A Blockchain-based File-sharing System for Academic Paper Review

Ian Zhou
University of Technology Sydney
Sydney, Australia
ian.zhou@student.uts.edu.au

Imran Makhdoom
Food Agility CRC Ltd
University of Technology Sydney
Sydney, Australia
imran.makhdoom@student.uts.edu.au

Mehran Abolhasan
University of Technology Sydney
Sydney, Australia
mehran.abolhasan@uts.edu.au

Justin Lipman
Food Agility CRC Ltd
University of Technology Sydney
Sydney, Australia
justin.lipman@uts.edu.au

Negin Shariati
University of Technology Sydney
Sydney, Australia
negin.shariati@uts.edu.au

***Abstract*—As a tool for human technological advancement, the peer-review system acts as a gateway for ensuring academic paper qualities. However, the system has proven to be slow and expensive. Also, biasedness remains an unsolved problem. Such issues could become a major bottleneck, which can adversely impact research progress and dissemination of knowledge. This paper aims to propose a double-blind paper review system to preserve the authors and reviewers anonymity. This system also addresses issues concerning the reviewers payment, inconsistent review metrics, and biased reviews. The proposed solution utilizes the Hyperledger Fabric blockchain with the InterPlanetary File System (IPFS). The blockchain smart contracts provide a base for financial transactions between paper publishers and the reviewers. Hence, we introduce AcadCoin, a novel cryptocurrency used for supporting said financial transactions. Also, the Hyperledger blockchain provides user access control to achieve double-blindness in reviews. Along with the Hyperledger blockchain, the IPFS is used to store the paper documents, review documents and open metrics documents to reduce the storage requirement of the blockchain. A broad system architecture is constructed to combine the blockchain and the file storage system. This system architecture distributes nodes of the system to related parties. Finally, the blockchain network is implemented and tested using the Hyperledger Composer Playground environment.**

***Index Terms*—Access control, blockchains, distributed databases, review systems**

I. Introduction

The peer-review system is an essential component in modern academic publications. As human technology advances with new ideas driven by academic papers, a requirement for improving the current peer-review system has emerged. The author in [1] pointed out that the current system is "slow and expensive." The primary cause of such inefficiency is reviewers. Even though the opportunity cost for reviewing a paper is high, most of the time, reviewers do not receive any payments for paper reviewing [1]. Thus, reviewers' lack of motivation could slow their review process. On the other hand, inconsistent review metrics created by the subjectivity and bias of different reviewers could also be another origin of the problem [1]. A decentralized peer-review system could be a solution to speed up the review process and also provide some reward to the reviewers.

The blockchain has some inherent benefits such as decentralization, immutability, and auditability [2]. These properties allow the construction of a decentralized paper review system. This would end the monopoly of academic journals on accepting paper reviews and prevent unfair decisions made by any centralized malicious party [3]. Also, this system could potentially give reviewers a choice for receiving rewards. The authors in [4] indicated that information sharing is more efficient using blockchain. This property makes blockchain a viable option to speed up the current peer-review process.

Cloud storage and distributed storage are considered as appropriate storage options to store the files shared on a blockchain-based peer-review system. The authors in [5] demonstrate that the issue of storage space on mobile devices can be solved by the use of cloud servers with cryptography schemes from the blockchain. In this way, data is no longer stored on the end devices, thus reducing storage requirements on devices like mobile phones. Another solution could be storing academic documents to different nodes of a distributed network [6]. This decouples the academic papers from the blockchain and results in a slim and efficient block architecture.

To address the problems and technical issues mentioned above, a blockchain-based paper review system is proposed. The proposed solution intends to resolve the opportunity cost for the reviewers, by introducing a novel cryptocurrency, the AcadCoin. Another significant contribution of this paper is to address the issue of subjectivity and biasedness of the reviewers. This problem is tackled by the double-blinded review induced by the blockchain user access control and most importantly by introducing the open metrics. As indicated above, the blockchain may have heavy storage requirements for mobile devices. In the proposed system, large files are decoupled from the blockchain and stored in the InterPlanetary File System (IPFS) in a distributed manner. To further reduce storage requirements on the user side, the proposed system contains a cloud gateway storing the blockchain on cloud storage.

The paper is organized into seven sections. Section II explores the current method of storing large records in blockchain systems and identifies flaws in current studies of online peer-review systems. Section III introduces a broad structure of the proposed blockchain peer-review system. Then, the different algorithms involved in the blockchain smart contracts are demonstrated in Section IV. Section V describes the sequence of events included in the review process. In Section VI, the test cases and results are presented to prove the system concept. Finally, the paper is concluded in Section VII.

II. Related Work

The authors in [7] mention that the reputation of academics is essential for any institution. A system to store educational record and reputation could be one type of application for blockchain. However, storing a large amount of data could lead to the problem of bulky blockchain, thus putting high storage requirements on the nodes. There is a need for storage optimization of blockchain [7]. Also, novel cryptocurrency schemes have been proposed to reduce the need for large storage sizes. As the size of data increases, demand for a novel method of data storage used by blockchain applications has emerged [8].

In another work [9], the authors used a centralized cloud system to store the blockchain. User access control is achieved by encrypting the data stored within the related blocks. These data are managed as temporary content on the user-side. Additionally, temporary content management is implemented on smartphone gateway apps, which are responsible for personal data management and verification of all data requests. This is a system with one centralized node accessed by many mobile gateways.

The authors in [10] further decouple the data and the record of modifying data. As a result, cloud storage is only used as a data warehouse. A distributed blockchain network is constructed to store operations such as queries and updates. Initiators of these operations are users, health care providers, and insurance companies. To provide security for user operations, the access control for user systems is achieved by using the Hyperledger fabric membership service and the channel scheme.

Similarly, the authors in [11] proposed a multi-module blockchain system. The client module is responsible for encrypting users uploaded data. The endorser checks the integrity of the data signature. Orderers in different hospitals ensure consensus between different nodes. Finally, the committer adds new nodes to the blockchain ledger. User access control is achieved by encrypting data with the users public key. Furthermore, the users private key with the departments private key creates signatures to ensure data integrity.

In another work, a distributed cloud storage approach is suggested to address the storage problem [8]. In the proposed file trading system, file storage and the blockchain trading system are separated, whereas users files are separated into chunks. Each chunk is encrypted and uploaded to a P2P network, where users can locate these file chunks with the file hashes and file location URLs stored in the blockchain.

The use of the Interplanetary File System (IPFS) for file storage enhances data integrity and availability [12]. To model the ownership of data, the data owner and data user entity are implemented for performing data transactions. A pair of shared key and secret key is used for controlling data access on smart contracts.

Incompressible imaging studies are another issue inducing high storage demand [13]. As a result, this limits node deployment on mobile devices and it is also vulnerable to successful attacks on the encryption algorithms used on health records. If an external party penetrates the encryption algorithms, all medical data of this block will become public. Therefore, this system separates the imaging data and the block meta-data that is stored in a distributed public database.

