Contemporary Readings in Law and Social Justice
ISSN: 1948-9137, e-ISSN: 2162-2752
Vol 17 (1), 2025
pp. 214 - 225

Smart Contracts: Essence, Legal Nature, Advantages and Disadvantages

Ben Moussa Abdelmadjid

benmoussamadjid@univ-adrar.edu.dz

University of Adrar

Abstract

Amidst the rapid digital revolution and the swift proliferation of blockchain technologies and their diverse applications, smart contracts have become a focal point for researchers, legislators, and economic practitioners alike. These contracts, with their ability to automate the execution of agreement terms without intermediaries, reduce transaction costs, enhance transparency, and accelerate contractual procedures, offer numerous advantages. Nevertheless, they also raise several legal concerns and challenges, including the lack of consensus regarding their legal nature and their incompatibility with traditional contract theory. Additionally, technical challenges arise due to vulnerabilities to security breaches, inflexibility, and difficulties in modification, among others. Accordingly, this study aims to shed light on this type of contract by defining its concept, examining its legal nature, and reviewing its advantages and disadvantages.

Keywords: smart contracts, blockchain technology, legal challenges, decentralized finance (DeFi)

Received: 03/02/2025 **Accepted:** 15/04/2025 **Published:** 08/05/2025

Introduction

Smart contracts represent one of the most prominent manifestations of technological and technical advancement, particularly amid the accelerating pace of digital transformation across various sectors of society. They have introduced a new model of contracting that is expected to reshape transactions globally in the modern era, owing to their ability to redefine trust and automation through blockchain technology, which serves as their fundamental infrastructure.
Smart contracts consist of self-executing computer protocols, with the terms and execution mechanisms encoded in the form of software code. This allows contracting parties to dispense with traditional intermediaries such as banks or legal institutions.
In terms of significance, smart contracts have become a cornerstone of the decentralized finance (DeFi) ecosystem and intelligent systems, providing a secure and transparent framework for executing agreements across borders and disciplines. They now play vital roles in e-commerce, financial exchanges, property ownership, inventory tracking, automated profit distributions, real estate, media and entertainment, and even government sectors. Smart contracts have enabled a wide range of operations and solutions to challenges that were previously unattainable.

Research Problem

Despite the widespread adoption of this type of contract due to its unique characteristics, smart contracts remain experimental and require further development, study, and research. As an emerging technology, they incorporate numerous terms and codes unfamiliar to many, with knowledge of them largely confined to computer scientists and software engineers. This has left the field open to many questions, particularly given the complexity of its terminology. Additionally, smart contracts possess distinctive features that set them apart from traditional contracts and operate through a digital electronic medium—blockchain technology—which is characterized by decentralization, immutability, and resistance to modification. Accordingly, the research problem can be formulated through the following questions:

1. How did smart contracts originate and develop?
2. What is their true nature?
3. What is their technological foundation?
4. What is their legal nature?
5. What are their advantages and disadvantages?

Research Objectives

Based on the above, the study aims to achieve the following:

1. Define the concept of smart contracts as newly emerging contracts in both practical and legal contexts.
2. Identify the elements and components of smart contracts and how they differ from traditional contracts.
3. Clarify the characteristics of blockchain as the technological foundation of smart contracts.
4. Outline the advantages and disadvantages of smart contracts.

Research Structure

To answer these questions and address the research problems, this study adopts a structure consisting of three sections and a conclusion, as follows:

- **Section One:** Conceptual Framework
- **Section Two:** Legal Approach to Smart Contracts
- **Section Three:** Advantages and Disadvantages of Smart Contracts
- **Conclusion**

Section One: Conceptual Framework

First Requirement: The Concept of Smart Contracts

Although the topic of smart contracts is relatively recent, dating back to the mid-1990s:

First Branch: The Emergence of Smart Contracts

Smart contracts are considered a modern subject, first discussed in the 1990s by the American legal scholar and programmer Nick Szabo. In his 1997 article, Szabo promoted the concept, drawing on his combined technical and legal background in both philosophy and practical application[i]. He believed that smart contracts would enhance the contractual process from start to finish by increasing trust, security, and efficiency. Szabo proposed that contracts could be programmed and executed automatically by computers without the need for intermediaries such as lawyers or institutions.
To illustrate his point, he used simple examples such as the *vending machine*: you insert money and receive a product—no intermediaries and no negotiation. Szabo emphasized that this type of contract could revolutionize legal systems, finance, and e-commerce.
Despite the significance of this revolutionary idea, it remained largely theoretical at the time due to the absence of the necessary infrastructure for its practical implementation.

