STUDY AND ANALYSIS OF SCALABILITY, INTEROPERABILITY AND TRANSITION DIFFICULTY IN PERMISSIONED BLOCKCHAIN

Thesis

Submitted in partial fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

SWATHI P

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575 025

May, 2022

DECLARATION

by the Ph.D. Research Scholar

I hereby declare that the Research Thesis entitled **Study and Analysis of Scalability, Interoperability and Transition-Difficulty in Permissioned- Blockchain** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of the requirements for the award of the Degree of **Doctor of Philosophy** in Department of Computer Science and Engineering is a bonafide report of the research work carried out by me. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

Swathi P, 187052 187CO005

Department of Computer Science and Engineering

Place: NITK, Surathkal.

Date: 17/05/2022

CERTIFICATE

This is to certify that the Research Thesis entitled **Study and Analysis of Scalability, Interoperability and Transition-Difficulty in Permissioned- Blockchain** submitted by **Swathi P** (Register Number: 187052 187CO005) as the record of the research work carried out by her, is accepted as the Research Thesis submission in partial fulfilment of the requirements for the award of degree of **Doctor of Philosophy**.

17/05/2022
Dr. M Venkatesan
Research Guide

(Signature with Date and Seal)

Chairman - DRPC
(Signature with Date and Seal)

ACKNOWLEDGEMENTS

Expressing my gratitude is the finishing touch on my dissertation. Research transformed me as a better individual, not only in the technical domain but also on a personal level. Such improvements would not have been possible without the support of the people who have helped me.

First, I would like to express my sincere gratitude to my research supervisor Dr. M. Venkatesan for the continuous support, guidance and motivation throughout my Ph.D. study and related research. His vast knowledge, encouragement and patience definitely helped me to reach to this height. I thank him for identifying my capabilities and giving this opportunity to purse research under him. The amount of consideration that he given during the stages of research paper and thesis writing are flawless. In fact I could not have imagined having a better advisor and mentor for my Ph.D study.

Now I would like to thank my Research Progress Committee (RPAC) members Dr. Jeny Rajan and Dr. P Jidesh for their insightful assessment and suggestions to further improve my work. The points raised by them were extremely valuable and helped me to look at my work in a different angle. I wish to show my appreciation to the Head of the Department Dr. Shashidhar G Koolagudi and other faculty members for their continuing support. I thank all the teaching and non-teaching staff of Computer Science Department, NITK for their help during my research period.

I will always cherish the kind of affection, care and support received from my friends Alkha Mohan, Victor Daniel, Nitesh Naik, Amit Praseed, Anoop B.N, Akhila, Rajani K. V, Sheetal Cyriac, Ashwin K.S and Joel Padamadan. Now I thank my fellow lab mates and group members for the stimulating discussions, knowledge transfer and team work. In particular, I am grateful to Dr Chirag Modi Navinchandra from NIT Goa, for enlightening me the first glance of research, without him I would not reach this level.

I would like to extend my special gratitude to the teachers, staffs and friends of SNDP school Nallalam, Saraswathi Vidya Nikethan Pantheerankavu, Government Ganapath Girls Higher Secondary School Chalappuram, Govt. Engineering College Bartonhill, National Institute of Technology Goa and National Institute of Technology Karnataka for their valuable support and guidance in molding me personally and professionally. Without them I wouldn't be in this position to write this.

Last but not the least, I would like to thank my father Mr. Dileepkumar P, mother Mrs. Jyothishkumari K, my father-in-law Mr. Suresh kumar,my mother-in-law Mrs. Thankamma Suresh, My brother Gokul Suresh and sister Sruthilaya Haridas for their love and guidance throughout my life. My family had to go through a lot of sacrifices for my studies and achievements and words cannot express how hard those were and how much I am grateful for those sacrifices. Most importantly, I wish to thank my loving and supportive husband Mr. Nandu Suresh who provide unending support and inspiration throughout my work. Finally I praise the almighty for giving me the vision and strength to continue with confidence. Thank you God.

Swathi P

ABSTRACT

Through the Bitcoin application, the innovative technology was miraculously launched in the markets, influencing numerous industries. Bitcoin is an innovative and path-breaking technology that has influenced numerous industries across the globe. Bitcoin is nothing but a form of digital currency (cryptocurrency) that can be used for trading in place of fiat money, where the underlying infrastructure is called Blockchain. The Blockchain is an open ledger that provides decentralization, transparency, immutability, and confidentiality. Blockchain can be used in massive, beneficial applications such as healthcare, logistics, supply chain management, the Internet of Things (IoT), etc. Most of the industrial applications rely on the permissioned Blockchain. However, the permissioned Blockchain fails in some aspects, such as scalability, interoperability and transission difficulty among consensus. This dissertation suggests a system to solve the scalability issue,interoperabilt issue and transission difficulty of permissioned Blockchain by incorporating data science techniques. The scalability analysis of the proposed solution is done in the hyperledger fabric framework with a variable number of transactions and results in scalability improvement.This work suggests a sustainable system to solve the interoperability issue of permissioned blockchain by designing a new infrastructure and this work has been tested in ethereum and hyperledger frameworks and which obtained a success rate of 100 percentage. Finally the transition among different consensus mechanism has been made possible by analysing the requiements of each algorithm and incoorperating it in MLP classifier.

Keywords: Blockchain; Scalability; Permissioned Blockchain; Hyperledger Fabric; Data science; Bitcoin; Cryptocurrency; Permissioned - Blockchain; Sustainable; Ethereum; Hyperledger

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	Expansion
Addy	Address
AI	Artificial Intelligence
AIF	Alternative Investment Fund
AIFM	Alternative Investment Fund Manager
AIFMD	Alternative Investment Fund Managers Directive
ALT	Alternative Cryptocurrency
Altcoin	Alternative Cryptocurrency
AML	Anti-Money Laundering
AMLD4	Fourth Anti-Money Laundering Directive
AMLD5	Fifth Anti-Money Laundering Directive
API	Application Program Interface
APS	Actions Per Second
ASIC	Application Specific Integrated Circuit
ASIM	Abstract State Interaction Machine
ASM	Abstract State Machine
ATH	All-Time High
ATL	All-Time Low
ATOM	Cosmos
AUR	Auroracoin
B1	Block.one
B2B	Business-to-Business
B2C	Business-to-Consumer
BAU	Business As Usual
BBO	Blockchain Based Obligation
BBOD	Blockchain Board of Derivatives
BCC	BitConnect
BCCC	Blockchain Collaborative Consortium
BCH	Bitcoin Cash
BCN	Bytecoin
BCOS	Blockchain Open Source
BFA	Brute Force Attack
BFT	Byzantine Fault Tolerance
BIP	Bitcoin Improvement Proposal

Abbreviations	Expansion
BIS	Bank for International Settlements
BNT	Bancor
BP	Block Producer
BT	Bithumb Coin
BTC	Bitcoin
BTC	Currency symbol for Bitcoin
BTD	Buy The Dip
BTG	Bitcoin Gold
BTM	Automatic Teller Machine for Bitcoin
BTM	Bitcoin Teller Machine
BTS	BitShares
CCICADA	Command, Control and Interoperability Center for Advanced Data Analysis
CDD	Customer Due Diligence
CDL	CoinDeal Token
CEA	Commodity Exchange Act
CEX	Centralized Exchange
CFD	Contract for Differences
CFTC	Commodity Futures Trading Commission
CIS	Collective Investment Scheme
CJEU	Court of Justice of the European Union
CLI	Command Line Interface
CMC	Coinmarketcap
CoT	Contract of Things
CPMI	Committee on Payments and Market Infrastructures
CPU	Central Processing Unit
CT	Crypto Twitter
CTO	Chief Technology Officer
CVC	Civic
DAC	Decentralized Autonomous Corporation
DAC	Distributed Autonomous Corporation
DAC	Distributed Autonomous Company
DAG	Directed Acyclic Graph
DAICO	Decentralized Autonomous Initial Coin Offering
DAO	Decentralised Autonomous Organisation
DAO	Decentralized Autonomous Organization
dApp	Decentralised App
DAPP	Decentralized Application
dApp	Decentralized Application
Dapp	Distributed App
DAPPs	Decentralized Applications
DAX	Digital Asset Exchange

Abbreviations	Expansion
DB	Database
dBFT	Delegated Byzantine Fault Tolerance
DCA	Dollar Cost Averaging
DCE	Digital Currency Exchange
DCR	Decred
DCT	Decent
DDoS	Distributed Denial of Service
DeFi	Decentralized Finance
DEVCON	Developers Conference
DEX	Decentralized Exchange
DGBB	Decentralized Governance by Blockchain
DGP	Decentralized Governance Protocol
DHT	Distributed Hash Table
DIACC	Digital ID and Authentication Council of Canada
DLT	Decentralised Ledger Technology
DM	Direct message
DNS	Deferred Net Settlement
DOGE	Dogecoin
DOJ	Department of Justice
DoS	Denial of Service
DPA	Data Protection Act
DPIA	Data Protection Impact Assessment
DPO	Data Protection Officer
dPoS	Delegated-Proof-of-Stake
dPoW	delayed Proof of Work
DSP	Digital Service Provider
EBA	European Banking Authority
ECB	European Central Bank
EDSE	European Decentralized Stock Exchange
EEA	Enterprise Ethereum Alliance
EIOPA	European Insurance and Occupational Pensions Authority
EIP	Ethereum Improvement Proposal
EMIR	European Market Infrastructure Regulation
ENISA	EU Agency for Network and Information Security
ERC	Ethereum Request for Comments
ESMA	European Securities and Markets Authority
ETC	Ethereum Classic
ETF	Exchange-Traded Fund
ETH	Ether
ETH	Ethereum
ETN	Electroneum

Abbreviations	Expansion
ETP	Exchange-Traded Product
EVM	Ethereum Virtual Machine
EWF	Energy Web Foundation
FA	Fundamental Analysis
FATF	Financial Action Task Force
FCA	Financial Conduct Authority
FDA	Food and Drug Administration
FDAS	Federated Distributed Agreement System
FDIC	Federal Deposit Insurance Corporation
FinCEN	Financial Crimes Enforcement Network
Fintech	Financial Technology
FIPS	Federal Information Processing Standard
FIU	Financial Intelligence Unit
FOMO	Fear Of Missing Out
FPML	Financial Products Markup Language
FSMA	Financial Services and Markets Act
FTC	Federal Trade Commission
FTR	Funds Transfer Regulation
FUD	Fear, Uncertainty and Dou
bt FX	Foreign Exchange
GDPR	General Data Protection Regulation
GNT	Golem
GPSG	Global Payments Steering Group
GPU	Graphical Processing Unit
HW	Hardware Wallet
IaaS	Identity-as-a-Service
IBC	Inter-Blockchain Communication
IBLT	Invertible Bloom Lookup Table
IBO	Initial Bounty Offering
IBS	Institute of Blockchain Singapore
ICO	Initial Coin Offering
ICT	Information Communications Technology
ICX	ICON
IEO	Initial Exchange Offering
ILP	Interledger Protocol
IMF	International Monetary Fund
IoT	Internet of Things
IOT	Iota
IPFS	Interplanetary Files System
IPO	Initial Public Offering
ITO	Initial Token Offering

Abbreviations	Expansion
JBA	Japan Blockchain Association
JBBA	Journal of the British Blockchain Association
JS	Javascript
JVM	Java Virtual Machine
KOI	Coinye
KYC	Know Your Customer
LE	Leader Election
LFT	Loop Fault Tolerance
LIB	Last Irreversible Block
LN	Lightning Network
LOC	Letter of Credit
LP	Liquidity Provider
mBTC	Millibitcoin
MCAP	Market Capitalization
MoE	Medium of Exchange
Multi-sig	Multi-Signature
NMC	Namecoin
NONCE	Number Used Only Once
OCO	One Cancels the Other
OTC	Over the Counter
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
PnD	Pump-and-Dump
PoA	Proof of Authority
PoB	Proof of Burn
PoD	Proof of Developer
PoS	Proof-of-stake
POT	PotCoin
PoW	Proof of Work
POW	Proof Of Work
PPC	Peercoin
REKT	Wrecked
ROI	Return On Investment
SATS	Satoshis
SC	Smart Contract
SEC	Securities and Exchange Commission
SegWit	Segregated Witness
SoV	Store of value
STO	Securities Token Offering
TA	Technical Analysis
TA	Trend Analysis
TIT	Titcoin
TLT	Think Long Term

Abbreviations	Expansion
TOR	The Onion Router
TPS	Transactions Per Second
Tx	Transaction
TxID	Transaction Identification
uBTC	MicroBitcoin
UoA	Unit of Account
USDC	USD Coin
USDT	Tether
UTC	Coordinated Universal Time
UTXO	Unspent Transaction Output
UXTO	Unspent Transaction
VTC	Vertcoin
WP	Whitepaper
WWDC	Worldwide Developers Conference
XBT	Bitcoin
XDG	Dogecoin
XEM	NEM
XLM	Stellar
XMR	Monero
XPM	Primecoin
XRP	Ripple
XVG	Verge
YTD	Year to Date
ZEC	Zcash
ZK	Zero Knowledge
2FA	Two Factor Authentication

CHAPTER 1

INTRODUCTION

Transaction systems in today's world must be decentralized, more transparent, and incorruptible. Instantaneous transactions and borderless ownership transfers are possible with digital money. According to IBM, Blockchain is a decentralized, immutable ledger that makes it easier to track assets and record transactions in a corporate network. A tangible asset (a home, vehicle, cash, or land) is different from an intangible asset (intellectual property, patents, copyrights, branding). On a blockchain network, virtually anything of value may be monitored and sold, lowering risk and lowering costs for all parties involved." A blockchain is made up of numerous blocks that create a ledger, with each block including essential components such as a hash, timestamp, additional data, and primary data. When a user completes a transaction, it is hashed and sent to all nodes in the network. The block of each node can contain a large number of transaction records. Blockchain generates a final hash value (Merkle tree root) that is stored in the block header (hash of current block) using a Merkle tree structure (Nakamoto (2009)). The timestamp indicates when the block was created. Data contains the block's signature, Nonce, and any other data defined by the user. Transaction records are included in the primary data, which is service-dependent. Anyone can look at the ledger, but they can't change it. Because the ledgers are generated by software, which must solve a mathematical problem to produce a legitimate result. All of these outputs are hashed, and a consensus is reached. A consensus mechanism in a blockchain is a fault-tolerant method for dispersed nodes to agree on a single network state. These are protocols that

ensure that all nodes are in sync with one another and agree on valid transactions to be added to the blockchain. Even if we try to modify a single ledger record, it will not be approved.

The following criteria may be used to classify blockchain networks: a) which clients are permitted to submit transactions, b) which peers are permitted to arrange transactions (including consensus), and c) how new clients are permitted to join the network. Anyone without a specified identification can join a public or permissionless blockchain network. A native coin or other economic incentives are frequently used in such networks. Bitcoin (Nakamoto (2009)) and Ethereum (Buterin (2015)) are two popular examples. Take bitcoin, for example, which spreads the business of producing money throughout the Internet. It employs computer algorithms to verify that funds are transferred safely from buyer to sale. Bitcoin's underlying technology, blockchain, provides transaction transparency and decentralized verification. In this case, Bitcoin is used by a network of computers to maintain the collective public database (Nakamoto (2009)). When Bitcoin is uploaded to the blockchain, all information about the transaction is locked. The transactions are verified and validated by bitcoin miners. If someone tries to tamper with the transaction, the node refuses to continue on the blockchain. On the user's node, a digital wallet is generated for each user. Each wallet has a unique address, which serves as a network node's effective identification. When someone wishes to send bitcoin to someone else, they send a message to the network of minors. Often, the system will automatically package some of these transactions into a difficult arithmetic problem. The transaction will be written down in the blockchain by whoever solves it first. The miners should be informed of the response. They check the transaction for accuracy and validate it. New bitcoins are used to vote for the winner, while current bitcoin transactions are not affected. A permissioned blockchain network is managed by well-known entities, such as members or stakeholders in a particular business environment (Cachin and Vukolic (2017)). All of the participants are on a safelist and are constrained by strong contractual requirements to perform in the network (T.Swanson (2015)). New peers can be added with consent from current peers or with permission from a regulator (IBM India). Voting-based consensus protocols,

Table 1.1: Comparison of Permissioned and Permission-less blockchain

Permissioned Blockchain	**Permission-less Blockchain**
They do not have to be transparent, but can opt transparency	Transparent
Free to choose the consensus algorithm	Strict consensus
They can be fully centralized or partially decentralized	Decentralized
More efficient performance	Comparitively less performance
Defined governance structures	No governance
High customizability	Customizable
Better Scalability	Less scalable
Security is entirely reliant on the integrity of its members	Members are anonymous
Less Transparent	More Transparent
Vulnerable to hacks and manipulation	Secure
Less anonymous	More Anonymous

such as Practical Byzantine Fault Tolerance (PBFT) (Castro and Liskov (2002)), are used in these networks. In such networks, a cryptocurrency is not required. A private blockchain network is a type of permissioned blockchain that is managed by one entity. The potential to automate corporate transactions using smart contracts (T.Swanson (2015)) is a key factor for companies considering moving to a blockchain network. A smart contract (Szabo (2018)) is a set of business rules that can be discussed and approved by a group of stakeholders before being put on a blockchain(IBM Corporation (2019a)). Smart contracts make it easier to automate and trust corporate operations. A transaction that refers to a smart contract triggers it. As a result, a transaction is a request to the blockchain to do a smart contract-based operation on the ledger. The comparison of permissioned and permission-less blockchain is given in table 1.1

Consider the blockchain scenario in supply chain management: Cashless Trade will digitize and automate administrative submissions by allowing end-users to submit, validate securely, and approve records across organizational boundaries, thereby reducing clearance and freight movement time and cost. Smart contracts built on the blockchain guarantee that all needed permissions are in place, speeding up approvals and eliminating errors.

Even though blockchain networks provide several advantages, there are questions regarding their ability to match the performance of traditional systems. The block frequency for public networks like Bitcoin is ten minutes (unknown (2020)), which implies

it takes 10 minutes or longer to complete a transaction. According to recent research (Croman et al. (2016)), Bitcoin has a maximum throughput of 7 transactions per second, which is horrible when compared to other payment systems. Permissioned blockchain networks, on the other hand, maybe built to employ efficient consensus methods like PBFT, resulting in considerably greater throughput and reduced latency while using far less compute, bandwidth, and storage. Another issue is scalability, or if the performance can keep up with a growing number of competitors. Hyperledger Fabric has been shown to reach a maximum throughput of 3500 transactions per second with a latency of less than a second (IBM Corporation (2019a)). Unlike trustless blockchains, permissioned blockchains don't rely on proof of work or stake. Instead, they rely on delegated consensus and transaction validation. Instead of relying on consensus, permissioned blockchains trust the validation and consensus processes to a group of nodes, which reduces the burden on consensus algorithms. Despite the current bottleneck, it is still being addressed in various works to motivate us to step beyond consensus and identify further performance improvement. According to the findings, there is a need to improve companies' ability to handle more transactions or workloads. The scalability of the hyper ledger fabric is depicted and broken down in this study using a generic framework. The system tries to figure out what impulses, ambiguities, and irregularities are essential for processing. This study critically examines Hyperledger Fabric 2.0, which is a fast-growing open-source permissioned blockchain platform. This work has implemented architectural optimizations that help improve the end-to-end transaction processing speed of networks. These techniques can increase the number of transactions that can be processed in a given time frame.

Another barrier preventing companies from adopting blockchain is platform interoperability. The capacity to freely exchange value across all blockchain networks without intermediaries are referred to as interoperability. It's about making it easier for different processes and units to communicate, collaborate, and coordinate." Interoperability levels include technical interoperability, legal interoperability, semantic interoperability, integrated public service governance, organizational interoperability, and interoperability governance. As the blockchain industry grows and evolves, interoper-

ability will become more crucial for any enterprise to thrive. It makes no sense to have hundreds of different blockchains that are entirely disconnected from one another. Interoperability is the capacity to exchange information over blockchain networks freely. If a user from another blockchain sends something on the blockchain, you will be able to quickly read, comprehend, and connect with them with no effort in an interoperable world. This work has considered two blockchain platforms: Hyperledger Fabric, a permissioned blockchain, and Ethereum, a permissionless blockchain. By integrating EVM Chaincode and fabric VM, the research proposed a novel architecture for resolving the interoperability issue. This research includes implementing the solution in the hyperledger fabric and Ethereum framework, as well as assessing the resulting system in terms of throughput, latency, and transaction success rates.

Last but not least problem is managing the transition among consensus mechanisms. A consensus mechanism is a fault-tolerant mechanism used in computer and blockchain systems to achieve the necessary agreement on a single data value or a single state of the network among distributed processes. In a dynamically changing status of the blockchain, these publicly shared ledgers need an efficient, fair, real-time, functional, reliable, and secure mechanism to ensure that all the transactions occurring on the network are genuine, and all participants agree on a consensus on the status of the ledger. If a transition is required, it is impossible to do so among the consensus. If a consensus transition occurs, the entire network will be disrupted, and the whole system will have to be rebuilt to continue with the new consensus. This scenario may be avoided if a correct categorization is carried out before the consensus is carried out. Classify the consensus using previous information, and then choose the suitable class for executing the consensus. In this way, the difficulties of transitioning from one consensus to another may be avoided. If a company switches from one consensus technique to another, it must abandon the entire network and restart. As a result, the question of transitioning from one consensus mechanism to another must be addressed. This thesis introduced and developed novel techniques to overcome the challenges in accepting Blockchain technology in industries. The work designed and create a hyperledger fabric-based framework that handles the scalability, transition difficulty, and interoperability issues

in the permissioned blockchain.

1.1 MOTIVATION

Decentralization of transaction processes, transparency, autonomy, and anonymity of users are all advantages of blockchain. Blockchain's promise extends beyond financial applications. Blockchain, a decentralized technology, has a wide range of applications, including healthcare, logistics, supply chain management, and the Internet of Things (IoT). Only a tiny fraction of people have utilized blockchain. These figures are expected to rise over time, and studying the factors which pull back people might be an intriguing research topic.

1.2 APPLICATIONS

The use of (i)Distributed Ledger Technology (DLT) (Nakamoto (2009))(ii) cryptographical verification and (iii) the ability to have smart contract reasoning embedded into it are instances of specific highlights of blockchains. This means that blockchains can allow many untrustworthy clients to execute directly on a record (smart contracts self-execute trades) without the need for a trusted intermediary. Transparency, decentralization, automation, and immutability are some of the innovative characteristics of blockchain technology. These divisions may be used for a variety of businesses, resulting in several application cases.

1.2.1 Banking

According to (Nguyen (2016)), Banks must lend, but only to borrowers who are at grave risk. This motivates banks to collect detailed, individually identifiable information from everyone who applies for a loan, such as annual salary, date of birth, voter id, aadhar card, or passport information. Finally, banks use this information to determine a candidate's creditworthiness. Specific data may be required to be shared with experts under certain circumstances, such as to prevent unlawful tax evasion. However, with so much personal information on hand, each bank becomes a tempting target for hackers. Instead of disclosing any confidential information, loan applicants can construct ZKPs stating that their past year income charges are above a certain threshold, that they have a valid

government identification number, and that their financial evaluation exceeded a specific point within the preceding week. Distributed ledger identification creates a global source of truth, which benefits a large number of people. Candidates can grant consent, and everyone can agree on how and when it was given. Banks can make adjustments to rules and keep a record of their actions. As a result, the market will be more efficient: banks will be able to provide loans with more precision, and applicants will be able to protect their personal information more effectively.

