid switch, "true" means that it can be traded, and "false" means that it can no longer be traded. The details are shown in Figure 5. A mapping relationship is declared here to map the patient's Ethereum address to the patient so that each patient depends on the Ethereum address to correspond to a unique patient structure.

Now the insurance provider can initialize the patient's insurance contract with the patient structure, similar to the contract formulation in real life. The details are shown in Figure 6. When the contract is initialized, ensure that the caller of the function is the provider of insurance, that is, the issuer of the contract, and then pass each variable parameter to the structure of the corresponding patient. Among them, "contract Sum" (insurance contract amount) is set by the contracting party; the initial value of "patient-Paymentcontract Sum" is 0. The initial value of "patient-Payment" (treatment costs) is 0; the initial value of "insurancePayment" (acquired insurance payment amount)

```
struct Disease{
unit diseaseId;
String diseaseName;
bool disease;
struct Patient{
    string name;
    mapping (uint=>Disease) diseases;
    uint256 contractSum;
    uint256 patientPaymentcontractSum
    uint2_56 patientPayment;
    uint256 insurancePayment;
    uint2_56 endtime;
    bool raiseRight;
    }
    mappin} (address=>Patient) patient;
```

Figure 5: Contracts and disease institutions.

```
function Insurance Init (address addr,string memory _name, uint256
_contract Sum,unit256 _ endtime) public {
    require (msg.sender== insurance);
    patient[addr].name=_name;
    patient[addr].contract Sum= _contract Sum;
    patient[addr].patient Paymentcontract Sum= 0;
    patient[addr].patient Payment= 0;
    patient[addr].insurance Payment= 0;
    patient[addr].endtime =now+_ endtime;
    patient[addr]. raise Right =false;
    }
}
```

Figure 6: Administrator and medical institution initialization.

is 0; "endtime" (insurance expiration period) is set by the contracting party; the initial value of "raiseRight" (transaction authorization) is false. Then we can control these variables by setting various functions. Pay special attention to "memory" means to mark this as value transfer. If we want to use reference transfer, the "storage" keyword is used and the function type of the function is changed to internal or private.

3.3.2. Realization of Patient Purchase Insurance Contract. After the patient initializes the patient's contract structure at the insurance provider, the medical institution uploads the patient's latest health information. The patient can purchase an insurance contract by calling a purchase function of the contract. The specific function implementation is shown in Figure 7.

This function sets a "bool" type value to record whether the patient is healthy. By verifying the patient's medical health information, if the patient has health problems, this value is set to "false," which means that the patient has a health problem. Therefore, we can rely on this value. When the value is true, it means that the patient can purchase the contract and vice versa. This function finally modifies the value of the caller's "patient Payment contract Sum" to the

```
function pay Contract () public {
    bool sign=true;
    for(uint
dis=0;dis<patient[msg.sender].diseases.len
gth;dis++){
if (patient[msg.sender].diseases[dis].diseas
e== true){
      sign=false;
      break;
}
}
  if(!sign) return;
    patient[msg.sender].
patient Paymentcontract
Sum+=patient[msg.sender].contract Sum;
    patient[msg.sender]. raise Right=true;
}
```

Figure 7: Purchase medical contract.

contract price and opens the transaction for this contract. Since this call will change the parameter value in the contract. Therefore, the data will be permanently recorded in the blockchain in the form of an ether transaction when it is called.

3.3.3. Payment Compensation Provided by the Insurer. Next, the insurance provider needs to make reasonable compensation for the corresponding patient through the compensation function in the system according to the treatment cost in the contract. The specific function code implementation is shown in Figure 8.

At the beginning of the function, permission authentication is required to confirm that the external caller is an insurance provider. After it is judged that the patient has paid the treatment cost, and the patient has subscribed to the insurance smart contract, the same amount of the treatment cost will be paid to the patient's insurance compensation amount. Similarly, due to data modification, similar to a Bitcoin transaction, it will be permanently recorded in the blockchain when this method is called.