In 2009, the term *smart contracts* regained momentum with the advent of blockchain technology, whose creator developed the cryptocurrency Bitcoin. The Bitcoin network enabled the execution of basic, conventional smart contracts, thus overcoming the obstacle that had hindered the concept for years. The idea became feasible when enthusiasts realized that Bitcoin's blockchain possessed unique and secure capabilities for conducting transactions, storing, and transferring data in a decentralized manner[ii].

In late 2013, Vitalik Buterin published a white paper proposing the creation of the Ethereum[iii] platform as a new generation of programmable smart contracts powered by a more advanced blockchain. The network was officially launched in 2015, enabling the execution of smart contracts using the high-level programming language *Solidity*. However, the *DAO hack* in June 2016 exposed critical security vulnerabilities in Ethereum contracts, prompting the community to adopt *Formal Verification* methods to ensure code safety and support rapid innovation[iv]. In the following years, competing platforms such as *Cardano* and *Polkadot* emerged, focusing on self-governance and execution efficiency, supported by formal verification tools to enhance security, including *thirdweb*.
Today, smart contracts are widely used in decentralized finance (DeFi), non-fungible tokens (NFTs), and decentralized autonomous organizations (DAOs), reflecting their development and global proliferation across various digital and financial sectors[v].

Second Branch: Definition of Smart Contracts

Nick Szabo defined smart contracts before the emergence of blockchain[vi] technology as: *"Computer programs capable of automatically executing contract terms"*. This means that contract terms can be converted into code instructions and executed independently by a computer, reducing the need for intermediaries and increasing trust through verifiable, tamper-resistant records.

Others have defined smart contracts as: *"A set of tamper-resistant algorithms that are self-executing and self-verifying[vii]."* Smart contracts incorporate algorithms that cannot be altered once deployed and operate automatically when certain conditions are met. Their execution is confirmed across the blockchain network without human intervention.

According to this definition, smart contracts provide a means of executing agreements and transactions without intermediaries by encoding the terms and conditions as computer code. This code is stored and executed automatically on a distributed blockchain network. Owing to its decentralized nature, no central authority or third party is involved, ensuring precise and tamper-resistant execution and fostering mutual trust between parties, even if they have no prior acquaintance.

A similar definition appears on *Investopedia*: *"A smart contract is a self-executing program that automates the actions required in a transaction on a blockchain. Once the transaction is completed, it becomes traceable and irreversible[viii]."*

Based on the above, it can be concluded that smart contracts are automated, self-executing agreements that do not require human intervention. They operate through blockchain using programming languages and are designed to be irreversible, thereby enhancing trust and security in contractual relationships.

It should also be noted that some French legal scholars express reservations about describing these contracts as "smart," considering the term provocative[ix]. In their view, it implicitly creates a negative impression that undermines the value of the traditional concept of a contract. This criticism raises questions about the accuracy and appropriateness of the term—whether it satisfies the legal requirements of a binding contract (as defined by well-established contract law principles) and whether the attribute of "intelligence" ascribed to these contracts truly reflects characteristics of artificial intelligence or independent algorithmic interaction, rather than being a promotional exaggeration.

Second Requirement: The Technological Foundation of Smart Contracts

Blockchain forms the infrastructure that transformed smart contracts from a theoretical concept into reality. This type of contract is a programmable model that leverages blockchain's capabilities to convert legal agreements into automated processes that do not require an intermediary. Thus, it is essential to explain the true nature of this technology as the foundation and background of smart contracts—without delving too deeply into technical aspects beyond what serves the purpose of this study.

First Branch: Blockchain Technology

Blockchain is a *digital decentralized ledger* that records all transactions across a network of interconnected devices using peer-to-peer technology. Through this system, participants can validate transactions without the need for a central authority[x].

Blockchain is considered one of the revolutionary inventions because of its ability to eliminate the need for intermediaries and to dispense with the concept of an untrusted third party, such as banks, notaries, or administrative bodies. This is achieved through the cryptographic technologies it employs. The system allows every user to access the ledger while ensuring a high degree of security.

The **Blockchain system** consists of the following components:

1. **Block**:
This is the fundamental unit where data and transactions are recorded. It can be compared to a page in a digital ledger where a set of transactions is documented. Each block accommodates a certain amount of related transactions and does not accept more than its capacity. Once full, a new block is created and linked to the previous one. This prevents fictitious transactions within a block, which could freeze the chain or obstruct its role in recording or completing transactions[xi].

