



Contents lists available at ScienceDirect

Blockchain: Research and Applications

journal homepage: www.journals.elsevier.com/blockchain-research-and-applications

Research Article

Blockchain-enhanced hydrogen fuel production and distribution for sustainable energy management

Yash Madhwal^{a,*}, Yury Yanovich^{a,b}, Matteo Coveri^c, Ninoslav Marina^d^a Skolkovo Institute of Science and Technology, Moscow 121205, Russia^b Faculty of Computer Science, HSE University, Moscow 109028, Russia^c Energy Efficiency Lease AG, Zug 6300, Switzerland^d University of Information Science and Technology "St. Paul the Apostle", Ohrid 6000, Macedonia

ARTICLE INFO

Keywords:

Blockchain
CO₂
Hydrogen fuel
Renewable energy
Energy distribution
Energy transition
H₂ mobility
Offset certificates
Sustainable technology
Tokenization

ABSTRACT

Renewable energy projects, particularly wind and solar farms, have garnered significant attention as a potential solution to global energy challenges. Despite the energy production obstacles, the steady availability of fossil fuels continues to compete due to established distribution systems, as societies increasingly rely on electricity. Hydrogen fuel has emerged as a promising avenue for energy production, storage, and distribution, involving converting surplus renewable electricity into hydrogen through electrolysis, storing it, and distributing it. Our novel approach leverages blockchain technology to enhance the efficiency and traceability of hydrogen fuel production, offering a unique synergy of transparency, security, and decentralized governance. We showcase its viability and effectiveness using the ERC-1155 token standard to tokenize renewable resources and convert them into hydrogen fuel. Within our tokenized fuel blockchain architecture, we simulate the forecasted growth in hydrogen production and vehicle demand, highlighting our approach's efficiency, traceability, and transparency. This integration showcases the potential for a sustainable hydrogen fuel ecosystem. The gas consumption data analysis indicates that the daily gas consumption remains below 194 million Gas for refilling 3245 vehicles (including the cost of one-time contract deployment), demonstrating the feasibility and efficiency of our approach.

1. Introduction

The rise of renewable energy projects like wind and solar farms has gained global attention due to their potential to address energy challenges [1]. More than 130 countries worldwide have set a target to achieve net-zero emissions by 2050. The Paris Agreement, a legally binding international treaty on climate change, has prompted 110 of the 191 participants to submit updated national action plans to reduce emissions and provide financial assistance to more vulnerable nations [2]. The European Union (EU) has set emission reduction targets for its 27 members. In July 2021, the EU released proposals to revise its climate legislation, including an emissions trading system focused on carbon dioxide (CO₂), nitrous oxide (N₂O), and perfluorocarbons emissions in the energy, chemical, and aviation sectors. National plans are expected to submit reports on reducing emissions [3]. However, a significant challenge for effectively storing and utilizing intermittent energy production remains [4]. As society increasingly adopts electricity for various appli-

cations, fossil fuels like coal, oil, and gas remain competitive due to their ease of storage and established distribution infrastructure [5,6]. The contrast between the intermittency of renewable energy and the constant availability of fossil fuels highlights the need for innovative and sustainable energy storage solutions.

Hydrogen is the most promising clean energy source for realizing low-carbon energy transformation in the 21st century [7]. Hydrogen Council forecasted that the global demand for hydrogen energy is projected to reach over 18% of the terminal energy demand by 2050, with the hydrogen market valued at over USD 2.5 trillion [8]. Incorporating hydrogen storage methods with Fuel Cell technology is a promising avenue for energy production, enhancing future material development and manufacturing processes [9–11]. The strategy entails the conversion of excess renewable electricity into hydrogen via electrolysis, subsequent storage, and eventual utilization through Fuel Cells to reconvey the stored hydrogen into electricity during peak demand [12,13]. This approach aims to bridge the temporal disparity between sporadic renew-

* Corresponding author.

E-mail addresses: yash.madhwal@skoltech.ru (Y. Madhwal), y.yanovich@skoltech.ru (Y. Yanovich), mcoveri@eelease.com (M. Coveri), ninoslav.marina@uist.edu.mk (N. Marina).

<https://doi.org/10.1016/j.bcr.2024.100229>

Received 5 October 2023; Received in revised form 9 June 2024; Accepted 8 August 2024

Available online 13 August 2024

2096-7209/© 2024 THE AUTHORS. Published by Elsevier B.V. on behalf of Zhejiang University Press. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

able energy generation and consistent energy consumption, thereby curtailing wastage and enhancing the efficacy of renewable energy sources as a steadfast and dependable energy reservoir.

The intersection of this energy paradigm with blockchain technology presents a distinctive synergy. The qualities intrinsic to the blockchain, including transparency, cryptographic security, and decentralized governance, have the potential to amplify the efficiency and reliability of this multifaceted energy framework [14–17]. Deploying blockchain-driven smart contracts can automate secure transactions encompassing storing energy acquisition, distribution, and sale [18]. Furthermore, blockchain inherent distributive ledger attributes can facilitate meticulous traceability of energy fluxes, ensuring precise bookkeeping and alleviating fraudulent practices [19–21]. The amalgamation of hydrogen storage, Fuel cell technology, and blockchain systems is poised to inaugurate a seminal era in the energy sector, potentially ushering in a paradigm shift toward a more ecologically sustainable and progressively viable energy landscape.

This paper contributes to the process of hydrogen fuel production by proposing a novel approach that leverages blockchain technology to enhance the efficiency and traceability of the production process. Through blockchain integration, we introduce a tokenization mechanism that represents these water and electrical resources as tokens. We address the hydrogen production process using electrolysis, enabling secure and transparent transfer and creating new hydrogen tokens distributed to designated filling stations, facilitating a seamless and auditable distribution of hydrogen fuel to vehicles.

Furthermore, our proposed tokenized fuel blockchain architecture is used to conduct a simulation-based analysis of the forecasted growth in hydrogen fuel production, distribution, and corresponding demand for the vehicle. This simulation is conducted within the framework of our proposed tokenized fuel blockchain architecture, allowing us to quantify the required production volume and resource allocation to meet future demands. By running a simulated scenario, we demonstrate the feasibility and effectiveness of our approach in ensuring a sustainable and well-regulated hydrogen fuel production and distribution ecosystem while meeting the requirements of hydrogen-produced vehicles. Our research contributes to advancing hydrogen fuel technologies in a blockchain-enabled paradigm.

This paper is organized as follows: a comprehensive literature review (Section 2), an overview of blockchain technology and related information to the architecture (Section 3), blockchain implementation using the presented methodology (Section 4), a case study and application development (Section 5), a simulation demonstrating the effectiveness of this method (Section 6), and a discussion of this work and potential future works (Section 7). Finally, Section 8 provides concluding remarks of the paper.

2. Literature review

This section presents a comprehensive literature review that synthesizes and analyzes existing knowledge of the subject matter. It explores various scholarly works, research articles, and relevant publications. The section is divided into two distinct parts. The first part examines the energy conversion and distribution landscape literature, providing valuable insights into current research in this domain. The second part investigates the domain of blockchain-based tokenization in supply chain assets, drawing parallelization to the hydrogen fuel production process for our current research. This segmented approach ensures a thorough exploration of the underlying themes while facilitating a systematic understanding of the reviewed literature.

2.1. Blockchain and sustainable fuel development

Andoni et al. [22] comprehensively explored the applicability of blockchain technology in the energy industry. It evaluates ongoing business cases, categorizes initiatives based on their field, platform, and

consensus approach, and analyzes potential drawbacks and technical challenges. The review also provides an overview of current research, industrial projects, and startups in various scenarios. It concludes by addressing challenges and market barriers that blockchain must overcome to establish its commercial viability and broader adoption. Wang and Su [23] reviewed the intersection of blockchain and energy using visual bibliometric analysis and the Scopus database from 2014 to 2020. The study identifies increased publications related to blockchain in the energy sector, particularly since 2018, suggesting a growing research domain. Developing countries are emerging as key contributors to energy blockchain research, focusing on renewable energy. Blockchain technology is expected to promote renewable energy advancement and energy sustainability. The paper examines 783 articles in the energy blockchain field, evaluating global and national perspectives. Empirical evidence supports the positive impact of energy blockchain on renewable energy development, with a surge in related literature since 2017.