The authors in [5] worked out a solution for data storage using multiple clouds. A virtual mobile terminal is associated with each mobile device to share files and applications across the network on mobile devices. These virtual mobile devices run on clouds, which provide users with enough storage and processing power. Blockchain is used to facilitate the data sharing process between users.

The authors in [14] introduced a cloud-based knowledge sharing system. Each enterprise or organization is required to host a private cloud database for knowledge data storage, whereas the data is stored in a knowledge blockchain and a transaction blockchain. The knowledge blockchain stores a preview for the knowledge, the knowledge access URL, and a smart contract with knowledge sharing conditions. The transaction blockchain has records of transaction approval and validation between different parties.

In order to integrate the reputation system with a file-sharing system, a blockchain-based system to store single dimension reputation was created [15]. This primitive reputation system only provides two numbers of scores to represent the successful or failed reception of the requested file.

Correspondingly, the authors in [16] enhanced the reputation system by creating incentives for data sharers. This Ethereum-based system awards data sharers the Ether cryptocurrency for sharing data with others. Data sharers can define the types of data to be shared and the applications accessible to the data. An institutional review board (IRB) assesses the eligibility of users to access data.

Similarly, the authors in [6] extended the idea of an incentive network and implemented an online publication process using blockchain. This allows authors to create smart contracts for granting permission to publishers for publishing digital content. These digital contents are stored in the IPFS file-sharing system in a distributed network.

The BMIF journal uses a centralized system for peer-review [17]. In this system, the peer-review process for each paper is double-blinded for reviewers and authors, thus mitigating biasedness from reviewers. However, this system does not provide public metrics for reviewers to follow. It also fails to address the opportunity cost of the reviewers.

The authors in [18] proposed a procedure to extend the ScholarOne Manuscript system. The system adopted a double-blinded review process to ensure openness and fairness during peer-review. To further mitigate biasedness and subjectivity, the journal editors are able to connect directly with the reviewers and the authors through the review system. Also, through the editor, the reviewer is able to obtain thoughts from the authors. This creates a relatively transparent peer-review process. On the other hand, interactions between editor, reviewer, and the author could increase the time of the review process and ultimately increase the unmentioned opportunity cost for reviewers.

To create more blindness between conference chair members, reviewers, and authors, the Privacy-Preserving peer-review System (P3ERS) was introduced [19]. This distributed system adds another blindness to the double-blind review process. It is achieved with the group signature scheme. Moreover, the third blind property ensures that the program chair does not know the list of members of the authors and the exact assignment of papers to reviewers. This increases objectivity during the review system. Unfortunately, the system failed to cover the cost for reviewers to review a paper.

Similarly, the authors in [20] provided a cloud-based solution for peer-review systems. It protects academic work and data from the cloud owner through encryption schemes. In summary, this provides a review system similar to the double-blind process. It also ensures privacy and security over the authors and reviewers.

The authors in [21] designed and implemented a web-based centralized peer-review system. The system contains four main modules. The guest module describes any unregistered users. The guests will be upgraded to the authors after registration and authors can submit papers. To become a reviewer, a request should be sent to the editor from the author. Ultimately, this system provides double-blindness during review through the usage of PHP and SQL based web interface.

Erie, a review system implemented in Python, partially automated the process of paper distribution to reviewers [22]. The system used Latent Semantic Indexing to calculate a suitability score between a paper and reviewers. The indexing system uses the parameters calculated from the reviewer's past publications. Further digitization of the paper allocation process increases the speed of the paper review process.

MaRSChain is a blockchain-based peer-review system consists of two types of blockchains [3]. The conference blockchains (CBC) maintain a list of papers submitted to different channels and the publishing house blockchain (PHBC) contains a list of published work in all channels. Also, the PHBC keeps a list of under-review papers. Double-blindness is achieved with the encapsulation of data in a smart contract.

TABLE I
COMPARATIVE ANALYSIS OF THE EXISTING LITERATURE ON PEER-REVIEW SYSTEM

System Name/Reference	**Distributed**	**Double-blindness Review**	**Reviewer Cost**	**Open Metrics**
BMIF Journal System [17]	X	✓	X	X
ScholarOne-based [18]	X	✓	X	X
P3ERS [19]	✓	✓	X	X
ConfiChair [20]	X	✓	X	X
Zakho System [21]	X	✓	X	X
IEEE INFOCOM System [22]	X	✓	X	X
MaRSChain [3]	✓	✓	X	X
The proposed System	✓	✓	✓	✓

Table I analyses and compares the peer-review systems mentioned above. These systems focus on a centralized approach with the double-blind review for mitigating reviewers' bias. However, these literature did not address reviewers' costs and did not provide an open platform for presenting metrics to mitigate bias further. Also, only one solution is implemented in a distributed manner, spreading some trust among the users.

To address the issues mentioned above, we propose a blockchain-based review system that utilizes Hyperledger Fabric blockchain for storing access information to different papers and reviews. Such a solution distributes trust among the authors and reviewers. This approach also pushes conference organizers to address the opportunity cost of reviewers. Moreover, due to the immutability of the blockchain, changes to the review metrics will be noticed by every user in the system, thus reducing bias from reviewers.

III. Proposed Architecture for the Academic Paper Review System

In this section, a high-level architecture of our proposed blockchain-based paper review system is presented. Fig. 1 demonstrates the file storage and access mechanisms in the proposed system. As can be seen, the IPFS system is used to store all the papers, reviews, and review metrics files. Users are able to find the relevant fingerprint of the files from the blockchain system and access these files from the IPFS system using respective fingerprints. The rest of this section describes the role of the IPFS system and Hyperledger blockchain.

The IPFS file system is a distributed file storage system that allows users to access files from multiple sources using a content-related hash code [23]. In the proposed system, there are three types of files stored using IPFS. The first file type includes the research papers for review and the published full papers. To ensure double-blindness, authors should not include any information related to their identity on the papers submitted for review. Also, both files are encrypted to preserve the confidentiality of user data. The second file type is the review metrics. This document is provided by the conference organizers to provide open review metrics for mitigating review bias. The format and layout of this file depend upon the organizers of the conference. The final file type is reviews, which are written by chosen reviewers. There

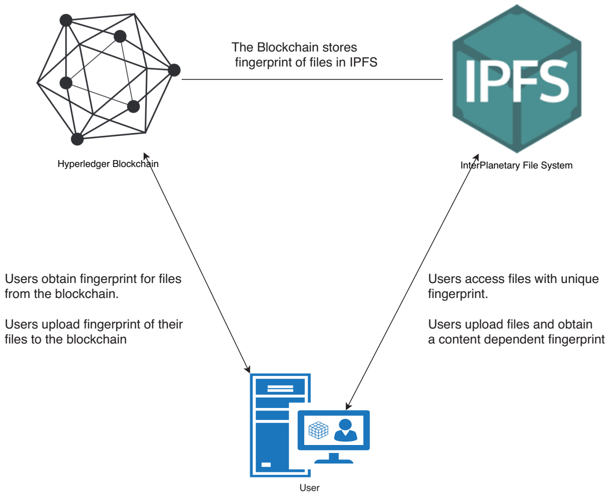

The Blockchain stores
fingerprint of files in IPFS
IPFS
Hyperledger Blockchain
InterPlanetary File System
Users obtain fingerprint for files
from the blockchain.
Users upload fingerprint of their
files to the blockchain
Users access files with unique
fingerprint.
Users upload files and obtain
a content dependent fingerprint
User

Fig. 1. File storage architecture.

is no restriction on the format of the reviews, as long as it follows the metrics provided by the conference organizers.