2. **Information**:
This refers to digital data recorded in the blocks that form the chain. This data is stored in a decentralized and secure manner using encryption technologies. It includes various types of records such as financial transactions (e.g., cryptocurrency transfers), smart contracts, digital identities, and others[xii].

3. **Hash**:
The hash is a code or cipher that encrypts transactions through complex mathematical operations using an algorithm within the block's program. It converts the written data into a digital message that is difficult for anyone to decipher. The hash serves the following key functions:
– Distinguishes one blockchain from others with a unique hash.
– Identifies each block by generating a unique hash code that represents its digital fingerprint.
– Links the blocks together within the chain, with each block connected to the hash of the previous and next blocks, making it virtually impossible to alter a block without detection.

4. **Timestamp**:
This is the date and time at which a transaction is carried out within the chain. The timestamp is associated with the hash, giving each transaction on the blockchain a unique chronological marker. Together, the timestamp and the hash enhance the security and accuracy of the information transferred on the blockchain, thereby increasing trust in its ability to securely store, preserve, and protect data[xiii].

Second Branch: Characteristics of Blockchain

Blockchain possesses three main characteristics:

1. **Open Ledger**:

This feature refers to the availability of the information stored on the blockchain—regardless of its type—to all users, promoting a high degree of transparency. While everyone can view the assets, users' real identities remain concealed, as the system allows individuals to use pseudonyms instead of their actual names.

2. **Distributed Ledger**:

A distributed ledger means the network operates in a decentralized manner. Copies of the stored data and information are distributed across all devices connected to the network. Therefore, no single entity controls the blockchain. Altering any data requires making the same change across all linked blocks through the sequential hash, greatly enhancing security[xiv].

3. **Mining**:
Mining is the process of verifying transactions by consensus[xv] and adding them to the blockchain ledger. This is achieved by performing complex mathematical operations to determine the transaction's specific hash. Miners around the world use specialized computer devices (nodes) to perform these calculations and obtain the correct hash that links the new transaction to the previous one in the chain.

Third Branch: Types of Blockchain

Blockchain is generally categorized into three types:

- **Public Blockchain**
- **Private Blockchain**
- **Hybrid Blockchain**

1. Public Blockchain

public blockchain is a decentralized network that is not controlled or managed by any single entity. It is open to everyone, allowing any individual to join and participate in verifying and broadcasting transactions without prior permission[xvi]. In this type of blockchain, all participants share in publishing transactions to a common ledger and benefit from the security provided by a large number of participants who protect the network against breaches. Public blockchains are primarily used for exchanging and mining cryptocurrencies such as Bitcoin, Ethereum, and Litecoin.

2. Private Blockchain

A private blockchain is a network that is not accessible to everyone and can only be joined with the permission of the entity responsible for managing it. This entity—whether governmental or non-governmental—oversees its administration. The process of verifying data is handled by a limited number of participants. Private blockchains are distinguished by their alignment with legal accountability and governance requirements, as they operate in a centralized and organized internal manner[xvii].

3. Hybrid Blockchain

The hybrid blockchain combines features of both public and private blockchains. For example, institutions like banks and financial organizations establish an internal private network to control sensitive data while allowing for limited interaction with a public blockchain. In hybrid networks, sensitive data (such as employee records) is stored in a closed environment, while some contracts or portions of data are made publicly accessible and verifiable through smart contracts. This approach enables regulators to benefit from the transparency of public blockchains in specific areas without compromising the confidentiality of internal information, making it suitable for sectors that require private ownership alongside flexible public data sharing[xviii].

Branch Two: Mechanism of Smart Contract Formation

The process of forming smart contracts involves specific stages that integrate legal and technical aspects, relying on the decentralized Blockchain mechanism. The main stages, supported by academic references, are as follows:

Stage One: Agreement on Terms

This stage involves the parties defining the fundamental terms of the contract and giving initial approval. These terms are written in natural language using a clear format that facilitates their later encoding. The terms may be drafted by one party, requiring the other party's consent through digital means such as the Internet and modern communication applications[xix].

Stage Two: Coding

Once the terms are agreed upon, they are converted into a programming code using a language specialized for smart contracts, such as Solidity used on the Ethereum network. These contractual terms are translated into executable program commands, ensuring that the contract is performed automatically without intermediaries, in full compliance with the agreed-upon obligations and requirements[xx].