Barcelo et al. [24] explored how blockchain technology can be integrated into the energy sector, offering the benefits of transparency, security, and efficiency. The combination of blockchain and renewable energy can lead to innovative business solutions. This study analyzes the evolving landscape of blockchain in the energy sector. Blockchain can help overcome challenges and promote energy independence by focusing on renewable energy. The study also highlights the potential for secure energy transactions with decentralized trading systems. Mould et al. [25] explored the potential benefits of combining certification and blockchain technology to ensure that gases such as hydrogen and biogas originate from environmentally friendly sources. This integration could lead to a more sustainable future. Despite blockchain's energy-intensive consensus mechanisms, they are being redefined to promote energy efficiency with renewables, making them a viable addition to the energy industry. By integrating blockchain with certificates, this technology can go beyond transportation and foster standards and consumer awareness. Hydrogen and biogas are emerging as promising renewable fuels.

Wang et al. [26] presented a new approach to pricing hydrogen fuel at renewable electricity filling stations. The proposed method involves a two-way decentralized trading system integrating real-time information to improve profits and energy efficiency. Through cross-chain interoperability, the approach is made more effective. A case study shows that the dynamic pricing approach is more advantageous than a constant pricing model. The article also presents a blockchain-based decentralized application (DApp) demonstrating how this approach can be implemented in electric-hydrogen coupling systems. By eliminating third-party intermediaries, the approach reduces operating costs and promotes credible transactions, leading to a more vibrant market. This new approach has the potential for wider applicability and further research in comprehensive energy-demand studies.

2.2. Tokenization

In supply chain management, tokenization refers to creating unique digital tokens on the blockchain to represent physical or digital assets. These tokens act as digital versions of real-world assets and commodities, such as products, resources, or materials, and are assigned a specific digital identity that is securely recorded on the blockchain. Tokenization offers multiple advantages for supply chains. It allows for precise tracking of each asset's origin and movement, increasing transparency and decreasing the risk of fraud or counterfeiting. Additionally, tokenization simplifies and automates various supply chain processes to prove operational efficiency. Madhwal [27] proposed using a private permissioned blockchain with tokenization to permanently record transaction data by establishing a transparent supply chain network, fostering trust at all levels and mitigating corrupt practices and product counterfeiting. Harish et al. [28] presented a system that uses blockchain technology to improve traceability in logistics financing through digital asset tokenization, aiming to enhance visibility and traceability in the physical and digital aspects of supply chain transactions. The primary objec-

tive suggests a blockchain-powered solution that utilizes digital asset tokenization to create unique blockchain tokens, resulting in more precise traceability in logistics financing. It has also explored integrating this system with stakeholder devices, privacy concerns, and practical implementation. The proposed system facilitates the sharing, tracking, and tracing of digital assets in logistics financing. Madhwal et al. [29] presented a new architecture for representing real-world production processes using a tokenized directed acyclic hypergraph (DAG). The proposed approach involves using smart contracts for various entities in the supply chain and managing tokens representing products in their possession. Token standards ERC-1155 simplify integration processes, enabling efficient tracking and assessment of contractor reliability. This approach combines blockchain data integrity with clear representation of tokenization products and their states, fostering a trustworthy and efficient supply chain ecosystem.

After thorough research, we suggest implementing a tokenization system for hydroproduction in electric-hydrogen systems, specifically for hydrogen filling stations. This model involves a mechanism for tokenizing renewable energy sources and a method for converting hydrogen. Utilizing blockchain technology can ensure the smooth and efficient operation of this approach.

3. Background

Blockchain technology, a decentralized system facilitating secure and transparent transactions without intermediaries [30], employs a digital ledger distributed among various entities or nodes, each maintaining an immutable record copy [31]. Its distributed nature eliminates central points of failure or vulnerability [32], ensuring high resistance to hacking, fraud, and manipulation. Transactions within a block are organized into pairs to form a Merkle root hash, computed using a cryptographic hash function like SHA256 [33]. Originating from Satoshi Nakamoto's white paper on Bitcoin [33], blockchain technology extends to manage cryptocurrencies like Ethereum [34], operating as a peer-to-peer (P2P) network without central authority. Each node independently updates the ledger, creates new blocks, and broadcasts them to others using the gossip protocol [35]. Various consensus algorithms, such as proof-of-work (PoW), proof-of-stake (PoS), or Byzantine fault-tolerant, verify block validity [36–39]. Beyond cryptocurrency, blockchain finds applications in diverse domains like supply chain management, healthcare, and real estate [19,40,41], enhancing logistics by ensuring responsible and sustainable goods movement [19,42].

Ethereum [34], the second-largest blockchain platform following Bitcoin, provides a robust ecosystem for building DApps on its decentralized infrastructure without a central point of control. Central to its functionality is the Ethereum virtual machine (EVM), an essential component within every Ethereum node, which facilitates the execution of arbitrary bytecode and ensures deterministic outcomes in a trustless environment [43]. Deploying Ethereum clients like Geth or Parity enables transaction synchronization, validation, and execution, with the EVM's Turing completeness allowing for the execution of any conceivable algorithm. Smart contracts deployed within the network interact via network explorers or DApps [44,45]. DApps operate on a P2P blockchain network, are open-source, and support P2P interactions [46]. While network explorers serve as conduits for blockchain transactions, DApps interact through clients like Geth and OpenEthereum, although they may face increased complexity due to storage demands. Third-party platforms like Infura and Alchemy [47,48] provide APIs to streamline Ethereum blockchain interactions. Blockchain token standards, such as ERC-20, ERC-721, and ERC-1155, define rules and protocols for token operation within an ecosystem, ensuring interoperability and consistency [49–51]. These standards facilitate the creation, management, and interaction of both fungible and non-fungible tokens within a single smart contract [52,53].

4. Blockchain implementation

4.1. Tokenized representation of fuel

ERC-1155 is a token standard on the Ethereum blockchain that enables the creation of a single contract capable of managing multiple types of tokens. Tokens are organized into classes, each representing a particular asset type. These classes can be either fungible or non-fungible. ERC-1155 reduces the overhead of deploying individual contracts for each type, saving computational resources and gas fees for developers and users. Various items in the supply chain can be represented using tokens, with fungible tokens representing quantities like raw materials or components, while non-fungible tokens represent unique items like finished products or high-value assets. Fig. 1 shows the fuel production and distribution supply chain. Different system parts are converted into tokens to make fuel generation easier. When the conditions for fuel generation are met, a new token is created to represent the produced fuel.

4.2. System design

The concept of tokenization is fundamental to our approach. We leverage the ERC-1155 standard to facilitate hydrogen fuel production. A precise combination of resources is necessary to generate 1 kg of hydrogen fuel, i.e., 55 kWh of electricity and 10.9106 liters of water [54,55], procured exclusively from renewable sources. By implementing ERC-1155 standards, we intend to establish an indexed representation of water, electricity, and hydrogen quantities. This approach ensures that the minting of hydrogen tokens is intrinsically linked to the availability of corresponding balances of water and electricity from the designated renewable sources. Utilizing this framework, we aim to create a transparent and traceable process for hydrogen fuel production, effectively tokenizing the production process to align with sustainable and accountable practices. The smart contract is designed to execute these specific actions.

The system has five main roles:

1. **Administrator:** To effectively manage access levels, the administrator can authorize and define three roles of the user's wallet: Water Supplier/Grid, Electricity Supplier/Grid, and Fuel (Hydrogen) Generator. As the smart contract owner, the administrator can assign roles and transfer ownership to another wallet address, allowing for role transitions if needed.
2. **Water Grid:** The water grid manages and facilitates water-related activities essential for producing hydrogen fuel. The "Mint" action records the amount of water drawn from a designated grid and inputs this information into the smart contract. The "Transfer" action allows the water plant to transfer a quantity of water (measured in liters) within its storage capacity to the hydrogen producer station, enabling the subsequent hydrogen production process. The water grid is crucial in maintaining accurate water usage records and facilitating water resource movement for sustainable hydrogen fuel creation.
3. **Electricity Grid:** The electricity grid manages and facilitates electricity-related operations crucial for hydrogen fuel production. The "Mint" action records the amount of electricity the plant generates and inputs the data into relevant smart contracts, focusing on renewable sources like solar and wind. The "Transfer" action enables the plant to transfer a specific amount of electricity (measured in kilowatt-hours) to the hydrogen producer station, powering the hydrogen production process. The plant uses smart contracts to accurately transfer electricity to the intended station, much like the water station. Ultimately, the electricity plant's role is to document electricity generation accurately and distribute it efficiently and in an environmentally friendly way for hydrogen fuel production.