The advantage of using the IPFS to store large file types is to reduce the size of the blockchain. As sharing files on the blockchain create redundancy on all nodes, the IPFS storage significantly reduces block size when executing file-related transactions. Therefore, the blockchain can function in a faster and more efficient manner.

A. The Hyperledger Blockchain

The Hyperledger blockchain is a consortium blockchain that provides access control through certificate authorities (CA) [24]. Including the CA, there are four types of nodes in the Hyperledger architecture. The CA is responsible for issuing digital identities to different entities in the system. There is a root CA issuing certificate to intermediate CAs. The intermediate CAs issue certificates to end-users. In this manner, the system can spread out the processing load and create a chain of trust. The peer node is a node with a complete copy of the blockchain. It is also responsible for running transactions. A special type of peer node is the endorser node, which is responsible for validating the submitted transactions [24]. Finally, orderer nodes ensure that all transactions are in the correct sequence. Orderer nodes are also responsible for consensus in the system by broadcasting changes to all peer nodes. Overall, the role of the Hyperledger blockchain in the proposed system is to achieve double-blindness through the user access scheme, mitigate review bias through publishing an open review metrics and address reviewer's cost through issuing of the AcadCoin, an integral digital currency.

B. The Broad System Architecture

Fig. 2 demonstrates the broad architecture of the paper review system. The proposed architecture consists of four essential components with different roles. The system admin maintains the root CA, which is responsible for issuing intermediate CA certificates, network updates, and cryptocurrency management. The admin is the only entity that can create new coins for the system.

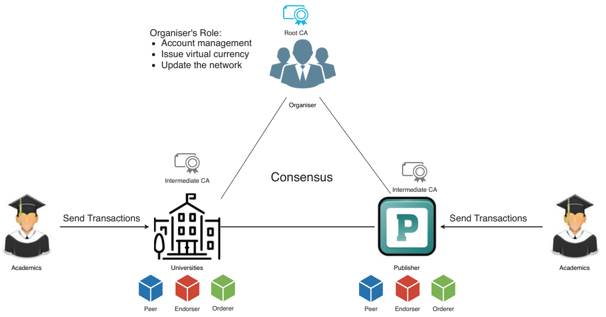


Fig. 2. The broad organizational architecture.

The universities and publishers components of the system all have intermediate CAs to invite other users to join the network. Also, these organizations provide the peer, endorser, and orderer nodes for running basic blockchain functionality. Finally, the academics and other users can access the blockchain network through the peers owned by Universities and Publishers. All of these components in the system have an IPFS server for uploading and downloading documents. As all users are known and controlled by the CA, the blockchain system can trace back to the accounts used by attackers. Furthermore, accounts compromised by the attackers and used for DDOS attacks can be identified based on the certificates provided by the CA. These accounts will be temporarily cast out of the system to protect the availability of the services to other users.

IV. Smart Contract

This section describes the algorithm for every smart contract in the proposed system. In order to implement these smart contracts, the entities and participants involved in every contract are defined. It is followed by the creation of an access control scheme. Based on these participants and schemes, the smart contracts of this blockchain system are defined by seven algorithms. These algorithms thus form the basis for building a review system with open review metrics, double-blindness, and addressing reviewer cost.

A. Blockchain Resources Model

In Hyperledger, users of the system are modeled as Participants. Participants can interact with other entities in the system. These other entities are modeled as Assets, which can be modified by the Participants and can also be transferred among different Participants. Figure 3 demonstrates the relationship between Participants and Assets. There are three types of participants or user types, including the Publisher, the Academic, and the Reader. All of these user types can buy papers to obtain the IPFS hash code and a decryption key for accessing the paper. However, only the Publishers can create a conference, assign reviewers to different submissions of papers and publish papers from these submissions. The Academics can request to be a reviewer for a conference and

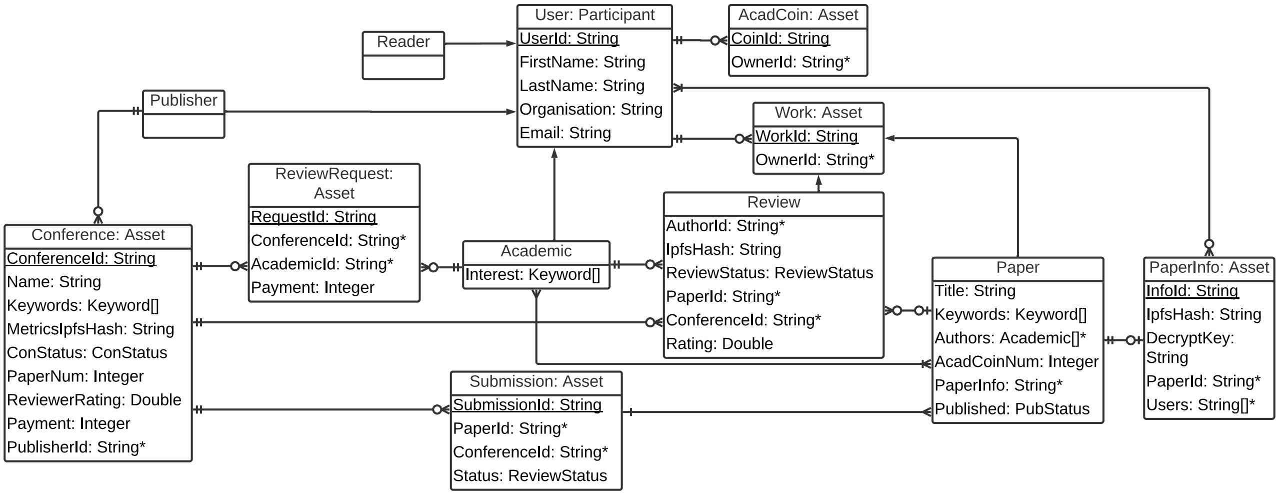


Fig. 3. Hyperledger blockchain resource model.

review an allocated paper. Finally, Readers only have the basic functionality of buying and reading papers.