1.2.2 Healthcare

Enrollment and retention are the most challenging aspects of clinical trials, and despite several attempts over the years, progress remains mostly hidden. For these use cases, blockchain offers clinical preliminary data exchange and the ability for study individuals to experience esteem disclosure (counting individual well-being data adaptation). Patient data management is one of the most common blockchain use cases in human services. Health organizations will separate therapeutic documents (Kuo et al. (2017)), making it difficult to pick a patient's remedial history without contacting their previous care provider. This approach can take a long time to complete and is prone to errors owing to human error. Identity management of individuals (e.g., doctor, patient, member, and supplier), unique device IDs for vital gadgets in the medical production network, or authoritative members or validators in a system is the essential aspect of blockchain use case in medicinal services. The most common use case for DLT across organizations is item supply, from inception to end-of-life. Blockchain is used in social insurance, medicine, clinical supplies, blood products, and therapeutic devices for activities, consistency, and gauging among pharmaceutical producers, suppliers, pharmacy retailers, blood donation centers, and payers.

1.2.3 Agriculture

Before it reaches the buyer, the typical agriculture store network involves complicated, interrelated operations between many parties, such as the farmer, distributor, processing agencies, examination and insurance agencies, planning and transit organizations, banks, and vendors. There are a few challenges in this lengthy operation (Kamilaris

(2018)). Distributors believe it is challenging to track provenance to determine the origin and type of imported goods. The custody or custodial details becomes challenging to follow as objects travel between spouses. Global traders are wary of centralized entities, especially private organizations that certify crops. The data stream among stakeholders is causing a potential delay in fundamental leadership downstream. These problems can be solved with blockchain technology. Ranchers may suffer significant agricultural losses due to natural disasters such as floods, torrential rain, wind, earthquake, and landslide. When the crops are ruined, they must appeal to the government for compensation using a complicated procedure, and the approval is subject to some inspections. This architecture avoids complications by combining blockchain technology. Customers may also use blockchain data to pay tips directly to the farmers who have provided them with the best service.

1.2.4 Supply chain management

The industrial network that transports fish from the ocean to the table is highly complex and opaque. It has a large number of members from diverse businesses and administrative controls that span national borders. As a result, this business structure is an appropriate gateway for blockchain advancements (Apte and Petrovsky (2016)). Oceana, a non-governmental organization dedicated to protecting the oceans, theorized that a common platform for fish recognition would increase labeling precision and reduce private fishing: 'Despite significant challenges, fish discernibility is well within reach.' This method can make ground against offshore trawling by tracking where our fish come from at each stage of the production chain. When a fish is captured, sensors are attached to it to record information such as the location from where it was caught, temperature, and moisture content. This information, as well as other events in the treatment of the fish, is recorded in the blockchain: stockpile temperature runs, ownership transfers, transportation organization, and so on. The record may also be used to investigate both administrative and logical aspects of fish harvesting and use.

1.2.5 Others

Blockchain has a wide range of applications in the commercial world, including aviation, telecommunications, and IoT (Alam (2019)). Several features of blockchain must be embraced by many companies. Among these, decentralization is critical. A concentrated server is used in business to store all of an organization's data. Anyone who wants access to the data consults the database about the various levels of data. Because everything is dependent on the server, it is essential to keep it running at all times. There will be congestion at some point. A single server failure will destabilize the entire system (Wang et al. (2018)). Unchanging nature is another aspect of a blockchain that companies must accept. There will be a complete transformation of reality, which anybody may check to see whether the change in information causes a difference in the hash value, which can detect the change. As a result, no one can change the information about the company. To put it another way, everyone is looking at the data. Therefore no one will try to change it. The blockchain's third characteristic is its cost-effectiveness—the benefits of blockchain cause it to adapt in the corporate world (Kshetri and Voas (2018)). The elliptic curve digital signature computation is used to mark blockchain trades carefully. The distribution of exchanges between hubs effectively demonstrates their origins. The validation process is computationally challenging and is the critical bottleneck. When an association has been created via separation in concentrated databases, there is no need to examine every sale that arrives across it separately. The transaction's independent processing takes place in a separate location. Depending on the consensus method, this might include objective forward and backward communication (Ben et al. (2017)), as well as the oversight of forks and the associated rollback.

1.3 CHALLENGES

Recognizing the significant obstacles of blockchain development will help us to mitigate those challenges.

1.3.1 Data Privacy

One of the most compelling features of corporate blockchain is that it decentralizes processes, removing the need for third-party intermediaries. It is, nevertheless, one of the blockchain implementation difficulties. Many permissioned blockchain scenarios are used in sectors with strict data privacy regulations. As a result, one of the essential requirements of permissioned blockchain systems is protecting and enforcing access restrictions on on-chain data.

1.3.2 Insufficient Blockchain Literacy

Blockchain is still in its infancy as a technology. Notably, the bulk of the solutions being proposed are innovative, and they are primarily technical to the majority of industry participants. Organizations, on the other hand, lack personnel who are well-versed in the idea of blockchain and, by implication, business blockchain. As a result, this is just another of the blockchain implementation problems impeding the sector's growth. Specifically, companies lack knowledge on issues such as selecting the appropriate corporate blockchain technology to use.

1.3.3 Security

Blockchain deployment challenges come in a variety of shapes and sizes. One of the most exciting features of blockchain is its ability to withstand assaults. However, unethical actors pose a threat to the blockchain industry. Notably, the speed with which the technology is being implemented appears to be exposing portions of the business to hackers. Technology has various weaknesses as a result of immature processes and defenses. For example, the likelihood of being a victim of phishing schemes is highly significant. Furthermore, the lack of defined development standards indicates a lot of malware circulating within the sector.

1.3.4 Scalability

The inability to service many users is a challenge that blockchain technology, and by extension enterprise blockchain technology, is dealing with. Companies that can effectively scale their business blockchain platforms will enjoy significant benefits as de-

mand for enterprise blockchain, and related applications grow. Notably, the throughput must be adequate for the technology to acquire widespread acceptance by mainstream organizations. For businesses with a significant number of clients, it is evident that scalability is an essential factor to consider.

1.3.5 Lack of Regulations

Considering the inherent tragedies, some political organizations are hesitant to give technology-free rein to develop. Furthermore, given the technology's complexity, regulators are having difficulty defining the legal framework for it. A blockchain network, for example, is made up of nodes all over the world. As a result, it is difficult for authorities to correctly identify the jurisdiction and, as a result, the proper legal duties of the parties to the transaction once a transaction occurs on the platform.

1.4 CONTRIBUTIONS OF THE DISSERTATION

For our work on modeling the PBFT consensus process for Hyperledger Fabric v2.0, the research contributions are:

- Practical Byzantine Fault Tolerance (PBFT) consensus with high throughput.
- Scalable model of PBFT process.
- Analysis of the consensus process with a large number of transactions with large number of peers.
- Better confirmation times for transactions over 30000.
- The Proposed work has utilised Apache-spark to handle more transactions, thus increases scalability.
- The proposed solution directive usage increased the scalability 10 times higher than that of the existing scalability, that is an improvement from 3000 to 30000 transactions.
- The incorporation of parallelism in the solution directive increased the opportunity to add more transactions in the system, leading to a reduction in overhead.

- The consensus mechanism can utilize the opportunity to handle more transactions.

Next, for our work on Managing the interoperability . The research contributions are:

- Introduced the blockchain interoperability study topic by providing background information and emphasise definitions that are appropriate for both business and academics.
- Described blockchain interoperability and explore the architecture of blockchain interoperability.
- Hyperledger fabric has designed to interact with ethereum smartcontract
- Interaction is achieved through EVM Chaincode and fabric vm.
- Ethereum vm chaincode wraps the hyperledger fabric in a GO chaincode- together named as Fabreum.
- The Ethereum chaincode acts as the smart contract runtime and stores the deployed contract on the ledger.
- Combining both fabric and ethereum will act as twins so that features can be incorporated.
- Obtained 100 percent success rate in 500 transactions with better latency and throughput.

Finaly, for our work on Handling the transition among consensus mechanism. The research contributions are:

- Analysis of Different consensus mechanisms.
- Comparative study of different classification algorithms on blocktivity dataset.
- Introduced a new algorithm to handle the transition among consensus based on the requirements

- Selection of consensus was analysed and cross-checked

Since various blockchain networks are built with different use-case assumptions, system metrics produced for one type of system may not be applicable to another. This thesis, gives clear and exact definitions of performance measures that are relevant to all blockchain networks.

1.5 ORGANIZATION OF THE THESIS

The thesis advances in 6 chapters. An outline of each chapter is given below.

- **Chapter 1 : The Introduction** section covers the need and difficulties of solving the issues of scalability, interoperability and transition difficulties of permissioned blockchain. The chapter ends with a brief overview of research contributions and a thesis outline.
- **Chapter 2 : Literature Review** section mainly consists of a detailed review of the scalability in permissioned blockchain, interoperability of permissioned blockchain, and transition difficulty among consensus.
- **Chapter 3 : Solving scalability issues in permissioned- blockchain** includes the proposed solution for scalability issues in permissioned- blockchain and their design details and result analysis.
- **Chapter 4 : Manage interoperability among different blockchain platforms** covers the proposed method of making the blockchain platforms interoperable and disuss model performance.
- **Chapter 5 : Handle the transition difficulty among consensus** discuss the model and design of blockchain which can handle consensus irrespective of the alteration of application requirements.
- **Chapter 6 : Conclusions and Future Scope** chapter summarize the contributions and findings of this research work with future scope.

CHAPTER 2

LITERATURE REVIEW

Chapter 1 introduced the need for the scalability improvement , managing interoperability and handling the transition difficulty among consensus in permissioned blockchain. Motivation for the research work along with significant applications of Permissioned blockchain are listed in chapter 1. This chapter aims to give the key concepts of Blockchain technology and a deep perception of recent methodologies proposed and developed in the literature of scalability, interoperability and transition difficulty of consensus in blockchain. Blockchain is familiarizing in this chapter. A set of research gaps evolved from a thorough literature review is listed at the end of the chapter, along with the problem definition and objectives.

2.1 KEY DEFINITIONS

- **Address**: Cryptocurrency addresses are used to receive or send transactions on the network. A string of alphanumeric characters called as an address.

- **Block**: Blocks are a group of data meant to form a blockchain.

- **Block Reward**: It is an encouragement for the miner for calculating the hash successfully.

- **Chaincode**: Chaincode refers to the software that runs on top of the blockchain to implement the business logic that governs how apps interact with the ledger. When a transaction is suggested, chaincode is triggered, which determines which

state modification to apply to the ledger.

- **Confirmation**: Addition of a successful transaction into a blockchain.
- **Consensus**: Consensus achievement happens by ensuring that the ledgers are exact copies of each other and all participants in the network agree on the transaction validity.
- **Cryptocurrency**: They are digital assets.
- **Cryptographic Hash Function**: From variable size transaction inputs, Cryptographic hashes produce a unique hash value. Even a small change in the input will change the entire hash value. The SHA-256 is one of the examples for a cryptographic hash.
- **Dapp**: It is a decentralised application. It operates autonomously and has its data stored in the blockchain.
- **DAO**: Decentralised Autonomous Organizations are corporations that work without human interaction.
- **Distributed Ledger**: Those are the ledgers in which the storage of data across a network of decentralised nodes.
- **Distributed Network**: It is a type of network where there is a distribution of processing power and data across the nodes.
- **Difficulty**: This refers to how easily a miner can mine a block successfully.
- **Digital Signature**: It is a digital code generated by public key encryption that is attached to e-document to verify its contents and the sender's identity.
- **Double Spending**: It happens if the same money is spent more than once.
- **Mining**: Mining is the technique of verifying and validating the blockchain transactions. Miners get the reward according to the validation, usually in the form of coins.

- **Multi-Signature**: Using one key or authorisation multi-signature addresses provide an added layer of security.

- **Node**: The participant of the blockchain network carries a copy of the ledger, which is called a node.

- **Wallet**: It is a software that keeps private keys. It allows access to view and creates transactions on a specific blockchain.

2.2 KEY CONCEPTS

2.2.1 Consensus

The distributed consensus mechanism is the most basic component of a distributed ledger system (Panda et al. (2019)) The consensus mechanism guarantees that all of the network's transactions are agreed upon and performed in a sequential order. The term "consensus" refers to a broad agreement reached by all of the blockchain network's participating nodes or blocks. Blockchain offers dependability and confidence in the network amongst anonymous nodes in a distributed computing setting by using the consensus method. In essence, the consensus method assures that the information on the distributed ledger is not tampered by anybody (Alsunaidi and Alhaidari (2019)) It's important recognizing the distinction between a public blockchain network like Bitcoin and a permissioned blockchain network like Hyperledger Fabric in terms of consensus. Anyone may join a public network, which increases the danger of a Sybil attack (Swathi et al. (2019)) Bitcoin overcomes this problem by making it computationally costly for a peer to propose a new block of transactions (a process known as "mining"). Proof-of-work (PoW) is a method in which each peer must discover the correct random number (nonce) in the block header so that the hash value has a specified high number of leading zeroes (Zoican et al. (2018))There is no need for an expensive consensus process in a permissioned network because all members are whitelisted and bound by stringent contractual commitments to act adequately.

2.2.2 Smart Contract

A smart contract is a collection of instructions that are executed when a message is received. These instructions may modify the assets and create new messages when they are executed. A basic form of smart contracts can be inserted within a transaction as an executable script in first-generation blockchains like Bitcoin. Smart contracts, which are used in second-generation blockchains like Ethereum, make it easier to store and manipulate data on the blockchain. Smart contracts, unlike stored procedures in databases, ensure that the data they contain can only be changed by executing the permitted functions (Yli-Huumo et al. (2016)). Smart contracts (also known as chaincode in HLF) are used to carry out all activities in the HLF. A smart contract (SC) is a set of business rules that are communicated and validated by a group of stakeholders (Wood (2014)). Smart contracts make things easier to automate and trust corporate operations. Within a smart contract, many functions may be specified based on the business logic. When a client sends a transaction request to the peers, the smart contract is activated. As a result, a transaction is a request to the blockchain to run a chaincode-implemented function on the ledger (Sukhwani et al. (2018)). Fabric enables chaincode in general-purpose programming languages (e.g., Go, Java, Node.js) running in ordinary Docker containers, unlike other blockchain systems (such as Ethereum) that require smart contracts to be written in a specialised programming language (Schäffer et al. (2019)) System chaincodes, which are integrated into peer executables and have the same programming model as application chaincodes, are likewise supported by Fabric.

2.2.3 Hyperledger Fabric

Hyperledger is an open-source, network-oriented effort made to propel cross-industry blockchain developments. It is a worldwide facilitated exertion remembering pioneers for banking, cash, Internet of Things, manufacturing, supply chains, and advancement. The Linux Foundation has Hyperledger under the establishment (Fan et al. (2020)). Hyperledger business blockchain structures are utilized to assemble undertaking blockchains for a consortium of associations. They are unique in relation to open records like the Bitcoin blockchain and Ethereum. Hyperledger broods and advances a

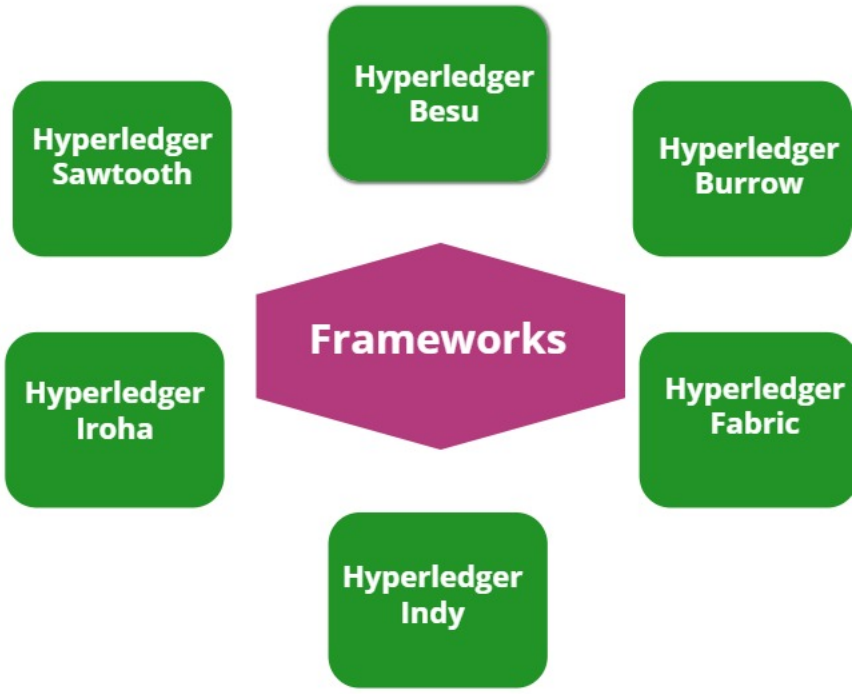

Figure 2.1: Hyperledger Frameworks

scope of business blockchain technology, including:

- Test applications
- Distributed ledger framework
- Smart contract engines
- Utility libraries
- Graphical interfaces
- Customer libraries

Some of the important hyperledger frameworks are shown in the figure 2.1 Hyperledger Fabric is an open-source for enterprises. It is a permissioned distributed ledger technology (DLT), which is intended for use in big business settings, that conveys some key

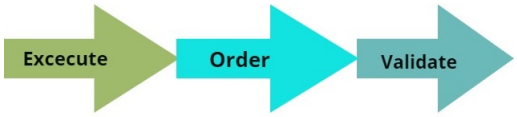

Figure 2.2: Hyperledger Fabric Transactionflow

separating capacities over other well known DLTs (Nathan et al. (2018)) Fabric has a profoundly particular and configurable design, empowering development, flexibility and improvement for an expansive scope of industry use cases including banking, IoT, music, cinema, healthcare, supplychain etc. The Fabric is permissioned, implying that, the members are known to one another , and in this manner completely untrusted. This implies while the members may not completely confide in each other, a system can be worked under an administration model that is worked off of trust in members. Fabric presents a new design for transactions as execute-request approve (Baliga et al. (2018)). It tends to flexibility, adaptability, versatility, execution and privacy challenges of order-execute model by isolating the transaction flow as shown in figure 2.2: Fabric makes channels, which empower the coordination of individuals to make an alternate record of trades. This is especially noteworthy for frameworks where a couple of individuals might be contenders who needn't bother with each trade. In an event that a gathering of clients makes a channel, only those individuals and no others have copies of the record for that channel. Hyperledger Fabric has an accounting systems including two sections: the world state and the exchange log. Each part has a copy of the ledger. The world state fragment shows state of the record at a given purpose of time. It's the database of the record. The transaction log part records all trades which have realized the present estimation of the world state; it's the update history for the world state. The ledger, by then, is an amalgamation of the world state database and the exchange log history. The smart contracts of fabric are written in chaincode and are summoned by an application which is external to the blockchain when that application needs to interface with the record. Generally speaking, chaincode works together just with the database portion of the record, the world state, and not the trade log. Chaincode can be executed in a couple of programming vernaculars. Starting at now, Go and Node is maintained. Many smart-contract-based blockchain systems have architectures that are quite similar to the

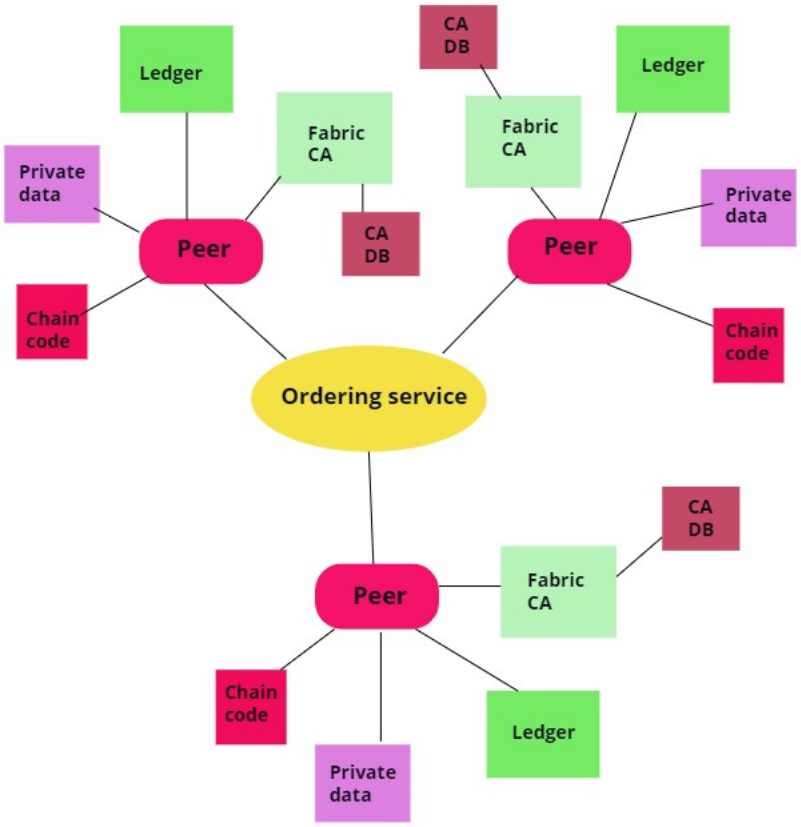

Figure 2.3: Hyperledger fabric architecture

classic state-machine replication method (IBM Corporation (2019a)) . These systems use active replication: first, the consensus protocol organizes the transactions and propagates them to all peers; second, each peer executes all of the transactions in the order in which they were received. This design is known as an order-execute architecture. Hyperledger fabric architecture is depicted in figure 2.3.

2.2.4 Hyperledger Caliper

Hyperledger Caliper is a blockchain benchmark apparatus or tool, it enables clients to quantify the exhibition of a blockchain execution with a lot of predefined use cases. Hyperledger Caliper will deliver reports containing various execution pointers to fill in as a source of perspective when utilizing the accompanying blockchain solutions such as: Ethereum, Hyperledger Besu, Hyperledger Burrow, Hyperledger Fabric, Hyperledger Iroha, FISCO BCOS, and Hyperledger Sawtooth. The key segment in Hyperledger Caliper is the adaptation layer, which is acquainted with coordinate different blockchain

solutions into the Caliper structure (Sukhwani et al. (2017)). A connector is implemented for each blockchain framework under test called system under test (SUT), the connector is answerable for interpretation of Caliper NBIs into comparing blockchain convention. Caliper NBI is a collection of basic blockchain interfaces, which contains tasks to connect with backend blockchain framework. Hyperledger Caliper will create reports containing various execution markers, for example, TPS (Transactions Per Second), exchange idleness, asset usage and so on. The purpose is for Caliper results to be utilized by other Hyperledger extends as they work out their systems, and as a source of perspective in supporting the decision of a blockchain execution reasonable for a client's particular needs.