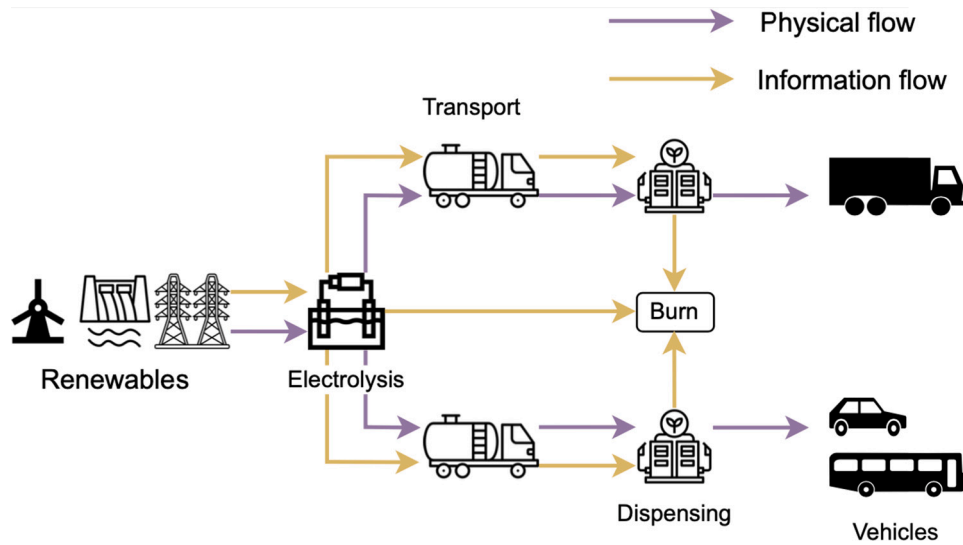


Fig. 1. Tokenized supply chain of fuel production and distribution.

4. **Hydrogen Producer:** The hydrogen plant effectively manages and facilitates all hydrogen production operations. The system administrator assigns specific wallets to carry out two crucial activities. Firstly, “Mint H₂” calculates and records hydrogen production using available water and electricity. Secondly, “Transfer” moves the produced hydrogen fuel to the filling station, maintaining the fuel balance.
5. **Fuel Station:** The fuel station carries out vehicle refills by the predefined requirements. The system has established specific filling amounts for different vehicle types, including cars, buses, and trucks.

4.2.1. Establishing access

Entities are created to grant access to the hydrogen production system. In order to gain access, an entity must input its wallet address. However, only the smart contract admin is authorized to execute transactions. If an attempt is made to execute a transaction as a non-admin, the smart contract will throw an error, and the transaction will fail to execute.

4.2.2. Production

A designated entity must have resources to produce hydrogen fuel, and it can adjust the amount according to specific demands. The entity responsible for fuel production is responsible for making any necessary adjustments. If the sender's address does not match the entity's address, a transaction reversion process is triggered, and an authorization error occurs, indicating that the sender is not authorized to proceed with the transaction. The entity's resource utilization depends upon the quantity of water extracted or electricity produced from the water and electricity grids, and this utilization is tokenized. The corresponding amounts are then attributed to the respective wallets and recorded onto the blockchain.

4.2.3. Data transfer

In production and distribution, entities can transfer inventory digitally and record these transactions on the blockchain. However, only the responsible entity is authorized to initiate transfer actions. Any attempt by an unauthorized entity to initiate such actions will result in a revert error, indicating a lack of authorization.

Note: There is a difference in timing between digital and physical transfers. Digital transfers happen immediately, assuming the time of transaction commitment to the blockchain is negligible compared with transportation, while physical transfers may experience delays because of the need for physical movement.

For more information on this situation, please refer to Section 7.

4.2.4. Fuel production

$$\begin{aligned} \text{maxWater} &= \text{_balanceWater} / \text{waterRequired} \\ \text{maxEnergy} &= \text{_balanceEnergy} / \text{electricityRequire} \\ \text{hydrogenProduction} &= \min(\text{maxWater}, \text{maxEnergy}) \end{aligned} \quad (1)$$

Algorithm 1 describes the fuel production process. The fuel production process algorithm utilizes ERC-1155 to manage token balances. It begins by verifying that the sender's address matches the authorized entity address. Next, it checks the available water and electricity balances in the hydrogen plant and compares them with the required amounts. If there are sufficient balances, the algorithm calculates the maximum amount of hydrogen that can be produced based on the available resources. Eq. (1) shows the calculation of the maximum amount of hydrogen fuel that can be produced based on the available resources. However, if the available resources are limited, this will constrain the estimated hydrogen production from water and electricity resources.

To accurately track resources and maintain balance, the algorithm mints new tokens to record hydrogen production and burns the corresponding amount of water and electricity tokens. This automated approach aligns with the initial text description and provides a structured method for the hydrogen plant to calculate, produce, and account for hydrogen fuel based on available resources.

4.2.5. Fuel filling

In fueling vehicles, Algorithm 2 is essential for efficiency and precision. It verifies sender authorization to ensure that only authorized entities can execute fueling. Once authorized, the algorithm identifies different entities, such as cars, buses, and trucks, based on the input provided.

Building upon this, the algorithm describes how the system establishes predefined fueling requirements for each type of vehicle: 6 kg for cars, 23 kg for buses, and 50 kg for trucks [56–58]. These predefined requirements are ingrained in the system and are tailored to vehicles with empty fuel tanks. After each successful fueling operation, the system automatically adjusts the hydrogen fuel balance, enabling seamless future transactions. Depending on the underlying system architecture, the filling amount can be sourced from the smart contract or off-chain resources. This algorithm and textual explanation highlight the system's ability to precisely and effortlessly manage hydrogen fuel distribution across various vehicle types.

Algorithm 1 Fuel production.

Import: Setting Parameters, ERC-1155 Instance,
Input: address `_entity`, uint `_waterIndex`, uint `_electricityIndex`

```

1: ERC-1155 h2bal
2: dataString ← '0x0'
3: if (msg.sender ≠ _entity) then
4:   Revert with an authorization error
5: else
6:   waterBalance ← h2bal.balanceOf(msg.sender, _waterIndex)
7:   requiredWater ← waterRequirement
8:   electricityBalance ← h2bal.balanceOf(msg.sender, _electricityIndex)
9:   requiredElectricity ← electricityRequirement
10:  if (waterBalance < requiredWater) then
11:    Revert with "Insufficient Balance" error
12:  else if (electricityBalance < requiredElectricity) then
13:    Revert with "Insufficient Balance" error
14:  else
15:    maxWater ← waterBalance / requiredWater;
16:    maxEnergy ← electricityBalance / requiredElectricity
17:    fuelProduced ← min(maxWater, maxEnergy)
18:    waterConsumed ← fuelProduced · requiredWater
19:    electricityConsumed ← fuelProduced * requiredElectricity
    _mint(fuelProduced)
    _burn(waterConsumed)
    _burn(electricityConsumed)
20:  end if
21: end if

```

Algorithm 2 Fuel filling.

Input: Setting Parameters, ERC-1155 Instance,
Output: string `_entity`

```

1: if (msg.sender ≠ _entity) then
2:   Revert with an authorization error
3: else
4:   if (_entity = 'Car') then
5:     _burn(6)
6:   else if (_entity = 'Bus') then
7:     _burn(23)
8:   else if (_entity = 'Truck') then
9:     _burn(50)
10:  end if
11: end if

```

4.3. Security

The blockchain platform uses smart contracts to ensure reliable, transparent, and secure decentralized systems. Smart contracts are predefined rules and conditions based on algorithms that describe the system workflow. These algorithms maintain security by preventing unauthorized users' unintended actions or malicious activities. Transactions in blockchain networks require private key signatures to prove ownership and authorization, ensuring that only legitimate parties can initiate actions on the blockchain.

Smart contracts provide a secure and reliable system by ensuring consistent and tamper-proof execution based on workflows. This eliminates the risks associated with errors or fraudulent manipulation in the production process. Additionally, predefined requirements are automatically enforced, which adds more security. Integrating blockchain technology allows for an immutable record of all fueling transactions. Each transaction is documented on the blockchain, forming an audit trail that remains transparent and resistant to tampering. This transparency enhances accountability and traceability, simplifying the verification of the legitimacy of each fueling operation while thwarting unauthorized access or fraudulent activities. Integrating blockchain and smart contract technology adds security and trust to fuel filling.