For users to perform these activities, different assets need to be defined. In the proposed review process, a Conference asset is created by a publisher for modeling a real-world conference. Academics can create Paper assets and the PaperInfo assets that define the hidden information of a paper. The hidden information includes the IPFS hash code and the decryption key. With an unpublished paper, an Academic can create a Submission asset to submit the unpublished paper to a conference. If academics are interested in becoming a reviewer at a conference, they can create a ReviewRequest asset. With all these requests, publishers can assign reviewers to review submissions and pay them AcadCoins for reviewing papers. Moreover, AcadCoins can also be used by all users to buy papers from the paper owners.

B. Custom Resource States and Variable

To prevent human input errors during the operation of the blockchain system, there are four custom enums defined. The first enum is Keyword, that is used to describe the keywords for a paper and a conference. Also, it is used to define an academic's interest. The Keyword enum contains keywords related to different topics, such as “IoT”, “Blockchain”, “AI”.

The second enum ConStatus depicts the conference status. A conference can be “Open” for paper submissions and review requests, or “Close” for no further submissions and requests. Finally, when the ConStatus is “End”, no modifications to the conference are allowed. The third enum ReviewStatus describes the status of a certain review and the status of a paper submission asset. All reviews and submissions start with a “Pending” status waiting for people to review. After publisher's or academic's revision, they can set the status to “Fail” or “Pass”, representing rejection or acceptance of the paper respectively. The final PubStatus enum defines the publication state of a paper. Papers should start with the “NotPub” state, indicating that it is not published. In this state, only the author of the paper and publishers can read the abstract information in that asset. After a paper is submitted to a conference, the status changes to “Submitted”. This state ensures that no one can modify the content of the paper. Finally, a paper will be “Published” if it is accepted by a conference and the information in the Paper asset will be available to all users.

TABLE II
ACL RULE DEFINITION [25]

Variable	Description
description	A comment to describe the ACL rule.
participant	Defines the initiator of the operation.
operation	Defines the CRUD operations to be considered for the current rule.
resource	Defines the resource to operate on.
transaction	An optional variable. If defined, this rule must be considered in a transaction.
condition	The condition to be met for the action to be performed. Optional
action	To deny or allow the operation when the condition is met.

C. Asset Accessibility

This subsection illustrates different access rules for numerous assets, controlling different participants. Hyperledger uses Access Control Language (ACL) [25] to exercise access control over the information stored on the blockchain. Table II defines the building blocks of an ACL rule. The blockchain system consists of multiple ACL rules to control access to the information. When there is an operation to perform, the system will check the ACL list from top to bottom until it finds a match for the participant, operation, and resource. If there is no matching ACL rule for an operation, the operation will be denied [25].

There are eight sets of rules corresponding to eight different types of assets. The first set is shown in Table III defines the

TABLE III
User ACL Rules

Participant	Resource	Operation	Condition
User	User	UPDATE	The resource matches the identifier of the participant.
User	User	READ	N/A

TABLE IV
Conference ACL Rules

Participant	Resource	Operation	Condition
Publisher	Conference	CREATE, UPDATE	The conference publisher matches the identifier of the participant. **AND** Conference status cannot be “End”.
User	Conference	READ	N/A

TABLE V
Paper ACL Rules

Participant	Resource	Operation	Condition
Academic	Paper	CREATE, UPDATE	The paper owner matches the initiator of the Action. **AND** The paper is not submitted or published.
Academic	Paper	READ	The paper owner matches the initiator of the Action. **OR** The paper is published.
Publisher	Paper	READ	N/A
Reader	Paper	READ	The paper is published.

TABLE VI
PaperInfo ACL Rules

Participant	Resource	Operation	Condition
User	PaperInfo	UPDATE	The paper owner of the info matches the initiator of the Action. **AND** The paper is not submitted.
User	PaperInfo	READ	User exists in the subscribed users list.

TABLE VII
Submission ACL Rules

Participant	Resource	Operation	Condition
Publisher	Submission	READ	The publisher is the initiator of the conference. **OR** The conference has ended.
Academic	Submission	READ	The Academic is the initiator of the submission. **OR** The conference has ended.
Academic	Submission	DELETE	The Academic of the submission is the initiator of the Action. **AND** The conference is in open state.
User	Submission	READ	Conference has ended.

user information accessibility. There are two rules in this set. The first rule defines that all users can only update themselves. Rule two dictates that all users are able to read each other’s information.

Table IV sets the access control rules for a particular conference. The first rule ensures that the Publisher can only create, and update conferences initiated by them or for which they are the publishers. Also, they can only create and update conference when the status is “Open” and “Close”. This ensures that when the conference ends, nothing is modified to destroy the finality of an ended conference asset. To ensure everyone knows about the conference, READ permission is granted to all the users on the Conference assets.

The Paper asset only stores basic details of a paper. It does not store information related to the accessibility of a paper. However, Paper assets contain information related to the authors of the paper, which could be a threat to double-blind review. Moreover, readers and academics can only read published papers. For paper management, Publishers can read all published and unpublished papers. Also, only Academics are allowed to create and update their paper if the paper is not submitted or published. These two rules are demonstrated in Table V.

The PaperInfo asset contains confidential information to access and decrypt a paper document. This information should only be available to the paper owner and users who bought the paper. ACL rules (Table VI) ensures that only the owner of the paper can update the PaperInfo and only subscribed users can read the PaperInfo. Also, following the paper asset rule, this portion of information should not be changed during the submission stage but could be changed after published for version update purposes by the publisher.

The accessibility of paper submission is an essential part of the double-blindness scheme. As shown in Table VII, there are four rules to limit access to the paper submissions. Rule 1 ensures that publishers can only read submissions to their conferences or the conferences that have already ended. For academics, as they are also the potential reviewers, they are restricted to only reading their submissions. Academics can also delete their submissions during the “Open” state of a conference if they decide not to submit to that conference. Finally, all submissions are available to the public after the conference ends.

The Review ACL rules (Table VIII) allow only the author of the review and the publisher of the conference to access the review before the conference ends. This is another pedestal to achieve double-blindness. After the conference ends, all reviews will be readable for all the users of the system. This prevents the reviewers from plagiarizing the contents of the paper they have reviewed. Review requests are confidential and can only be read by the publisher of the conference and the submitting academic. The academics can also delete their requests during the “Open” stage of a conference. Table IX presents the conditions of these ACL rules.

The AcadCoin is the integral currency of this system for users to buy papers and publishers to pay reviewers. An AcadCoin can only be created by the system administrator.

TABLE VIII
REVIEW ACL RULES

Participant	Resource	Operation	Condition
Academic	Review	READ	The author of the review is the initiator of the Action. **OR** The conference has ended.
Publisher	Review	READ	The publisher is the owner of the review. **OR** The conference has ended.
Publisher	Review	UPDATE	The publisher is the owner of the review. **AND** The conference has not ended.
User	Review	READ	Conference has ended.