3.4. Smart Contract Payment Linkage Clause Control Rules. In order to make the smart contract clauses perfectly reflect the contract law norms, it is necessary to follow the next steps: (1) listing and screening out contract law issues in the order of daily contract transactions, including contract establishment, effectiveness, and breach of contract relief; (2) extracting relevant contract law rules based on issues at different stages of the contract; (3) focusing on the extracted contract law rules, identifying and confirming the input content that may be required; (4) identifying the relevant interests behind the different contract law rules; and (5) evaluating the possibility of the automatic execution of the contract for each rule of the contract and the result. There may be two situations. One is that the contract code

```
function treatment Payment (address addr) public {
        require (msg.sender== insurance);
        if (patient[addr]. patient Payment==0||!patient[addr]. raise Right) return;
        patient[addr]. insurance Payment += patient[addr]. patient Payment;
}
```

Figure 8: Compensation for medical insurance money.

completely conforms to the intention of the parties and does not cause ambiguity; the other is that the transaction situation is relatively complicated, and the code semantics needs to be further explained; (6) for more complicated transactions, we can choose a variety of ways to deal with it. Evasion, such as negotiating to establish a "highest rule," that is, the contract code must not violate the principle of contract purpose; otherwise, it cannot be executed automatically or relies on a trusted third party to control the execution of the contract. In order to make the above method more intuitive, Professor Eric carried out the purchase of a car in installments as an example and wrote the corresponding contract code program. The smart contract not only includes the basic rights and obligations of buyers and sellers but also makes assumptions about possible defaults and sets the content of notifying the counterparty of the default. Compared with the previous smart contract code, the smart contract is more comply with the basic norms of the contract law, helps protect the legal rights of the parties, and provides a good contract template for the transaction type of installment car purchases. However, the use of template contracts is still the result of the free choice of the parties. As far as contract laws and regulations are concerned, it should still be an arbitrary standard. However, in practice, the terms frequently used in template contracts have been proven to effectively protect the interests of contract parties. The method of entering into the contract can be regarded as an effective way of regulation.

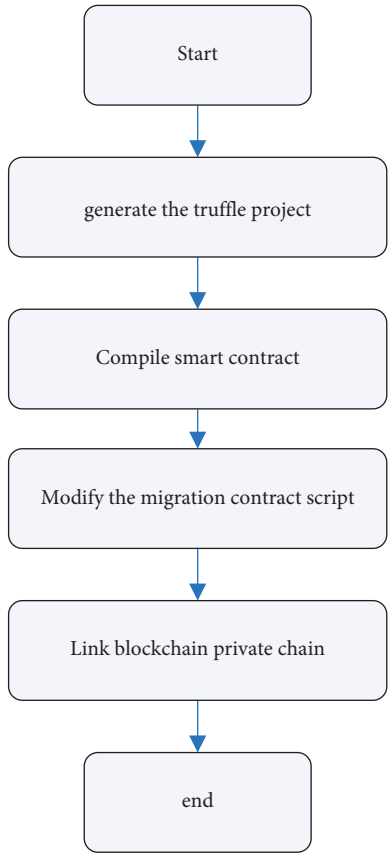

Figure 9: Truffle development flowchart.

4. Experiments and Results

4.1. System Construction. Taking the medical insurance payment system as an example, it is easy to use Truffle to build a distributed application of Ethereum. The basic process of Truffle development is shown in Figure 9. The first is to generate the truffle project and then compile the smart contract that was written before, then use the migration script to publish the smart contract to the Ethereum network, and finally build the blockchain application.

(1) Create and initialize the project. First, create a folder where you want to store the project. Since this system is completed under the windows platform, so enter the "power Shell" under this folder and then use the "truffle unbox webpack" command in the "power Shell" console to generate the truffle project, and you can get all the file directories. The front-end pages of the project are stored in the App directory, mainly JavaScript files and CSS files. The Contracts directory is where the truffle stores the smart contracts we wrote by default. The smart contracts we wrote in Section 3 are stored in this directory. It is also necessary to delete the default generated "Meta Coin.sol" file and "Convert Lib.sol" file in the directory, leaving only the "Migrations.Sol" file. The function of this file is to help us deploy the contract.

4.2. Smart Contract Compilation. Now add the "2_deploy_contracts" deployment script under the migration files, remove the deployment process that comes with it, and modify it to the smart contract written in Section 3. After the modification is completed and saved, there will be an error that the contract cannot be found. Next, in the project directory, enter "truffle compile" in the Pow Shell console. After these preparations are completed, the last step