5. Application prototype

The application is developed using a combination of programming languages, frameworks, and tools, consisting of two main components: a frontend and a blockchain-based backend. The backend consists of a smart contract deployed on the Binance smart chain (BSC) testnet [59], utilizing the EVM for executing smart contracts. The smart contract is developed in Solidity, a programming language designed explicitly for EVM-based smart contract development [60]. The Hardhat framework is employed to develop, test, and deploy the smart contract on the blockchain [61].

The frontend utilizes Vue3, an open-source JavaScript framework for creating user interfaces and applications [62]. The design aspect is accomplished using the utility-first CSS framework Tailwind CSS [63]. The frontend component interacts with the blockchain-based smart contract to facilitate data read and write operations, ensuring a seamless connection between the user interface and the underlying blockchain infrastructure.

The application is available at <https://hydrogenProduction.web.app/>.

5.1. Smart contract: development

The smart contract is developed using the Solidity programming language based on the ERC-1155 token standard.

5.2. Smart contract: testing

The Hardhat network offers a local blockchain environment for development and testing purposes. Automated smart contract testing is executed by implementing predefined test cases that replicate the simulated workflow of the use case. These tests are conducted using the ethers.js library, enabling interaction with the smart contract and ensuring a comprehensive evaluation of its functionality.

The rigorous testing of the smart contracts is carried out into different categories.

5.2.1. Smart contract testing: basic

On deployments, the following basic functionalities of the smart contract are tested:

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
contracts/ HydrogenProductions.sol	100 100	79.17 79.17	100 100	100 100	
All files	100	79.17	100	100	

Fig. 2. Solidity code coverage.

- Basic Contract Deployment
 - ✓ Contract deployed address is not Null
 - ✓ WATER generator address is Null
 - ✓ Renewable generator address is Null
 - ✓ WATER generator address is Null
 - ✓ Decimal Value
- Initializing the Roles
 - ✓ Transaction Fails as No initialization for all roles
 - ✓ Only the owner can set roles
 - ✓ Cannot set the Null address as roles
 - ✓ Setting roles and verifying the information
 - ✓ Able to change the roles and verify the information
- Production Phase
 - ✓ Renewables cannot generate Zero Energy
 - ✓ Non-authorized renewable energy makers cannot make energy
 - ✓ Renewables cannot generate Zero Energy
 - ✓ Non-authorized renewable energy makers cannot make energy
 - ✓ Renewables can create tokens
 - ✓ When roles are changed, the new entities can mint, but the previous ones cannot

5.2.2. Smart contract testing: hydrogen fuel production

The production testing tests that the process of hydrogen production is correct:

- ✓ Renewables cannot generate Zero Energy
- ✓ Non-authorized renewable energy makers cannot make energy
- ✓ Renewables cannot generate Zero Energy
- ✓ Non-authorized renewable energy makers cannot make energy
- ✓ Renewables can create tokens
 1. Minting as per requirements
 2. Minting
 3. Transferring
 4. Checking balance of water and renewables after transferring, should be 0
 5. Minting H₂
 6. Checking balance of H₂
- ✓ When roles are changed, the new entities can mint, but the previous ones cannot
- ✓ Only the new entities can set new requirements to mint

5.2.3. Smart contract testing: workflow

- ✓ All balances, etc., should be at the initial state
- ✓ Producing 1 kg (10000) of H₂
 1. Minting: water: 109106 liters; electricity: 550000 kWh
 2. Transferring
 3. Minting H₂ = 1 kg (10000 units)
 4. The balance of electricity at electrolysis should be 0, and the balance of water at electrolysis should be 0.
- ✓ Producing 1 kg (10000) of H₂
 1. But Minting: water: 2 × 109106 liters; electricity: 550000 kWh

2. Transferring all the units
3. Minting H₂ = 1 kg (10000 units)
4. The balance of electricity at electrolysis should be 0, and the balance of water at electrolysis should be 109106 liters.

We evaluate test effectiveness by measuring code coverage. Fig. 2 shows the coverage of the proposed smart contract.

The code with instructions is available here [64].

5.3. Smart contract: deployment and verification

Hardhat Deploy simplifies the process of deploying smart contracts onto a blockchain network. This process involves compiling the Solidity smart contract code into bytecode and creating a transaction that initializes the contract's state on the blockchain. Hardhat Deploy interacts with the chosen blockchain network and submits the deployment transaction during deployment. This tool streamlines the deployment process by automating various steps, ensuring accurate contract initialization, and providing essential insights into the deployment procedure.

Smart contract verification guarantees the reliability and security of the system's operations. Verification of the smart contract ensures that the smart contracts governing these interactions are free from vulnerabilities, bugs, and logical errors. If the smart contracts are verified, the system gains enhanced trustworthiness, reducing the chances of unauthorized access, manipulation, or unauthorized changes. This increased reliability bolsters the overall confidence of stakeholders, users, and participants.

5.4. Frontend: deployment

Interacting directly with a smart contract via a blockchain explorer might not be the most efficient approach, prompting the preference for web applications for better user experience (UX). Blockchain networks can be accessed through DApps, enabling users to execute network transactions. A web application is developed using the Vue framework and Tailwind CSS for design. Custom web pages corresponding to users' roles were created and integrated, ensuring seamless workflow actions. External wallets like MetaMask are also utilized, enabling users to interact with blockchains running on EVMs (such as Ethereum, Binance, Polygon, etc.). The wallet's interaction streamlines user activities by managing the private keys necessary for transaction signing and interacting with the blockchain application.

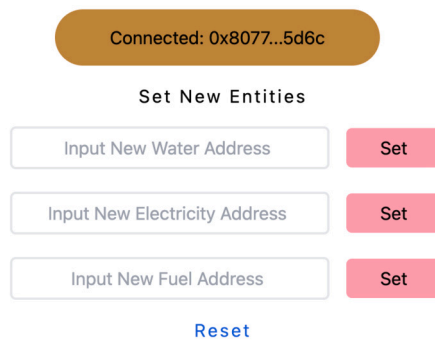
The application consists of the following pages.

5.4.1. Administrator

The administrator can define three roles: Water Supplier/Grid, Electricity Supplier/Grid, and Fuel (Hydrogen) Generator. They have key control over the authorization and own the smart contract, which can transfer ownership to another wallet address (Fig. 3).

5.4.2. Water supplier/grid

The water plant's user interface (Fig. 4) facilitates two key actions the assigned wallet role enables. The role includes two main functions: firstly, it allows the writing of the amount of water drawn from a water grid into the associated smart contract, a process known as "Mint"; secondly, the water plant can transfer water to the hydrogen producer



Connected: 0x8077...5d6c

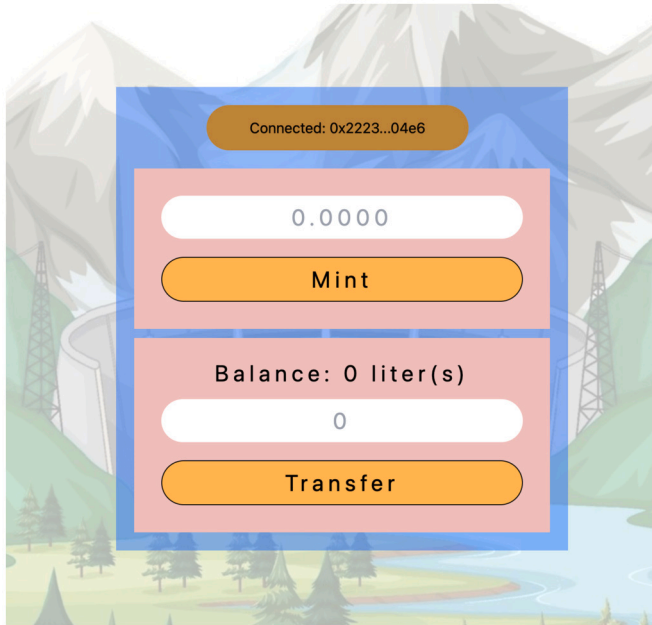
Set New Entities

Input New Water Address

Input New Electricity Address

Input New Fuel Address

Fig. 3. Interface for Admin, who can set roles for performing actions.



Connected: 0x2223...04e6

0.0000

Balance: 0 liter(s)

0

Fig. 4. Interface for a water plant to mint and transfer water.

