Carboncoin: Tokenization of Carbon Emission with ESG-based Reputation

Oscar Golding
School of Computer Science
University of New South Wales
Sydney, Australia
oscargolding17@gmail.com

Guangsheng Yu
Data61
CSIRO
Sydney, Australia
saber.yu@data61.csiro.au

Qinghua Lu
Data61
CSIRO
Sydney, Australia
qinghua.lu@data61.csiro.au

Xiwei Xu
Data61
CSIRO
Sydney, Australia
xiwei.xu@data61.csiro.au

***Abstract*—Recent blockchain based carbon markets focus on permit-based trading requiring manual application processes to grant the right for carbon emission. A decentralized blockchain-based carbon market without relying on off-chain permits is yet to be explored. In this paper, we present a new design of blockchain-based carbon trading through the introduction of *Carboncoin* - a blockchain asset which tokenizes the right of energy producers to emit carbon. Instead of relying on off-chain and centralized permits, producers are allowed to freely exchange *Carboncoin* with each other for fiat currency. By using an on-chain divisible asset, carbon production can be automatically expensed whenever a producer records new energy production which is certified on the blockchain. Moreoever, we present a system which enables generic ESG (Environmental, Social and Governance) data to be used to provide a more holistic reputation score inclusive of ESG initiatives undertaken by market participants. Through the construction of a completely decentralized platform for carbon trading we explore how system performance generally tends to suffer when too many assets are placed on-chain for a blockchain trading platform. We conclude that entirely blockchain-based carbon markets can be made more comprehensive using ESG data and divisible assets, but at the cost of reduced performance.**

***Index Terms*—blockchain, tokenization, carbon emission,**

I. INTRODUCTION

Carbon markets provide a mechanism for energy producers to pay for expensive carbon usage, thereby providing a direct incentive to not become reliant on unclean sources of energy. Of key concern to the successful creation and operation of carbon markets is the issue of them being politically contentious. Carbon prices are open to government manipulation and are often the subject of heated debates in both political and economic fields. As an example, consider the 'axe the tax' political campaign which resulted in the removal of the Australian Emissions Trading Scheme (ETS) in 2014 [1]. Despite political concerns, carbon markets have grown over recent decades to act as an important tool in combatting climate change. Some of the largest economies in the world, such as the European Union and China run some form of an ETS [1].

The blockchain, which acts as an immutable ledger, has the potential to make carbon assets more transparent to both energy regulators and market participants. As opposed to being run by a centralised authority such as a government or large firm, a carbon market can instead be run openly on the blockchain so it becomes visible to all stakeholders in a society. The openness of the blockchain can help energy producers to use carbon assets more efficiently to direct investment into greener sources of energy. The transparency of the blockchain can also allow market participants to more easily track their energy consumption and carbon assets, whilst regulators can trustfully ensure carbon reduction goals are being met. The blockchain is a useful tool for not only energy certification but also general Environmental, Social and Governance (ESG) data [2]. Energy certification can be used as a form of ESG data which can be submitted to a blockchain carbon market automatically using on-chain certification. Subsequently, the blockchain has seen a lot of academic interest for the creation of a platform for carbon markets.

Presently, creating a blockchain architecture for carbon markets which incorporates all forms of on-chain ESG data is unexplored. Instead, previous attempts to create a blockchain carbon market have relied on a non-divisible asset such as a permit which can provision the right to emit carbon. Despite being able to trade permits with one another, important questions for both the regulator and market participant remain. How can market participants ensure they are purchasing permits from firms who engage in positive ESG initiatives which align with their own carbon reduction goals? Moreover, how can a regulator trustfully ensure participants are reducing their carbon footprint with ESG initiatives outside of the on-chain carbon market?

Carbon market blockchains have the ability to be refined in two important ways. First, instead of relying on permits a general fungible asset (such as a coin) is introduced to be automatically expensed whenever environmentally costly energy is reported through a blockchain certification scheme. Second, providing a traceable on-chain reputation score to market users with blockchain ESG data can help both participants and regulators trustfully ensure carbon reduction goals are being met. Our proposed architecture invokes a series of smart contracts which first collates on-chain ESG data and then utilises blockchain patterns to transform such data into a usable reputation score.

In this paper, we use the token economy of the blockchain to create an architecture for carbon trading with ESG data.

We design smart contracts using industry standard patterns to allow for distributed carbon trading where market participants are assigned reputation scores based off their ESG initiatives. Algorithms contained within the smart contracts apply a transparent weight to a market participant's ESG contribution to provide such a reputation score.

The remainder of the paper is organised as follows. Section two goes over the related work for blockchain carbon markets. Section three explains the proposed ESG architecture. Section four details the implementation of such an architecture. Section five outlines the performance of our implemented system whilst section six provides a discussion of our findings.