Stage Three: Deployment

After the programming code of the smart contract is finalized, it is deployed on a specialized Blockchain platform, such as the Ethereum network. Distributed nodes across the network review and verify the programming code before storing it permanently and immutably within the blockchain[xxi].

Stage Four: Execution and Processing

When the agreed-upon conditions in the contract are met, the nodes automatically execute the programming code without human intervention. For instance, if the contract stipulates that one party must transfer ownership of a specific asset or deliver a defined service, the smart contract verifies the fulfillment of these conditions. Once verified, the corresponding obligation—such as transferring a payment to the other party—is executed automatically. This process ensures precise and swift execution, while maintaining necessary administrative safeguards[xxii].

Section Two: The Legal Approach to Smart Contracts

First Requirement: Characteristics of Smart Contracts

Smart contracts are distinguished from other types of contracts by the following characteristics:

Branch One: Electronic and Conditional Nature

Smart contracts are characterized by their exclusive execution through computer systems. They consist of codes or scripts that outline the obligations of the contracting parties in preparation for execution. The parties are bound by a digital agreement, meaning that the contract aims "to execute the negotiation or contractual obligations, and it does not take effect unless the execution requirements are fulfilled." The electronic format in smart contracts may appear in digital assets, such as cryptocurrencies, or in the digital representations of assets whose ownership is recorded on the blockchain. By their very nature, smart contracts require the use of digital electronic signatures and rely on encryption technology[xxiii].

Additionally, smart contracts are inherently conditional agreements, where their execution is suspended until the occurrence of certain predefined future events or conditions. These conditions are written in a programming language such as *Solidity*, commonly used on the Ethereum network, and are executed automatically without human intervention once fulfilled. For instance, in the context of a car loan guarantee, if a specified payment is not received by a predetermined date, the car may then be repossessed. This rule forms the basis for the execution of a smart contract, meaning that the contract becomes effective from the moment it is concluded[xxiv].

These conditions or events may also rely on external data—such as currency exchange rates or weather conditions—which are integrated into the contract through the **Oracle** technology to determine whether the specified condition has been met.

Branch Two: Self-Verification and Self-Execution

Self-verification is one of the most prominent features of smart contracts. The contract automatically verifies whether the programmed contractual conditions have been fulfilled through the blockchain network[xxv]. This process operates in a decentralized manner according to consensus mechanisms, without relying on the will of the parties or any central authority for oversight.

Smart contracts also feature **self-execution**, meaning that the contract is automatically carried out once the required conditions or events are met, without human intervention or mediation[xxvi]. This is governed by the principle of **determinism**, where execution is irreversible and cannot be delayed or modified. For instance, whereas a traditional real estate sales contract requires the involvement of a municipal authority or land registry, and financial transactions typically require a bank or financial institution as an intermediary, in smart contracts, there is no need for such intermediaries. Once a smart contract is concluded on a blockchain platform, algorithms autonomously and securely execute the terms and conditions of the agreement.

Branch Three: Immutability

Once a smart contract is executed automatically according to its code, it is deployed onto the blockchain. The code is stored in an **immutable ledger**, preventing any subsequent modifications.

Branch Four: Specialized Language and Automated Notarization

Smart contracts use specialized programming languages, primarily **Solidity** on the Ethereum platform, which serve as precise software intermediaries expressing the contract's terms explicitly through conditional "if…then" statements that leave no room for interpretation. This contrasts with traditional contracts, which are written in natural language and may be subject to ambiguity.

The **notarization of a smart contract** occurs automatically without human or third-party intervention. The contract's execution is distributed and viewable across all devices participating in the blockchain network. The encrypted nature of this process makes alteration, forgery, or destruction extremely difficult.

By contrast, traditional contract notarization relies on signed paper documents or notarized certificates and the registration of contracts with land registries or other authorities, often maintained in paper records[xxvii].

Accordingly, **proof of obligation** in smart contracts relies on the blockchain's immutable ledger, which guarantees the documentation of every invocation and a timestamp. Traditional contracts, however, depend on paper originals or commonly accepted electronic signatures.

Second Requirement: The Legal Nature of Smart Contracts

Scholars have differed regarding the legal nature of smart contracts. These differing views can be presented through the following subsections:

Branch One: The Nature of Smart Contracts in French Doctrine

French legal scholars have expressed divergent opinions about the legal nature of smart contracts. Some scholars, including Bruno Dondero, argue that the smart contract constitutes an autonomous contract according to its legal definition and is integrated into the blockchain platform[xxviii].