TABLE IX
REVIEWREQUEST ACL RULES

Participant	Resource	Operation	Condition
Academic	Review-Request	DELETE	The academic is the initiator of the request. **AND** The conference is still open.
Academic	Review-Request	READ	The academic is the initiator of the request.
Publisher	Review-Request	READ	The publisher is the initiator of the conference.

TABLE X
ACADCOIN ACL RULES

Participant	Resource	Operation	Condition
User	AcadCoin	READ, UPDATE	The owner of the coin is the action initiator

Furthermore, Users in the system can only read and give their coins to another user through the change of ownership (Table X). All the ACL rules demonstrated in this subsection are disabled in a transaction. Assess control within transactions is achieved by algorithms in the next subsection.

D. Transaction Algorithms

Transactions are predefined smart contract algorithms. We have developed the following seven algorithms. The algorithm for creating PaperInfo (*createPaperInfo*) is used by authors for creating a PaperInfo asset containing the IPFS hash code to the paper document. To access PaperInfo, users buy the papers with the algorithm for paper subscription (*paperSubscription*). Moreover, the algorithm for paper submission (*submitPaper*) is initiated by academics to submit a paper to a conference. The algorithm also checks if the academic has already requested to be a reviewer to prevent any potential bias. To submit a request to be a reviewer, an academic needs to start the algorithm for reviewer request (*requestReview*), which also checks if the academic has a paper submitted in the conference to prevent any unfairness during the review process. When there are enough papers and reviewers for a conference, the conference status would be set to “Close” and the algorithm for paper allocation to reviewers (*allocatePapers*) would automatically allocate papers to reviewers and ensure all papers are reviewed. Then, the algorithm for review submission (*submitReview*) is utilized by reviewers to submit their review. Finally, the algorithm for paper submission assessment (*assessPaper*) helps publishers to accept or deny a paper. The source code of the algorithms is located in the GitHub repository: https://github.com/BlockchainReview/BlockchainReview.git

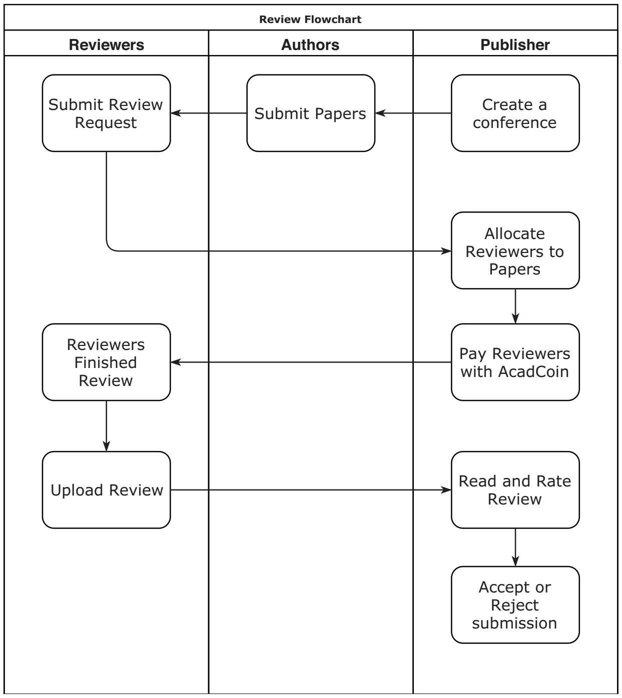


Fig. 4. The review process.

V. THE REVIEW PROCESS

This section describes the complete process of the paper review. Figure 4 presents the flowchart for the review process. First of all, a conference asset is created by a publisher. Then, the authors submit papers using the *submitPaper* algorithm. At the same time, academics submit requests to become the reviewer using the *requestReview* algorithm. Once there are enough papers and review requests, publishers use the *allocatePapers* transaction to pay the reviewers and allocate papers to the reviewers. Once all the reviewers finish their review, they upload it to the IPFS server and update the IPFS hash code through the *submitReview* transaction. Next, the publisher reads and rates these reviews. Finally, the publisher uses the *assessPaper* smart contract to accept or reject the submitted papers.

VI. EXPERIMENTS AND TESTING RESULTS

The proposed paper review system is evaluated using the Hyperlegder Composer Playground, a simulator for blockchain business networks. The evaluation is based on three test items. The objective of the first test item is to ensure that the system provides an open review metrics. Secondly, the system ensures double-blindness between paper authors and paper reviewers. Finally, it is tested whether the system accurately sends the

TABLE XI
RELEVANT ASSETS AND PARTICIPANTS DURING TESTING

Name	Asset/Participant Type	Description	Asset ID
Publisher	Publisher	The Publisher of the Conference.	1
Conference	Conference	The conference owned by the publisher to accept papers.	1
Paper	Paper	A paper submitted to the conference.	1
PaperInfo	PaperInfo	The paperInfo of the paper.	1
Submission	Submission	The paper Submission to the conference.	4befd1b4-feeb-42c4-a320-b29e7198f1ab
ReviewRequest	ReviewRequest	A reviewRequest to the Conference.	fdbaa4b5-27c9-47e3-b01c-e78cd5bcd86d
Paper Academic	Academic	The Academic Who Wrote and submit the paper.	2
Neutral Academic	Academic	An Academic not a part of the review process.	3
Reviewer Academic	Academic	The reviewer who sent the ReviewRequest.	4
N/A	AcadCoin	The AcadCoin to pay the reviewer.	1

payments to the reviewers. The Hyperledger Composer has the provision to define certain assets and participants for the test scenario. Table XI shows a list of assets and participants, defined for the test environment. Asset IDs are demonstrated partially only to be distinguished within an asset/participant type. To ensure readability, the font size and structure of all figures in this section are adjusted from the actual Hyperledger Composer Playground output. Also, to limit the paper length, empty results and registry are not demonstrated in the form of figures.

The sequence of events for building the test scenario encompasses the following steps. In step 1, the system administrator (Admin) creates all the relevant participants. This includes the Publisher for hosting a conference, the Paper Academic for writing and submitting a paper, the Reviewer Academic to act as a reviewer for a conference and the Neutral Academic for testing the accessibility of review-related assets. In step 2, the Admin creates an AcadCoin and allocates it to the Publisher. Then, with the identity of the Publisher, a Conference asset is created. The Conference has one keyword "Blockchain" with the *PaperNum* equal to 1 and *ReviewerRating* as 1. To test open review metrics, a .docx file is uploaded to the IPFS file system with the IPFS hash code "QmWrkYX5sdUXMbnDgSDDSd3muquc7HDUJYaTEZH9T mgwtx". The publisher of this conference is the predefined Publisher. After switching the role to the Paper Academic, this participant creates a test Paper asset with the keyword "Blockchain". This keyword matches the keyword of the Conference created above. A PaperInfo is also created using a transaction by the Paper Academic. The PaperInfo contains the IPFS hash "QmfC7t5rStBGCdZujU4t5Af1j9nefyqGxVn82QPqGkQPY5" for the Paper document. The decryption key is blank in this test case. Next, this paper is submitted with a transaction to the Conference in step 5. Then, in the role of the Reviewer Academic, a review request is submitted to the Conference for that academic to become a reviewer. Finally, the Publisher runs a transaction for allocating reviewers to submitted papers. This transaction also pays the reviewers a predefined amount of AcadCoins.