2.3 SOLVING SCALABILITY ISSUES IN PERMISSIONED- BLOCKCHAIN : A REVIEW

The scalability concerns of blockchain have also been revealed as a result of Bitcoin's dominance in cryptocurrencies. Kyle Croman and his colleagues (Croman et al. (2016)) looked at a number of critical parameters to assess Bitcoin's scalability, including maximum throughput, latency, bootstrap time, and cost per verified transaction . The two most essential performance indicators that have a major influence on the user's quality of experience are maximum throughput and latency. We may divide the existing popular methods for addressing blockchain scalability into three layers: Layer 1, Layer 2, and Layer 0. Layer 1 is concerned with the blockchain's consensus, network, and data structure, which are all carried out on-chain. Layer2, on the other hand, is interested in using off-chain technologies like as off-chain channels, side-chains, and cross-chain protocols to scale up blockchain. Layer 1 (on-chain) solutions like Bitcoin-Cash increase block size, Compact block relay compresses blocks, Sharding methods, and many enhanced consensus algorithms boost transaction throughput and minimise transaction delay. Payment channel (McCorry et al. (2016)) and side chain layer2 solutions are still under development. Layer2 scaling solutions rely heavily on the cross-chain solutions that have evolved in recent years. Cosmos (Kwon and Buchman (2016)), which seeks to link several separate blockchains to create an integrated blockchain network and achieve scalability, is one of the most representative alternatives. Although previous solutions

increase scalability, it should be noted that most of them compromise the most essential characteristic of blockchain, namely decentralization, as well as introduce new security concerns. The Segregated Witness (SegWit) (Lombrozo et al. (2015b)) protocol, as described in BIP141 (Lombrozo et al. (2015a)), is intended to avoid unintentional Bitcoin transaction malleability and to ease the blockchain size limit, which slows down Bitcoin transactions. It accomplishes the objectives by dividing the transaction into two parts, deleting the unlocking signatures from the original transaction hashes, and storing both the scripts and signatures in the new Witness structure.

Bitcoin had a hard fork (Sompolinsky and Zohar (2015)) in 2017 as a result of the scalability issue, and was divided into two blockchain branches, Bitcoin and Bitcoin-Cash. The block size of Bitcoin-Cash has been raised to 8MB, which is significantly higher than the previous version (only 1MB in size). After that, Bitcoin-Cash was updated again, with the block size increased to 32MB. The Bitcoin-Cash average block interval has remained unchanged at 10 minutes. The transaction throughput can theoretically be significantly improved. The stress test done in September 2018 corroborated this. Since intra-blockchain bandwidth is limited, limitless growth increases the size of each block, making it difficult to transfer. As a result, just raising the block size is not a long-term solution. Other studies (Rohrer and Tschorsch (2019), Chawla et al. (2019)) claim that larger blocks can lead to centralization because individual network users are unable to propagate blocks efficiently and have difficulty verifying a large number of transactions in a short period of time. CUB (Xu et al. (2018)) offers a system for grouping nodes into Consensus Units. Each node holds a portion of the block data in each unit. To reduce the overall query cost, the blocks of the whole chain are assigned to nodes in the unit. They call this process the block assignment issue, and they offer techniques for solving it that decrease each node's storage cost while maintaining throughput and latency.

Other research (Sompolinsky et al. (2016), Eyal et al. (2016), and Zhou et al. (2020)) focused on enhancing the initial PoW method. Bitcoin-NG (Eyal et al. (2016)), for example, is a blockchain system based on the Nakamoto consensus (Nakamoto (2009)). It splits time into epochs, with a single leader in charge of transaction serialization for

each epoch. Sharding (Bagui and Nguyen (2015)) is a conventional database technology that was initially developed for the optimization of big commercial databases. This approach divides a huge database's contents into a number of pieces, which are subsequently stored on different servers to relieve the load on a single server, increasing search speed and expanding storage capacity) of the overall database system. Divide-and-conquer is the core concept of sharding technology. Elastico (Luu et al. (2016)) is the first permission-less blockchain sharding system. Participants in each Elastico consensus epoch must solve a PoW problem, which will be used to choose the consensus committee. At the beginning of each epoch, Elastico creates identities and committees. The efficiency of transaction execution may be harmed by such frequent operations. Despite the fact that each node only needs to validate transactions inside its own shard, each node must nevertheless retain all network data. Elastico demands a modest size to reduce the expense of executing PBFT in each committee, resulting in a high failure probability with only a 1/4 proportion of defective nodes tolerated. The atomicity of cross-shard transactions is not guaranteed by Elastico. RapidChain (Zamani et al. (2018)) is a sharding-based public blockchain technology that can withstand Byzantine failures affecting up to 1/3 of the users. RapidChain demonstrates that in prior sharding-based protocols, communication overhead per transaction is a substantial impediment to transaction speed and latency (Luu et al. (2016), Kokoris-Kogias et al. (2018)).

Transactions are stored in blocks in a conventional blockchain, which are arranged in a single chain structure. Because blocks cannot be produced concurrently under this form, transaction throughput is limited. To address this issue, DAG (Pervez et al. (2018)), a proposal for changing the structure of blockchain, has been presented. Y. Lewenberg et al. (Sompolinsky et al. (2016)) use a Directed Acyclic Graph of Blocks (blockDAG). In contrast to the usual structure of blockchain, a new block in this protocol refers to many previous blocks. To choose a primary chain for the generated DAG, an inclusive rule is provided (Sompolinsky and Zohar (2018)). Furthermore, the ledger can incorporate the contents of off-chain blocks that do not contradict with prior blocks. The system gets a greater throughput using the suggested protocol. There are numerous DAG-based initiatives in industry as well. Dagcoin (Pervez et al. (2018)), a DAG-based

cryptocurrency, considers each transaction as a block and prioritises quicker security confirmation and higher throughput. Another line of research, like Dagcoin, is to create DAG-based distributed ledgers, such as IOTA (Swathi et al. (2019)), Byteball (Uddin et al. (2021)), and Nano (Gatteschi et al. (2018)). DAG-based systems have a different ledger structure and transaction-confirmation mechanisms than blockchain-based platforms. However, several doubts have been raised regarding IOTA (Silvano and Marcelino (2020)), concentrating on the purported outstanding qualities of IOTA, such as its lack of transaction fees and high scalability. Meanwhile, treating each transaction as a block necessitates the addition of additional information. As a result, it cannot be used as an efficient approach for building a scalable system.

A Payment Channel Network (PCN) (McCorry et al. (2016)) is used to perform off-chain transactions between two parties who have not established a direct payment channel. One participant can make indirect transactions by routing to another via the channel between them. This network allows for quick and low-cost payment. The lightning network's faults, on the other hand, are extremely evident. The off-chain channel, for starters, necessitates both participants being online at the same moment. Second, a big transaction success rate has been reported to be low (Yapa et al. (2021)), implying that the present Network is not capable of processing high-value transactions. Pegged Sidechain (Back et al. (2014)) is the first sidechain that allows assets in blockchains like Bitcoin to be moved across blockchains while keeping the assets safe from attackers and maintaining atomicity.

Summary of research works done in scalability of blockchain is listed in table 2.1

2.4 MANAGE INTEROPERABILITY AMONG DIFFERENT BLOCKCHAIN PLATFORMS: A REVIEW

Interoperability is quickly becoming one of the most important aspects of blockchain technology, yet the expertise required to achieve it is dispersed. This makes it difficult for academics and business to create flawless interoperability between blockchains. Interoperability is a top concern for decision makers interested in building blockchain solutions. Organisations do not want to find themselves on a blockchain platform that may

Table 2.1: Summary of Research Works Done in Scalability of Blockchain

Research work	**Approach used**
(Croman et al. 2016)	Off-chain channels, side-chains, and cross-chain protocols
(McCorry et al. 2016)	Payment channels and side-chains
(Kwon and Buchman 2016)	Cosmos
(Lombrozo et al. 2015b)	SegWit
(Lombrozo et al. 2015a)	Adjustment in block size
(Sompolinsky and Zohar 2015)	Grouping nodes into consensus unit
(Eyal et al. 2016)	Bitcoin-NG
(Chawla et al. 2019)	Enhanced PoW
(Xu et al. 2018)	Enhanced PoW
(Bagui and Nguyen 2015)	Sharding
(Luu et al. 2016)	Elastico
(Zamani et al. 2018)	Rapidchain
(Pervez et al. 2018)	Directed Acyclic Graph (DAG)
(Sompolinsky et al. 2016)	Directed Acyclic Graph of blocks (BlockDAG)
(Pervez et al. 2018)	Dagcoin
(Swathi et al. 2019)	IOTA-DAG
(Uddin et al. 2021)	Byteball-DAG
(Gatteschi et al. 2018)	Nano-DAG
(McCorry et al. 2016)	Payment Channel Network (PCN)
(Yapa et al. 2021)	Offchain channels
(Back et al. 2014)	Pegged Sidechains
(Zhou et al. 2020)	Segwit
(Rohrer and Tschorsch 2019)	Adjustment in blocksize
(Silvano and Marcelino 2020)	Iota
(Sompolinsky and Zohar 2018)	Restructured blockchain

limit their options for external collaboration. When constructing a blockchain, developers frequently disregard standards in order to gain more flexibility, but this can lead to interoperability and communication concerns. Multiple blockchain networks with various properties such as consensus models, smart contract functionality, and transaction algorithms are the most significant barrier to interoperability. Several standardisation projects are now underway to address this issue. Researchers has put forward some solutions for the interoperability of blockchain. Herdius is a decentralised exchange platform that focuses on the private keys, which are the common thread that connects all blockchains. As a result, Herdius facilitates cross-chain transfers by allowing them to be shared. No assembler node can fully decrypt the native private key, which adds another degree of protection. Rather, homomorphic cryptography operations are used to sign the transaction. Herdius can decentralise the notary-scheme by employing this structure (Albert Callarisa Roca (2017)). One method of achieving interoperability across several blockchains is to use a sidechain (Qasse et al. (2019b)). The amount of total assets does not increase, hence sidechain interoperability is restricted to asset transfers in a one-to-one connection. Another disadvantage of sidechain implementation is that if a hacked sidechain is present in the network, vulnerability in the main chain or other sidechains may rise (Sztorc (2015)).

A blockchain router was invented by Wang et al. (Wang et al. (2017)), which allows several blockchains to connect with one another. The approach's architecture includes four participants: a connector, a validator, a nominator, and a surveillant. This technique uses a consensus algorithm that is similar to PBFT. Qasse et al. (2019b) proposed a private token-based inter-Blockchain communication system to enable cross-chain communication without the use of middlemen. Chen et al. employed PBFT as the consensus method and a routing algorithm. The key constraint of this study is that it had a significant impact on system throughput. Anlink Blockchain (Tech (2017)) proposed a corporate blockchain architecture that links various blockchains and allows cross-chain communication via an inter blockchain communication protocol (CBCP). Ann-Router, AnnChain, and other blockchain technologies make up the proposed architecture. The consensus algorithm used in this method is Delegated Stake-PBFT.

Kan et al. (Kan et al. (2018b)) presented numerous blockchain topologies for transferring assets reliably across various blockchain networks. The article presented an interblockchain connection model for network routing management. There are four levels in the suggested architecture: the fundamental layer, the blockchain layer, the multi-chain communication layer, and the application layer. A single packet for transaction and routing was also introduced in the paper. Interchain is a system created by Ding et al. (Ding (2018)), which allows any pair of blockchains to communicate with each other. Subchain, InterChain, interchain nodes, validating nodes, and gateway nodes are all part of the proposed framework design. Three handshaking methods are utilised to complete asset transfers across various blockchains. However, no consensus algorithm was included in the publication to support the framework.

P. Bennink et al. (Bennink (2018)) investigated and compared the various methods for performing atomic swaps on Ethereum blockchain systems. Transferring or trading assets between numerous parties across various blockchain platforms, such as swapping ether for bitcoin, is referred to as a cross-chain atomic exchange. Swap contracts for single usage were also devised by engineering to be established for each swap. Dagher et al. (Dagher and Enderson (2018)) studied the use of smart contracts to achieve interoperability between different blockchains. The suggested approach consists of a smart contract that permits data exchange between heterogeneous blockchains that are independent of one another. The proposed on two Ethereum networks, one public and the other private, as proof of concept. The authors were unable to successfully apply their method to two hybrid systems. Li et al. (Li et al. (2017)) developed a satellite chain, a blockchain architecture that complies with industry norms. The design includes of multiple subchains that run their own individual consensus algorithms, as well as a regulator that uses smart contracts to manage the whole network and specialised responsibilities. Different sub chains on the satellite chain can execute heterogeneous consensus methods in parallel.

Block Collider (overline (2017)) is a multi-chain platform based on a collection of already exported blocks from other blockchains, allowing cross-chain functionality. Blocks are collected from connected blockchains by peer-to-peer decentralised miners

without the use of centralised validators. The proof of distance consensus mechanism, which is a modified version of the proof of work consensus process, is used by the Block Collider. The interledger protocol (Thomas and Schwartz (2016)) is implemented in Java by Hyperledger Quilt (linux foundation project (2017)). The protocol is intended to offer interoperability by transferring value across systems. The project is still in its infancy, and there is no whitepaper accessible. The Polkadot project (wood (2016)) is a solution that allows diverse blockchains to communicate with one another. Dot is the token utilised in this project. Polkadot's architecture is divided into three categories: parachains, relay chains, and bridges. The parachains represent diverse blockchains, the relay chains handle transaction consensus and delivery, and the bridges link the parachains to their consensus.

The Aion Project (Spoke (2019)) intends to provide cross-chain interoperability by allowing different blockchain systems to connect. The protocols that distinct and independent blockchains might utilise to interconnect inside the AION Platform are known as connecting networks. Interchain transactions allow data to be transferred between the ecosystem's linked blockchains. The bridges, which are a group of validators, validate interchain transactions. While any blockchain network may become a participating network provided it meets the Aion ecosystem's specifications. Through its platform, the ICON Project (Foundation (2019)) aims to link various blockchain businesses and groups, including financial institutions, government offices, hospitals, and colleges. Nexus and ICON Republic make up the platform. Nexus is a collection of decentralised blockchain entities linked via ICON Republic portals. ICON's consensus mechanism is Loop Fault Tolerance (LFT), and its official token is ICX. LFT is a tendermint-based enhancement on BFT consensus methods. The project's key constraint is that it is focused on and created for Korea, and it adheres to the rules governing blockchain and crypto firms in Korea. Wanchain blockchain (Louie (2017)) is a financial infrastructure-based fork of the Ethereum project. The project's goal is to enable asset transfer across blockchains that are interconnected and unrelated. Wanchain, like Ethereum, will employ a proof of stake consensus method. The WANchain project's token is WAN.

The ARK project (Wood (2017)) intends to accelerate blockchain adoption by developing a framework that allows anybody to create their own blockchain in a short amount of time. The project's main feature is smart bridges. Smart bridges are used to connect incomplete and independent blockchains, with ARK acting as an intermediary layer between the blockchains. The ARK token is called "ARK." The Delegated Proof of Stake (dPoS) consensus method was employed in this project as the consensus algorithm. The Blocknet (Belchior et al. (2021a)) protocol connects cryptocurrency and token-based blockchains by providing inter-blockchain services like decentralised exchange (DEX). The protocol is compatible with the majority of today's cryptocurrencies. Blocknet's architecture is made up of three primary parts: a blockchain router, a decentralised asset exchange system, and an inter-chain data transit protocol. The router is used to choose the appropriate service nodes to which the requested service should be directed. The exchange component's aim is to enable cross-chain transactions between various cryptocurrencies. Data may be transferred from one chain to another using the third component. Metronome (Belchior et al. (2021a)) is a project that tries to improve current cryptocurrency systems in order to develop a superior cryptocurrency solution. Metronome also provides crossblockchain transfer, which allows a user to move their token from one blockchain to another via a proof-of-exit receipt. MTN is the token utilised in this project. Ripple (Thomas and Schwartz (2016)) is a protocol that allows for instantaneous swaps between several blockchain systems. It isolates the sender and recipient to eliminate the danger of an intermediate failure. The protocol allows for a safe transfer by employing hash locking, in which the payment is conditionally locked until the transfer is completed.

Summary of research works done in interoperability of blockchain is listed in table 2.2

2.5 HANDLE THE TRANSITION DIFFICULTY AMONG CONSENSUS: A REVIEW

The scope of blockchain networks has grown significantly during the last decade, moving beyond tamper-evident distributed ledgers. However, most extant general assessments and surveys on blockchains focus primarily on scenarios of deploying blockchain

Table 2.2: Summary of Research Works Done in Interoperability of Blockchain

Research work	**Approach used**
(Albert Callarisa Roca 2017)	Blockchain Connecters
(Qasse et al. 2019b)	Sidechain
(Tech 2017)	Inter blockchain communication protocol
(Kan et al. 2018b)	Inter blockchain connection model for network routing
(Ding 2018)	Interchain
(Bennink 2018)	Cross-chain atomic swap
(Dagher and Enderson 2018)	Smartcontract based interoperability
(Li et al. 2017)	Satellite chain
(overline 2017)	Feature Band set+Object oriented approach
(linux foundation project 2017)	Interledger Protocol
(wood 2016)	Polkadot
(Qasse et al. 2019b)	Token based inter blockchain communication
(Spoke 2019)	Interchain transactions- AION platforms
(Belchior et al. 2021a)	Metronome: Used proof of exit receipt
(Foundation 2019)	ICON Project: Adhers to the rules governing Korean firms
(Louie 2017)	Wanchain Blockchain: Enable asset transfer
(Wood 2017)	ARK Projects: Fastest blockchain creation
(Belchior et al. 2021a)	Blocknet:Provide inter-blockchain services
(Thomas and Schwartz 2016)	Ripple Protocol: Allows safe transfer

networks as the backbone technology for cryptocurrencies, particularly market-dominant ones such as Bitcoin and Ethereum, due to the recent market excitement around cryptocurrencies. Several research works exist to address ML usages for blockchain-based applications, however they have not yet been fully explored. In this research, we looked at how machine learning may be used to solve the problem of transition difficulty of consensus mechanism in blockchain. With the fast advancement of blockchain technology, the need for higher-quality services from blockchain-based applications has increased, posing new problems in the development of blockchain protocols (Dinh et al. (2018)),(Bonneau et al. (2015)). The difficulty of preserving the canonical blockchain state throughout the P2P network may be translated as a fault-tolerant state-machine replication problem in the context of distributed systems (Raynal (2010)). Consensus nodes are supposed to reach an agreement (i.e., consensus) on the unique shared view of the blockchain in the event of Byzantine/arbitrary failures. Various blockchain networks have different consensus protocols. Because permissioned blockchain networks allow for finer control over consensus node synchronisation, they can use traditional Byzan-

tine Fault-Tolerant (BFT) protocols to achieve the requisite consensus features (foundational algorithms described in (Miller and LaViola (2014)),(Sun and Duan (2014)).

Satoshi Nakamoto (2009), the founder of Bitcoin, devised POW, the earliest and most well-known consensus method. In POW, the miner who discovers the hash first is authorised to add a new block to the blockchain containing the transaction. Because mining is a computationally costly activity, having a high hashrate is essential for miners to compute the hash and so get the rewards. They will need a significant amount of effort to complete. A successful assault needs a significant amount of processing power as well as a significant amount of time to complete the computations. As a result, the attack is doable but somewhat pointless due to the enormous expenses. Miguel Castro and Barbara Liskov created Practical Byzantine Fault Tolerance (PBFT) at the MIT Laboratory for Computer Science in 1999 (Castro and Liskov (1999)). One of the proposed answers to the Byzantine Generals' Problem, a classic distributed system issue (Lamport et al. (2002)), is PBFT. The purpose of PBFT is to determine whether or not a piece of information contributed to the blockchain should be accepted. PBFT, like the conventional Byzantine Generals' Problem, may accept 1/3 node treachery. Because PBFT depends on the number of nodes to validate trust, a large hashrate is not necessary in this procedure. The transaction is validated to be legitimate once enough answers have been received. In permissioned blockchain, PBFT is a representative consensus process. However, because of the large number of message exchanges, it must accept the risk of centralization and limited scalability.

King and Nadal (2012) were the first to create PoS (Proof of Stake) in 2012. This solves the problem of Bitcoin mining consuming a lot of electricity. Every miner spends part of their coins as stake in the system's currencies to create new transactional blocks under PoS . The monopoly problem of PoW is also solved by this algorithm. Furthermore, because this technique is resistant to a 51 percent attack, penalties may be enforced if any validators do incorrect verification (Nakamoto (2009)). Decred, Ethereum, and Peercoin are among the crypto currencies that have used PoS as a consensus mechanism. A variant of PoS is Distributed proof of stake (DPoS) (Shala et al. (2019)). With DPoS, currency holders may use their balance to vote for a list of nodes that will

be authorised to potentially add new transaction blocks to the blockchain. Changes to the network parameter can also be voted on by coin holders. DPoS provides all coin holders more power and ownership in the network, whereas PoS is more like winning a lottery. Those with more money or tokens will have more influence on the network than those with less. Token holders in DPoS don't vote on the validity of the blocks directly; instead, they vote to elect delegates to validate the blocks on their behalf.

PoET is backed by Hyperledger Sawtooth (IBM Corporation (2019b)), an Intel-developed modular blockchain technology. It may be used on both private and public platforms. It allows users on a permissioned blockchain to reach agreement even if they don't know each other, whereas most permissioned blockchains need users to know and trust one another. PoET is similar to PoW, except it does not consume as much resources. PoA (Cachin and Vukolić (2017)) is a consensus process in which transactions are verified by authorised accounts, which operate as the system's "admins." PoA is a modified version of PoS in which a validator's identity serves as the stake instead of a monetary value. Validators, or authorised accounts, validate transactions and blocks in PoA-based networks. Validators use software to organise transactions into blocks. The process is automated, so validators don't have to keep an eye on their computers all the time. However, it does need keeping the computer secure. Waves suggested LPoS (Leased Proof of Stake) (De Angelis et al. (2017)). Waves developed a bespoke coin on a decentralised blockchain network that uses less energy. LPoS establishes a centralised environment within a decentralised network, giving smallholders the opportunity to stalk. The coin holder might gain from this by leasing the coins. This currency boosts the probability of being permitted to contribute a new block to the chain by making the node stronger or giving it more weight .

The majority of the time, machine learning models are employed to make predictions. A good prediction model aids in making the best decisions and analysing data. In addition, Valenkar et al. (Velankar et al. (2018)) suggested an ML model for predicting bitcoin values. With numerous characteristics such as block size, total bitcoins, day high, number of transactions, and trading volume, this model employs Bayesian regression and random forest. Log, z-score, and box-cox normalisation methods were

used to normalise the learned dataset. A price prediction research was also conducted for numerous cryptocurrencies (Saad and Mohaisen (2018)).

Classifier comparison is significant in both academic and industrial settings. A lot of studies comparing data categorization methods have been published. Choosing the best method for a particular classification job based on a priori knowledge about the classifiers' behaviour across domains is risky unless the evaluation is done in a systematic way that allows the findings to be repeated and generalised.