However, the majority of French jurists, including Mustapha Mekki and Christophe Roda, express skepticism about its classification as a contract in the traditional legal sense. They view the smart contract merely as a technological tool that executes previously agreed-upon terms within a computer program. This perspective holds that a pre-existing traditional contract is necessary to regulate the relationship between the parties[xxix].

Branch Two: The Nature of Smart Contracts in American Doctrine

American legal scholarship also reflects differing views regarding the nature of smart contracts. Some scholars consider the smart contract to be a genuine contract. This is reflected in the position of U.S. lawmakers, particularly recent legislation in the State of Nevada, which explicitly recognizes smart contracts as legally binding agreements stored in an electronic format, in accordance with applicable laws.

Conversely, other scholars argue that a smart contract does not constitute a contract in the legal sense. Among them is Nick Szabo, the creator of the concept, who views it merely as an informational framework designed to modernize the traditional concept of a contract. According to this view, a smart contract involves embedding pre-agreed contractual clauses into a digital framework. Thus, the smart contract, as an information technology tool, serves as a companion to the classic contract and operates based on a conditional approach—specifically, the “If this... then that” logic. This approach must be governed by actual conditions that stipulate both the requirements for its implementation and the resulting legal effects[xxx].

Section Three: Advantages and Disadvantages of Smart Contracts

Smart contracts have witnessed widespread adoption across various societies and have become the focus of attention and scrutiny by stakeholders in diverse sectors—including economic, financial, administrative, and even governmental domains—due to their revolutionary role in reshaping traditional operational mechanisms in these sectors. Accordingly, this chapter will explore, through two sections, the advantages that have attracted such keen interest and the disadvantages that have prompted continued observation and evaluation.

First Requirement: Advantages of Smart Contracts

Smart contracts offer numerous advantages, outlined as follows:

Branch One: Automation

Automation is one of the most prominent advantages of smart contracts. Through blockchain technology, contract terms can be executed automatically once the programmed conditions are met, eliminating the need for human intervention or external mediation. Automation enables the smart contract to operate autonomously once specific inputs or pre-defined conditions are satisfied, thus facilitating the execution of agreements—such as transferring funds upon the delivery of a service or product[xxxi].

Branch Two: Reduced Costs

The absence of intermediaries such as lawyers and notaries in smart contracts results in significant cost savings during the contracting process. By automating the review and execution of contracts, smart contracts eliminate the need for human intervention in tasks such as reviewing documents submitted by the contracting parties and drafting the contract—processes that traditionally incur financial costs.

Branch Three: Transparency and Trust

Smart contracts, through blockchain technology, provide **transparency** by allowing all parties involved to view and verify the terms, conditions, and documentation immediately upon their placement on the platform[xxxii]. Once the contract is executed, the results are permanently recorded. This transparency fosters **trust** among the parties, ensuring that the contract terms are clear, immutable, and tamper-proof.

Branch Four: Accuracy and Security

Thanks to blockchain technology, smart contracts execute automatically based on predefined conditions, guaranteeing precise and faithful implementation of the agreed-upon terms. The data storage and preservation functions of blockchain offer a high level of **security**, supported by its decentralized nature. This means that no single entity controls data verification. Data is securely stored in an encrypted ledger, making it resistant to hacking and tampering[xxxiii].

Branch Five: Speed

The automation provided by blockchain technology significantly reduces timelines and deadlines. Smart contracts execute much faster than traditional contracts because they eliminate the time and effort involved in manually processing paper documents and do not require intermediaries or third parties—such as agents or consultants[xxxiv].

Some critics of promoting smart contracts argue that the blockchain system does not entirely eliminate the intermediary role but rather redefines it within a digital framework that is open and accessible to all. In other words, the intermediary—though transformed—still exists within the blockchain system itself[xxxv].

Branch Six: Instant Enforcement

By their very nature, smart contracts enforce compliance automatically. Once the predefined conditions of the contract are met, the contract executes itself, ensuring that all parties fulfill their obligations. The system, being automated and non-negotiable, operates on a fixed algorithm based on the principle: *"If this... then that."* This ensures the **inevitability** of completing the contractual transaction and eliminates concerns about potential delays, non-performance, or interference by either party, as execution is beyond the control of both parties[xxxvi].

Branch Seven: Environmental Impact

Smart contracts reduce paper usage by eliminating the need for physical documents. This not only lowers costs but also has a positive environmental impact by promoting more sustainable practices and reducing the carbon emissions associated with the global paper industry.