II. Related Work

A. Blockchain Carbon Markets

An early adoption of blockchain carbon pricing was the creation of a bitcoin-based emissions trading infrastructure model [3]. The model proposed a blockchain called Decentralised Carbon Emissions Trading Infrastructure (D-CETI), which inherited the arhitectural innovations created with Bitcoin in 2009. The authors proposed that a significant barrier limiting the effectiveness of the major Emissions Trading Schemes (ETS) was a lack of security and privacy in carbon trading. A downside to the proposed solution was its slow performance and reliance on environmentally expensive cryptographic puzzle solving.

With the development of more advanced blockchains, such as Ethereum, there has been renewed interest in deploying smart contracts specialised in carbon trading. In 2017, the suitability of Ethereum was explored as a platform to develop smart contracts for carbon trading [4]. The researchers concluded that a public blockchain may not be the most suitable platform to host a carbon market.

In 2018, the blockchain framework *Multichain* was explored as a platform to host a carbon market [5]. A major advantage of the new implementation was the focus on a blockchain supporting a higher transactions per second (TPS) along with optional market reputation. Of notable significance is the requirement where all users must have their carbon sale offers ranked by reputation. Furthermore, the work was important in introducing the notion of having an on-chain reputation score for traders in a blockchain carbon market. Where *Carboncoin* differs is the decision not to force a reputation score for all users, and instead use reputation to nudge users into making better decisions on which producers to trade carbon with.

In 2018 another carbon market was deployed on the at the time new permissioned blockchain Hyperledger Fabric [6]. The researchers identified a significant barrier to creating a trustful carbon market on the blockchain was the lack of trust and security issues associated with blockchain carbon trading. Since Hyperledger is a permissioned blockchain, the researchers concluded it would allow for more trustful trading since participants would be allowed to openly identify trading partners on the platform. The proposed architecture required the manual submission of individual emissions permits for each user. Users would then be allowed to trade permits amongst each other on-chain. A downside is the manual intervention required by regulators to approve carbon permits for users. Instead, a more elegant solution would allow for on-chain certificates to expense an on-chain permit automatically without manual intervention.

Our application of carbon trading on the blockchain differs in two key ways. First, energy production resulting in carbon emissions when reported (such as through a smart meter) will automatically deduct a token called *Carboncoin* from a user's account and limit their ability to engage in costly production. Second, the reputation of a user is based on general ESG data that can be automatically submitted and verified by the carbon market blockchain.

B. Industry Case Studies

To facilitate online trading of carbon the industry has proposed the creation of commodity exchanges such as carbon trading platforms. One such example is *Xpansiv* - an online market for trading energy, carbon and water using ESG data [7]. Although offering a similar set of features to *Carboncoin* - it differs importantly in using a centralised architecture requiring a high degree of trust with the platform.

The industry has recently moved to support energy on the blockchain. One such example is the Australian firm *PowerLedger* [8]. The firm offers a private decentralised blockchain product to enable peer-to-peer energy trading. To help support such a goal a *POWR* token provides users with electrical units (in kilowatt-hour units). A core difference to our proposed blockchain solution is *PowerLedger* not allowing for the decentralised trading of the right to emit carbon.

C. Blockchain Certification

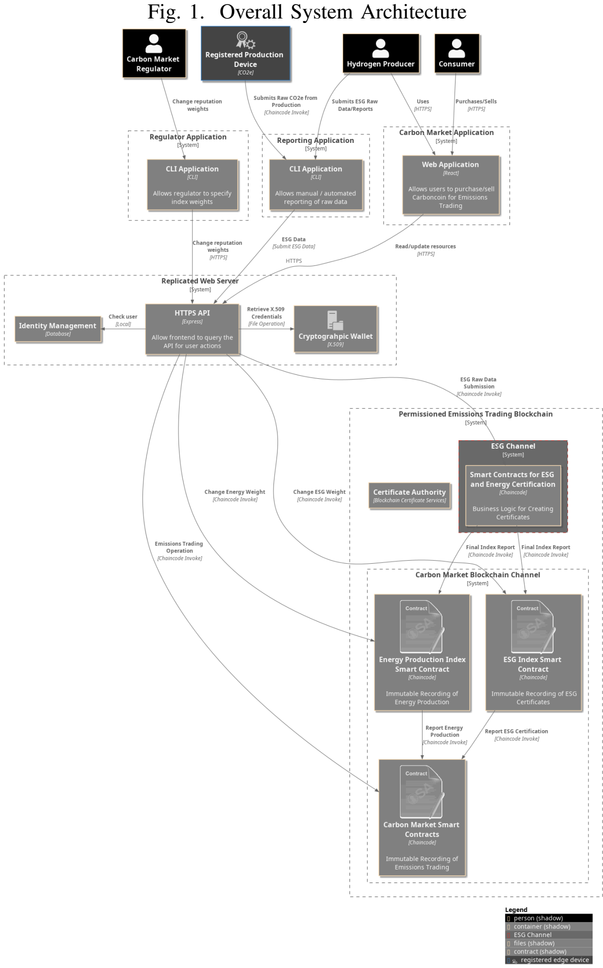
Energy production is traditionally certified by a centralised regulator who develops certification schemes. One of the most popular certification schemes is CertifHy - a European scheme for Hydrogen certification [9]. Centralised certification schemes provide services such as the issuance of hydrogen certificates to verify production. Often attached to the certificates are crucial information such as greenhouse gas emissions. Ideally, a certification scheme on-chain should be used to generate trustful certificates of energy production - which can subsequently be used to expense permits on a blockchain carbon market.