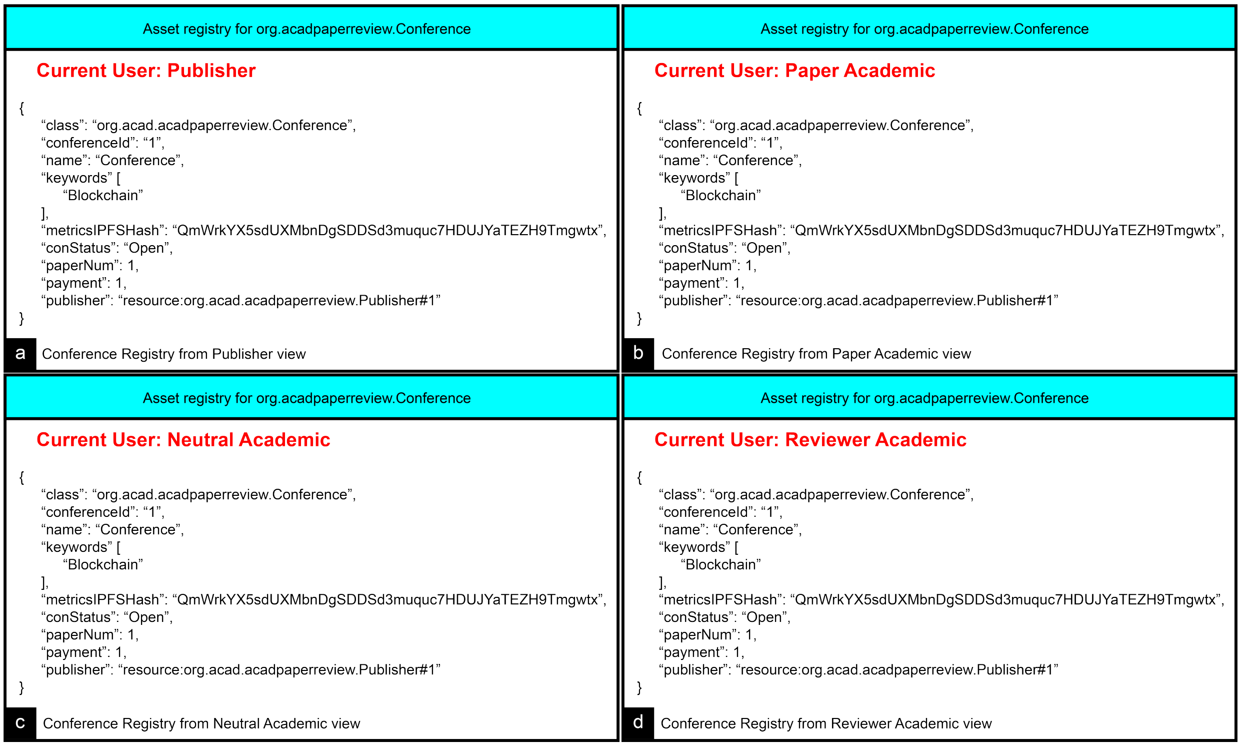
A. *Test Item 1: Open Review Metrics*

This test item demonstrates an open reviewer metrics document that is accessible by the public. The result is collected after step 3, in which the Publisher creates a conference asset. Fig. 5 is the conference view of the Publisher, Paper Academic, Neutral Academic and Reviewer Academic. To conclude, all of these participants have near identical views. This indicates that all parties can obtain information on the conference and retrieve the review metrics using a common IPFSHash data field.

B. *Test Item 2: Double-blindness*

Double-blindness ensures that paper authors and reviewers are unaware of the identity of each other during the review process. In the proposed system, the paper asset and submission asset are the two types of assets that contain author information. Consequently, the reviewer's identity is available in the request for reviewing and the review document asset. Results for this test item are obtained after step 7 of the test process when all reviewers are allocated to a paper for review. Fig. 6a is the Paper Academic's view of his/her paper. It contains the system identifier, which leads to the author's information. Before paper publication, only the paper authors and the publishers can view a Paper asset. Therefore, in the test case, reviewers were not able to access this information and the registry remained empty. For the Submission asset, the initiator of the submission is the owner of a paper. The owner's identifier is recorded in the Submission asset (Fig. 6b). A submission can only be read by the publisher of the conference and the owner of the paper. Similar to the Paper asset, access is also denied for reviewers to read this information. This also returned with an empty registry in the test scenario. Therefore, the author's identity has been proven to be hidden from the reviewers during the review process.

The reviewers' identity should also be hidden from the authors. Fig. 6c shows that the identifier of the reviewer can be found in the academic data field of a ReviewRequest asset. With the limitation of ACL rules, an author is not able to retrieve this information and trace back to the identity of the reviewer. In the test scenario, the access of the ReviewRequest asset from the Paper Academic returned with an empty registry. Moreover, only the conference publisher and the request initiator can access this asset. After the publisher sends a paper



Asset registry for org.acadpaperreview.Conference

Current User: Publisher

```
{
    “class”: “org.acad.acadpaperreview.Conference”,
    “conferenceId”: “1”,
    “name”: “Conference”,
    “keywords” [
        “Blockchain”
    ],
    “metricsIPFSHash”: “QmWrkYX5sdUXMbnDgSDDSd3muquc7HDUJYaTEZH9Tmgwtx”,
    “conStatus”: “Open”,
    “paperNum”: 1,
    “payment”: 1,
    “publisher”: “resource:org.acad.acadpaperreview.Publisher#1”
}
```

a Conference Registry from Publisher view

Asset registry for org.acadpaperreview.Conference

Current User: Paper Academic

```
{
    “class”: “org.acad.acadpaperreview.Conference”,
    “conferenceId”: “1”,
    “name”: “Conference”,
    “keywords” [
        “Blockchain”
    ],
    “metricsIPFSHash”: “QmWrkYX5sdUXMbnDgSDDSd3muquc7HDUJYaTEZH9Tmgwtx”,
    “conStatus”: “Open”,
    “paperNum”: 1,
    “payment”: 1,
    “publisher”: “resource:org.acad.acadpaperreview.Publisher#1”
}
```