Branch Eight: Scalability

Smart contracts can be easily scaled to handle large volumes of transactions. Traditional contract management can become cumbersome and inefficient as contract volumes grow. However, smart contracts can process thousands of transactions simultaneously, making them ideal for businesses that require the management of a high volume of contracts.

Branch Nine: Interoperability

Smart contracts can interact with other smart contracts and systems, enabling seamless integration with various platforms and technologies. This interoperability allows businesses to create complex, multi-party agreements that can be executed automatically without manual coordination.

Second Requirement: Disadvantages of Smart Contracts

Despite the many promising aspects of smart contracts, they are not without shortcomings—leading many stakeholders to remain cautiously optimistic about their future. These disadvantages include the following:

Subsection One: Security Vulnerabilities and Breaches

Smart contracts are known to be susceptible to **security vulnerabilities and breaches**. Vulnerabilities may allow malicious exploitation, enabling the system to produce outcomes contrary to its intended objectives—commonly known as "exploits." Not all vulnerabilities stem from coding errors, and not every coding error can be exploited.

Conversely, attacks and breaches targeting smart contracts, often resulting in financial losses, are common occurrences. Among these, **reentrancy attacks** became infamous after the 2016 attack on the DAO (Decentralized Autonomous Organization), which caused losses exceeding USD 60 million. Since then, attacks on smart contracts have become recurrent phenomena, frequently causing significant financial damage[xxxvii].

Subsection Two: Lack of Flexibility and Difficulty in Modification

Once a smart contract is deployed on the blockchain, it cannot be modified—even if errors are discovered or conditions change. Making amendments requires creating an entirely new contract, which increases both costs and complexity. The immutability of smart contracts is seen as a disadvantage because programming errors and vulnerabilities are almost inevitable. However, introducing the ability to modify smart contracts would undermine their core purpose—by making them "changeable," and thus less reliable.

Subsection Three: Dependence on External Data (Oracles)

A further drawback is the reliance of smart contracts on **oracle systems** as data sources. Oracles supply data but neither generate nor verify its accuracy. Consequently, if the data provided by an oracle is incorrect or compromised, the smart contract may operate erroneously. In other words, even if the smart contract itself is 100% secure, it can be manipulated by feeding it false information via the oracle system. This vulnerability is commonly referred to as **"The Oracle Problem."**[xxxviii]

Subsection Four: Legal and Regulatory Challenges

A major issue lies in the lack of **legal clarity** regarding liability in cases of programming errors or fraud, making it difficult to identify responsible parties. Furthermore, smart contracts often conflict with numerous local laws, as many jurisdictions are still reluctant to recognize them and, in some sectors, even criminalize their use. There is also uncertainty regarding their admissibility as evidence in legal disputes.

Branch Five: Technical Complexity and Coding Errors

The design of smart contracts requires advanced programming skills and a deep understanding of blockchain technology. Any programming error can prevent the contract from achieving its intended purpose. Smart contracts execute precisely what is coded—not necessarily what the programmer or contracting parties intended. In other words, if the code contains faulty logic or programming that does not align with the original intent, it can result in incorrect or unexpected outcomes[xxxix].

Moreover, smart contract programming languages are often Turing-complete, offering high flexibility to accommodate numerous variables. However, this also makes them difficult to verify. Contracts may contain complex logical errors that lead to vulnerabilities and weaknesses. The extensive expressive power of such languages comes at the expense of security and predictability[xl].

Branch Six: High Costs

Drafting a smart contract requires specialized expertise in blockchain programming languages such as *Solidity* and others. Developers with experience in smart contracts typically command high fees due to the complexity of the task and the need to write secure, error-free code. The more complex the contract or the more special features it requires, the higher the associated costs. Furthermore, because errors in smart contracts cannot be corrected after deployment—as previously discussed—extensive security auditing is necessary prior to launch, which can be highly expensive depending on factors such as the code's length and complexity, the auditing firm's expertise, and the required assurance level[xli].

Branch Seven: Consensus and Timestamp Challenges

Consensus mechanisms and timestamping are two critical pillars ensuring the reliability and security of smart contract execution. However, both present shortcomings that may negatively impact the trustworthiness of smart contracts.

Regarding consensus, issues such as temporary forks, chain reorganizations, and even 51% attacks may arise, potentially reversing or canceling contract executions after they have ostensibly been completed[xlii].

Regarding timestamps, miners can manipulate timestamps within a limited range, enabling attacks that exploit *block.timestamp* to inflate or disrupt time-sensitive functions. Additionally, the accuracy of timestamps can be affected by network delays and the lack of full synchronization across the blockchain[xliii].