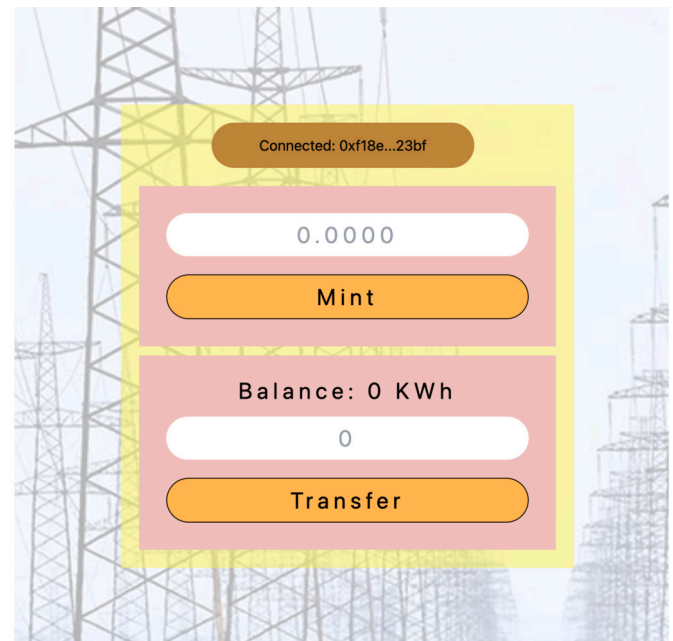
station, limited to its storage amount. This transfer is bound by the condition that the transferred water should be less than or equal to the stored amount. Additionally, the smart contract determines the transfer's recipient address, ensuring consistent delivery even if the hydrogen station's address changes. The amount of water stored in the plant is updated when it draws or transfers water.

5.4.3. User: electricity grid

The user interface of the electricity plant (Fig. 5) has two functions for the wallet role assigned by the administrator (discussed in Section 5.4.1). First, the wallet role allows the plant to "Mint" its electricity by updating the smart contracts, assuming it is generated from renewable sources. Second, the plant can transfer electricity (in kWh) from its storage to the hydrogen producer station, with the transfer amount limited to the stored amount. The smart contract determines the recipient address for this transfer to ensure accurate delivery to the intended user. The plant's stored electricity balance is continuously updated based on generation and transfer activities.

5.4.4. User: hydrogen producer

The user interface of the hydrogen plant, depicted in Fig. 6, explains its operational capabilities based on the role assigned by the administrator (as discussed in Section 5.4.1). This role enables two fundamental activities. Firstly, the hydrogen plant calculates and records the amount of hydrogen produced based on its available water and electricity resources using the "Mint H₂" function. Secondly, the hydrogen plant can



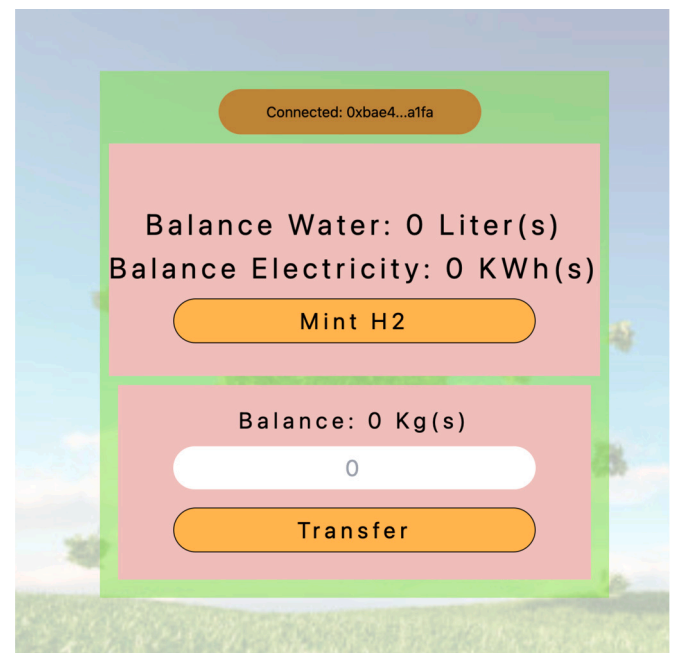
Connected: 0xf18e...23bf

0.0000

Balance: 0 KWh

0

Fig. 5. Interface for an electricity plant to mint and transfer electricity.



Connected: 0xbae4...a1fa

Balance Water: 0 Liter(s)

Balance Electricity: 0 KWh(s)

Balance: 0 Kg(s)

0

Fig. 6. Interface for a hydrogen plant to mint and transfer hydrogen fuel generated.

transfer the hydrogen fuel it has produced (limited by the fuel balance) in kilograms to the designated filling station. The recipient address for this transfer is not derived from the smart contract in the current version, but incorporating such functionality in future iterations is possible. The hydrogen plant's stored balance of hydrogen fuel changes dynamically due to its production and transfer activities.

5.4.5. User: fuel station

Each vehicle type has a hard-coded filling requirement of 6 kg for cars, 23 kg for buses, and 50 kg for trucks, assuming empty fuel tanks. The balance of hydrogen fuel is updated after each fill. Fig. 7 shows the filling station interface.

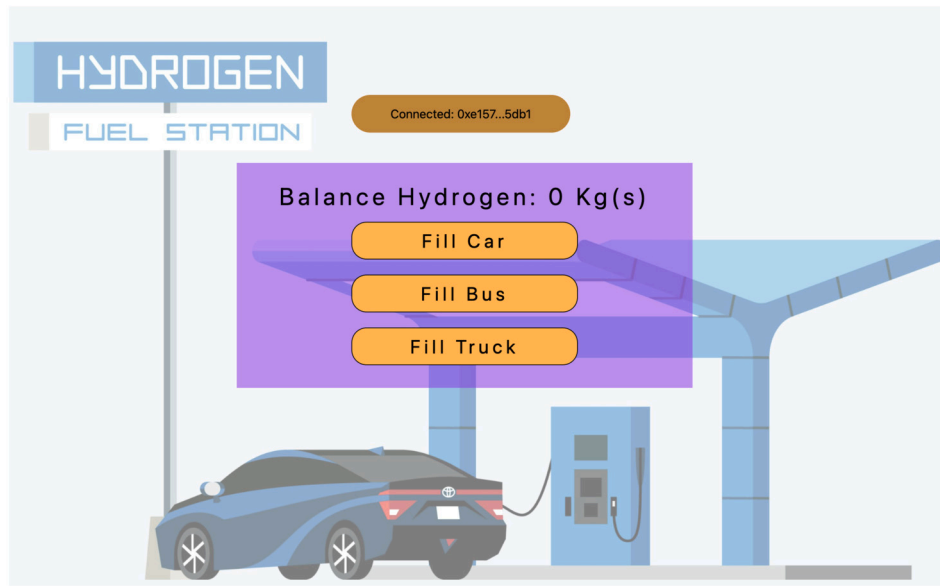


Fig. 7. Interface for fuel station to transfer hydrogen fuel to car, bus, and truck.

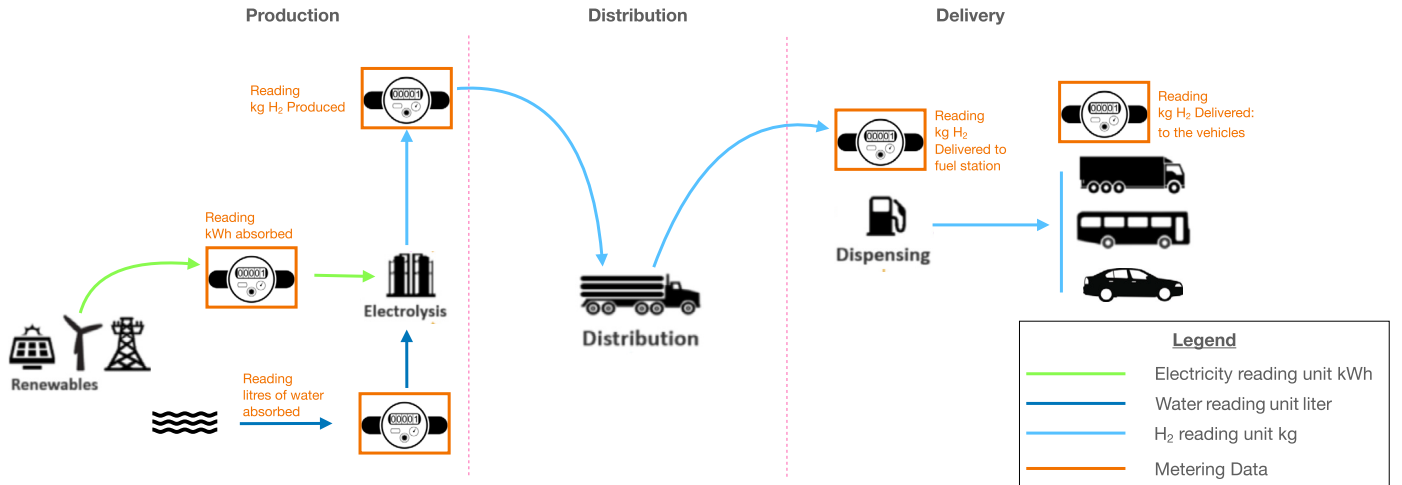


Fig. 8. Hydrogen production workflow.