Zheng et al. (2017) (1993) advocated using 16 dimensions (accuracy, number of attributes/classes/instances/data set density, etc.) to assess the accuracy of three classification algorithms established prior to 1993, including C4.5 and ID3. On 8 real-world credit scoring data sets, Lessmann et al. (2015) compared various classifiers. Individual classifiers (such as C4.5, ELM, LR, and NN) predict substantially less accurately than RF, with a few outliers. Brown and Mues (2012) investigated the effectiveness of eight classifiers for credit scoring, which is a binary (2-class) unbalanced classification issue, including SVM, RF, GBDT, NN, C4.5, KNN, LR, and LDA. The findings of their trials demonstrate that GBDT and RF worked effectively with samples that had a significant class imbalance. On large real-world issues, King and Nadal (2012) used the StatLog project to examine different classification algorithms from symbolic learning (including C4.5), statistics (including NB, KNN, LR), and neural networks (NN). They discovered that performance is highly dependent on the data set being studied, and that there is no one optimum algorithm. This argument is supported by the No-Free-Lunch theorem (Wolpert (1996)), which argues that the best classifier for each data set will be different. Macià and Bernadó-Mansilla (2014) compared the accuracy rates obtained by 8 classifiers from different learning paradigms that represent some of the fundamental algorithms of Machine Learning, including C4.5, RF, Multilayer Perceptron (MLP), SVM, LR, and NB, based on the type, complexity, and use of UCI data sets. They discovered that the accuracy of C4.5, LR, NB, and RF over the UCI repository is relatively comparable for most data sets, and the divergence of accuracy rates is minor, using the implementations given by Weka.Jones et al. (2015) used a large sample of worldwide credit rating changes data sets from 1983 to 2013 to investigate the predic-

tion performance of a variety of binary classifiers. Generalised boosting, AB, and RF outperformed SVM, LDA, and LR, according to their findings. This research, however, is confined to binary classifiers. Lim et al. (2000) compared 22 decision trees, 9 statistical, and 2 neural net-work methods in terms of classification accuracy and training time using 32 data sets. The best accurate classifiers were determined to be (Multinomial) LR algorithms, according to the authors. However, GBDT, RF, ELM, SVM, SRC, and DL were not among the algorithms examined. Fernandez-Delgado et al. (2014) tested 179 classifier parameters from 17 families using C, Weka, R, and Matlab, and discovered that parallel RF in R performed best, followed by SVM. There are two NN settings among the top-10 performers. The authors show that all of the classifiers from the RF and SVM families are in the top 25 best classifiers, with accuracies of more than 79 percent (the greatest is 82.3 percent), indicating that both families are the best. These two families outperform C4.5, AB, LR, and NB settings and ensembles in general. GBDT, ELM, SRC, and DL, on the other hand, were not among the algorithms examined.

Summary of research works done in consensus mechanism of blockchain is listed in table 2.3

2.6 RESEARCH GAPS

Based on our review there are clear research gaps in the blockchain technology.

- The research is needed in the area of latency, throughput, size and bandwidth, versioning, hard fork, since it doesnot exist in the current literature.The size of blockchain is small now, But as blockchain accepted widley then there will be a need of research in such topics.

- The majority of current research is conducted either in the bitcoin environment or in other cryptocurrencies, rather than in other blockchain environment like permissioned blockchain (Eya (2016))
  Therefore, this research proposal will consider these key challenges such as scalability, transition difficulty and interoperability in permissioned blockchain.

Table 2.3: Summary of Research Works Done in Consensus Mechanism of Blockchain

Research work	**Approach used**
(Nakamoto 2009)	Explained PoW
(Dinh et al. 2018)	Explanation of PBFT algorithm
(King and Nadal 2012)	Introduced PoS- Solved the problem of electricity wastage
(IBM Corporation 2019b)	PoET
(Cachin and Vukolić 2017)	PoA
(De Angelis et al. 2017)	LPoS
(Velankar et al. 2018)	ML model for predicting bitcoin values
(King and Nadal 2012)	Distributed PoS
(Saad and Mohaisen 2018)	Price prediction models
(Zheng et al. 2017)	Compared Classification algorithms
(Lessmann et al. 2015)	Comparison of
(Brown and Mues 2012)	Investigated the effectiveness of classifiers
(King and Nadal 2012)	Examined different classification algorithms
(Wolpert 1996)	Compared the accuracy rates of different classifiers
(Macià and Bernadó-Mansilla 2014)	investigated the prediction performance of binary classifiers
(Miller and LaViola 2014)	Explanation of basic consensus algorithms
(Sun and Duan 2014)	Consensus analysis
(Jones et al. 2015)	Binary classifiers
(Lim et al. 2000)	Examined different classification algorithms
(Fernandez-Delgado et al. 2014)	comparison of classifiers
(Raynal 2010)	Fault tolerent problem
(Castro and Liskov 1999)	Practical byzantine fault tolerent algorithm (PBFT)
(Lamport et al. 2002)	Analysis of PBFT
(Shala et al. 2019)	Distributed proof of stake (DPoS) an analysis

2.7 PROBLEM STATEMENT

To design and develop a hyperledger fabric based framework which handles the scalability, transition difficulty and interoperability issues in the permissioned blockchain.

The objectives of the work are:

1. **Solve the scalability issues in permissioned blockchain:** Scalability is necessary for organizations that have a large number of customers. The existing enterprise blockchain networks are untested to that extent. It is imperative to address the scalability issue for putting blockchain into practice.

2. **Manage interoperability among different platforms :** It is impossible for users on one platform to interact with users on other platforms. Handle this interoperability issue using both permissionless and permissioned blockchian.

3. **Handle the transition difficuty among consensus:** If an enterprise is changing from one consensus strategy to another, it has to give up the entire network and has to start from scratch again. So it is important to address the issue of transition from one consensus mechanism to another.

CHAPTER 3

SCALABILITY OF PERMISSIONED- BLOCKCHAIN

3.1 INTRODUCTION

The technology which has a significant impact on the next generation has arrived. It can be incorporated in any field where it is needed, But it is not a bigdata, AI, or robotics. It is the underlying technology of a cryptocurrency called bitcoin. This technology is termed as a blockchain. For instance, bitcoin (Nakamoto (2009)) disperses the matter of making cash around the Internet. It utilizes algorithms to guarantee that the payment has moved safely from the purchaser to the merchant. Bitcoin uses blockchain as an essential innovation to offer transparency on exchanges with disseminated verification. Here, a system utilizes Blockchain, which keeps up the aggregate open database (Dinh et al. (2017a)). Bitcoin is included in the blockchain; the exchange blots all data about it. Bitcoin miners verify and validate the transactions. Suppose there is an attempt to degenerate the transaction; at that point, the hub will refuse the transaction to proceed in the blockchain. For every client, an advanced wallet is made on the end-user hub. Every wallet has a unique address that is considered an effective identity of the system's hub. Think about a situation, a health insurance agency is giving insurance compensation, and it needs to guarantee that the given cash is utilized distinctly for a clinical reason (Swathi et al. (2019)). Here comes the blockchain scope; blockchain guarantees that the money is used for the same reason; else, the given cash can be returned consequently to the guarantor if it isn't utilized for a specific timeframe or the referenced reason. In this way, blockchain guarantees transparency to the transactions and offers significant

preferences like decentralization of the transaction process, transparency, autonomy, and anonymity of the users (Nathan et al. (2018)). The potential of blockchain is beyond financial applications.

Blockchain can be used in substantial valuable applications such as medicinal services, academics, banking marketing, and much more. The cryptocurrencies mentioned before are coming under the permissionless blockchain. Permissionless blockchains are also known as public or decentralized blockchains. Anyone can create and access the blockchain in which anyone can publish the self-executing contract(smart contract; which will be explained in 3.2.1). Moreover, anyone can run the blockchain node with 100 percent transparency. But organizations require an entirely different type of blockchain, which can safeguard their terms and policies. It should incorporate only preapproved nodes . This type of blockchain is called permissioned blockchains.

Permissioned blockchain requires every peer to execute every transaction, maintain a ledger and run consensus(which will be explained in 3.2.2), a fault-tolerant mechanism. It can't support the valid private transaction with confidential contracts. Hyperledger Fabric is one of the best blockchains which can deliver the modular and secure foundation for industrial blockchain. Hyperledger fabric generally uses a practical Byzantine Fault Tolerance (PBFT) consensus algorithm. But some factors pull back the industries from adapting the blockchain in full fledge. Scalability is one among them. Scalability indicates the ability of a system to entertain a growing amount of work and contain the enlargement. Scalability will be one of the points in the bucket list of organizations.

"I examined aspects of scalability, but did not find a useful, rigorous definition of it. Without such a definition, I assert that calling a system 'scalable' is about as useful as calling it 'modern'. I encourage the technical community to either rigorously define scalability or stop using it to describe systems." —Mark D. Hill, What is Scalability? [15:scalability analysis]

Dinh et al. (2017b) presented Blockbench, which estimates the throughput, latency, scalability, and adaptation to internal failure. They used Blockbench to direct an extensive assessment of three significant private blockchains: Ethereum, Parity, and Hy-

perledger Fabric. The outcomes exhibit that these frameworks are still far away from uprooting current database frameworks in conventional data handling. Besides, there are flaws in execution among the three frameworks assigned to the structure decisions at various layers of the blockchain's product stack.

Baliga et al. (2018) adopted a test strategy, where they studied the throughput and latency of Fabric by exposing it to various arrangements of workloads. Through a set-up of smaller scale benchmarks, specially worked for Fabric, the authors tune diverse transaction and chaincode parameters and study how they influence latency.

Duboc et al. (2006) presented a structure for absolutely describing and examining a product framework's adaptability. The system regards scalability as a multi-measures streamlining issue and catches the reliance connections. This research work presents the aftereffects of a contextual investigation where the structure and examination strategy were applied to a simple framework, showing that it is conceivable to build up an exact, deliberate portrayal of scalability and to utilize the report to analyze the scalability of alternative framework plans.

Croman et al. (2016) examined how essential and arbitrary bottlenecks in bitcoin limit its present distributed overlay system's capacity to help significantly higher throughputs and lower latencies. The outcomes recommend that block size and interims' reparameterization should be seen distinctly as a first increase toward accomplishing people to come, high-load blockchain conventions. Significant advances will additionally require a fundamental re-evaluation of specialized methodologies. This method is resulting in an organized point of view on the plan space for such scenarios.

Gorenflo et al. (2019) re-constructed permissioned blockchain framework, Hyperledger Fabric, to build exchange throughput from 3,000 to 20,000 exchanges for every second. They concentrated on the existing execution bottlenecks in the hyperledger Fabric and made changes in the parameters which determine the calculation and I/O overhead.

Dinh et al. (2017c) proved the difficulty of scaling permissioned blockchain applications to serve many customers without hitches efficiently. It describes Blockbench, the

assessment structure for breaking down private blockchains. The author fills a reasonable method for correlation for different stages and empowers further comprehension of other frameworks.

The investigation point outs there is a demand of improving the capability of organizations to handle more number of transactions or workload. This chapter presents a general structure for portraying and breaking down the scalability of the hyper ledger fabric. The system endeavors to determine the instincts, ambiguities, and irregularities fundamental to the utilization of processing.

The Research Contributions are:

- Practical Byzantine Fault Tolerance (PBFT) consensus with high throughput.
- Scalable model of PBFT process.
- Analysis of the consensus process with a large number of transactions with large number of peers.
- Better confirmation times for transactions over 30000.
- The Proposed work has utilised Apache-spark to handle more transactions, thus increases scalability.
- The proposed solution directive usage increased the scalability 10 times higher than that of the existing scalability, that is an improvement from 3000 to 30000 transactions.
- The incorporation of parallelism in the solution directive increased the opportunity to add more transactions in the system, leading to a reduction in overhead.
- The consensus mechanism can utilize the opportunity to handle more transactions.

The rest of the chapter is organized as follows, Section 3.2 contain the important terminologies related to this work, Section 3.3 deals with the proposed solution direc-

tives, Section 3.4 describes results and discussion, Section 3.5 summarizes the proposed method and its significance.

3.2 MATERIALS AND METHODS

This section introduces some important terminologies related to this work.

3.2.1 Smart contract

Smart contracts are just like contracts in the real world. The ultimate difference is that they are entirely digital. A smart contract is a compact computer program that is stored inside a blockchain. The smart contract will hold all the received funds until a particular goal is reached. For example, consider the execution of a project. The supporters of the project can transfer their money to the smart contract. If the project gets fully funded, the smart-contract passes all the money to the project's creator(Szabo (2018)). If the project fails to meet the sufficient fund within the time frame, the money automatically goes to the supporters. Since the smart contract is inside the blockchain, everything is distributed and immutable; hence the smart contract is completly trustable. Based on the business logic, several functions can be defined within a smart contract.

3.2.2 Consensus

Consensus mechanisms ensure the records are genuine and honest. Consensus is the basic building block of a distributed ledger (Ongaro and Ousterhout (2014)).The consensus mechanism ensures that all the transactions occurring on the network are genuine, and all participants agree on an agreement on the ledger's status. Public blockchains such as bitcoin use Proof of Work (PoW) as the consensus mechanism. There are vast variants of consensus mechanisms such as Proof of Stake (PoS), Proof of Authority (PoA), Proof of capacity(PoC), Practical Byzantine Fault Tolerance (PBFT), etc. Hyperledger fabric uses PBFTSousa et al. (2018).

3.2.3 Hyperledger fabric v2.0

Hyperledger Fabric is a stage for distributed record arrangements supported by a private design conveying high levels of secrecy, strength, adaptability, and versatility. It is

intended to help pluggable segment's usage and bind the unpredictability and complexities that exist across the financial ecosystem. Hyperledger Fabric has been updating for the last few years. Currently, it is on the v2.x version.

3.2.3.1 Nodes

A blockchain contains few nodes which interact with each other for processing the transactions. Since hyperledger fabric is a permissioned network, the nodes have a unique identity provided by the membership service provider(MSP). A node can run in physical hardware, a container, or a virtual machine. According to hyperledger fabric, there are three types of nodes, namely, peers, orderers, and clients. The noticeable change in the hyperledger is its peers. The peers are decoupled into endorsers, committers, and consenters. Peers are the nodes that run the transactions and maintain them in the ledger. Peers will receive ordered state update as a block from the ordering service and maintain it in the ledger, so by default all peers are committers. Peers have an additional duty as an endorser. They will execute the smart-contracts and simulates the transactions. The consenters verify whether the peers have exchanged some assets. Orderers order the transactions. The collection of orderes are termed as ordering service. And finally, the end-users will be clients; they will send the transaction request to the peers. The clients will coordinate the orders and committers during the verification process.

3.2.3.2 System overview

Consider a small network shown in Figure 3.1, where Three organizations say O1, O2, O3, would create a distributed ledger among them. Each organization can be considered as a validating peer (Androulaki et al. (2018)). One among the validating peers is assigned as the network initiator(O1). The clients send transaction requests through validating peers via multiple channels(C1, C2, C3). Validating peers validate the transaction and broadcast this transaction.. Peer node P1 stores a copy of the ledger L1 associated with C1. Peer node P2 supports a copy of ledger L2 associated with C2. Peer node P3 has a copy of the ledger L3 associated with C3. Channel C1 is governed according to channel configuration CC1; the channel is under the control of O1

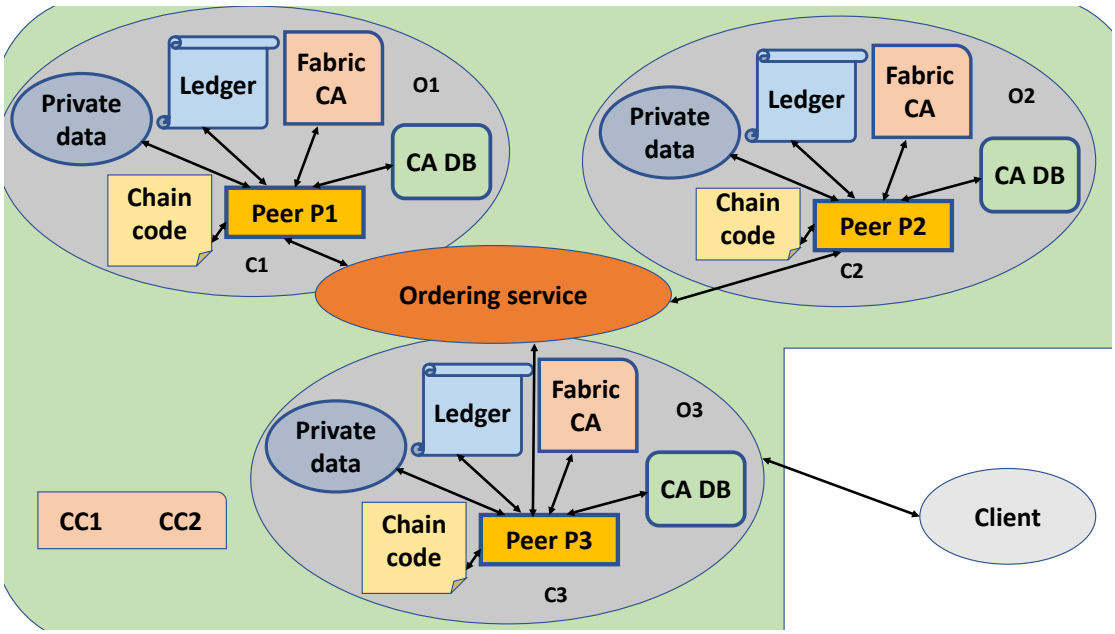

Figure 3.1: System overview of Hyperledger Fabric

and O2. Channel C2 is under O2 and O3, and governed according to the policy rules specified in channel configuration CC2. There is an ordering service that services as a network administration point and uses the system channel (Krstić and Krstić (2020)). Each of the organizations has a preferred Certificate Authority(CA), which will provide certificate-services to their peers.

3.2.3.3 Transaction flow

The transaction flow of hyperledger Fabric with three endorsing peers and one committing peer will occur as per the convenience of protecting data confidentiality in the transaction. The chaincode references the data collection. Figure 3.2 explains the transaction flow. The client application submits a proposal request to invoke a chaincode function to endorsing peers that are a part of approved organizations. The endorsing peers simulate the dealing and store the non-public information in a temporary data store and send the proposal response back to the client. The response consists of the supported read/write set and a hash of keys. The client application submits the transaction to the ordering service. The hashed transaction gets added to the block and is distributed among peers. The peers will validate the data by checking whether they can access the data during the commit time. If they have the authority to do so, the peers

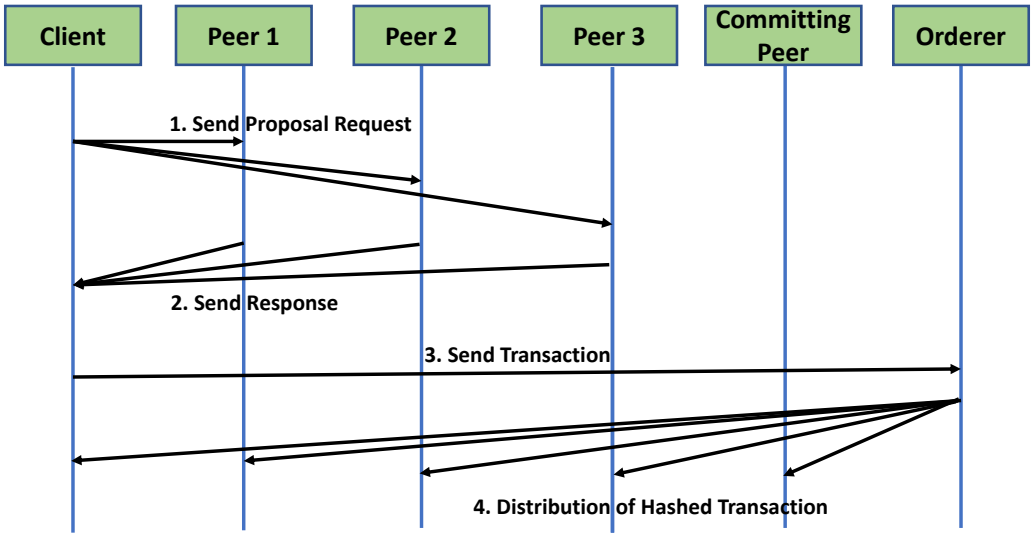

Figure 3.2: Transactionflow of Hyperledger Fabric

will check in the temporary datastore whether their data is already received. If not, they will pull the data from their peers and validate the data. After validation, the data's copy is moved to private storage and deleted from the temporary storage.

3.3 PROPOSED METHODOLOGY

The proposed framework shown in Figure 3.3 helps to analyze the scalability through transaction latency and throughput. If the throughput remains constant or increasing with the increase in number of transaction, then the framework is scalable. In another way, if the latency remains constant or decreasing in the rise in transactions, then also it is considered a scalable framework (Hill (1990)). Scalability indicates the capacity of a framework to oblige the developing volume of work and suit amplification.The users have a choice to use the proposed framework according to their business use-cases. The framework's performance assessment leads to the scalability investigation by differing at least one independent variable and estimating the reliant factors. In the real world, the measurements are shifted between every production trial. However, a portion of the measures can be moved during a trial. Distributed Machine Learning (ML) can be a solution to the scalability issue.

Distributed machine learning refers to algorithm for multi-node systems to enhance

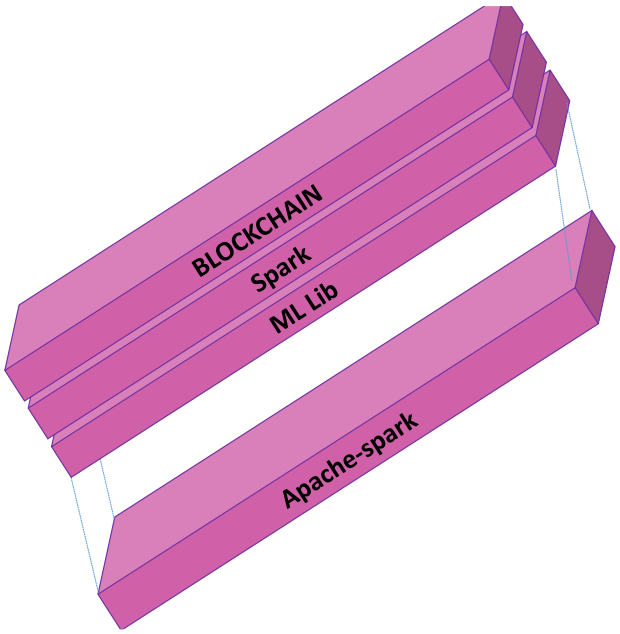

Figure 3.3: Proposed Methodology

their performance, precision, and larger input data sizes. Distributed ML algorithms are accessible to deal with enormous data sets and create efficient and scalable algorithms concerning exactness and calculation requirements (Shanahan and Dai (2015))(memory, time, and correspondence needs). Apache-spark MLLib can be the solution to the scalability issue. Apache Spark is an open-source cluster processing system for constant information preparation. The principle highlight of Apache Spark is, in-memory cluster figuring that speeds up an application. Spark gives an interface to programming whole bunches with certain information parallelism and adaptation to non-critical failure. It is intended to cover a broad scope of workloads, for example, batch applications, iterative calculations, intelligent inquiries, and streaming. Blockchain has been built over apache-spark, as shown in Figure 3.3. This forms the biggest coordinated standard for a large number of transactions and will be undoubtedly led to empowering the effectiveness of workflows. This work has implemented memory management outside the java virtual machine (JVM) and runtime code to bring Data and SQL exhibition. A few parts of transaction approval can be parallelized utilizing spark. Fabric's transaction validation service has been redesigned, and it has built over spark. This technique has parallelized as many validations as possible (Hill (1990)). Blockchain has four layers namely; Contract layer, network layer, consensus layer and application layer. The ac-

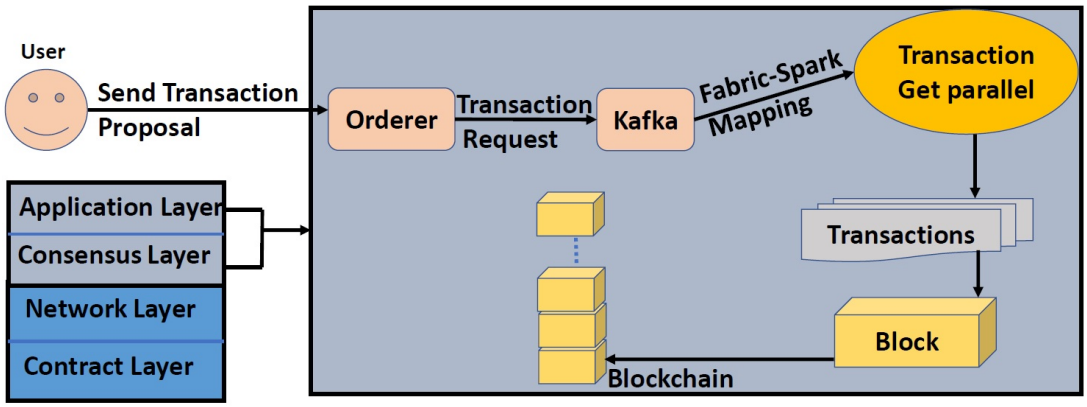

Figure 3.4: Detailed Structure of Blockchain in Proposed Methodology

tion of spark is on the application and consensus layer of the blockchain as shown in Figure 3.4. That is, When the orderer gets a transaction proposal, it checks whether the client is approved to present the transaction. The orderer will then send the particular transaction request to the Kafka cluster (Explanation in section 3.4.1), where each Fabric is mapped to the spark, to create respective orders of transactions. Prior to the mapping the transactions are passed through the Machine learning algorithm called Random Forest. Random Forest algorithm is selected because of its less training time and high accuracy. Which will help the transactions to take decision on selection of the channels, hence the transaction will occur parallelly. It will finally be collected into a block-based on the maximum number of transactions allowed in a block (Sukhwani et al. (2017)), which will uniquely identify a group of transactions. The transaction work flow and the mapping process of fabric and spark is shown in Figure 3.5. Since the process executes parallel, it can accommodate more transactions and make the system scalable.