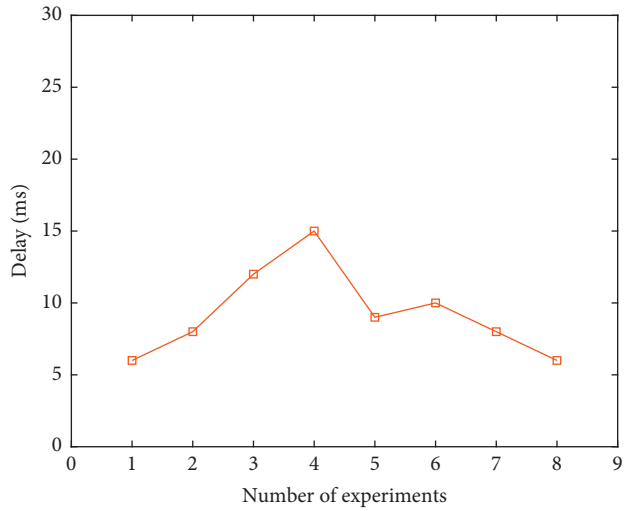

Figure 10: System delay test result.

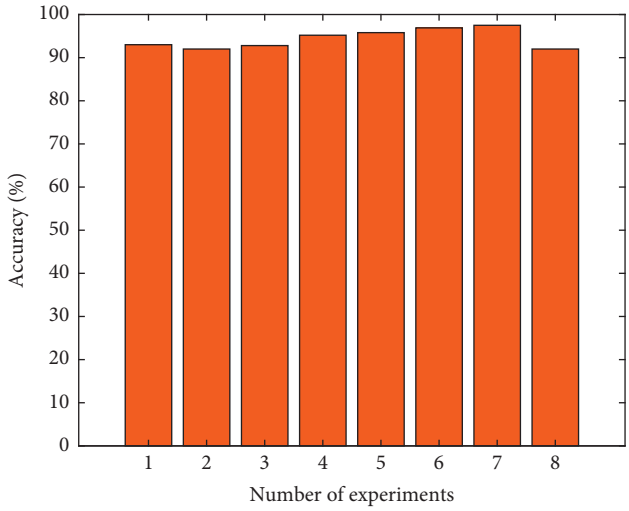

Figure 11: System accuracy test result.

is to deploy the truffle project we created to the private blockchain built in Sections 3 and 4. The specific method is to switch to the truffle project and find "truffle-config.js" in it. Edit this file and add the following content to it. Among them, "8545" refers to the port number, which is the default port number of Ethereum RPC; "99" is the "network_id," and the value here must be consistent with the private chain of the blockchain started by the configuration. Next, we can deploy our smart contract to the previously created blockchain private chain and just enter the command such as "truffle migrate–network privatenetwork" in the console. After this success, we successfully deployed our truffle project to our blockchain private chain network. The next step is to test the performance of the various indicators of the system. The results are shown in Figures 10–13.

(1) Delay test

(2) Accuracy test

(3) Recall rate test

(4) Flexibility test

It can be seen from the above experimental results that the system has a certain degree of stability and reliability.

4.3. System Achievements. First of all, this system guarantees the user's personal privacy. The system only grants the customer's target organization the right to have its information. At the same time, only the customer can view their own information and the status of insurance compensation. Using the security and transparency of the blockchain can effectively avoid unauthorized access, use, destruction, and modification. The second system guarantees the authenticity and reliability of the data. This system uses the decentralization of blockchain and smart contract technology and the advantages of nontamperability and stores data in the blockchain structure to ensure that everything is open and transparent. In this system, customers only have access to read their own medical data, which eliminates the possibility

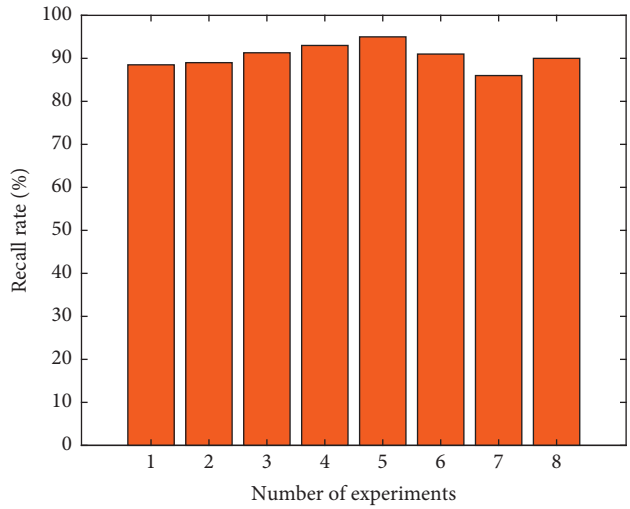

Figure 12: System recall rate test result.