Table 1
Requirements of Water and Electricity to generate 1 kg of Hydrogen.

Source	Amount	Unit
Water	10.9106	liter
Electricity	55	kWh

6. Simulation results and discussion

6.1. Workflow

6.1.1. Production

Table 1 presents the minimum amounts of water and electricity electrolysis that should be used to generate 1 kg of hydrogen fuel (Fig. 8). The hydrogen production phase includes the following stages.

1. Renewable procurement:

- **Water Minting:** *WATER PROVIDER* generates water tokens, representing the amount of water available from renewable sources such as water reservoirs or dams.

- **Energy Minting:** *RENEWABLES PROVIDER* produces energy tokens, symbolizing the energy harnessed from renewable sources like solar panels or wind turbines.

2. Renewable transfer:

- **Water Transfer:** *WATER PROVIDER* transfers the water tokens to *FUEL GENERATOR*, which will be used in the hydrogen production process.

- **Energy Transfer:** Similarly, *RENEWABLES PROVIDER* sends the energy tokens to *FUEL GENERATOR* for further use.

3. Fuel Creation:

Assuming that *FUEL GENERATOR* has an adequate supply of water and energy tokens, it utilizes an electrolysis process to create hydrogen fuel. This is where water is broken down into hydrogen and oxygen using the supplied energy.

4. Resource Consumption:

FUEL GENERATOR consumes the energy tokens and water tokens during the electrolysis process to generate the hydrogen fuel. This ensures that the consumed resources are accounted for and used in the fuel production process.

6.1.2. Distribution

After producing hydrogen fuel through electrolysis at the plant, it is transported to the filling station by trucks and distributed to end users. Specialized containers ensure safety and integrity during transit. **Note:**

Table 2

Daily requirement for electrolysis to produce the minimum amount of Hydrogen production to fill the demand in 2030.

Type	Size	Single unit requirement	Total requirement
NEXO	3000	6	18000
BUSSES	200	23	4600
TRUCK	45	50	2250

Although, while transferring the fuel, the transporting trucks also consume some amount of hydrogen (depending on the distance), for the experiment, we do not consider such a scenario but assume the amount of fuel loaded at the source is the same as the amount of fuel offloaded at the destination.

6.1.3. Delivery

Upon arrival at the filling station, hydrogen fuel is unloaded and stored in designated tanks to maintain its integrity and safety. The stored fuel is distributed to trucks, buses, and cars. The filling station adheres to predefined vehicle fueling requirements and accurately dispenses the appropriate amount of hydrogen fuel to each vehicle, ensuring that the produced hydrogen fuel contributes to a sustainable energy ecosystem.

6.2. Simulation

Based on the predicted demand for the vehicles, we consider the following scenario for the year 2030:

- Production of plant 1 is 50 MWh.
- One production plant fills 20 filling stations that fill 3000 NEXO, 200 BUS, and 45 TRUCK vehicles.

6.2.1. Experiment setup

We conducted a blockchain-based simulation on a local machine to evaluate the prototype's performance. We used Hardhat, a development-focused local Ethereum network node. The smart contract is deployed on the local blockchain network, and the simulation is conducted using Python. For interacting with Ethereum and EVM-based environments, we use Web3.py, a Python library that allows sending transactions, interacting with smart contracts, reading block data, and various other use cases. EVM backs the Hardhat implementation for an Ethereum-based environment. The experiments are conducted on a computer with 16 GB 2400 MHz DDR4 memory and an Intel Core i9 processor running at 2.3 GHz.

Table 2 shows the number of accounts allocated within each type of vehicle and the daily requirement for each vehicle to fill complete tanks with hydrogen fuel. In a blockchain network, we run experiments with 3250 wallets, of which 5 wallets represent hydrogen production and distribution entities and the remaining to the fleet of the vehicles.

The daily requirement for hydrogen fuel to meet the fleet requirement is 24850 kg, i.e., a minimum of 271128.41 liters of water and 1366750.00 kWh of electricity will be required to meet the daily demand.

6.2.2. Experiment

The experiment follows the following steps:

1. The smart contract owner deploys the smart contract on the local network.
2. We make the instance of the deployed smart contract by passing the contract address and the application binary interface (ABI) to interact [65].
3. The entities of the fuel production and distribution perform the activities as described in Section 6.1.
4. Assuming that 1 fuel station can accommodate the daily fuel requirement required to fill a fleet of 3245 vehicles.

Table 3

Average Gas consumption of the fuel filling process, including the contract deployment.

Role	Action	Gas used
Admin	Contract Deployment	3658896
	Water Authorization	46430
	GRID Authorization	46399
	Electrolysis Authorization	46444
Water	Mint Water	50196
	Transfer Water	53714
GRID	Mint Energy	50186
	Transfer Energy	53750
Electrolysis	Mint Fuel	72323
	Transfer Fuel	53738
Fuel Station	Fill Car	58513.136
	Fill Bus	58524.92
	Fill Truck	58417.734
Total		193865993

5. When analyzing the gas consumption of each vehicle type, we use a loop to record the data according to the requirements detailed in Table 2.

6.2.3. Result

Gas consumption data are presented in Table 3, which can be used to estimate the computational complexity using a specific formula:

$$\text{Complexity} = 3364 + 17n + (342 + 41.5n)d \quad [10^3 \text{ Gas}]$$

where

- d is the system operation days number,
- n is the fleet size,
- Gas is the EVM computational complexity unit.

Each block has a target size of 15 million Gas, but the size of blocks will increase or decrease per network demand until the block limit of 30 million Gas (2x target block size) is reached. The current average is 15000000 [66]; therefore, the daily consumption of Gas is less than 194000000 Gas or 13 Ethereum blocks (2 min and 36 s of Ethereum network time) or 7 Ethereum blocks during the high network. The smart contract is deployed once initially, but on each repetitive process, the block is deployed.

7. Discussion and future works

In our architecture, the electrolysis entity defines the water and electricity required. Since these requirements can be variable and dependent on multiple factors, like operational changes or demand fluctuations, we have entrusted the electrolysis component with control over these resources to ensure a responsive and adaptable system. If entities, such as water and electricity suppliers, determine these requirements, it could cause issues, such as supply shortages or inefficiencies; thus, it is challenging to meet the daily demand for vehicles. We can better manage and optimize the allocation of these resources to the electrolysis plant to cater to the system's specific needs, enhancing its efficiency and effectiveness and meeting the daily requirements.

There can be a time difference between digital and physical resource transfers. In a blockchain, transactions done through tokenized representations are much faster than physical resource transfers that require actual movement, like transporting fuel by vehicle or moving products through pipelines. This causes a time lag because of logistical constraints. We can incorporate a proof-of-delivery mechanism as an integral component of our system to ensure the exact amount of resources dispatched from the source and received at the destination [67]. Imagine a scenario where water reservoirs are transporting a specific volume of water, but during transit, a leakage occurs, resulting in a

shortfall between the dispatched and received quantities. In such cases, the electrolysis process can accept or reject the delivery based on irrefutable evidence provided by the proof-of-delivery mechanism. This multifaceted approach ensures the accuracy of resource transfers and accelerates the rectification of disparities, thus maintaining our architect's overall efficiency and reliability.

8. Conclusion

This paper presents a new approach to producing hydrogen fuel using blockchain technology, improving the production process's efficiency and traceability. The tokenization mechanism utilizes the ERC-1155 token standard, which tokenizes renewable resources like water and electricity, converting them into hydrogen fuel, thereby enabling secure and transparent transfers. This technique is applied via electrolysis to hydrogen production, resulting in new hydrogen tokens allocated to designated filling stations. Blockchain technology ensures that the distribution is accountable and tamper-resistant, providing a seamless and transparent distribution of hydrogen fuel to vehicles, leveraging cryptographic mechanisms and a distributed ledger to ensure the secure and transparent distribution of hydrogen fuel.