3.4 RESULTS AND ANALYSIS

3.4.1 Setup

For the performance evaluation and validation of the proposed solution directive, a private small blockchain network has been created at NIT Karnataka using hyperledger fabric installed on a 64-bit Ubuntu operating system with 8 GB RAM. The experiments are carried out with a blockchain framework version 1.4.0 and version 2.0 of

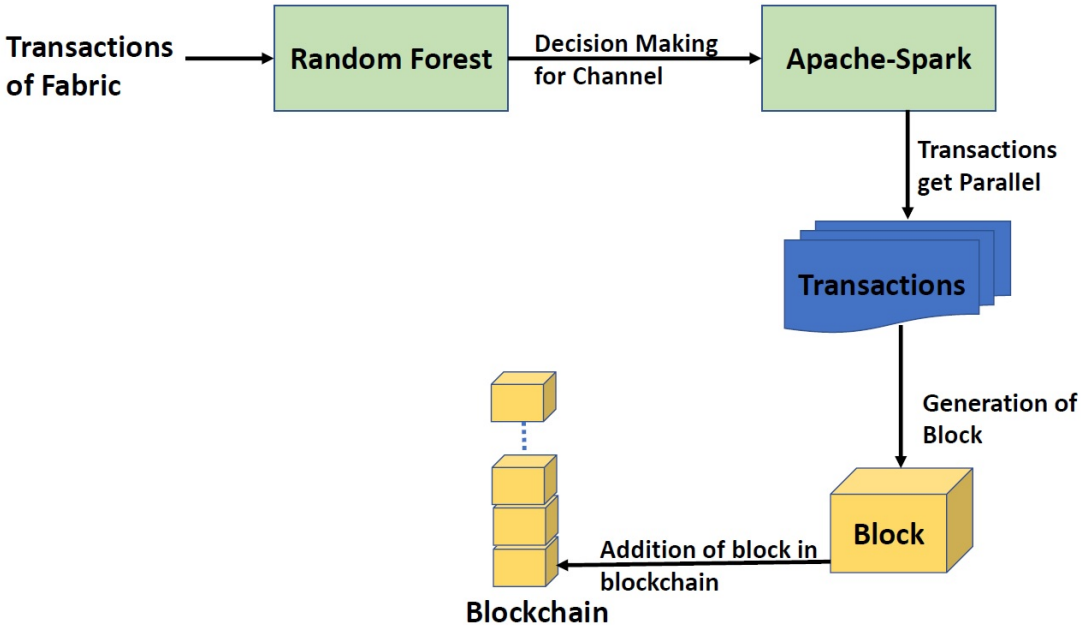

Figure 3.5: The Transaction Workflow and the Mapping Process of Fabric and Spark

Hyperledger Fabric, which is the latest version available during our investigations by considering three organizations. Each organization has one peer, one Certificate Authority client (CA), and one Membership Service Provider(MSP), which means that there are N organizations in a network of N peers. All organizations are connected using a single channel, as shown in Figure 3.1. The chaincode used for the experiments is written in Golang language. There are two different ordering services implemented in Fabric , SOLO and Kafka-based ordering service. SOLO is only meant for testing and not built for a production environment; therefore, the proposed work uses Kafka in the experiments. This ordering service consists of a variable number of Kafka servers and Zookeeper nodes. ZooKeeper is a cluster (a group of nodes) used to communicate and preserve shared data using robust synchronization techniques. ZooKeeper itself is a distributed application that offers distributed application writing services. There needs to be an odd number of Zookeeper nodes to avoid split-head-decisions. A minimum of four Kafka servers is recommended for fault tolerance. The Blockchain network is the assortment of hubs that run the system. It incorporates various equipment, programming, structure, and design of each required course (Chung et al. (2019)). The test is the arrangement of hubs that execute the performance assessment. These hubs

are customers that can play two broad classes of jobs, namely, Load-generating clients and Watching customers. Load generating clients submit exchanges for the end-client, and Watching customers will question their peers or get warnings regarding the transaction's completion status (Yasaweerasinghelage et al. (2017)). The test also gathers and examines the required datasets to assess performance metrics.

3.4.2 Parameters

3.4.2.1 Transaction Latency

Transaction latency is the time taken between the exchange submission and the exchange confirmation over the system. This inertness incorporates the proliferation time and any settling time because of the agreement calculation on a consensus mechanism. Transaction latency is the measure of time produced for an exchange's results to be usable over the system (Zhou et al. (2020)). The transaction latency can be considered in two perspectives: The number of peers at which the exchange is seen to be settled and the percentage of perceptions equivalent to or beneath which the estimation is substantial(Percentile). Transaction latency is generally reported as average latency, which is determined as follows:

Average Transaction Latency = sum of transaction latency / total committed transactions

3.4.2.2 Transaction throughput

Transaction throughput is controlled by the block interim and the block size. A more significant block can store more transactions, legitimately raising throughput; however, it additionally causes an expansion in block proliferation time (Pongnumkul et al. (2017)). To guarantee the current block to be spread to most peers in the entire system before the following block is produced. Parallelism can be a solution to the above problem. Transaction throughput is the rate at which the blockchain arrange submits legitimate exchanges in the defined timeframe. The throughput of transactions is the rate at which legitimate transactions are committed. Therefore,

Transaction Throughput = total committed transaction/ total time.

3.4.3 Result analysis

3.4.3.1 Analysis of Transactions and Confirmation times

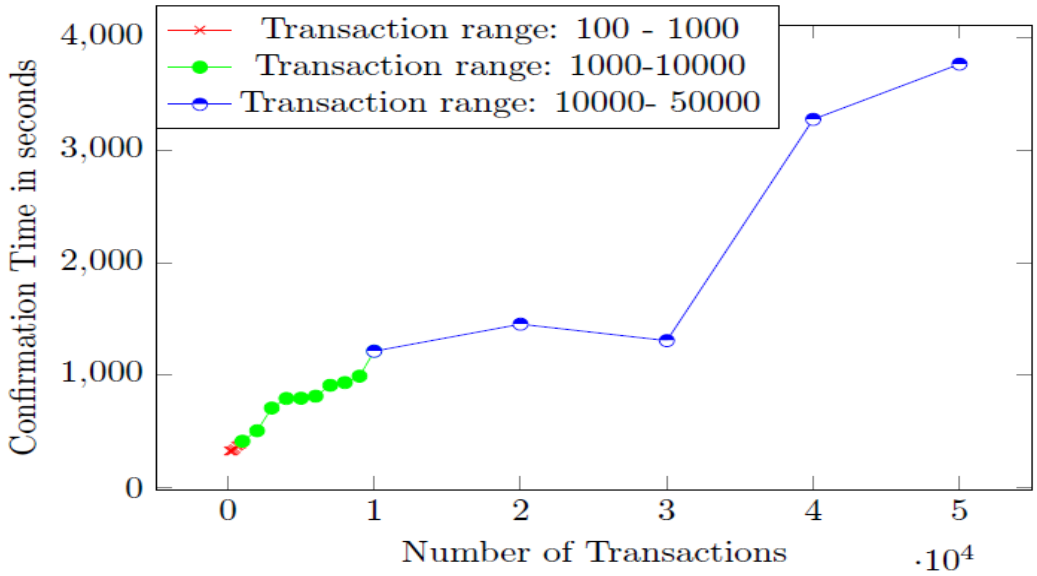

Figure 3.6: No.of transaction vs confirmation time

The graph has been plotted for transactions ranging from 0 to 50000 is shown in Figure 3.6. The confirmation time has measured in seconds. From the observation, it is clear that the confirmation time is almost the same till 30000 transactions. Then there is a gradual increase in the confirmation times with an increase in the number of transactions from 30000 transactions. These values have been gathered from the above explained experimental setup. The findings from the graph are:

- There is a steep increase in confirmation times for transactions over 30000.
- An increase in the number of transactions increases the throughput of the system at the cost of increasing network latency.
- An increase in the number of transactions boosts the overhead system for extraction of further transactions into blocks

3.4.3.2 Analysis of Transactions and Throughput

The throughput of the blockchain framework is characterized by the number of verified transactions per second. Most of the modern payment systems have a throughput of

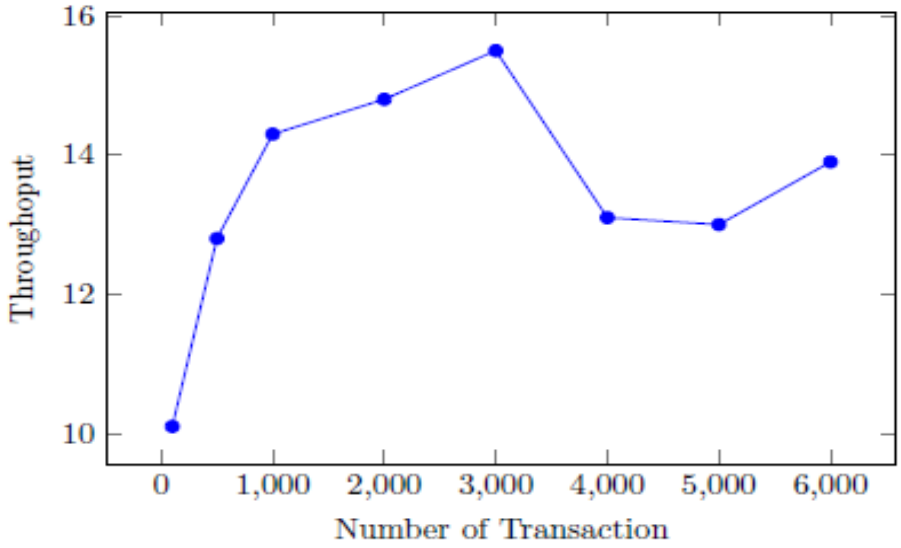

Figure 3.7: Throughput- before applying solution

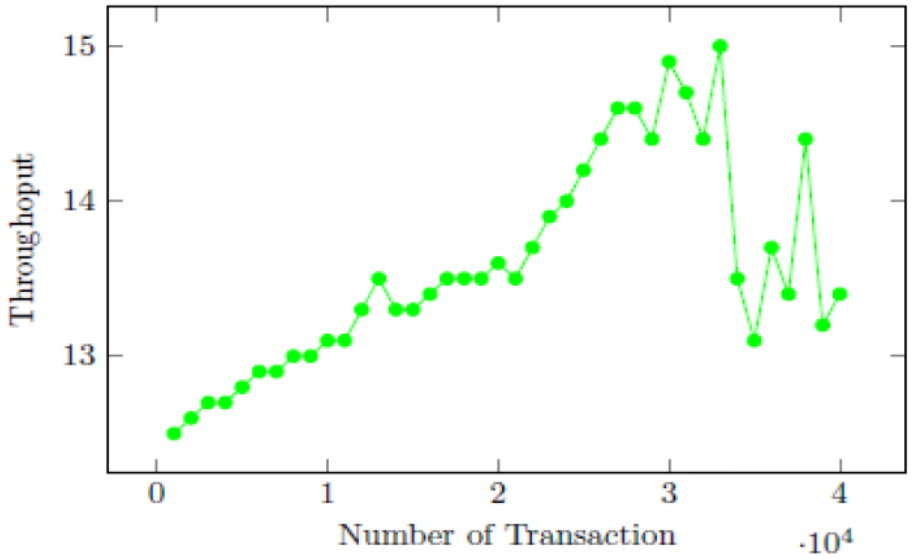

Figure 3.8: Throughput- after applying solution

an average of 2000 transactions per second. Bitcoin-based blockchain systems have an average throughput of processing only seven transactions per second. 3000 transactions per second are the average throughput of an hyperledger fabric system. An increase in the transaction load on the network and larger block size leads to hard forking. An increase in the block size of the hyperledger Fabric to achieve higher throughputs will compromise blockchain security and decentralization. So the increment of blocksize is not practical, however as per our solution directive, the apache spark did the process. From Figure 3.7 and Figure 3.8, it is clear that the proposed framework reduces the workload of Fabric to a great extend. The findings from graph are:

- The hyperledger Fabric shows a good transaction throughput to 30000 transactions. After 30000 transactions, the throughput is decreasing abruptly.
- The proposed solution directive usage increased the throughput of the blockchain up to 30000 transactions, which is 10 times higher than that of the existing throughput.
- The incorporation of parallelism in the solution directive increased the opportunity to add more transactions in the system, leading to a reduction in overhead.

3.4.3.3 Analysis of Transactions and Latency

The delay caused by propagating the blocks in the network is termed as the blockchain network's latency. The time taken to validate a transaction has a direct effect on the latency of the network. Faster transaction confirmation time means either lower latency or quicker transmission of the network.. While analysing and comparing the results in Figure 3.9 and Figure 3.10, the following observations are noticed.

- The hyperledger Fabric shows a good transaction latency up to 30000 transactions and an abrupt increase in latency after 30000 transactions.
- It would mean the rise in the number of transaction load on the system will increase the latency. It is expected that the reason for the increase in the latency is the increase in the block size.

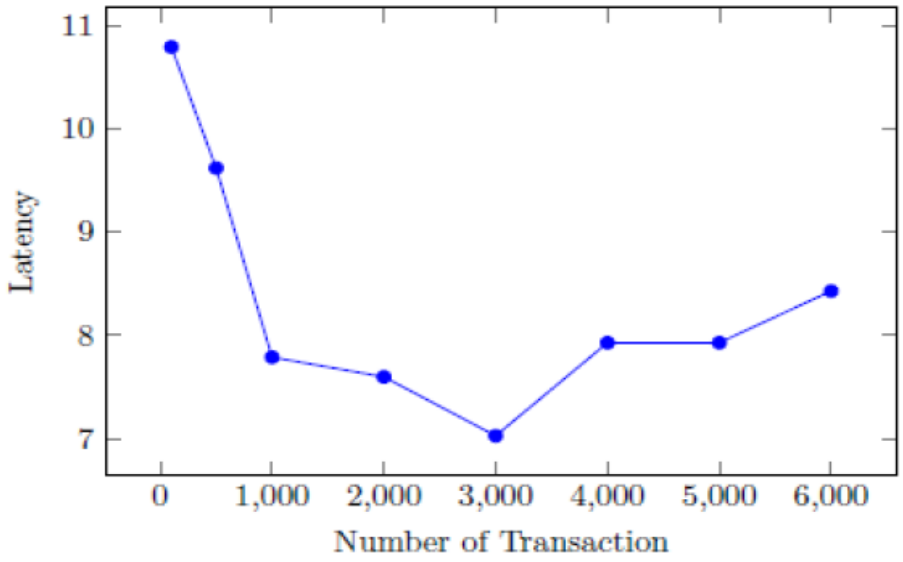

Figure 3.9: Latency- before applying solution

- More network bandwidth or more resources to propagate heavier blocks may lead to network congestion and thus increase the latency.

3.4.3.4 Analysis of scalability

It is evident from the above two results that the introduction of the proposed framework keeps the throughput and latency stable upto 30000 transactions. After parallelizing the transaction process using spark, it could handle more transactions, thus increases scalability. The speedup of the confirmation time also means the improvement in scalability. Since blockchain is built over spark, the consensus mechanism can utilize the opportunity to handle more transactions. As a result, the consensus in hyperledger Fabric (PBFT) is said to be scalable.

This paper discussed about different parameters of permissioned blockchain, and from the experiments it is clear that the parameters such as scalability, transaction throughput, transaction latency and the confirmation time are improved to an extend from the existing trait. This observation is really an encouragement to the industry to welcome blockchain in to their field and to make use of the enormous possibilities of blockchain. Even though the paper improves the scalability, it opens the door for the researchers to improve the scalability from 30000 to an unlimited number of transac-

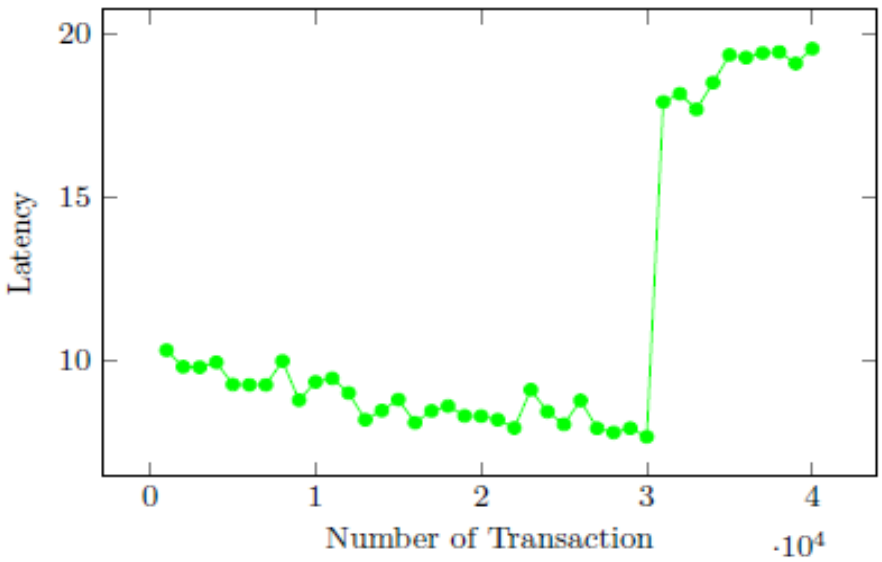

Figure 3.10: Latency- after applying solution

tions.

3.5 SUMMARY

This chapter mainly focused on the scalability of permissioned blockchain. Through the Bitcoin application, the innovative technology was miraculously launched in the markets, influencing numerous industries. Bitcoin is nothing but a form of digital currency (cryptocurrency) that can be used for trading in place of fiat money, where the underlying infrastructure is called Blockchain. The Blockchain is an open ledger that provides decentralization, transparency, immutability, and confidentiality. Blockchain can be used in massive, beneficial applications such as healthcare, logistics, supply chain management, the Internet of Things (IoT), etc. Most of the industrial applications rely on the permissioned Blockchain. However, the permissioned Blockchain fails in some aspects, such as scalability and throughput. This chapter suggests a system to solve the scalability issue of permissioned Blockchain by incorporating data science techniques. The scalability analysis of the proposed solution is done in the hyperledger fabric framework with a variable number of transactions and results in scalability improvement.

CHAPTER 4

INTEROPERABLE PERMISSIONED- BLOCKCHAIN

4.1 INTRODUCTION

Blockchain is a decentralized computational and information-sharing platform enabling multiple authoritative domains which do not trust each other to cooperate, coordinate, and collaborate in rational decision-making processes (Nakamoto (2009)). It is an electronic, decentralized ledger that keeps a copy of all the transactions that take place within the network, which is peer-to-peer. It is a continuous list of transaction records stored in encrypted form, called a ”block”. Each block is uniquely connected with the previous block by digital signature, so that the record cannot be altered or tampered with without disturbing the records in the previous block of the chain, which makes the blockchain immutable. The unique feature of Blockchain is that there is no need for a third-party authentication mechanism. The transaction becomes valid if the entire peer-to-peer in the network agrees the transaction. One of the applications of Blockchain in crypto currency is bitcoin. Let us see the workings of Blockchain in terms of the Bitcoin transaction life cycle. Consider the scenario where Alice wants to send some coins to Bob—initially, Alice opens her Bitcoin wallet and provides the address of Bob and amount to transfer. Then she presses the send button, and the wallet constructs the transaction which is signed using Alice’s private key. By applying digital signature techniques, the wallet signs the transaction made by Alice and broadcasts it over the network. Depending upon the network, all hubs in the system, or the majority of the hubs in the system receive that particular transaction. After receiving the transaction,

the nodes in the network will validate the transaction based on the existing blockchain. Once this transaction is validated, then It is propagated to some particular nodes called miners (Swathi et al. (2019)). The miner collects all the transactions for a duration of time, and they construct a new block and try to connect it with the existing blockchain through some cryptographic hash computation, and then they propagate the updated blockchain in the network.

Blockchain can be used in substantial valuable applications, such as medicinal services, academics, banking marketing, and much more. The cryptocurrencies mentioned previously come under the permissionless blockchain. Permissionless blockchains are also known as public or decentralized blockchains. Anyone can create and access the blockchain in which anyone can publish the self-executing contract (a smart contract, which will be explained in Section 4.2.1). Moreover, anyone can run the blockchain node with 100 percent transparency. However, organizations require an entirely different type of blockchain, which can safeguard their terms and policies. It should incorporate only pre-approved nodes. This type of blockchain is called permissioned blockchains.