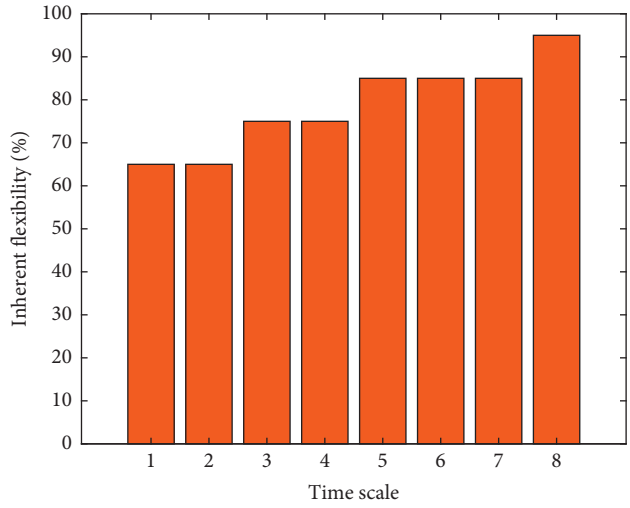

Figure 13: Inherent flexibility with multiple time scales.

of fraud. Only medical institutions approved by the insurer have the right to read and write the patient's medical data. Finally, due to the characteristics of the smart contract, it is guaranteed that the patient does not need a cumbersome process to obtain compensation. Under the guarantee of the authenticity and reliability of the data of the blockchain technology, the manual review process is subtracted so that the insurer can quickly pay the insurer the insurance compensation. This system can greatly improve the shortcomings of the traditional commercial insurance system, allowing users who use this system to enjoy convenience and safety and fully embody the practical value of this system.

5. Conclusion

The main work of this thesis introduces blockchain technology in depth. After introducing the underlying principles of the blockchain, these characteristics of the blockchain are used to study the application of the blockchain technology in the smart contract payment contract mechanism. Blockchain technology is not a sudden emergence of new technology, but the integration of existing technologies, including privacy protection technology based on cryptographic algorithms, that is, the SHA256 function algorithm in the hash algorithm, the asymmetric encryption technology in the encryption algorithm, and distributed consensus mechanism to solve the Byzantine problem. In this article, the development of the system is based on the Ethereum platform, which is mainly based on the remix platform, writing smart contracts, using smart contracts for medical insurance to implement the main functions of the kernel in the system, and then using the truffle framework to easily build the entire distributed application system. In the medical insurance system, the contracts purchased by users are permanently recorded in the blockchain, and at the same time, the characteristics of the blockchain are used to ensure

the authenticity and reliability of these data. Because smart contracts and medical insurance contracts are naturally self-consistent, they help insurers to easily formulate insurance contracts. Contract execution can be transformed into the effective execution of smart contract functions, and the task of insurance compensation can be completed intelligently. The medical insurance system based on the blockchain smart contract uses the nontampering of the smart contract to ensure the safety and reliability of the data, while saving the cumbersome verification process of traditional medical insurance and the high manual review fee and saving the operating cost of the insurer. In addition, as a form of contract transactions, smart contracts are still inseparable from the scope of contract law in legal regulation. Although the smart contract payment linkage clause can connect many smart contracts and form various types of contract interactive systems and even form a company-like contract linkage organization, forming a different traditional single contract form, the use of contract law thinking is correct. Its understanding and application are still the basic path that legal regulations should follow. In fact, different types of smart contract payment linkage clauses are closely similar to the corresponding contract clauses in the contract law. The basic legal principles and related legal provisions of the contract law are fully used, and the analysis is carried out in conjunction with the technical characteristics of the smart contract. It can effectively solve the problem of judging the contract validity of the smart contract payment linkage clause.

Data Availability

The datasets used are available from the corresponding author on reasonable request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] M. Swan, *Blockchain: Blueprint for a New Economy*, O' Reilly Media, Inc., Sebastopol, CA, USA, 2015.

[2] M. A. Jan, J. Cai, X. C. Gao et al., "Security and blockchain convergence with Internet of Multimedia Things: current trends, research challenges and future directions," *Journal of Network and Computer Applications*, vol. 175, pp. 1–23, 2020.

[3] Z. Hongmei, "A cross-border E-commerce approach based on blockchain technology," *Mobile Information Systems*, vol. 2021, Article ID 2006082, 10 pages, 2021.

[4] T. T. Thwin and S. Vasupongayya, "Blockchain-based access control model to preserve privacy for personal health record systems," *Security and Communication Networks*, vol. 2019, Article ID 8315614, 15 pages, 2019.