We have conducted an analysis using simulations within the framework of our proposed tokenized fuel blockchain architecture. This analysis assesses the projected growth in hydrogen fuel production, distribution, and demand for hydrogen-powered vehicles. The simulation allowed us to accurately quantify the required production volume and resource allocation to meet future needs. The presented gas consumption data offers insights into computational complexity, aiding in estimating the required resources. Each block has a target size of 15 million gas, with the flexibility to adjust based on network demand. The daily gas consumption remains below 194 million Gas, equating to approximately 13 Ethereum blocks or 7 during periods of high network activity. These findings provide tangible evidence of the feasibility and efficiency of our approach.

Through our simulated scenario, we have demonstrated the practicality and effectiveness of our approach in ensuring the sustainability and well-regulated growth of the hydrogen fuel production and distribution ecosystem. This aligns with the need for hydrogen-powered vehicles, making hydrogen fuel technologies more advanced and sustainable within a blockchain-enabled paradigm. Our research has paved the way for a cleaner, more sustainable energy future.

CRedit authorship contribution statement

Yash Madhwal: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Conceptualization. **Yury Yanovich:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Investigation, Formal analysis, Conceptualization. **Matteo Coveri:** Writing – review & editing, Writing – original draft, Validation, Investigation, Formal analysis, Data curation. **Ninoslav Marina:** Writing – review & editing, Writing – original draft, Supervision, Formal analysis, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] V. Smil, The long slow rise of solar and wind, *Sci. Am.* 310 (1) (2014) 52–57, <https://doi.org/10.1038/scientificamerican0114-52>.
- [2] United Nations, Net zero coalition, <https://www.un.org/en/climatechange/net-zero-coalition>, 2022. (Accessed 9 September 2023).

- [3] R.S.J. Tol, Eu emissions trading system, <https://ideas.repec.org/p/sus/susvid/2106.html>, 2021. (Accessed 9 September 2023).
- [4] T. Gür, Review of electrical energy storage technologies, materials and systems: challenges and prospects for large-scale grid storage, *Energy Environ. Sci.* 11 (10) (2018) 2696–2767, <https://doi.org/10.1039/C8EE01419A>.
- [5] F. Crotogino, Traditional bulk energy storage—coal and underground natural gas and oil storage, in: T.M. Letcher (Eds.), *Storing Energy*, Elsevier, Amsterdam, Netherlands, 2022, pp. 633–649, <https://doi.org/10.1016/b978-0-12-824510-1.00021-0>.
- [6] C. Zou, Z. Yang, S. Huang, et al., Resource types, formation, distribution and prospects of coal-measure gas, *Pet. Explor. Dev.* 46 (3) (2019) 451–462, [https://doi.org/10.1016/s1876-3804\(19\)60026-1](https://doi.org/10.1016/s1876-3804(19)60026-1).
- [7] G. Pan, W. Gu, Y. Lu, et al., Optimal planning for electricity-hydrogen integrated energy system considering power to hydrogen and heat and seasonal storage, *IEEE Trans. Sustain. Energy* 11 (4) (2020) 2662–2676, <https://doi.org/10.1109/TSTE.2020.2970078>.
- [8] C. Tarhan, M. Çil, A study on hydrogen, the clean energy of the future: hydrogen storage methods, *J. Energy Storage* 40 (2021) 102676, <https://doi.org/10.1016/j.est.2021.102676>.
- [9] U. Qazi, Future of hydrogen as an alternative fuel for next-generation industrial applications; challenges and expected opportunities, *Energies* 15 (13) (2022) 4741, <https://doi.org/10.3390/en15134741>.
- [10] H. Mehrjerdi, Peer-to-peer home energy management incorporating hydrogen storage system and solar generating units, *Renew. Energy* 156 (2020) 183–192, <https://doi.org/10.1016/j.renene.2020.04.090>.
- [11] J. Graetz, New approaches to hydrogen storage, *Chem. Soc. Rev.* 38 (1) (2009) 73–82, <https://doi.org/10.1039/b718842k>.
- [12] R. Kothari, D. Buddhi, R. Sawhney, Comparison of environmental and economic aspects of various hydrogen production methods, *Renew. Sustain. Energy Rev.* 12 (2) (2008) 553–563, <https://doi.org/10.1016/j.rser.2006.07.012>.
- [13] G. Kakoulaki, I. Kougias, N. Taylor, et al., Green hydrogen in Europe — a regional assessment: substituting existing production with electrolysis powered by renewables, *Energy Convers. Manag.* 228 (2021) 113649, <https://doi.org/10.1016/j.enconman.2020.113649>.
- [14] I. Bashir, *Mastering Blockchain*, 4th ed. Packt Publishing Ltd, St Paul's Square, Birmingham, 2017.
- [15] M. Atzori, Blockchain technology and decentralized governance: Is the state still necessary?, *J. Gov. Regul.* 6 (1) (2017) 45–62, https://doi.org/10.22495/jgr_v6_i1_p5.
- [16] C. Pop, T. Cioara, M. Antal, et al., Blockchain based decentralized management of demand response programs in smart energy grids, *Sensors* 18 (1) (2018) 162, <https://doi.org/10.3390/s18010162>.
- [17] L. Malyavkina, A. Savina, I. Parshutina, Blockchain technology as the basis for digital transformation of the supply chain management system: benefits and implementation challenges, in: *Proceedings of the 1st International Scientific Conference "Modern Management Trends and the Digital Economy: from Regional Development to Global Economic Growth" (MTDE 2019)*, Atlantis Press, 2019, pp. 10–15, <https://doi.org/10.2991/mtde-19.2019.3>.
- [18] S. Asefi, Y. Madhwal, Y. Yanovich, et al., Application of blockchain for secure data transmission in distributed state estimation, *IEEE Trans. Control Netw. Syst.* 9 (4) (2022) 1611–1621, <https://doi.org/10.1109/tcms.2021.3134135>.
- [19] Y. Madhwal, P. Panfilov, Blockchain and supply chain management: aircrafts' parts' business case, in: *Proceedings of the DAAAM International Symposium on Intelligent Manufacturing and Automation*, DAAAM International, 2017, pp. 1051–1056, <https://doi.org/10.2507/28th.daaam.proceedings.146>.
- [20] Y. Madhwal, Y. Yanovich, I. Chumakov, CoVID-19 vaccination certificate supply verification based on blockchain, in: *Proceedings of the 2021 4th International Conference on Blockchain Technology and Applications*, ACM, 2021, <https://doi.org/10.1145/3510487.3510500>.
- [21] Y. Madhwal, E. Pieber, Y. Yanovich, et al., Smart contract based honey production supply chain, in: *Proceedings of the 2022 5th International Conference on Blockchain Technology and Applications*, ACM, 2022, <https://doi.org/10.1145/3581971.3581981>.
- [22] M. Andoni, V. Robu, D. Flynn, et al., Blockchain technology in the energy sector: a systematic review of challenges and opportunities, *Renew. Sustain. Energy Rev.* 100 (2019) 143–174, <https://doi.org/10.1016/j.rser.2018.10.014>.
- [23] Q. Wang, M. Su, Integrating blockchain technology into the energy sector — from theory of blockchain to research and application of energy blockchain, *Comput. Sci. Rev.* 37 (2020) 100275, <https://doi.org/10.1016/j.cosrev.2020.100275>.
- [24] E. Barceló, K. Dimić-Mišić, M. Imani, et al., Regulatory paradigm and challenge for blockchain integration of decentralized systems: example—renewable energy grids, *Sustain.* 15 (3) (2023), <https://doi.org/10.3390/su15032571>.
- [25] K. Mould, F. Silva, S.F. Knott, et al., A comparative analysis of biogas and hydrogen, and the impact of the certificates and blockchain new paradigms, *Int. J. Hydrog. Energy* 47 (93) (2022) 39303–39318, <https://doi.org/10.1016/j.ijhydene.2022.09.107>.
- [26] L. Wang, S. Jiao, Y. Xie, et al., Two-way dynamic pricing mechanism of hydrogen filling stations in electric-hydrogen coupling system enhanced by blockchain, *Energy* 239 (2022) 122194, <https://doi.org/10.1016/j.energy.2021.122194>.
- [27] Y. Madhwal, Implementation of tokenised supply chain using blockchain technology, in: *Proceedings of the 2020 IEEE 21st International Symposium on "A World of*