The permissioned blockchain requires every peer to execute every transaction, maintain a ledger, and run a consensus (which will be explained in Section 4.2.2), a fault-tolerant mechanism. It cannot support the valid private transaction with confidential contracts. Hyperledger Fabric is one of the best blockchains which can deliver the modular and secure foundation for the industrial blockchain. The hyperledger fabric generally uses a practical Byzantine Fault Tolerance (PBFT) consensus algorithm. However, some factors pull back the industries from adapting the blockchain fully fledged. Interoperability among platforms is one among them. Interoperability indicates the possibility to freely share value across all blockchain networks without the need for intermediaries. Interoperability among enterprise systems is defined by Vernadat as ”a measure of the capacity to execute interoperation between entities” (processes, software, systems, business units) (Vernadat (2007)). The issue is to make it easier for various processes and units to ”communicate, cooperate, and coordinate”. Technical interoperability, legal interoperability, semantic interoperability, integrated public service

governance, organizational interoperability, and interoperability governance are some of the interoperability levels. Technical interoperability, for example, is concerned with the technical processes that enable blockchain integration, whereas organizational interoperability is concerned with whether different organizations can work together across different blockchains a shown in Figure 4.1.

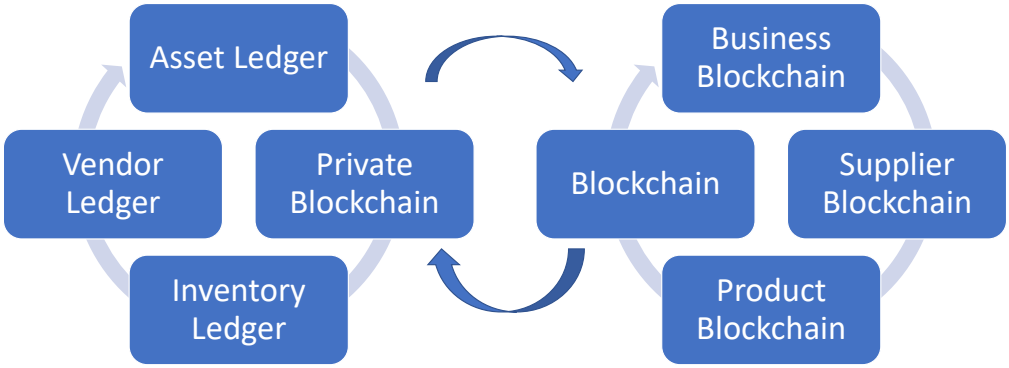

Figure 4.1: Interoperability among blockchain platforms.

Centobelli et al. (2021a) provided a broad qualitative and quantitative review of blockchain research using bibliometric analysis methodologies, and also bridged a research gap concerning the absence of a thorough overview of blockchain using a rigorous analytical method. It provides deep insights into current debates, advances the research area linked to the contextual and multilayered phenomena of Blockchain, and leads to future research paths. It is clear that Blockchain technology and its practical use require further scientific, helpful, and legal implementation (Karpenko et al. (2019)). The problem of establishing the status of cryptocurrencies and distributed registry technology is now on the table, and both good and bad instances may be used to show the necessity for a balanced and clearly defined economic policy in this field. Despite the fact that interoperability has a broad scope, we primarily focus on technical and organizational interoperability because this is where the majority of blockchain

interoperability efforts is centred.

Interoperability will become an important aspect for any project to succeed as the blockchain industry continues to expand and advance. To have hundreds of blockchains totally isolated from each other makes no sense. The ability to exchange knowledge openly through blockchain networks is interoperability. In a completely interoperable world, you will be able to easily read, understand, and communicate with little effort if a user from another blockchain sends something on the blockchain (Jolma and Rizzoli (2003)). To allow the above-mentioned use cases, there are three main techniques: notary systems, side-chains, and hash-locking notary schemes. Notary schemes use a trusted party between two blockchains as an intermediary. Therefore, the notary's job is to verify that a blockchain event took place and to feed this information to a second blockchain. Clarity is the key benefit of the notary scheme, as no changes in the underlying blockchains are needed. As one possible solution, a set of notaries they trust might be selected by all parties involved. By using consensus algorithms, such as BFT, the performance of notaries could then be generated. There would be no need to trust every single notary, but just two-thirds of the sidechain community (Mockapetris and Dunlap (2001)). A sidechain is a blockchain that has the potential to verify and collect data about the status of other blockchains. Although the data need to be fed from one blockchain to the other externally, due to the cryptographic properties of blockchains, this process does not involve trust. It would be easy to produce evidence that the headers were tampered with. By being able to enter the state from other blockchains, sidechains allow for a variety of use cases. To build a sidechain, however, smart contract capabilities are required. In addition, each blockchain will require a sidechain to attain maximum interoperability, which in turn needs to support every other blockchain. Interoperability between various blockchains, interoperability between dApps utilising the same blockchain, and interoperability between blockchain and other technologies were all highlighted by Besancon et al. (2019). According to Buterin (2014), the interoperability solutions sought to enable compatibility between cryptocurrency systems. This category catalogues and specifies several chain interoperability techniques used by public blockchains that allow cryptocurrencies, such as

hash time hashlocks, sidechains, and notary schemes. Centobelli et al. (2021b) pointed out that in the field of circular supply chains, there is a growing corpus of blockchain literature. An increasing interest in the issue necessitates additional practical study on the design and execution of blockchain systems, in addition to the substantial theoretical contribution. The authors (Centobelli et al. (2021b)) analysed six key clusters of blockchain-related research contributions and divided research themes into motor themes, fundamental themes, emerging or fading themes, and specialized themes based on the centrality and density metrics. Even though the majority of contributions are in computer science, many papers on technology management provide valuable information to scholars. While many standards address various aspects of interoperability, there is still space for improvement.

Let us see these concepts in detail.

4.2 MATERIALS AND METHODS

This section introduces some important terminologies related to this work.

4.2.1 Smart Contract

Smart contracts are just like contracts in the real world. The ultimate difference is that they are entirely digital. A smart contract is a compact computer program that is stored inside a blockchain. The smart contract will hold all the received funds until a particular goal is reached. For example, consider the execution of a project. The supporters of the project can transfer their money to the smart contract. If the project gets fully funded, the smart-contract passes all the money to the project's creator (Szabo (2018)). If the project fails to meet the sufficient fund within the time-frame, the money automatically goes to the supporters. Since the smart contract is inside the blockchain, everything is distributed and immutable; hence, the smart contract is completely trustable. Based on the business logic, several functions can be defined within a smart contract.

4.2.2 Consensus

Consensus mechanisms ensure the records are genuine and honest. Consensus is the basic building block of a distributed ledger (Ongaro and Ousterhout (2014)). The consen-

sus mechanism ensures that all the transactions occurring on the network are genuine, and all participants agree on an agreement on the ledger's status. Public blockchains, such as bitcoin, use Proof of Work (PoW) as the consensus mechanism. There are vast variants of consensus mechanisms, such as Proof of Stake (PoS), Proof of Authority (PoA), Proof of capacity (PoC), Practical Byzantine Fault Tolerance (PBFT), and so forth. Hyperledger fabric uses PBFT (Sousa et al. (2018)).

4.2.3 Hyperledger Fabric v2.0

Hyperledger Fabric is a stage for distributed record arrangements supported by a private design conveying high levels of secrecy, strength, adaptability, and versatility. It is intended to help pluggable segment's usage and bind the unpredictability and complexities that exist across the financial ecosystem. Hyperledger Fabric has been updating for the last few years. Currently, it is on the v2.x version (IBM Corporation (2019a)).

4.2.3.1 Nodes

A blockchain contains a few nodes which interact with each other for processing the transactions. Since hyperledger fabric is a permissioned network, the nodes have a unique identity provided by the membership service provider (MSP). A node can run in physical hardware, a container, or a virtual machine. According to hyperledger fabric, there are three types of nodes, namely, peers, orderers, and clients. The noticeable change in the hyperledger is its peers. The peers are decoupled into endorsers, committers, and consenters. Peers are the nodes that run the transactions and maintain them in the ledger. Peers will receive an ordered state update as a block from the ordering service and maintain it in the ledger, so by default, all peers are committers. Peers have an additional duty as an endorser. They will execute the smart contracts and simulate the transactions. The consenters verify whether the peers have exchanged some assets. Orderers order the transactions. The collection of orderers is termed as an ordering service. Finally, the end-users will be clients; they will send the transaction request to the peers. The clients will coordinate the orders and committers during the verification process.

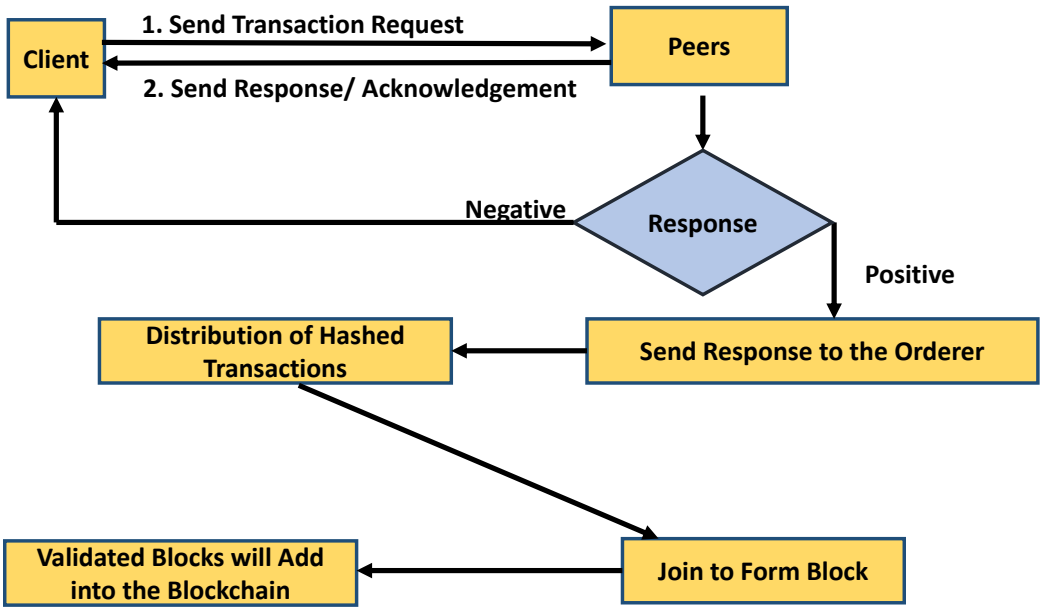

Figure 4.2: Transaction flow of hyperledger fabric.

4.2.3.2 Transaction Flow

The transaction flow of hyperledger fabric with three endorsing peers and one committing peer will occur as per the convenience of protecting data confidentiality in the transaction. The chaincode references the data collection. Figure 4.2 explains the transaction flow. The client application submits a proposal request to invoke a chaincode function to endorsing peers that are part of approved organizations. The endorsing peers simulate the dealing and store the non-public information in a temporary data store and send the proposal response back to the client (Androulaki et al. (2018)). The response consists of the supported read/write set and a hash of keys. The client application submits the transaction to the ordering service. The hashed transaction gets added to the block and is distributed among peers. The peers will validate the data by checking whether they can access the data during the commit time. If they have the authority to do so, the peers will check in the temporary data store whether their data have already been received (Krstić and Krstić (2020)). If not, they will pull the data from their peers and validate the data. After validation, the data's copy is moved to private storage and deleted from the temporary storage.

4.2.3.3 Ethereum

In Buterin's article (Buterin (2015)), Ethereum was presented to cover some shortcomings of Bitcoin. Ethereum then supports the status of the contract, as well as some other changes to the framework of the blockchain. Ethereum is made up of a network of cryptographic, or protected, public documents that are hard to alter because they are stamped with user data, time and date, and modifications that must be accepted by all users (Androulaki et al. (2018)). Anyone may establish a financial arrangement or hold debt or ownership registries on the ledger, removing the need for a third-party record-keeper or trust officer. They are called "trustless" transactions, and they do not include trusting the transaction's counter-party. Ethereum is a permissionless, non-hierarchical computer (node) network that builds and decides on an ever-growing sequence of "blocks" known as the blockchain. Whenever a node attaches a block to the chain, the transactions are always executed in their order and modify the Ethereum account storage values. A relatively small subset of the network, known as its peers, connects with each node. The transaction flow of Ethereum is depicted in Figure 4.3. Whenever a node tries to add a new transaction to the blockchain, it sends it to its peers, who send it to their peers, and so on. It travels across the network this way. Some nodes, known as miners, hold a list of all these recent transactions and use them to create new blocks, which are then sent to the rest of the network. Whenever a node receives a block, the validity of the block and of all its transactions is checked and, if correct, added to its blockchain, and all such transactions are executed. A node can obtain competing blocks, which may form competing chains, since the network is non-hierarchical. The network achieves unity on the blockchain by applying the "longest chain law", which specifies that the canonical chain is the one with the most blocks at any given time. Since miners do not want to spend their computing energy attempting to connect blocks to a chain that would be abandoned by the network, this rule achieves consensus.

By considering the above explanations of Bitcoin, Hyperledger Fabric and Ethereum, it is clear that all the platforms are entirely different and they work in their own way. There comes the role of interoperability.

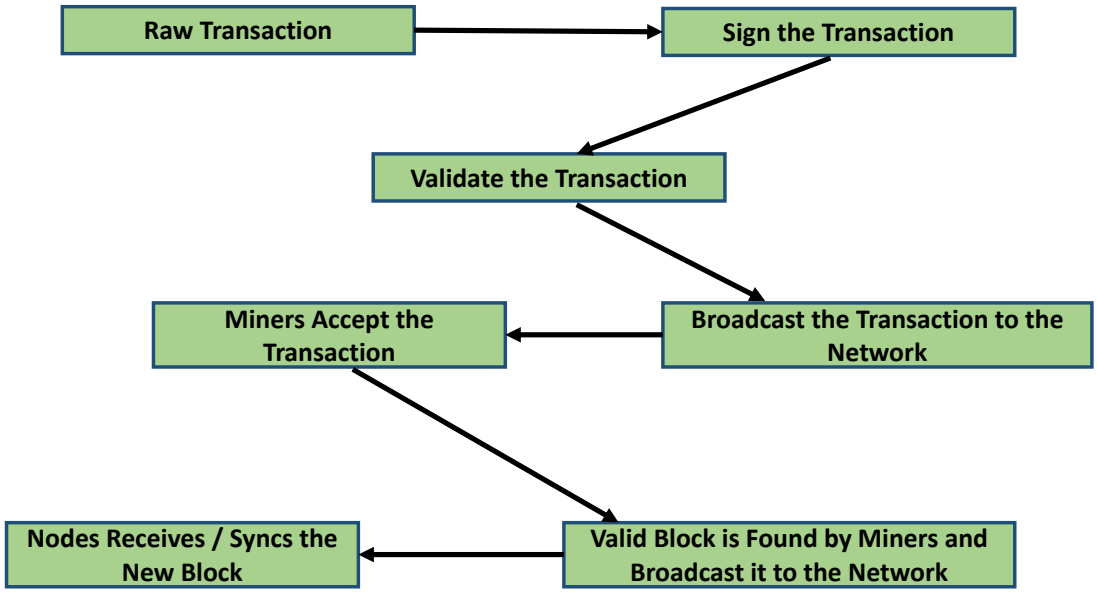

Figure 4.3: Transaction flow of Ethereum.

4.3 PROPOSED METHODOLOGY

Interoperability on the blockchain can be achieved using a variety of methods. According to the classes of strategies we defined, we divide blockchain interoperability into three categories: cryptocurrency-directed interoperability methods, Blockchain Engines, and Blockchain Connectors (Belchior et al. (2021b)). There are sub-categories within each division. It describes and establishes various chain interoperability techniques through public blockchains, the majority of which use cryptocurrencies. The next category focuses on general use-cases and heterogeneous systems, while the cryptocurrency-based interoperability approaches a category focused on cryptocurrency ecosystems, often homogeneous blockchain structures. Blockchain Engines are platforms that include reusable data, network, agreement, opportunity, and contract layers for building customizable blockchains that power decentralised apps. The use of tokens is included in this grouping, and is mostly used as an incentive tool for participants to adopt protocols and manage the network. The Blockchain Connector type includes non-cryptocurrency interoperability applications, as well as blockchain engines. Trusted Relays, Blockchain Agnostic Protocols, Blockchain of Blockchains, and Blockchain Migrators are some of the sub-categories we extracted from the studies (Belchior et al. (2021b)). Above all,

this work has utilised a notary scheme for structuring a new framework, so that it may use a trusted party between two blockchains as an intermediary. Therefore, the notary's job is to verify that a blockchain event took place and to feed this information to a second blockchain.

This work has considered two blockchain platforms: Hyperledger Fabric, a permissioned blockchain, and Ethereum, a permissionless blockchain. The notary scheme of Fabric and Ethereum is depicted in Figure 4.4.The supporting layers (e.g., networking, storage, and encryption) are used to build the consensus engine, which organises transactions and appends them to the chain of blocks (Kan et al. (2018a)). Hyperledger Fabric's consensus is modular and based on endorsement policies. A client (C) submits a transaction proposal to the peer nodes (P) and gets an endorsement (a signed transaction) in Fabric. The endorsements are checked by an orderer, who then produces a block of legal transactions that is added to the ledger. A node can suggest a block of transactions to be added to the ledger after discovering a PBFT solution. Because of the fundamental differences between the two types of blockchains, the interoperability challenge is unique. There are multiple layers for a blockchain (Qingyi et al. (2019)). The data layer defines how data on the blockchain are interpreted (e.g., transactions piled into blocks vs. transactions represented in a directed acyclic graph). The network layer defines the node category in a peer-to-peer network (Nakamoto (2009)). The consensus algorithm, as well as its security assumptions, are part of the consensus layer. The contract layer contains the smart contract execution environment, which provides the framework for the application layer, and includes blockchain-enabled corporate logic (Dinh et al. (2017a)).

This methodology considered each blockchain as a self-contained system that connects with others via a cross-chain protocol that includes a notary mechanism. Interoperability gateways are created by nodes on both public and private blockchains. To encourage interaction across blockchains, decentralised blockchain registries that can recognise and address blockchains and their components (e.g., smart contracts and certificate authorities) can be utilised (Vo et al. (2018)). A repository for both public and private blockchains might be created on a public blockchain with solid security

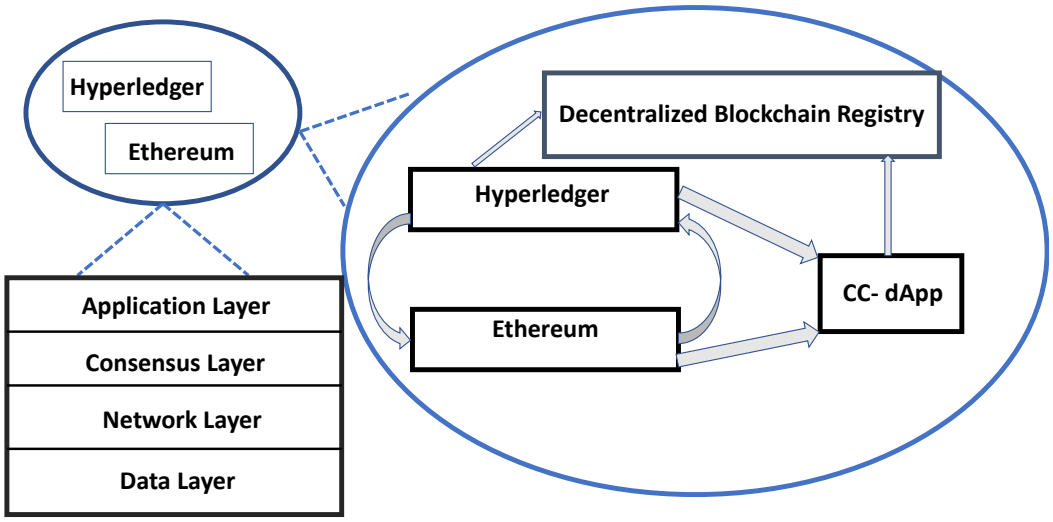

Figure 4.4: Network of Ethereum and Hyperledger Fabric in the notary scheme.

assumptions (Zheng et al. (2017)). The registry's information is maintained in a customised shared database administered by significant blockchain players (Hardjono et al. (2019)). The decentralised repository would work similarly to a decentralised domain name structure.

Consolidating all of these data resulted in the creation of a new architecture for addressing the interoperability issue. Figure 4.5 shows a concept for the Hyperledger Fabric chaincode, which includes Ethereum integration. Ethereum is a permissionless and EVM-based blockchain, while Hyperledger Fabric is a permissioned blockchain. An Ethereum blockchain node's interaction endpoint (i.e., IP address) is registered with the blockchain registry. After that, it looks for the address of a Hyperledger fabric server with which it is supposed to communicate. The Cross-Chain Communication Protocol (CCCP) and the Cross-Blockchain Communication Protocol (CBCP) provide for unidirectional or bidirectional interoperability (CBCP). Because the Hyperledger node interprets Ethereum's block headers but not the other way around, a CBCP allows the Ethereum and Hyperledger nodes to communicate arbitrarily . A CC-dApp that is already linked to Ethereum and Hyperledger fabric utilises the private blockchain to create the required credentials after getting its address from the blockchain registry. A CC-dApp protocol lets an end-user achieve semantic interoperability by utilising Hy-

perledger fabric and Ethereum. These actions establish blockchain connection, culminating in the establishment of connection among Hyperledger Fabric and Ethereum. Existing blockchains would require changes to multiple levels, including the network, consensus, contract, and application layers. In essence, this work uses the Hyperledger Fabric permissioned blockchain infrastructure to allow users to interact with Ethereum smart contracts written in an EVM (Ethereum virtual machine) compatible language called Solidity. To complete the integration, the EVM chaincode (EVMCC) and the web provider are utilised. The EVMCC is a Go chaincode that encapsulates the Hyperledger fabric EVM bundle and maps out the various ways between the peer and the EVM. The EVMCC acts as a smart contract runtime, placing the implemented contract code on the ledger in the EVMCC namespace. Users may connect with smart contracts running in the Fabric EVM using tools like Web3.js. A proxy that provides a subset of Ethereum-compliant JSON RPC APIs. The Fabric GO SDK allows the proxy to connect to the Fabric network and communicate with the EVMCC. The Ethereum Smart Contract Runtime and the Hyperledger Fabric Runtime are being rebuilt by the EVMCC and proxy. Applications that employ the Ethereum JSON RPC API and EVM smart contracts should be able to interact seamlessly with Hyperledger Fabric. Fabreum is the name given to this innovative design since it functions as both Ethereum and Fabric as shown in Figure 4.6.