b Conference Registry from Paper Academic view

Asset registry for org.acadpaperreview.Conference

Current User: Neutral Academic

```
{
    “class”: “org.acad.acadpaperreview.Conference”,
    “conferenceId”: “1”,
    “name”: “Conference”,
    “keywords” [
        “Blockchain”
    ],
    “metricsIPFSHash”: “QmWrkYX5sdUXMbnDgSDDSd3muquc7HDUJYaTEZH9Tmgwtx”,
    “conStatus”: “Open”,
    “paperNum”: 1,
    “payment”: 1,
    “publisher”: “resource:org.acad.acadpaperreview.Publisher#1”
}
```

c Conference Registry from Neutral Academic view

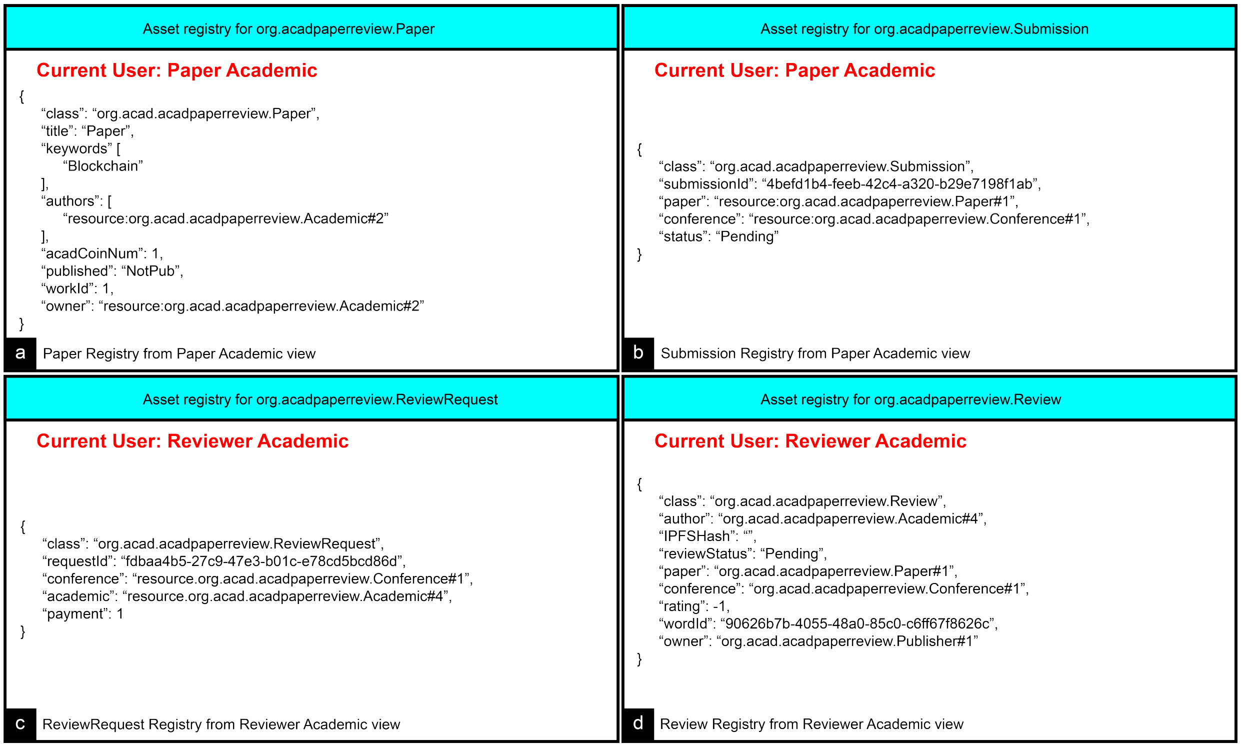
Asset registry for org.acadpaperreview.Conference

Current User: Reviewer Academic

```
{
    “class”: “org.acad.acadpaperreview.Conference”,
    “conferenceId”: “1”,
    “name”: “Conference”,
    “keywords” [
        “Blockchain”
    ],
    “metricsIPFSHash”: “QmWrkYX5sdUXMbnDgSDDSd3muquc7HDUJYaTEZH9Tmgwtx”,
    “conStatus”: “Open”,
    “paperNum”: 1,
    “payment”: 1,
    “publisher”: “resource:org.acad.acadpaperreview.Publisher#1”
}
```

d Conference Registry from Reviewer Academic view

Fig. 5. Conference access for the publisher and academics.



Asset registry for org.acadpaperreview.Paper

Current User: Paper Academic

```
{
    “class”: “org.acad.acadpaperreview.Paper”,
    “title”: “Paper”,
    “keywords” [
        “Blockchain”
    ],
    “authors”: [
        “resource:org.acad.acadpaperreview.Academic#2”
    ],
    “acadCoinNum”: 1,
    “published”: “NotPub”,
    “workId”: 1,
    “owner”: “resource:org.acad.acadpaperreview.Academic#2”
}
```

a Paper Registry from Paper Academic view

Asset registry for org.acadpaperreview.Submission

Current User: Paper Academic

```
{
    “class”: “org.acad.acadpaperreview.Submission”,
    “submissionId”: “4befd1b4-feeb-42c4-a320-b29e7198f1ab”,
    “paper”: “resource:org.acad.acadpaperreview.Paper#1”,
    “conference”: “resource:org.acad.acadpaperreview.Conference#1”,
    “status”: “Pending”
}
```

b Submission Registry from Paper Academic view

Asset registry for org.acadpaperreview.ReviewRequest

Current User: Reviewer Academic

```
{
    “class”: “org.acad.acadpaperreview.ReviewRequest”,
    “requestId”: “fdbaa4b5-27c9-47e3-b01c-e78cd5bcd86d”,
    “conference”: “resource.org.acad.acadpaperreview.Conference#1”,
    “academic”: “resource.org.acad.acadpaperreview.Academic#4”,
    “payment”: 1
}
```

c ReviewRequest Registry from Reviewer Academic view

Asset registry for org.acadpaperreview.Review

Current User: Reviewer Academic

```
{
    “class”: “org.acad.acadpaperreview.Review”,
    “author”: “org.acad.acadpaperreview.Academic#4”,
    “IPFSHash”: “”,
    “reviewStatus”: “Pending”,
    “paper”: “org.acad.acadpaperreview.Paper#1”,
    “conference”: “org.acad.acadpaperreview.Conference#1”,
    “rating”: -1,
    “wordId”: “90626b7b-4055-48a0-85c0-c6ff67f8626c”,
    “owner”: “org.acad.acadpaperreview.Publisher#1”
}
```

d Review Registry from Reviewer Academic view

Fig. 6. Access to the author’s and reviewer’s information.

to the reviewers, a Review asset is created containing the identifier of the reviewer (Fig. 6d). As the final step towards a double-blindness review system, authors are restricted from accessing this asset. Similarly, the system also returned an empty registry for Paper Academic’s access to the Review asset in the test environment.