- Wireless, Mobile and Multimedia Networks", IEEE, 2020, <https://doi.org/10.1109/WoWMoM49955.2020.00026>.
- [28] A.R. Harish, X. Liu, M. Li, et al., Blockchain-enabled digital assets tokenization for cyber-physical traceability in e-commerce logistics financing, *Comput. Ind.* 150 (2023) 103956, <https://doi.org/10.1016/j.compind.2023.103956>.
 - [29] Y. Madhwal, I. Chistiakov, Y. Yanovich, Logging multi-component supply chain production in blockchain, in: *Proceedings of the 2021 the 4th International Conference on Computers in Management and Business*, ACM, 2021, <https://doi.org/10.1145/3450588.3450604>.
 - [30] Z. Zheng, S. Xie, H. Dai, X. Chen, H. Wang, An overview of blockchain technology: Architecture, consensus, and future trends, in: *Proceedings of the 2017 IEEE international congress on big data (BigData congress)*, IEEE, 2017, pp. 557–564, <https://doi.org/10.1109/BigDataCongress.2017.85>.
 - [31] A. Yakubov, W. Shbair, A. Wallbom, et al., A blockchain-based PKI management framework, in: *Proceedings of the NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, IEEE, 2018, <https://doi.org/10.1109/NOMS.2018.8406325>.
 - [32] H. Min, Blockchain technology for enhancing supply chain resilience, *Bus. Horiz.* 62 (1) (2019) 35–45, <https://doi.org/10.1016/j.bushor.2018.08.012>.
 - [33] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, <https://bitcoin.org/bitcoin.pdf>, 2008. (Accessed 9 September 2023).
 - [34] G. Wood, et al., Ethereum: a secure decentralised generalised transaction ledger, *Ethereum project yellow paper*, <https://ethereum.github.io/yellowpaper/paper.pdf>, 2014. (Accessed 9 September 2023).
 - [35] X. Li, A. Scaglione, Robust decentralized state estimation and tracking for power systems via network gossiping, *IEEE J. Sel. Areas Commun.* 31 (7) (2013) 1184–1194, <https://doi.org/10.1109/JSAC.2013.130703>.
 - [36] A. Gervais, G.O. Karame, K. Wüst, et al., On the security and performance of proof of work blockchains, in: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 3–16, <https://doi.org/10.1145/2976749.2978341>.
 - [37] S. King, S. Nadal, PPOCoin: peer-to-peer crypto-currency with proof-of-stake, *Self-Publ. Pap.* 19 (1) (2012).
 - [38] M. Castro, B. Liskov, et al., Practical byzantine fault tolerance, *ACM Trans. Comput. Syst.* 20 (4) (2002) 398–461, <https://doi.org/10.1145/571637.571640>.
 - [39] Y. Yanovich, I. Ivashchenko, A. Ostrovsky, et al., Exonum: byzantine fault tolerant protocol for blockchains, https://exonum.com/theme/public/img/downloads/wp_consensus_181227.pdf, 2018. (Accessed 9 September 2023).
 - [40] I. Kotsiuba, A. Velvkhzanin, Y. Yanovich, et al., Decentralized e-health architecture for boosting healthcare analytics, in: *Proceedings of the 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, IEEE, 2018, pp. 113–118, <https://doi.org/10.1109/WorldS4.2018.8611621>.
 - [41] I. Karamitsos, M. Papadaki, N.B. Al Barghuthi, Design of the blockchain smart contract: a use case for real estate, *J. Inf. Secur.* 9 (3) (2018) 177–190, <https://doi.org/10.4236/jis.2018.93013>.
 - [42] P. Mamoshina, L. Ojomoko, Y. Yanovich, et al., Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare, *Oncotarget* 9 (2018) 5665–5690, <https://doi.org/10.18632/oncotarget.22345>.
 - [43] E. Hildenbrandt, M. Saxena, N. Rodrigues, et al., KEVM: a complete formal semantics of the Ethereum virtual machine, in: *Proceedings of the 2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, IEEE, 2018, pp. 204–217, <https://doi.org/10.1109/CSF.2018.00022>.
 - [44] Z. Zheng, S. Xie, H.-N. Dai, et al., An overview on smart contracts: challenges, advances and platforms, *Future Gener. Comput. Syst.* 105 (2020) 475–491, <https://doi.org/10.1016/j.future.2019.12.019>.
 - [45] L. Luu, D.-H. Chu, H. Olickel, et al., Making smart contracts smarter, in: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ACM, 2016, pp. 254–269, <https://doi.org/10.1145/2976749.2978309>.
 - [46] N. Barney, A. Gillis, C. Bernstein, Decentralized application (DApp), <https://www.techtarget.com/iotagenda/definition/blockchain-dApp>. (Accessed 9 September 2023).
 - [47] Infura: the foundation for decentralized applications, <https://infura.io/>. (Accessed 9 September 2023).
 - [48] Alchemy, <https://www.alchemyapi.io/>. (Accessed 9 September 2023).
 - [49] L. Oliveira, I. Bauer, L. Zavolokina, et al., To token or not to token: tools for understanding blockchain tokens, in: *Proceedings of the 39th International Conference on Information Systems 2018 (ICIS 2018)*, ICIS, San Francisco, CA, 2018, pp. 1–17, <https://doi.org/10.5167/uzh-157908>.
 - [50] V. Davydov, A. Gazaryan, Y. Madhwal, et al., Token standard for heterogeneous assets digitization into commodity, in: *Proceedings of the 2019 2nd International Conference on Blockchain Technology and Applications*, ACM, 2019, pp. 43–47, <https://doi.org/10.1145/3376044.3376053>.
 - [51] M. di Angelo, G. Salzer, Tokens, types, and standards: identification and utilization in Ethereum, in: *Proceedings of the 2020 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*, IEEE, 2020, pp. 1–10, <https://ieeexplore.ieee.org/document/9126009/>.
 - [52] F. Vogelsteller, V. Buterin, EIP-20: ERC-20 token standard, <https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md>, 2015. (Accessed 9 September 2023).
 - [53] Q. Wang, R. Li, Q. Wang, et al., Non-fungible token (NFT): overview, evaluation, opportunities and challenges, *arXiv*. 2021. preprint, [arXiv:2105.07447](https://arxiv.org/abs/2105.07447).
 - [54] W. treatment for green hydrogen: what you need to know, <https://hydrogentechworld.com/water-treatment-for-green-hydrogen-what-you-need-to-know>, 2022. (Accessed 9 September 2023).
 - [55] Cavendish Hydrogen, <https://nelhydrogen.com/hydrogen-fueling-equipment/>, 2022. (Accessed 9 September 2023).
 - [56] H. Newsroom, Hyundai motor's xcient fuel cell heavy-duty trucks to hit German roads, <https://www.hyundaimotorgroup.com/news/CONT000000000046014>, 2022. (Accessed 9 September 2023).
 - [57] P. Zsófia, New Dutch hydrogen hgv called Europa will have range over 1,000 kilometres, <https://trans.info/europa-hydrogen-hgv-325338>, 2023. (Accessed 9 September 2023).
 - [58] B. Heid, C. Martens, M. Wilthaner, Unlocking hydrogen's power for long-haul freight transport, <https://www.mckinsey.com/industries/travel-logistics-and-infrastructure/how-we-help-clients/global-infrastructure-initiative/voices/unlocking-hydrogens-power-for-long-haul-freight-transport>, 2022. (Accessed 9 September 2023).
 - [59] Binance testnet, <https://testnet.bscscan.com/>. (Accessed 9 September 2023).
 - [60] Solidity, <https://soliditylang.org/>. (Accessed 9 September 2023).
 - [61] Ethereum development environment for professionals, <https://hardhat.org/>. (Accessed 9 September 2023).
 - [62] The progressive javascript framework, <https://vuejs.org/>. (Accessed 9 September 2023).
 - [63] Tailwindcss, <https://tailwindcss.com/>. (Accessed 9 September 2023).
 - [64] Y. Madhwal, <https://github.com/yashmadhwal/h2-production-distribution>. (Accessed 9 September 2023).
 - [65] Contract ABI Specification, <https://docs.soliditylang.org/en/latest/abi-spec.html>. (Accessed 9 September 2023).
 - [66] The history of ethereum's block size & block gas limit, <https://blog.mycrypto.com/the-history-of-ethereums-block-size-block-gas-limit>, 2021. (Accessed 9 September 2023).
 - [67] Y. Madhwal, Y. Borbon-Galvez, N. Etemadi, et al., Proof of delivery smart contract for performance measurements, *IEEE Access* 10 (2022) 69147–69159, <https://doi.org/10.1109/access.2022.3185634>.