In 2021, a blockchain-based ESG reporting framework was proposed to trustfully register raw ESG data on the blockchain [2]. The proposed framework allows for fact-checking, consistency verification and evaluation of raw ESG data so a final index can be compiled and utilised by other actors on the blockchain. Importantly, it would allow for reputation on a carbon market to not just reflect energy production. Instead, reputation could reflect all ESG initiatives deemed to be beneficial to a society in general. Moreover, such ESG data could be automatically verified on-chain without manual submission - solving an issue with older blockchain markets which required permit creation by a regulator.

III. SYSTEM ARCHITECTURE

Figure 1 outlines the architecture for the emissions trading platform. The end user interacts with the blockchain platform through a web application, which invokes chaincode using an intermediary HTTPS API. The business logic for emissions trading occurs on the blockchain through deployed smart contracts. The blockchain runs on *Hyperledger Fabric* with an access-control list. The access-control list is a way for smart contracts running on Hyperledger to check the identities of invoking users using *Fabric* provided certificates. Lightweight command-line applications can be attached to smart devices - for example a smart meter - to automatically record emissions from energy production. Energy producers are given the opportunity to submit ESG data directly to the blockchain 'ESG Channel'. All submitted ESG data is utilised by the 'Carbonmarket' blockchain channel to generate a reputation score for traders on the platform. Energy producers engage in emissions trading through a web application - but are also given the freedom to directly invoke chaincode with credentials registered with *Fabric*.

Fig. 1. Overall System Architecture



A. Hyperledger Fabric

The architecture of the proposed carbon market uses the permissioned blockchain framework *Hyperledger Fabric* [10]. Importantly, *Fabric* allows for the division of smart contracts into individual 'channels', which allows for the creation of private subnetworks to facilitate confidential transactions. Such a feature allows for strict permissions to enforce rules on how chaincode is invoked by different users of the carbon market. As an example, access control on a smart contract would allow energy producers to submit chaincode inside the ESG channel and carbon market, but external users can only access the carbon market channel.

Access control and invoking chaincode is managed by the Hyperledger Certificate Authority. The Certificate Authority exists as a service for the blockchain to enrol and register users (with a secret) and invoke chaincode transactions. A policy contract pattern is used by smart contracts in the carbon market to ensure users are only allowed to invoke chaincode specified by their role.

B. System Users

Of particular note - as detailed in Figure 1 - is the role of a 'Carbonmarket Regulator' to maintain the initial allocation of *Carboncoin* given to new producers along with the weight specifying the conversion of carbon dixoide equivalence to *Carboncoin*. The regulator can ensure through the control of initial supply and the weights mapping carbon dixoide equivalence with *Carboncoin* that inflation is controlled within the carbon market to meet the objective of reducing total emissions in energy production.

The energy producer is the primary agent in the market, and invokes chaincode through a web application. Using the web application, the energy producer can view sales of *Carboncoin* and optionally purchase *Carboncoin* from other users in a decentralised manner. Moreover, the energy producer is allowed to view their entire reputation breakdown (as derived from ESG data) along with the reputation scores of other energy producers on the platform. Furthermore, the energy producer is given the optional ability to submit raw ESG data into the carbon market. For example, details inside the producer's Annual Report could be submitted to boost the market reputation score of an energy producer.

Optionally, users who are not producers can use the market under the 'consumer' role. Such users cannot submit ESG data, but are allowed to engage in market speculation through the purchasing and selling of *Carboncoin*.

C. Order Book

All open market offers for the sale of *Carboncoin* are stored in an on-chain order book. The philosophy of the system is to have emissions trading be transparent to users interacting with the blockchain - including all open market offers for trading a fiat currency to *Carboncoin*. A secondary benefit of putting the order book on chain is that the market can continue to operate if the replicated HTTPS web server were ever to fail. Producers of energy are allowed to optionally operate as a *peer* in the

Hyperledger network and directly invoke chaincode to view offers for the sale of *Carboncoin* by other users. Executing the sale of *Carboncoin* for fiat currency requires an off-chain payment channel existing only on the HTTPS web server.

D. ESG Reputation of Energy Producers

A core feature of the implemented system is the idea energy producers are given a mandatory reputation recorded on the blockchain. The ‘reputation’ of the user is always associated with an open offer for the sale of *Carboncoin* for fiat currency. Users are also given the optional ability to view the entire reputation breakdown of other users to assist in the ethical exchange between fiat currency and *Carboncoin*.