4.4 RESULT AND ANALYSIS

Two blockchain networks were established at NIT Karnataka utilising Hyperledger Fabric and Ethereum installed on a 64-bit Ubuntu operating system with an 8 GB RAM for performance assessment and validation of the suggested solution direction. The tests were conducted using a Hyperledger Fabric version 2.0, as well as Ethereum 1.10.8, which is the most recent version accessible at the time of our research. Three organisations were considered in this study. There are N organisations in a network of N peers, since each has one peer, one Certificate Authority client (CA), and one Membership Service Provider (MSP). A single channel connects all of the organisations. The experiments' chaincode is written in the Golang programming language. SOLO and Kafka-based ordering systems are two distinct ordering services developed in Fab-

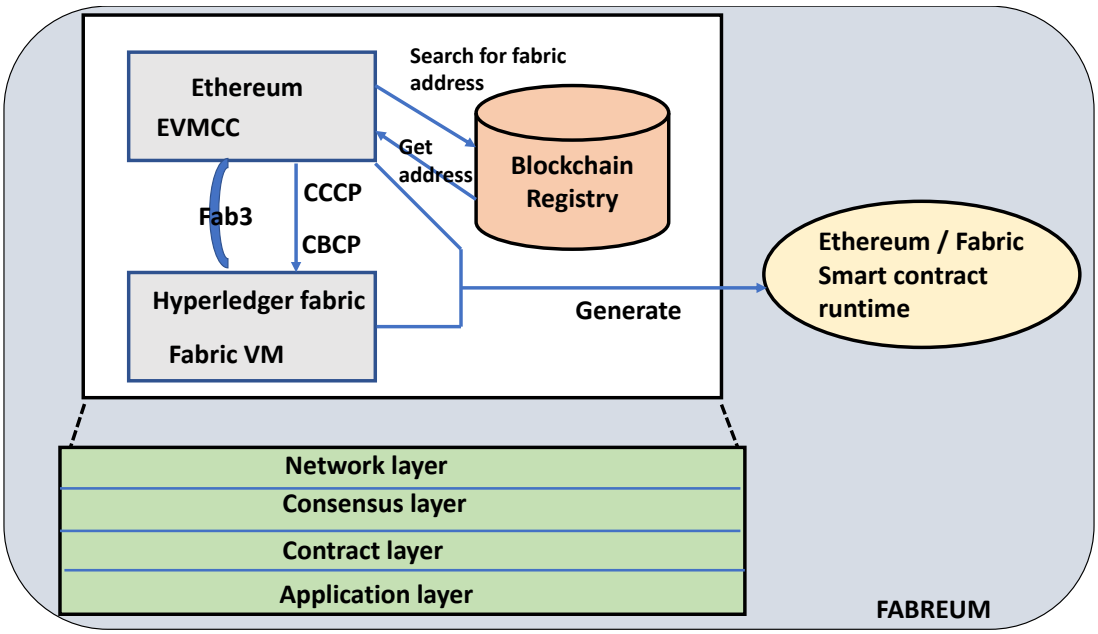

Figure 4.5: Interoperability framework of Ethereum and Fabric in detail.

ric. Since SOLO is only intended for testing and not for usage in a production context, the suggested work employs Kafka in the tests. The test gathers performance metrics using caliper.

The performance matrix obtained from caliper is shown in Table 4.1. The experiments were done by taking Ethereum and Hyperledger Fabric as the source and destination, respectively, and vice versa for 500 transactions. The same experiment was even done for Ethereum as a source as well as destination, and Hyperledger as a source as well as destination. The comparison of an output of interoperability before and after applying the solution directive is illustrated in Figure 4.7. All 500 transactions in each of the cases became successful, and each case obtained a good level of latency and throughput. This experiment has resulted in an average of a 25.55 tps send rate. The send rate is nothing but the number of transactions sent per second.

Send rate = total number of transactions send / total time.

All of the cases showed a similar pattern in send rate, which means the transactions happen irrespectively of the sender or receiver. Transaction latency is the measure of time produced for an exchange's results to be usable over the system (Zhou et al.

(2020)). There is a similar pattern of transaction latency visible in the matrix. The transaction latency can be considered from two perspectives: the number of peers at which the exchange is seen to be settled, and the percentage of perceptions equivalent to or beneath where the estimation is substantial (percentile). Transaction latency is generally reported as the average latency, which is determined as follows:

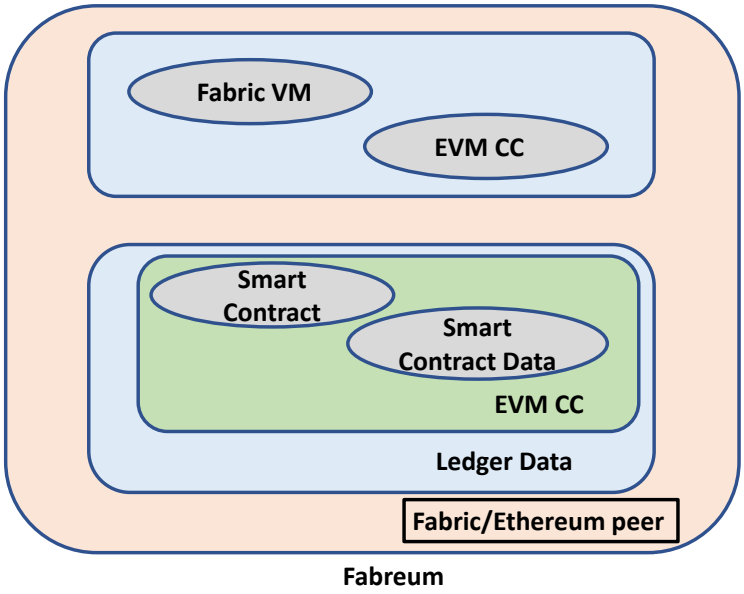

Figure 4.6: Consolidated interoperability framework of Ethereum and Fabric.

Average Transaction Latency = sum of transaction latency / total committed transactions,

and here we obtain an average of 8.96 s. In the same way, we collected an average transaction throughput of 13.25. The transaction throughput is the rate at which the blockchain arrangement submits legitimate exchanges in the defined timeframe. The throughput of transactions is the rate at which legitimate transactions are committed. Therefore,

Transaction Throughput = total committed transaction/ total time.

Vo et al. Vo et al. (2018) focused on interoperability architecture, providing some Blockchain of Blockchain and contract solutions. Buterin et al. Buterin (2014) gave

Table 4.1: Performance matrix of the system after applying Interoperability solution

Src	Destn	Succ	Fail	Send Rate	Max Latency	Min Latency	Throughput
Fabric	Ethereum	500	0	25.2 tps	18.39 s	0.78 s	13.2 tps
Fabric	Ethereum	500	0	25.3 tps	18.56 s	0.78 s	13.4 tps
Ethereum	Fabric	500	0	25.9 tps	18.43 s	0.79 s	13.2 tps
Fabric	Ethereum	500	0	25.4 tps	18.24 s	0.78 s	13.2 tps
Ethereum	Fabric	500	0	25.6 tps	18.37 s	0.77 s	13.2 tps
Ethereum	Fabric	500	0	25.7 tps	18.46 s	0.78 s	13.5 tps
Fabric	Ethereum	500	0	25.4 tps	18.95 s	0.79 s	13.1 tps
Ethereum	Ethereum	500	0	25.6 tps	18.21 s	0.78 s	13.3 tps
Ethereum	Ethereum	500	0	25.8 tps	18.32 s	0.79 s	13.1 tps
Fabric	Fabric	500	0	25.4 tps	18.54 s	0.77 s	13.4 tps
Ethereum	Fabric	500	0	25.7 tps	18.47 s	0.78 s	13.3 tps
Fabric	Fabric	500	0	25.7 tps	18.56 s	0.78 s	13.2 tps

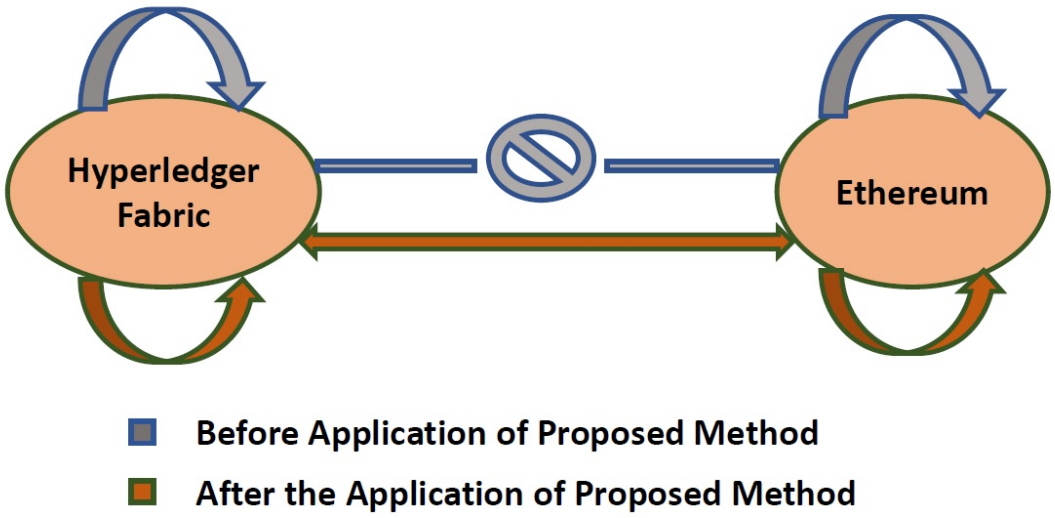

Figure 4.7: An illustration as to the comparison of interoperabiliy before and after applying the proposed solution.

an overview of public connectors, including notary methods, sidechains, and hash-time locking mechanisms. Conversely, other studies concentrated on public connections, with a particular focus on sidechains and hash lock time contracts (Abebe et al. (2019); Barber et al. (2012)). Meanwhile, Qasse et al. (2019a) arranged solutions across sidechains, blockchain routers, smart contracts, and industrial solutions. Siris et al. (2019) and, Kannengiesser et al. (2020) did a survey on interoperability problems, and

Johnson et al. (2019) and Koens and Poll (2019) concentrated on Ethereum as the framework that allows for interoperability across various types of applications. When looking at these literatures, it is apparent that they mainly focused on public blockchains, particularly cryptocurrencies. This study, on the other hand, is primarily focused on the interoperability of permissioned and permission-less blockchain blockchain networks. The following is a list of the observations made during this work.

Observations

- Hyperledger fabric has been designed to interact with Ethereum smart contract.
- Interaction is achieved through EVM Chaincode and fabric vm.
- Ethereum vm chaincode wraps the Hyperledger Fabric in a GO chaincode, together named Fabreum.
- The Ethereum chaincode acts as the smart contract runtime and stores the deployed contract on the ledger.
- Combining both Fabric and Ethereum will act as twins so that features can be incorporated.
- Obtained 100 percent success rate in 500 transactions with better latency and throughput.

4.5 SUMMARY

Blockchain innovations have developed quickly in the current decade. The involvement of Blockchain in lifestyles is not so far off. With the expanding reception of blockchain innovation, the quantity of clients has consistently expanded. However, its performance still needs much improvement compared with the mainstream processors. The system blockage of the existing framework is a common issue, and experts are cautiously considering how to settle down the interoperability issue. The proposed solution directive put forward a new framework for solving the interoperability issue by incorporating EVM Chaincode and fabric vm. This chapter included applying the solution in the Hyperledger Fabric and Ethereum framework and analyzing the resultant system in terms

of throughput, latency, and successful rates of transaction. The proposed method exhibits better throughput and latency for all cases of source and destination combos for all 500 transactions.

CHAPTER 5

TRANSITION- DIFFICULTY AMONG CONSENSUS

5.1 INTRODUCTION

Transaction systems in today's world must be decentralized, transparent, and incorruptible. Instantaneous transactions and borderless ownership transfers are possible with digital money. Take bitcoin, for example, which disseminates the business of producing cash over the Internet. It employs computer algorithms to verify that funds are transferred safely from buyer to sale. Bitcoin's underlying technology, blockchain, provides transaction transparency and decentralized verification. A network of computers uses Bitcoin to maintain the collective public database (Nakamoto (2009)). When Bitcoin is uploaded to the blockchain, all information about the transaction is locked. The transactions are verified and validated by bitcoin miners. If someone tries to tamper with the transaction, the node refuses to continue on the blockchain. On the user's node, a digital wallet is generated for each user. Each wallet has a unique address, which serves as a network node's effective identification. A blockchain is a database that records all network transactions (Dinh et al. (2018)). The validated transactions are uploaded to the blockchain as beads in a chain (Kamilaris (2018)). Each transaction is signed with confidential data called the private key, kept in the Bitcoin wallet. Every transaction is broadcast to all users. The network usually confirms it within the first 10 minutes, which is referred to as mining. In another way, mining is a distributed consensus method that is used to verify transactions that are awaiting inclusion on the blockchain.

The challenge of preserving the canonical blockchain state throughout the P2P network

may be translated as a fault-tolerant state-machine replication problem in the context of distributed systems. In other words, each consensus node keeps a local copy of the blockchain (i.e., a view) (Ben Hamida et al. (2017)). In the event of Byzantine/arbitrary failures, the consensus nodes are supposed to reach an agreement (i.e., consensus) on the unique shared view of the blockchain. Byzantine failures in blockchain networks lead defective nodes to behave in unpredictable ways, such as malicious assaults/collusions (e.g., Sybil attacks and double-spending attacks), node mistakes (e.g., unexpected blockchain fork owing to software incompatibility), and connection problems. A blockchain state transition occurs when a transaction is confirmed, and the sequence of blocks reflects the blockchain state (Cachin and Vukolić (2017)). According to the Byzantine setting, a blockchain update protocol achieves consensus if it satisfies the following characteristics. Authenticity (Correctness): Any honest node transitioning to a new local replica state accepts the blockchain led by that block if all the honest nodes active on the same state propose to enlarge the blockchain by the same block. Agreement: If an honest node verifies a new block header, the legitimate node updates it by updating its local blockchain view. Liveness (Termination): Verification of all transactions that will occur at some point initiated by honest nodes. Total order: Local blockchains will verify all the transactions, and all honest nodes accept the same order of transactions Permissionless blockchain networks, on the other hand, do not allow for identity identification or explicit synchronization methods; as a result, the consensus process should be scalable and tolerant to pseudo identities and synchronization issues. The fundamental aim of the consensus protocol in permissionless networks is to ensure that every consensus node follows the ”longest chain rule” by proposing the state transition with its candidate block for the blockchain header. The blocks are arranged as a linked list; only the longest chain may be recognized as the canonical state of the blockchain at any one time. A blockchain system may be considered a traditional distributed system that uses globally dispersed multiple types of networks to transfer data.

5.2 CONSENSUS

Proof of Work: Satoshi Nakamoto, the founder of Bitcoin, created POW, the earliest and most well-known consensus method. In POW, the miner who discovers the hash first is permitted to add a new block to the blockchain containing the transaction. Because mining is a computationally demanding operation, having a high hash rate is essential for miners to compute the hash and get the rewards. The primary advantages are the defense against DoS attacks and the low impact of stake on mining options. PoW imposes some restrictions on network activity. They will need a significant amount of effort to complete. A successful assault requires a considerable amount of processing power and a significant amount of time to complete the computations. As a result, the attack is viable but somewhat pointless due to the enormous expenses. It makes no difference how much cash you have in your wallet. It's essential to have a lot of computing power to solve the riddles and create new blocks. As a result, the owners of vast sums of money do not command the entire network's actions. Mining needs highly specialized computer gear to perform the complex algorithms. The expenses are uncontrollable. Mining is increasingly restricted to certain mining pools. These technical equipment use a lot of energy to run, which raises costs. Large fees pose a challenge to the system's centralization, which has several advantages. Miners put forth a lot of effort and use a lot of energy to create blocks. Their computations, however, are not relevant anyplace else. They provide network security, but they can't be used in business, science, or other sectors. Another issue with PoW is the 51% assault rate. A majority attack, also known as a 51 percent attack, occurs when one person or a group of users controls most mining power. The attackers gain enough ability to control the majority of network activities. Because they can prevent other miners from finishing blocks, they can dominate the generation of new blocks and earn incentives. They can reverse transactions.

Practical Byzantine Fault Tolerance (PBFT): Miguel Castro and Barbara Liskov created Practical Byzantine Fault Tolerance (PBFT) in 1999 at the MIT Laboratory for Computer Science (Hill (1990)). One of the proposed answers to the Byzantine Generals' Problem, a classic distributed system issue, is PBFT. PBFT aims to determine whether or not a piece of information contributed to the blockchain should be accepted. Each

party (the ”general”) keeps track of its internal condition. When a party gets a message, it combines it with its internal state to do a calculation. This calculation will influence this party's messaging decision. The choice will then be shared with all other participants in the network. The ultimate conclusion is based on the sum of all decisions made by all parties. PBFT, like the conventional Byzantine Generals' Problem, may accept 1/3 node treachery. Because PBFT depends on the number of nodes to validate trust, a significant hash rate is unnecessary in this procedure.

Proof of Stake: POW necessitates a large amount of energy. Unlike POW, POS is dependent on the coin stake of the players. The shareholder with the most coins will be more likely to upload a new transaction block to the blockchain. In POS, there is no block reward. The POS method is well suited for systems with static coin supply due to its reduced energy usage compared to POW (Szabo (2018)). Tokens are provided to validating nodes in the network from the beginning of the network's existence under the PoS method, which implies that tokens are not generated concurrently as new blocks are added to the ledger. Every few seconds or minutes, a particular node is chosen to commit the new block. However, a node with more coins has more control over what is deemed the truth on the ledger. As a result, people with the most currencies have a significant effect on the selection. PoS usually needs far less computing effort; therefore, the cost of implementing PoS is much cheaper. The ”nothing-at-stake” problem is one of the most often mentioned issues with PoS. On PoW blockchains, there is a financial incentive to keep mining the longest chain on the ledger because it will be regarded as the primary version of the truth. As a result, the miners are motivated to mine that one chain. However, with PoS, there is little to stop a miner from mining on several PoS chains simultaneously since the cost of mining is exceptionally cheap. As a result, a PoS miner running on several chains might make it impossible for the network to establish agreement, while a bad actor attempts to alter the past. Various PoS variants work around the drawbacks of PoS. Validators are randomly allocated the right to propose blocks in BFT-style PoS. At the end of the procedure, all validators agree that any given block is a chain member. The length or size of the chain has no bearing on a block's consensus.

Delegated Proof of Stake (DPoS): A variant of PoS is DPoS. With DPoS, currency holders may use their balance to vote for a list of nodes that will be permitted to add new transaction blocks to the blockchain potentially. Coin holders can also vote on changes to the network parameter. DPoS offers all coin holders more power and ownership in the network, whereas PoS is more like winning a lottery (Shala et al. (2019)) . Those with more money or tokens will have more influence on the network than those with less. Token holders in DPoS don't vote on the validity of the blocks directly; instead, they vote to elect delegates to validate the blocks on their behalf. The representatives are swapped regularly and given instructions on how to deliver their blocks. With fewer delegates, it is easier for them to arrange themselves and set aside time for each representative to publish their block. Stakers can vote delegates out and replace them with a better representative if they consistently miss their blocks or post incorrect transactions. Instead of competing as in PoW and PoS, miners in DPoS can work together to create blocks. DPoS is orders of magnitude quicker than most other consensus methods since it partially centralizes block generation.

Proof of Elapsed Time (PoET): Hyperledger Sawtooth, a modular blockchain technology created by Intel, supports PoET (Ongaro and Ousterhout (2014)) . It may be used on both private and public platforms. It allows users on a permissioned blockchain to reach an agreement even if they don't know each other, whereas most permissioned blockchains need users to know and trust one another. PoET is equivalent to PoW; however, it does not consume as many resources. Simply described, it uses trusted computing to ensure random block building wait times. Each member of the blockchain network is given an arbitrary length of time to wait. The first person to finish waiting becomes the new block's leader. Two conditions must be met for this to operate.The lottery winner should select a random wait time rather than purposefully choosing a short one. The lottery winner must then wait for the stipulated length of time to expire. PoET was developed by Intel and is based on a unique set of CPU instructions known as Intel Software Guard Extensions (SGX). Applications can use SGX to run trusted code in a secure environment. In the case of PoET, the trusted code guarantees that the two conditions are met for the lottery to remain fair.

Proof of Authority (PoA): PoA is a consensus process in which transactions are authenticated by authorized accounts, which function as the system's "admins." PoA is a modified version of PoS in which a validator's identity serves as the stake instead of a monetary value (Baliga et al. (2018)). Validators, or authorized accounts, validate transactions and blocks in PoA-based networks. Validators use software to organize transactions into blocks. The procedure is automated, so validators don't have to keep an eye on their computers all the time. However, it does need to keep the system secure. Individuals that acquire the privilege to become validators through PoA have a motivation to hold the position they have earned. Validators are encouraged to preserve the transaction process by attaching a reputation to their identities since they do not want their identities to be associated with a poor reputation. Because the incentives in PoS might be imbalanced, this is regarded more resilient than PoS.
Table 5.1 shows the comparison of various consensus mechanisms.

5.3 PROPOSED METHODOLOGY

The tests were conducted considering three groups. Each organization has one peer, one CA client, and one MSP, resulting in N organizations in a network of N peers. A single channel connects all of the organizations. The experiments' chaincode is a key-value store with methods for accessing the ledger and committing transactions, built-in Golang. For 3078 transactions, the tests used Ethereum, Hyperledger Fabric, Bitcoin, WAXP, TRX, TLOX, XLM, HBAR, EOH, MHC, HIVE, and LUNA as alternative application platforms. We've experimented with altering the application's requirements as well.