C. Test Item 3: Reviewer payment

In the proposed system, reviewers are paid with AcadCoin for their revisions. The ownership of an AcadCoin is determined by the "owner" data field of the coin. In the experiment, the AcadCoin belongs to the Publisher participant at the start of the conference. This Publisher is the one who started the conference (Fig. 5). The "owner" field was "org.acad.acadpaperreview.Publisher#1." After the allocation of reviewers to submitted papers, the AcadCoin was automatically paid to the Reviewer Academic. The "owner" data field of the AcadCoin changed to "org.acad.acadpaperreview.Academic#4." Therefore, this demonstrates that the proposed system is capable of providing payments to the reviewers.

VII. Conclusion

In this paper, a blockchain-based paper review system is proposed to solve the problems of current peer-review systems. The proposed framework provides a decentralized solution addressing open review metrics, double-blindness, and reviewers' opportunity cost issues. The paper also illustrates system architecture and smart contract algorithms for Hyperledger Composer. Furthermore, smart contract algorithms are implemented and tested in the Hyperledger Composer Playground. The test results indicate that all the users of the system can access the review metrics document. Moreover during the review process, authors and reviewers cannot read each other's identity. Also, AcadCoins can be rewarded to the reviewer as payment for paper revision. However, only the storage method is implemented for open metrics. In the future, a detailed definition of these open metrics will be proposed. Also, to extend the functionality of the back-end system, a plagiarism detection system will also be included as one of the processes of paper review. In addition, to increase the reviewers' motivation for writing meaningful reviews, an author's rating to the review document will be added. With the author's rating, the overall review rating will increase in accuracy. With this improved version of the review system, front-end interfaces will be designed and implemented for users to access the system.

References

[1] R. Smith, "Peer review: a flawed process at the heart of science and journals," *Journal of the royal society of medicine*, vol. 99, no. 4, pp. 178–182, Apr. 2006.

[2] S. Nakamoto. (2008) Bitcoin: A peer-to-peer electronic cash system. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[3] N. Emmadi, L. P. Maddali, and S. Sarkar. (2018) Marschain: Framework for a fair manuscript review system based on permissioned blockchain. [Online]. Available: https://wiki.hyperledger.org/_media/groups/requirements/use-case-inventory/marschain_usecase_hl.pdf

[4] S. Ølnes, J. Ubacht, and M. Janssen, "Blockchain in government: Benefits and implications of distributed ledger technology for information sharing," *Government Information Quarterly*, vol. 34, no. 3, pp. 355–364, Sep. 2017.

[5] Y. H. Ho, Z. Cheng, P. M. F. Ho, and H. C. Chan, "Mobile intercloud system with blockchain," in *Proc. of the International MultiConference of Engineers and Computer Scientists*, vol. 1, Hong Kong, China, Mar. 2018, pp. 100–105.

[6] N. Nizamuddin, H. R. Hasan, and K. Salah, "Ipfs-blockchain-based authenticity of online publications," in *Proc. International Conference on Blockchain*, Seattle, Wash., USA, Jul. 2018, pp. 199–212.

[7] Z. Zheng, S. Xie, H.-N. Dai, and H. Wang, "Blockchain challenges and opportunities: A survey," *International Journal of Web and Grid Services*, vol. 14, no. 4, pp. 352–378, 2018.

[8] J. Li, J. Wu, and L. Chen, "Block-secure: Blockchain based scheme for secure p2p cloud storage," *Information Sciences*, vol. 465, pp. 219–231, Oct. 2018.

[9] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: Found healthcare intelligence on blockchain with novel privacy risk control," *Journal of Medical Systems*, vol. 40, no. 10, p. 218, Aug 2016. [Online]. Available: https://doi.org/10.1007/s10916-016-0574-6

[10] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *Proc. 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, Montreal, QC, Canada, Oct. 2017, pp. 1–5.

[11] K. Fan, S. Wang, Y. Ren, H. Li, and Y. Yang, "Medblock: Efficient and secure medical data sharing via blockchain," *Journal of medical systems*, vol. 42, no. 8, Aug. 2018.

[12] S. Wang, Y. Zhang, and Y. Zhang, "A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems," *IEEE Access*, vol. 6, pp. 38 437–38 450, Jun. 2018.

[13] V. Patel, "A framework for secure and decentralized sharing of medical imaging data via blockchain consensus," *Health Informatics Journal*, Apr. 2018.

[14] Z. Li, L. Liu, A. V. Barenji, and W. Wang, "Cloud-based manufacturing blockchain: Secure knowledge sharing for injection mould redesign," *Procedia CIRP*, vol. 72, no. 1, pp. 961–966, 2018.

[15] R. Dennis and G. Owen, "Rep on the block: A next generation reputation system based on the blockchain," in *Proc. 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*, London, UK, Dec. 2015, pp. 131–138.

[16] A. K. Shrestha and J. Vassileva, "Blockchain-based research data sharing framework for incentivizing the data owners," in *International Conference on Blockchain*, Seattle, Wash., USA, Jul. 2018, pp. 259–266.

[17] Z. Constantinescu and M. Vladoiu, "The bmif journals online peer review system," *Bulletin of PG University of Ploiesti, Series Mathematics, Informatics, Physics*, vol. 62, no. 1, pp. 126–136, 2010.

[18] D.-Y. Du and Q.-B. Ling, "Role of scholarone manuscripts online peer review system in standardizing the edition and peer review process of manuscripts," *Chinese Journal of Medical Library and Information Science*, vol. 21, no. 6, p. 80, Jun. 2012.

[19] E. Aïmeur, G. Brassard, S. Gambs, and D. Schönfeld, "P3ers: Privacy-preserving peer review system." *Transactions on Data Privacy*, vol. 5, no. 3, pp. 553–578, Dec. 2012.

[20] M. Arapinis, S. Bursuc, and M. Ryan, "Privacy supporting cloud computing: Confichair, a case study," in *Proc. International Conference on Principles of Security and Trust*, Tallinn, Estonia, Mar. 2012, pp. 89–108.

[21] K. Jacksi, "Design and implementation of online submission and peer review system: A case study of e-journal of university of zakho," *International Journal of Scientific & Technology Research*, vol. 4, no. 8, pp. 83–85, Aug. 2015.

[22] B. Li and Y. T. Hou, "The new automated ieee infocom review assignment system," *IEEE Network*, vol. 30, no. 5, pp. 18–24, Sep. 2016.

[23] J. Benet. (2014) Ipfs - content addressed versioned p2p file system. [Online]. Available: http://arxiv.org:/abs/1407.3561

[24] (2018, Nov.) hyperledger-fabricdocs documentation. [Online]. Available: https://media.readthedocs.org/pdf/hyperledger-fabric/latest/hyperledger-fabric.pdf

[25] Hyperledger composer access control language. [Online]. Available: https://hyperledger.github.io/composer/latest/reference/acl_language