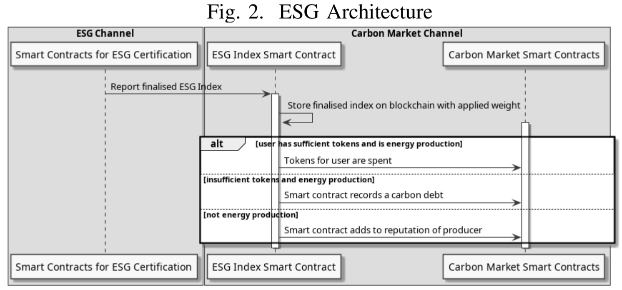
To aid in determining the overall quality of the ESG data the ‘Carbon Market Blockchain Channel’ from Figure 1 communicates with an on-chain ESG channel. The technical responsibility of the ESG channel is to compile raw ESG data, which can be submitted automatically in the case of a smart meter or manually when a company uploads an important document like their Annual Report. The ESG channel will apply classification techniques to determine the quality of the raw data and produce a final index score. The ESG channel responsible for orchestrating the verification of raw data is outlined with a dark grey background in Figure 1. The implemented system assumed the existence of such smart contracts, as the objective of the presented architecture is to show how blockchain recorded ESG data can improve the quality of carbon markets.

Once the raw data is compiled into an ESG index, smart contracts for ESG certification invoke chaincode in the carbon market blockchain channel. The producer identifier, ESG category and the raw index value are the input provided when invoking the carbon market channel. A weight for the submitted ESG category is provided by a regulator. The weight is multiplied with the raw index score to produce a reputation value. A final chaincode invoke calls the carbon market smart contracts to add the reputation value to the user’s total reputation score. Summary statistics are stored on the blockchain to make it transparent to producers the underlying index used along with the weight applied to produce a final reputation score. Equation 1 shows how each ESG Index value x_i is multiplied with a weight w_i to reach a reputation score for a user. The summation occurs over all n immutable ESG index scores for an energy producer.

$$\text{Reputation} = \sum_{i=1}^{n} x_i \cdot w_i \tag{1}$$

Figure 2 details the process of how raw ESG is finalised and subsequently used by the Carbon Market to give a reputation score. Of note is how the ‘ESG Index Smart Contract’ alerts the carbon market on whether the ESG data requires payment with *Carboncoin*. A primary benefit of the architecture is how producers of energy do not have to manually expense their carbon production - instead the smart contract on the blockchain automatatically deducts *Carboncoin* as production occurs. Conversely, if the raw ESG data reflects a positive environmental, social or governance development for the producer (such as improved water quality in hydrogen production) then a positive reputation score is calculated.

Fig. 2. ESG Architecture



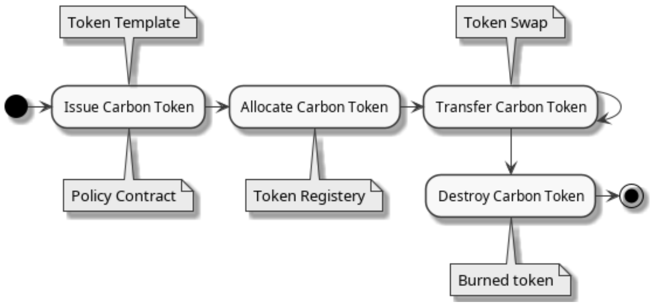
IV. Patterns

To motivate blockchain-based emissions trading, an on-chain currency called *Carboncoin* was developed. The smart contract controlling Emissions Trading is referred to as the ‘Carbonmarket Smart Contracts’ in Figure 1. An on-chain asset, such as one for the right to emit carbon, must go through a series of states to ensure the correct facilitation of blockchain payments. Recent work has analysed the importance of incorporating blockchain focused design patterns inside smart contracts [11]. Of particular note is the requirement for a token registry to map users in the system to owned assets and a policy contract to determine the immutable rules controlling the decentralised sale of a carbon asset.

Figure 3 details the life-cycle of *Carboncoin* from its allocation to destruction. Energy producers are initially allocated *Carboncoin* based on the size of the firm’s energy operations using an on-chain policy contract. Firms which are not registered with the policy contract are given a default amount of tokens. When allocating *Carboncoin*, a token registry pattern is used to track the amount of *Carboncoin* currently held by each user. A token swap pattern is used when energy producers trade in *Carboncoin* with one another. Trading is motivated by the incentive to sell excess *Carboncoin* for a fiat currency. Again, a policy contract pattern determines the precise rules regulating how trades of *Carboncoin* happen. A producer is never allowed to over-extend themselves when selling *Carboncoin* by creating sale offers with a quantity of *Carboncoin* greater than the balance of their account minus the quantity they have already offered on the market. The lifecycle of *Carboncoin* ends with a burned token pattern. *Carboncoin* is burned when a firm emits carbon through the means of energy production. In the case of hydrogen production - if an emissions heavy technique such as ‘coal gasification’ is used then the excess carbon as reported in carbon dioxide equivalence is converted to *Carboncoin* so it can be automatically expensed from a producer’s emissions trading account. Energy production and the requirement to have *Carboncoin* to pay for emissions drives the operation of the blockchain market.