Conclusion

Upon concluding this study, several key findings have emerged:

1. Smart contracts are digital programs capable of automatically executing contractual terms. They reduce the need for intermediaries and enhance trust through verifiable, tamper-resistant records.

2. The blockchain serves as the infrastructure for smart contracts, leveraging its capabilities to transform legal agreements into automated actions without the need for a mediator.

3. Blockchain operates as a decentralized, distributed digital ledger for all transactions across a peer-to-peer network of connected devices. This provides a high degree of transparency, party autonomy, and speed in performance.

4. The legal nature of smart contracts remains under debate among legal scholars. Some consider them to be genuine contracts, while others view them as technological tools that do not fully align with the rules and principles of traditional contract theory. This divergence arises from differing interpretations of the electronic environment in which they operate and their unique features—including self-verification, automatic execution, conditional logic, specialized programming language, and automated notarization.

Additionally, issues such as unclear liability for programming errors and the lack of robust legal frameworks in many jurisdictions contribute to ongoing uncertainty.

References

[i] Nick Szabo, Smart Contracts: Formalizing and Securing Public Networks, First Monday, sept. 1997, n° 9.

[ii] Halouani, Maher. Blockchain, digital currencies, and international financial law: A critical academic study of Bitcoin and digital currencies. Dar Tweeta for Publishing and Distribution, Egypt, 2018, p. 71.

[iii] https://blog.b9lab.com/the-dao-hack-in-eight-minutes-94919018692d9d

[iv] Desbonnet, J. & Vanunu, O. (2024). The rise of smart contracts and strategies for mitigating cyber and legal risks. In World Economic Forum. Retrieved from https://www.weforum.org/stories/2024/07/smart-contracts-technology-cybersecurity-legal-risks

[v] Schär, F. (2021). Decentralized finance: on blockchain and smart contract-based financial markets. Review of the Federal Reserve Bank of St Louis, 103(2), 153–174.

[vi] Nick Szabo, Smart Contracts: Formalizing and Securing Public Networks, First Monday, sept. 1997, n° 9.

[vii] Alketbi, W., Nasir, Q., & Al-Kuwaiti, A. (2023). Smart Contracts in Blockchain Technology: A Critical Review. Information, 14(2), 117. MDPI.

[viii] https://www.investopedia.com/terms/s/smart-contracts.asp.

[ix] J. Grimmelmann, All Smart Contracts Are Ambiguous, January 14, 2019, Penn Journal of Law and Innovation) Forthcoming (; Cornell Legal Studies Research Paper n° 19-20. P.-J. Benghozi, Blockchain: objet à réguler ou outil pour réguler? JCP E, 2017, n° 36, p. 1470.

[x] S. Sayadi, S.Ben Rejeb and Z. Choukair, "Blockchain Challenges and Security Schemes: A Survey", Seventh International Conference on Communications and Networking (ComNet), Hammamet, Tunisia, IEEE, 1-3 Nov. 2018

[xi] Hassane, Hossam Eddine Mahmoud Mohamed. Smart contracts concluded via blockchain technology: What challenges for the current contract system? The Legal Journal, Vol. 16, No. 1, 2023, p. 20.

[xii] Warsama Ghaleb, Abdelkader. Blockchain and the development of legal systems. Published in International Journal of Islamic Economics, No. 81, February 2019, p. 75.

[xiii] Warsama Ghaleb, Abdelkader. Op. cit., p. 75.

[xiv] Hassane, Hossam Eddine Mahmoud Mohamed. Op. cit., p. 22

[xv] The Consensus Mechanism is the protocol that ensures agreement among distributed nodes in a blockchain network on the validity of data added to the chain, without the need for a central authority. This mechanism is the cornerstone of the security and reliability of decentralized networks, preventing tampering and ensuring synchronization among all participants.

[xvi] Issa, Haitham El-Sayed Ahmed. Conclusion of smart contracts via blockchain technology. Journal of Legal and Economic Studies, Faculty of Law, Sadat City University, Vol. 7, 2021.

[xvii] Daoud Mansour, Smart Contracts Integrated in Blockchain: The Beginning of the End of Traditional Contracts, Algerian Journal of Legal and Political Sciences, Vol. 59, No. 1, 2022, p. 524.

[xviii] Merlinda Andoni et al., Blockchain Technology in the Energy Sector: A Systematic Review of Challenges and Opportunities, Renewable and Sustainable Energy Reviews, Vol. 100, 2019, p. 147.