[5] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *Decentralized Business Review*, Article ID 21260, 2008.

[6] M. Daniel and C. Cristian, "Smart contract applications within blockchain technology: a systematic mapping study," *Telematics and Informatics*, vol. 35, no. 8, pp. 2337–2354, 2018.

[7] N. Szabo, "Formalizing and securing relationships on public networks," *First Monday*, vol. 2, no. 9, 1997.

[8] M. Vasek, "The age of cryptocurrency," *Science*, vol. 348, no. 6241, pp. 1308-1309, 2015.

[9] H. Hodson, "Bitcoin moves beyond money," *New Scientist*, vol. 220, no. 2945, pp. 24–26, 2013.

[10] S. Nakamoto, "Bitcoin: a peer-to-peer electronic Cash system," hops://bitcoin.org/bitcoin.pdf, 2008.

[11] A. M. Antonopoulos, *Mastering Bitcoin: Unlocking Digital Crypto-Currencies*, O'Reilly Media, Inc., Sebastopol, CA, USA, 2014.

[12] M. Swan, "Blockchain thinking: the brain as a decentralized autonomous corporation," *IEEE Technology&Society Magazine*, vol. 34, no. 4, pp. 41–52, 2015.

[13] T. K. Mackey, T. T. Kuo, B. Gummadi et al., "'Fit-for-purpose?'-challenges and opportunities for applications of blockchain technology in the future of healthcare," *BMC Medicine*, vol. 17, no. 1, pp. 1–17, 2019.

[14] Q. Nasir, I. A. Qasse, and M. Abu Talib, "Performance analysis of hyperledger fabric platforms," *Security And Communication Networks*, vol. 2018, Article ID 3976093, 14 pages, 2018.

[15] P. Yuan, X. Xiong, and L. Lei, "Design and emission trading system," *IEEE access, Implementation on Hyperledger-Based*, vol. 7, no. 7, pp. 5112–5127, 2017.

[16] D. C. Wang, *The Empirical Study on Trust-Building Mechanism of Second-Hand Trading Platform in Sharing Economy*, Beijing University of Posts and Telecommunications, Beijing, China, 2019.

[17] G. Zyskind and O. Nathan, "Decentralizing privacy: using blockchain to protect personal data," in *Proceedings of the 2015 IEEE Security and Privacy Workshops*, pp. 180–184, IEEE, San Jose, CA, USA, May 2015.

[18] J. A. Akinyele, M. W. Pagano, M. D. Green, C. U. Lehmann, Z. N. J. Peterson, and A. D. Rubin, "Securing electronic medical records using attribute-based encryption on mobile devices," in *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*, pp. 75–86, Chicago, IL, USA, October 2011.

[19] X. M. Li, X. Li, and H. Q. Wu, "The application model and challenges of blockchain technology in education," *Modern Distance Education Research*, vol. 2, no. 34, p. r45, 2017.

[20] W. X. Xiong, *Research on Credit Credit Authentication System Based on Blockchain Technology*, Beijing University of Posts and Telecommunications, Beijing, China, 2018.

[21] S. Sony, "Global education develops technology using blockchain for open sharing of academic proficiency and progress records," 2018, https://www.sony.net/SonyInfo/News/Press/201810/18-1015E/index.html.

[22] H. Watanabe, S. Fujimura, and A. Nakadaira, "Blockchain contract: securing a blockchain applied to smart contracts," in *Proceedings of the 2016 IEEE international conference on consumer electronics (ICCE)*, pp. 467-468, IEEE, Las Vegas, NV, USA, January 2016.

[23] Z. Li, *Research on Construction of Electronic Invoice Cloud Platform Based on Blockchain*, Chinese Academy of Fiscal Sciences, Beijing, China, 2018.

[24] H. Q. Yang, L. Sun, and X. C. Zhao, "Build mutual trust supply chain information system based on blockchain," *Technological Progress and Countermeasures*, vol. 35, no. 5, pp. 21–31, 2018.

[25] Y. C. Liu, *Design and Implementation of Information Platform for Supply Based on Blockchain*, Mongolian university, Ulan Bator, Mongolia, 2019.

[26] X. R. Ye, Q. Shao, and R. Xiao, "A supply chain prototype system based on blockchain, smart contracts and the Internet of Things," *Science & Technology Review*, vol. 35, no. 23, pp. 62–69, 2017.