For the ‘ESG Channel’ in the carbon market, a token template pattern specifies a category and weight submitted by the regulator. Upon using the template, the regulator can subsequently provide a weight for all incoming ESG data. Another policy contract enforces that the weight must be multiplied with the index value on incoming ESG data to produce a carbon market score which is either added or subtracted from an energy producer’s total reputation. Again, the ‘ESG Index Smart Contract’ from Figure 1 of the architecture has a strict policy contract to alert the carbon market on whether the ESG data requires payment with *Carboncoin*.

Fig. 3. *Carboncoin* Issuance, Allocation, Transferral and Destruction



V. Implementation

A. Order Book Implementation

A user with the energy producer role interacts with the order book through the creation of decentralised offers which exchange *Carboncoin* for fiat currency - primarily Australian dollars. A policy contract requires the energy producer not to create more offers with a sale value of *Carboncoin* greater than the amount contained inside their account. The orders are subsequently submitted to the blockchain and recorded with a token template pattern which immutably records the energy producer identifier, the price offered per *Carboncoin* in fiat currency, the total amount of *Carboncoin* being sold and a unique offer identifier.

The order book chaincode was written to be deployed on *Hyperledger Fabric* with support for *couchDB* queries to efficiently retrieve a paginated version of the order book for viewing by potential actors in the system. Decentralised offers for the sale of *Carboncoin* are only allowed to be deleted by the producer who submitted the offer for the sale. Moreover, the use of *couchDB* allows for rich queries of immutable data on the blockchain. Producers seeking to view the order book are allowed to specify individual viewing preferences - such as an acceptable price per token in fiat currency or a range of reputation scores associated with another producer selling *Carboncoin*.

The finalisation of the sale of *Carboncoin* amongst producers requires an offer with a sufficient number of *Carboncoin* owned by the seller. The buyer specifies the total number of *Carboncoin* they wish to purchase from the offer. A token swap pattern results in the actual token swap amongst the producers.

Since the order book is impelemented with support for rich queries against an on-chain *couchDB* the active offers on the market are viewable by each independent actor in the system. Moreover, a user is allowed to delete their own offers and have them marked as inactive on the blockchain and subsequently no longer viewable to other energy producers.

B. Carboncoin Supply

The supply of *Carboncoin* is controlled through the business logic of the deployed smart contracts and cannot be directly mainpulated by a regulator. Regulators can specify how much *Carboncoin* producers in different energy production brackets are entitled to upon registration - but no chaincode can be invoked to inject a surplus supply of *Carboncoin* onto the blockchain. Energy producers lose *Carboncoin* automatically using a burned token pattern whenever the carbon dioxide equivalence of energy production is reported. Algorithm 1 details how production is registered for the carbon market. Importantly, the system requires chaincode invocation to come from the ‘Energy Production Index’ smart contract (as seen in Figure 1). Moreoever, *Carboncoin* is always expensed automatically for a user if there is a sufficient balance to pay for production.

Algorithm 1 Carbon Production Registration

```
procedure REGISTERCARBON(producer, CO2e)
    invokerType = getInvoker()
    if invokeType not ESGIndex then
        reject transaction
    producer = getProducer(producer)
    add carbon producer.AddCarbon(CO2e)
    carbonCoinBalance = producer.GetBalance()
    if carbonCoinBalance > CO2e then
        producer.deductToken(CO2e)
        reportProduction(producer, CO2e, paid=True)
    else
        reportProduction(producer, CO2e, paid=False)
```

In the scenario where a user requires a significant amount of *Carboncoin* to pay for expected future energy production the carbon market smart contract can provide a producer with a redeemable token to purchase *Carboncoin* at a set price per token. Equation 2 details how the price per *Caroncoin* in fiat currency is calculated for an energy producer wanting to purchase more *Carboncoin* outside of the open market. Each x_i represents the price per token for offer i presently active on the market. A constant value of c (in our implementation $c = 50$) is added to actively disincentivise purchasing *Carboncoin* outside of the open market. In general, any value of $c > 0$ should provide enough of an incentive not to directly purchase *Carboncoin* from the carbon market smart contract. The value of $c = 50$ was picked in the implementation to provide enough of a buffer to ensure there was a decent economic cost to purchasing *Carboncoin* off the open market.

$$\text{Price Per Carboncoin} = \max\left(\langle x_1, x_2, \ldots, x_n \rangle\right) + c \tag{2}$$

The implementation of emissions trading on the blockchain is carefully designed to make *Carboncoin* as scarce as possible. Energy producers are required to pay a steep price for large amounts of emissions heavy energy production - whereas producers who have clean sources of energy are allowed to trade away surplus *Carboncoin* and make a profit from emissions trading. The underlying philosophy of the blockchain-based ETS is to drive long term adaption of environmentally positive sources of energy by making it costly to have emissions heavy production.