[xix] Parizo, C., What Are the 4 Different Types of Blockchain Technology, in Blockchain for Businesses: The Ultimate Enterprise Guide, TechTarget, 2021.

[xx] Tanash Utamchandani Tulsidas, Smart Contracts from a Legal Perspective, Final Degree Project for the Bachelor of Law, University of Alicante, 2018, p. 9.

[xxi] Chris Dannen, Introducing Ethereum and Solidity (New York, US: Apress Media LLC, 2017), p. 2.

[xxii] Christian Sillaber and Bernhard Waltl, "Life Cycle of Smart Contracts in Blockchain Ecosystems", Datenschutz und Datensicherheit 41 (2017): p. 497-500: https://doi.org/10.1007/s11623-017-0819-7.

[xxiii] Hossam Eddine Mahmoud Mohamed Hassan, Previously Cited Reference, pp. 11–12.

[xxiv] Abdelrazak Wahba Sayed Ahmed Mohamed, The Concept of Smart Contracts from the Perspective of Civil Law: An Analytical Study, Journal of Economic, Administrative and Legal Sciences, National Research Center, Gaza, Vol. 5, No. 8, April 2020, p. 88.

[xxv] Haitham El-Sayed Ahmed Issa, The Emergence of Smart Contracts in the Blockchain Era, Dar Al-Nahda Al-Arabia, 1st Edition, 2021, p. 46.

[xxvi] Hossam Eddine Mahmoud Mohamed Hassan, Previously Cited Reference, p. 13.

[xxvii] Ahmed Saad Ali Al-Borai, The Formation and Execution of Transaction Contracts Between Traditional Methods, Blockchain Technology, and Smart Contracts: A Comparative Jurisprudential Study, Scientific Journal

of the Faculty of Islamic and Arabic Studies, Al-Azhar University, Part 4, No. 39, December 2020, pp. 2301–2302.

[xxviii] Bruno Dondero, Les Smart Contracts, in Le Droit civil à l'ère du numérique, LexisNexis, December 2017, p. 19.

[xxix] Moamer Ben Tria, Smart Contracts Integrated in Blockchain: What Challenges for the Current Contractual System?, Journal of Kuwait International Law School – Special Supplement – No. 4, Part 1, May 2019, p. 483.

[xxx] Nick Szabo, Smart Contracts: Formalizing and Securing Public Networks, First Monday, No. 9, September 1997.

[xxxi] See in more detail: Moamer Ben Tria, Previously Cited Reference, p. 480.

[xxxii] Ben Ali Saliha, Blockchain Technology as the Basis for Activating the Mechanism of Smart Contracts, Journal of Legal and Social Sciences, Vol. 7, No. 2, June 2020, p. 962.

[xxxiii] Hamsi Miloud & Mouna Meflati, Smart Contracts as a Mechanism for Remote Contracting, International Journal of Legal and Political Research, No. 2, October 2022, p. 142.

[xxxiv] Ghassan Salem Al-Talib, Digital Currencies and Their Relationship to Smart Contracts, Conference of the International Islamic Fiqh Academy, 24th Session, Organization of Islamic Cooperation, Dubai, 2019, p. 39.

[xxxv] Mohamed Irfan Al-Khatib, Smart Contracts... Credibility and Methodology: An In-Depth Critical Study in Philosophy and Foundation, Journal of Kuwait International Law School, Year 8, No. 2, Dubai, 2019, p. 163.

[xxxvi] Mohamed Irfan Al-Khatib, Previously Cited Reference, p. 164.

[xxxvii] Mik Eliza, Smart Contracts: Tales of Trust and Certainty. Technology and Regulation, 2022,pp 104-105.

[xxxviii] Mik Eliza, Smart Contracts: Tales of Trust and Certainty. Technology and Regulation, 2022, p 105.

[xxxix] Mik E, pp 104-105.

[xl] Massimo Bartoletti, et al, Verification of Recursive Bitcoin Contracts (2020) https://arxiv.org/abs/2011.14165 accessed 26 April 2025.

[xli] ( )-Rozario, A. M., & Vasarhelyi, M. A. (2018). Auditing with Smart Contracts. International Journal of Digital Accounting Research, 18.

[xlii] https://medium.com/%40RocketMeUpCybersecurity/understanding-blockchain-forks-security-risks-and-how-to-manage-them-e565bfd22368 accessed 27 April 2025.

[xliii] https://medium.com/coinmonks/smart-contract-security-block-timestamp-manipulation-baec1b95c921 accessed 27 April 2025.