C. ESG Market Reputation

The implementation of *Carboncoin* and blockchain-based emissions trading requires having an ‘ESG Index’ for each user on the platform to increase market information and drive more ethical carbon trading. Creating a complete blockchain recording of ESG data requires the authentication of raw data and additional comparisons with an ESG report and a set of locale specific standards. As part of the implementation, the market assumed the existence of such automated fact-checking for ESG data and instead focused on the application of compiled ESG certificates for a carbon market.

Energy producers are allowed to submit any form of raw ESG data (such as the water quality in hydrogen production or the amount of money donated to charitable causes). The data is submitted into a command-line application which gets transformed into immutable data on the blockchain through the compilation of a final value - referred to generally as an ESG Index. Upon compilation of such data, the carbon market has specialised chaincode invoked by the ESG smart contracts to become notified of the creation of an ESG Index for a producer on the carbon market. The value for the ESG Index certificate is then multiplied by a particular weight and immutably stored on the blockchain so it can be accessed by chaincode in the carbon market. Finally, to calculate a reputation score of a producer the summation of the ESG Index values multiplied with weights is calculated and associated with an open offer on the blockchain. Algorithm 2 details how weights are applied to ESG data to produce a reputation score which is subsequently passed into the carbon market. Of note is how reputation scores are updated live, meaning Equation 1 is iteratively calculated when a new reputation score is provided. Moreoever, a strict policy contract is used to ensure that an update to Equation 1 for a user is only callable by an authorised user on the ‘ESG Channel‘ in Figure 1.

Algorithm 2 ESG Data Weight Update

```
procedure REGISTERESG(certificateId, value, producer, rawStatistic)
    invokerType = getInvoker()
    if invokerType not ESGChannel then
        reject transaction
    carbonMarketWeight = getWeightToken(certificateId)
    minValue = carbonMarketWeight.Min
    maxValue = carbonMarketWeight.Max
    weight = carbonMarketWeight.Weight
    if value < minValue and value > maxValue then
        reject transaction
    reputationUpdate = value ∗ weight
    invoke chaincode registerReputation(producer, reputationUpdate)
```

Each weight for an ESG index category is specified by a blockchain actor with the carbon market regulator role. Carbon dioxide equivalence from energy production always has a score of negative one. The regulator is free to specify the weight for other ESG Index categories. For example, having a good water score in hydrogen energy production could contribute a weight of positive two to the reputation score of a producer.

To assist in making reputation scores more transparent for producers in the market, a reputation breakdown for all individual sellers is viewable on the web application. A user is able to rapidly determine how another producer’s reputation score is derived from their contributions to Environmental, Social and Governance initiatives. Moreoever, a complete breakdown is presented to show how the raw ESG data for a producer was transformed into an ESG Index value and subsequently multiplied with a weight to get an ESG score. The final reputation score would subsequently be a summation of each ESG score given by the carbon market to uniquely reported raw data for a producer.

The reputation system is implemented to nudge other producers to purchase *Carboncoin*, and therefore the right to create emissions, towards producers who engage in positive environmental, social and governance initiatives. Reputation is tied into the interaction of a producer with the order book. A producer can specify a preference to view only *Carboncoin* sale offers within a specified reputation range. Moreoever, producers are allowed to efficiently find offers to fill a budget of *Carboncoin* with only offers with the best reputation scores being shown.

D. X.509 Certificates

The Certificate Authority provides an *X.509* certificate to invoke chaincode. The certificate has attached attributes, including the role of the user in the system to enforce access control in the smart contracts. A producer is allowed to use their enrolment secret to join the blockchain as a peer and invoke chaincode without requiring an intermediate API or web application. The use of *X.509* certificates allows the carbon market to continue to function in the case either the web application or HTTPS API were to stop functioning.

VI. Performance

To test the performance of the system, artificial loads on the most commonly used smart contract functions were generated. The tests were conducted on a single machine running three Fabric nodes. Moreover, the testing machine was running the Linux kernel version 5.15.2. Fabric had access to a CPU with 12 threads and 16GB of system memory. Each test was run once for a total of sixty seconds before the results were collated into a final output. The testing methodology was a fixed load rate controller maintaining a target backlog of five transactions. The results of the tests are the maximum possible TPS whilst also continually maintaining the pending five transactions in the backlog.

The results of testing are presented in Table I for reads and Table II for write operations. The dependencies between each test and the on-chain reads are shown in Figure 4.

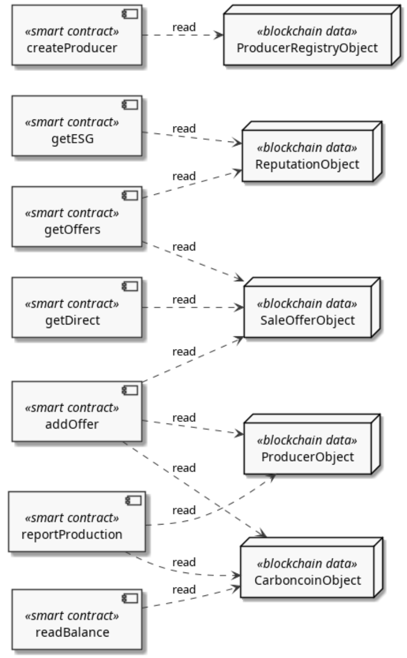
TABLE I
ON-CHAIN PERFORMANCE OF CARBON MARKET READ OPERATIONS

Test Name	Success	Fails	Send Rate (TPS)	Avg Latency (s)	Throughput (TPS)
getOffers	5937	0	100.4	0.06	100.4
readBalance	37166	0	629.8	0.01	629.8
getESG	25409	0	430.5	0.02	430.4

TABLE II
ON-CHAIN PERFORMANCE OF CARBON MARKET WRITE OPERATIONS

Test Name	Success	Fails	Send Rate (TPS)	Avg Latency (s)	Throughput (TPS)
createProducer	978	22	76.6	0.23	73.4
addOffer	869	37	15.1	0.45	14.8
reportProduction	774	60	14.1	0.48	13.6
getDirect	875	0	14.6	0.43	14.3

Fig. 4. Testing Dependencies



The smart contract functions *getOffers* and *getESG*, which make use of the performant *couchDB* on-chain index performed well. Moreover, these operations map to viewing the order book and retrieving the reputation breakdown of another energy producer on the platform. The reason why retrieving the offers is less performant than viewing production is a result of the decision to attach the market reputation of a firm to an active offer on the market, as seen in Figure 4. Performing an additional read for the market reputation of an offer slows down the query substantially compared to the 430 TPS of directly viewing the reputation.

Reading the balance of a user (*readBalance*) also performed well, since as Figure 4 shows only the user's *Carboncoin* stream of balance changes are read. *Carboncoin* assets are recorded as a stream of values which gets updated whenever emissions are recorded in energy production or users decide to purchase *Carboncoin* from other offers. Recording an asset as a stream of values ensures that multiple writes can happen to an on-chain asset when a block of transactions is submitted to the *Fabric* ordering service, providing more performance for write operations and avoiding concurrency errors.

Performance degrades when complicated write operations happen on the blockchain. The most performant write operation is the registration of a new energy producer in the system (*createProducer*), which allows for 73 TPS. Since the registration only has minimal reads in place to check for the existence of a producer. Figure 4 shows how *createProducer* only performs a final read to determine the *Carboncoin* allocation for an energy producer.

Performance begins to degrade substantially when doing more complex operations such as adding an offer for the sale of *Carboncoin* tokens. The function *addOffer* only exhibits a TPS of 15. The reason for such a low value is because of the extensive use of policy contract involved with the decentralised sale of *Carboncoin* to another user. The smart contract must first check if the user has enough tokens inside their account to create the sale. Moreover, the smart contract must also check existing offers on the market to ensure the user is not offering more tokens for sale than the amount contained inside their account balance. Such a sequence of reads is shown in Figure 4. Furthermore, adding an offer continually for the same producer can result in reads on information within the same block. Additionally, a recorded value such as the allowed sellable tokens may get updated multiple times. Such a scenario will result in a concurrency error which causes a transaction failure.

The carbon market operation *reportProduction* which writes carbon emissions for a user also exhibits low performance - specifically reaching a TPS of only 14. Figure 4 shows how *reportProduction* must first read a producer's details and then perform an additional read on their *Carboncoin* balance. Similarly, if an automatic deduction of *Carboncoin* has occurred due to carbon production then there will be a read on a range of values updated in the same block, which causes the transaction to fail. Moreover, the decision to use a single reputation score can result in frequent updates of the same value - leading to 60 out of 834 transactions failing.

Directly purchasing *Carboncoin* from the market (*getDirect*) further exhibits a low TPS value of 14. To create a direct purchase token the smart contract must first retrieve all open offers on the market and find the maximum sale value before providing a final price per *Carboncoin* to provide to the user, as seen in Figure 4.

VII. DISCUSSION

The proposed implementation of *Carboncoin* offers a complete blockchain application for the trading of carbon assets in a decentralised manner. Compared to centralised alternatives, it provides a more trustful application of carbon trading without the need to deal with centralised alternatives in a politically contentious market. Users are allowed to trade the right to emit carbon without the economic concern that the underlying asset will become manipulated through political intervention.

The market is made informationally efficient through additional blockchain assets. Producers are allowed to view the reputation breakdown of other participants to determine the complete ESG reputation of another user before making the decision to purchase carbon assets. Moreover, users are allowed to automatically submit carbon emissions through smart production devices (such as smart meters for recording energy production). After reporting production, producers can have *Carboncoin* automatically expensed and their reputation scores updated without any manual intervention. Producers are further allowed to refine their ESG reputation through manual submissions of ESG data.

A. Performance

Despite offering a complete implementation of a carbon market the performance of the application needs further refinement. The decision to host nearly the entirety of the business logic on-chain using industry standard patterns comes at the cost of reduced performance. Important write operations such as creating sale offers and reporting production exhibit unsatisfactorily low system performance as measured through transactions per second.

Such issues with performance can be mitigated by adopting more scalable blockchain architectures - for example deciding to record an asset as a stream of values. Although allowing for more performant writes, there is still the problem that in a complicated sequences of operations (such as a write then a read) errors will still occur. Despite representing some of the assets in the carbon market as a stream of values, phantom read conflicts were not able to be eliminated when testing the performance of the application.

A better alternative to deal with the low performance would be to adopt a hybrid architecture. Important assets such as *Carboncoin*, ESG data and reputation scores can remain on the blockchain. Instead of storing the entire order book on-chain, it would instead be more performant to move it off-chain with payment settlement being finalised on-chain. The order book is where the majority of the producer interaction with the system happens. If the decision was made to move the logic for interacting with the order book off-chain the performance could be substantially improved without largely impacting the trustfulness of the application.

B. Costs of Alternatives

The cost of any future decision to move the order book off-chain is the requirement to have a centralised component to ensure the continued operation of the carbon market. In the presented implementation of *Carboncoin* the application can continue to operate through the use of cryptographic credentials which are uniquely supplied to new users in the system. Producers are allowed to join the private carbon market blockchain as a peer and invoke chaincode similar to how they presently interact with the blockchain using a web application. Therefore, the price of centralisation is less user freedom, and subsequently trust in the operation of the market.

VIII. Conclusion

The proposed implementation for *Carboncoin* showed how blockchain carbon markets can have automatic expensing of divisible carbon assets through energy production. Moreover, it was able to show how the quality of market participant reputation scores can be refined through the use of ESG data which is stored on the blockchain. In the future it may be the case a blockchain framework will be released with high user trust and the ability to store the entirety of the order book on-chain. In such a scenario, having the entirety of the business logic for a carbon market on-chain would be reasonable. Future work on a blockchain carbon market employing ESG data will have to determine how the performance of the most commonly used chaincode operations for an ESG-based carbon market can increase. Regardless, having the ability to automatically operate a market without manual permits and also be able to succinctly observe the market reputation of other participants allows for a more comprehensive blockchain carbon market.

References

[1] The Economist, "The world urgently needs to expand its use of carbon prices," May 2020.

[2] X. Liu, H. Wu, W. Wu, Y. Fu, and G. Q. Huang, "Blockchain-enabled esg reporting framework for sustainable supply chain," in *Sustainable Design and Manufacturing 2020* (S. G. Scholz, R. J. Howlett, and R. Setchi, eds.), (Singapore), pp. 403–413, Springer Singapore, 2021.

[3] E. Kawasmi, E. Arnautovic, and D. Svetinovic, "Bitcoin-based decentralized carbon emissions trading infrastructure model," *Systems Engineering*, vol. 18, 03 2015.

[4] M. N. B. et al, "A blockchain-based smart grid: towards sustainable local energy markets," *Comput Sci Res Dev*, August 2017.

[5] K. N. Khaqqi, J. J. Sikorski, K. Hadinoto, and M. Kraft, "Incorporating seller/buyer reputation-based system in blockchain-enabled emission trading application," *Applied Energy*, vol. 209, pp. 8–19, 2018.

[6] P. Yuan, X. Xiong, L. Lei, and K. Zheng, "Design and implementation on hyperledger-based emission trading system," *IEEE Access*, vol. PP, pp. 1–1, 12 2018.

[7] A. Mumcular, *Blockchain meets natural gas: a case study of the University of British Columbia and Xpansiv from an operations research perspective*. PhD thesis, University of British Columbia, 2020.

[8] G. Kim, J. Park, and J. Ryou, "A study on utilization of blockchain for electricity trading in microgrid," in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 743–746, IEEE, 2018.

[9] F. Barth, "Certifhy-developing a european guarantee of origin scheme for green hydrogen," *Definition of Green Hydrogen*, vol. 19, 2016.

[10] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, EuroSys '18, (New York, NY, USA), Association for Computing Machinery, 2018.

[11] Q. Lu, X. Xu, H. M. N. D. Bandara, S. Chen, and L. Zhu, "Patterns for blockchain-based payment applications," 2021.