5.3.1 Data Preparation

The information utilized in the research came from Blocktivity.info, one of the most prominent blockchain browsers. It contains information on all network transactions, mined blocks, and user accounts. Various blockchain platforms have been reported. It was possible to get information about all transactions in which a specific wallet was involved using the API. Blocktivity aims to give the rawest data from as many blockchains as possible. It provides a means to understand better the projects accessible in the area

Table 5.1: Comparison of Consensus Algorithms

	PoW	**PoS**	**DPoS**	**PoA**	**PBFT**	**Paxos**
Type of Blockchain	Permission-less	Permissioned and permission-less	Permissioned and permission-less	Permissioned and permission-less	Permissioned	Permissioned
Election of Miners	Solving Difficulty hash	Stake owned	Stake owned	Solving Difficulty hash	Mathematical Operation	Proposal Number
Model of Trust	Un-trusted	Un-trusted	NA	NA	Semi-trusted	Semi-trusted
Transaction Finality	Probabilistic	Probabilistic	NA	NA	Immediate	Immediate
Decentralization Structure	Strong	Strong	Strong	Strong	Weak	Weak
Properties of Distributed Consensus	Probabilistic	Probabilistic	Probabilistic	Probabilistic	Deterministic	Deterministic
Reward	Yes	Yes	Yes	Yes	No	No
Control of Acceptance of the Adversary	<25% Computing Power	<51% Stack	<51% Validators	50% of Online Stake	¡33.3% Replicas	NA
Fees of Transaction	Yes for All Miners	Yes for All Miners	Yes for All Miners	Yes for Miner and Lucky Stakeholders	No	No
Speed of Verification(Per Sec)	>100 sec	<100 sec	<100 sec	NA	<10 sec	NA
Speed of Block Creation	Low	High	High	High	High	High
Throughput (Transaction per sec)	<100	<1000	<1000	NA	<2000	NA

by giving object data and tools to analyze and compare it. This dataset includes at least the top 100 most valuable blockchains by market capitalization, as well as any other blockchains that choose to be included. Table 5.2 shows the explanatory variables. The experiment uses 221 520 samples as a training set and 70 439 samples as a validation set to partition the dataset. The binary classification problem formulation given here is a typical example of a prediction issue. To assess the ability to make correct predictions for a particular dataset, we looked at the following classifiers: Decision Tree, Support Vector Machines, Naive Bayes, KNN, and Multi-layer Perceptron. Figure 5.1 shows

Table 5.2: Required Features of Dataset

Variable name	**Variable description**
Act 24	Activity 24h
TXN 24	Transaction 24h
Record	Record
Protocol	Protocol
Btime	Block Time
P-activity	Rate of activity with respect to bitcoin
Potential	Potential
Act 7	Activity 7d
VM	Virtual Mining
SLE	Simulating Leader Election
RLC	Rule of Long Chain
DBProp	Decoupling Block Proposal
Resource	Resourse Conception

the data and system architecture for the experiment. We began by downloading data, then aggregated it to produce the 13 variables shown in Table 4.1. The next stage was to use grid search with 10-fold cross-validation to select a set of parameters that would create the best results for the supervised learning algorithms we picked. Before entering the chain code, classify the consensus strategies. Run the chain code based on the resultant class. A consensus method is a fault-tolerant technique used in computer and blockchain systems to establish the required agreement among distributed processes on a single data value or network state. These publicly shared ledgers need an efficient, fair, real-time, functional, reliable, and secure method to verify that all transactions on the network are authentic and that all participants agree on a consensus on the ledger's status in a constantly changing state of the blockchain. The consensus mechanism, which is a system of rules that decides on the contributions of the many blockchain participants, handles this crucial duty. If a transition is required, it is impossible to do so among the consensus. If a consensus transition occurs, the entire network will be disrupted, and the whole system will have to be rebuilt to continue with the new agreement. The majority of the time, machine learning models are employed to make predictions. A good prediction model aids in making the best decisions and analyz-

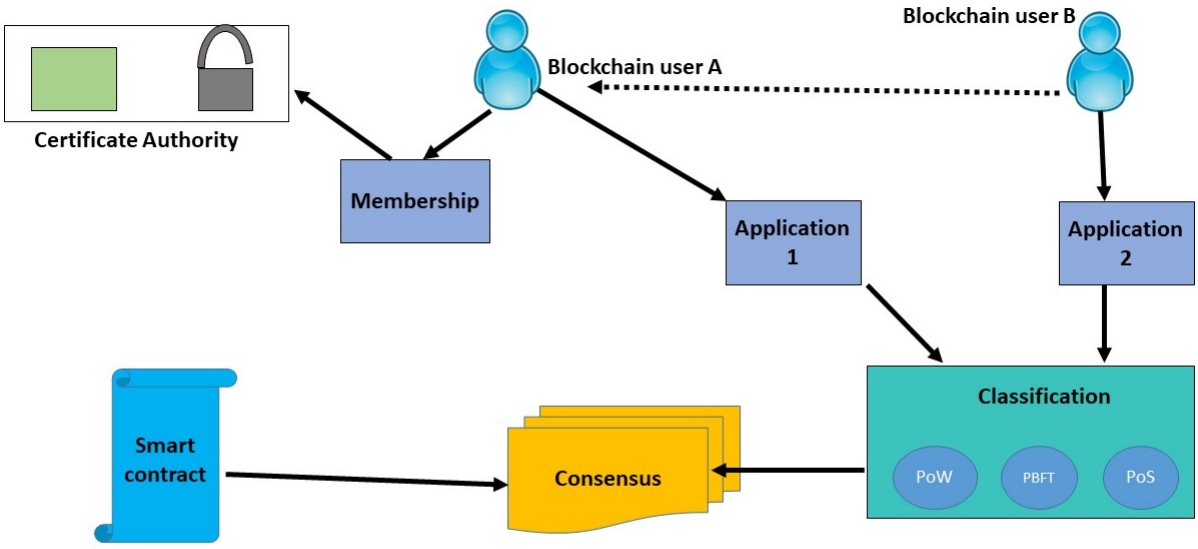

Figure 5.1: Design for Proposed Methodology

ing data. Valenkar et al. have also presented a machine learning approach to forecast bitcoin values. The experiment used Decision Trees, Support Vector Machines, Naive Bayes, KNN, and Multi-layer Perceptron. The MLP algorithm is used in this model, which considers block size, total bitcoins, day high, number of transactions, and trading volume. Log, z-score, and box-cox normalization methods were used to normalize the training dataset. A protocol prediction research was also conducted for many cryptocurrencies, including Ripple, Litecoin, Dash, Bitcoin, and Ethereum. The overall trends throughout the network were presented using correlation matrices for feature selection. Because it had greater accuracy, recall, and F1-score than previous algorithms, the suggested model employed multiple regression approaches on various blockchain platforms. The classification block's output will be a suitable consensus mechanism for the features of the entering application. The smart contract will then be executed on the blockchain using the proposed consensus method, as indicated in the algorithm 1.

Algorithm 5.1: Consensus Transition Algorithm

Input: The previous block hash, the current block number, timestamp, type of blockchain, election of miners, reward, speed of block creation

Output: Type of container, blockchain platform, used protocol, memory, CPU, traffic-in, traffic-out

Method :

Initialize all weights with small random numbers, typically between -1 and 1

repeat

for every pattern in the training set

Present the pattern to the network

for each layer in the network

for every node in the layer

1. Calculate the weight sum of the inputs to the node

2. Add the threshold to the sum

3. Calculate the activation for the node

end

end

for every node in the output layer

calculate the error signal

end

for all hidden layers

for every node in the layer

1. Calculate the node's signal error

2. Update each node's weight in the network

end

end

Calculate the Error Function

end

```
1 while ((maximum number of iterations < specified) AND (Error Function is >
    specified))
2 Get the appropriate consensus Protocol
3 Propose()
4 While true do
5 blockprotocol
6 If solve cryptopuzzle
7 Add block to the blockchain
8 Broadcast the chain updation
9 end
```

5.3.2 Classification Model

Support Vector Machine (SVM): In a high-dimensional space, the classifier is a binary classifier method that seeks an optimum hyperplane as a decision function (Demidova et al. (2016)). Support vector machines are algorithms that create a hyperplane or a group of hyperplanes in a high-dimensional space. SVMs may be used for a variety of tasks, including classification, regression, and other applications. Intuitively, any hyperplane provides no misclassification on all data points. Any of the considered classes achieve a separation between two linearly separable classes; that is, all points belonging to class A are labeled as +1, for example, and all points belonging to class B are marked as -1. SVM is a method for determining the optimal separation hyperplane. This technique generally ensures that the bigger the margin, the smaller the classifier's generalization error.

Decision tree: A decision tree is a tree-based approach in which a data separating sequence defines the path from the root to the leaf node until a Boolean conclusion is obtained (Safavian and Landgrebe (1991)). It is a hierarchical example of knowledge relationships that includes nodes and links. Nodes indicate purposes when relations are used to categorize. Systems that build classifiers are one of the most commonly utilized approaches in data mining. Classification algorithms are capable of managing a large amount of data in data mining. It may be used to create assumptions about category class names, categorize knowledge based on training sets and class labels, and categorize newly available data.

K-Nearest-Neighbor (KNN): The goal of the KNN model is to forecast the target class

label as the one that appears most frequently among the k most comparable training samples for a particular query point (Liao and Vemuri (2002)). The class label may be thought of as the ”mode” of the k training labels or as the result of ”plurality voting.” It's worth noting that KNN categorization is sometimes referred to as ”majority voting” in the literature. While the writers usually intend well, the phrase ”majority voting” is a misnomer because it usually refers to a 50 percent reference number for making a choice. There is always a majority or a tie in binary predictions (classification issues with two classes). As a result, a majority vote is also a plurality vote. In multi-class scenarios, however, we don't need a majority to generate a KNN prediction.

Naive Bayes: A classification method based on the Bayes rule and a set of conditional independence assumptions is known as the Naive Bayes algorithm (Mukherjee and Sharma (2012)). The Naive Bayes method assumes that each Xi is conditionally independent of each of the other Xks given Y, as well as independent of each subset of the other Xks given Y, to learn P(Y—X) where X = [X1..., Xn]. This assumption is helpful since it dramatically simplifies the representation of P(X—Y) and predicting it from training data.

Multi-layer Perceptron (MLP): MLP is a supervised learning technique that uses a dataset to train a function. It can learn a non-linear function approximator for either classification or regression given a collection of features and a goal. It differs from logistic regression in that one or more non-linear layers, referred to as hidden layers, might exist between the input and output layers (Gazzah and Essoukri Ben Amara (2006)). There can be more than one linear layer in a multi-layer perceptron (combinations of neurons). If we consider a three-layer network, the first layer is the input layer, the last layer is the output layer, and the intermediate layer is the hidden layer. The input layer receives our data, and the output layer gets the output. We may make the model as complicated as we wish by increasing the number of hidden layers. MLP uses backpropagation to train. It trains using some kind of gradient descent, with backpropagation used to determine the gradients. It minimizes the Cross-Entropy loss function for classification. Each input vector is connected with a label, or ground truth, specifying its class or class label in a supervised classification system. For each input, the

network's output is a class score or prediction. The loss function is used to evaluate the classifier's performance. If the projected class does not match the actual class, the loss will be considerable; otherwise, it will be minimal (Windeatt (2006)). When the model is being trained, the problem of overfitting and underfitting might emerge. Our algorithm performs admirably on training data but not on testing data in this situation. An optimization method is necessary to train the network, for which a loss function and an optimizer are required. The values for the set of weights, W, that minimizes the loss function will be found using this technique.

5.4 RESULT AND ANALYSIS

Our goal was to create a model that might be utilized as a real-world Blockchain consensus transition mechanism. We tested several categorization models, including Decision Trees, Support Vector Machines, Naive Bayes, KNN, and Multi-layer Perceptron, to see how well they could make correct predictions for a particular dataset. We opted to focus our evaluation of a specific method on recall and accuracy statistics due to the significant class imbalance. We got results with either high recall and low precision or low recall and high precision for various parameter combinations. On the other hand, it is meaningless in real-world applications where all transitions must be manually evaluated and restarted from the beginning by a person. We had a distribution in which the chance of achieving good consensus for a specific transaction was considerably more significant than in the actual world since we included virtually all consensus methods. Figure 5.2 shows that SVM, Naive Bayes, KNN, and Multi-layer Perceptron had 90 percent accuracy over the dataset, but the decision tree had only 50 percent accuracy. As a result, we could cross the decision tree off the list right away. Because all of the other classifiers had the same accuracy, we looked at their precision, recall, and F1 score to better understand. It can be seen from the findings in figures 5.3, 5.4, 5.5, 5.6, 5.7 that the MLP algorithm has a clear advantage in predicting the correct consensus on a dataset. Figure 5.8 depicts the decline in the loss value curve. On the blocktivity dataset with a reasonably large size, we can see that our suggested technique converges within 300 iterations, demonstrating its efficiency in the training step.

Figure 5.9 depicts the performance matrix generated using the caliper. The 3078

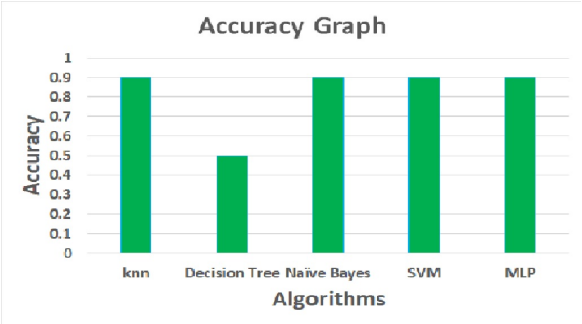

Figure 5.2: Accuracy graph

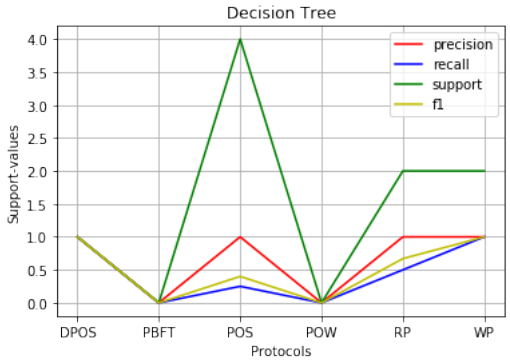

Figure 5.3: Decision Tree Algorithm

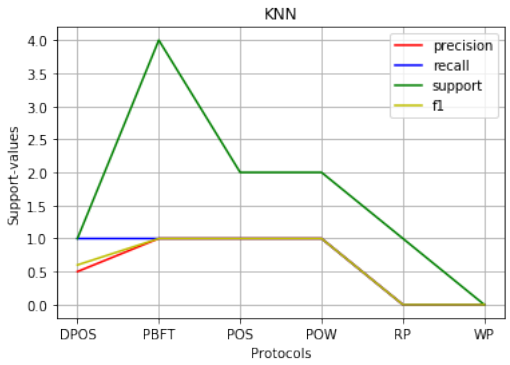

Figure 5.4: K-Nearest Neighbor Algorithm

transactions were carried out using Ethereum, Hyperledger Fabric, Bitcoin, WAXP, TRX, TLOX, XLM, HBAR, EOH, MHC, HIVE, and LUNA as platforms. The experiment was conducted by determining if the consensus was maintained even after the application's criteria were changed. It implies that, unlike in the past, a change in an application's needs does not disrupt the blockchain's operation. It will continue from where it left off. All of the transactions in each instance were completed successfully, and each case had a reasonable latency and throughput. This experiment yielded a trans-

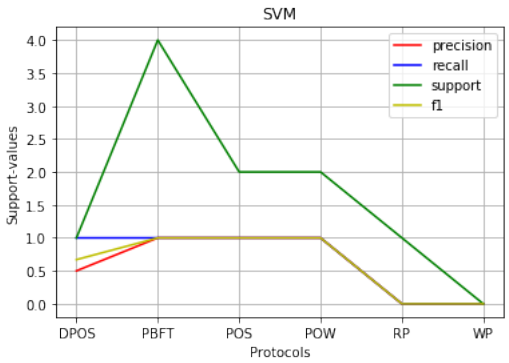

Figure 5.5: Support Vector Machine Algorithm

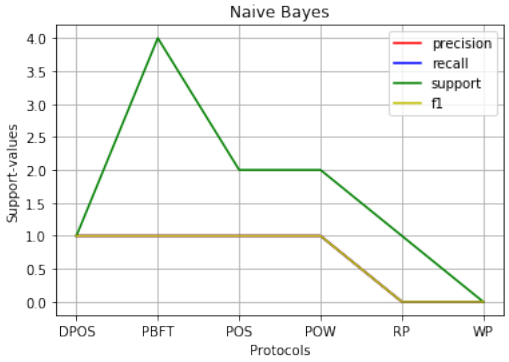

Figure 5.6: Naive Bayes Algorithm

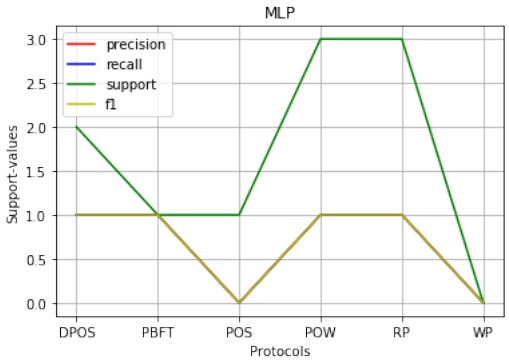

Figure 5.7: Multi-Layer Perceptron Algorithm

mit rate of 29.2 tps on average. The number of transactions sent per second is known as the send rate.

5.5 SUMMARY

Blockchain offers a peer-to-peer system in which dispersed nodes jointly confirm transaction provenance as a means to decentralize services, security, and verifiability . Blockchain,

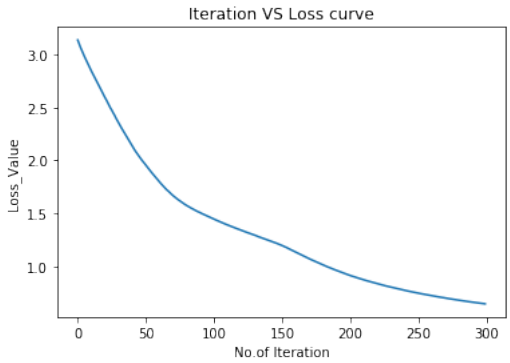

Figure 5.8: Iteration-Loss Graph

performance metrics

Name	Succ	Fail	Send Rate	Max Latency	Min Latency	Avg Latency	Throughput
init	3000	78	29.2 tps	16.37 s	0.94 s	7.93 s	13.1 tps

resource consumption

TYPE	NAME	Platform	Protocol	Memory(max)	Memory(avg)	CPU(max)	CPU(avg)	Traffic In	Traffic Out	Disc Read	Disc Write
Process	node fabricClientWorker.j...(avg)	Hyperledger fabric	PBFT	150.7MB	150.1MB	92.25%	9.21%	-	-	-	-
Docker	dev-peer0.org1.example.co...nk-v0	Ethereum	PoS	6.8MB	6.8MB	0.01%	0.00%	989B	240B	0B	0B
Docker	dev-peer0.org2.example.co...nk-v0	Bitcoin	PoW	6.9MB	6.9MB	0.00%	0.00%	744B	198B	0B	0B
Docker	dev-peer0.org1.example.co...le-v0	WAXP	PBFT	6.8MB	6.8MB	0.01%	0.00%	555B	282B	0B	0B
Docker	dev-peer0.org2.example.co...le-v0	TRX	DPoS 2.0	6.9MB	6.9MB	0.00%	0.00%	555B	282B	0B	0B
Docker	dev-peer0.org2.example.co...rm-v0	TLoS	DPoS	6.8MB	6.8MB	0.00%	0.00%	198B	198B	0B	0B
Docker	dev-peer0.org1.example.co...rm-v0	XLM	PoS	6.7MB	6.7MB	0.00%	0.00%	240B	240B	0B	0B
Docker	dev-peer0.org2.example.co...es-v0	HBAR	PoS	7.3MB	7.2MB	3.82%	1.64%	1.1MB	643.3KB	0B	0B
Docker	dev-peer0.org1.example.co...es-v0	EoS	DPoS 2.0	7.3MB	7.1MB	4.26%	1.78%	1.1MB	671.7KB	0B	0B
Docker	peer0.org2.example.com	MHC	MDPoS	205.8MB	204.3MB	15.72%	9.37%	3.4MB	2.0MB	0B	7.5MB
Docker	peer0.org1.example.com	HIVE	DPoS	206.4MB	204.9MB	15.39%	9.52%	3.4MB	2.7MB	0B	7.5MB
Docker	ca.org2.example.com	Ethereum	PoW	8.8MB	8.8MB	0.00%	0.00%	0B	0B	0B	0B
Docker	ca.org1.example.com	LUNA	PoS	8.9MB	8.9MB	0.00%	0.00%	0B	0B	0B	0B

Figure 5.9: Results from Caliper for the experiments done using Ethereum, Hyperledger fabric, Bitcon, WAXP,TRX, TLOX, XLM, HBAR, EOH, MHC , HIVE and LUNA as application platforms

in particular, mandates the ongoing preservation of transaction history, which is protected by digital signatures and confirmed by agreement. Blockchain technology is likely to substantially influence a wide range of sectors, not the only cryptocurrency. The industrial application will run on a permissioned blockchain rather than a permissionless blockchain. A permissioned blockchain is a secure distributed ledger maintained by a group of trusted and certified nodes. Current blockchain systems, particularly permissioned blockchains, have limitations that discourage companies from using the technology . One of the critical challenges is the transition difficulty among consen-

sus. Improving the functional qualities of blockchains, such as consistency and secrecy, and its non-functional properties, such as performance and scalability, requires overcoming these restrictions. This chapter suggests a method that uses categorization algorithms to overcome the challenge of achieving consensus across multiple platforms. The studies were carried out using Ethereum, Hyperledger Fabric, Bitcoin, WAXP, TRX, TLOX, XLM, HBAR, EOH, MHC, HIVE, and LUNA as multiple application platforms for 3078 transactions, and the performance was assessed by altering the application requirements. The suggested method employs a Multi-Layer Perceptron classifier to achieve consensus with more accuracy.

CHAPTER 6

CONCLUSIONS AND FUTURE SCOPE

In this dissertation, we investigate permissioned blockchain from a performance standpoint. Blockchain technology has advanced significantly in recent years, and it will be used to additional applications in a variety of areas in the not-too-distant future. The number of users has constantly risen as blockchain technology has been more widely adopted. Permissioned blockchains' scalability is more attainable, as they trade trust freedom for scalability. Permissionless blockchains' scalability necessitates unique concepts, and recent research appears to be quite promising from a practical standpoint.

A thorough literature study is carried on the scalability , interoperability and transition difficulty among consensus and observed clear research gaps.This research work focused on objectives in an account of these concerns.

Our key contribution is a model that captures the important features such as scalability, interoperability and transition of consensus. A full examination of model parameterization and validation is also provided. Blockchains have brought a new age of system development, and this thesis lays the groundwork for future modelling and analytic research in the field. Both system developers and architects implementing permissioned blockchain in the field will benefit from the models and analysis presented in this thesis. This study delivers intriguing insights by methodologically investigating each solution.

6.1 FUTURE SCOPE

As the techniques proposed in this thesis are performed better than existing models for permissioned blockchain, this area is still in its infant stage. Therefore, there is significant scope for future works. Further research directions exist in this area are:

- Develop a system which can handle unlimited users and achieve great scalability
- Develop a blockchain model which could handle resource consumption
- The proposed model could perform interoperability among ethereum and hyperledger fabric. So there is an excellent research scope in making the entire blockchain platforms interoperable.

In conclusion, this dissertation proposes a new model for permissioned blockchain. Our findings lead us to believe that the requirements for blockchain research have been met, enabling for a slew of new applications. As a result, we anticipate a significant increase in interest in this field of study. This initiative aims to make the blockchain ecosystem more realistic for developers and academics by making their jobs easier. We anticipate that this study will serve as a solid and reliable foundation upon which developers and academics may build in the field of blockchain research.

BIBLIOGRAPHY

(2016). *Open access to the Proceedings of the 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI '16) is sponsored by USENIX. Bitcoin-NG: A Scalable Blockchain Protocol Bitcoin-NG: A Scalable Blockchain Protocol* *.

Abebe, E., Behl, D., Govindarajan, C., Hu, Y., Karunamoorthy, D., Novotny, P., Pandit, V., Ramakrishna, V. and Vecchiola, C. (2019). "Enabling enterprise blockchain interoperability with trusted data transfer (industry track)." .

Alam, T. (2019). "Blockchain and its Role in the Internet of Things (IoT)." *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 151–157.

Albert Callarisa Roca, B. D. (2017). "Herdius: A next-generation decentralized blockchain financial infrastructure." Accessed: 2018-08-22.

Alsunaidi, S. J. and Alhaidari, F. A. (2019). "A survey of consensus algorithms for blockchain technology." *2019 International Conference on Computer and Information Sciences (ICCIS)*, 1–6.

Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Singh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., Vukolic, M. and Yellick, J. (2018). "Hyperledger fabric: A distributed operating system for permissioned blockchains." .

Apte, S. and Petrovsky, N. (2016). "Will blockchain technology revolutionize excipient supply chain management?." *Journal of Excipients and Food Chemicals*, 7(3), 76–78.

Back, A., Corallo, M., Dashjr, L., Friedenbach, M., Maxwell, G., Miller, A. K., Poelstra, A., Timón, J. and Wuille, P. (2014). "Enabling blockchain innovations with pegged sidechains." .

Bagui, S. and Nguyen, L. (2015). "Database sharding:: To provide fault tolerance and scalability of big data on the cloud." *International Journal of Cloud Applications and Computing*, 5, 36–52.

Baliga, A., Solanki, N., Verekar, S., Pednekar, A., Kamat, P. and Chatterjee, S. (2018). "Performance characterization of hyperledger fabric." 65–74.

Barber, S., Boyen, X., Shi, E. and Uzun, E. (2012). "Bitter to better — how to make bitcoin a better currency." *Advances in Water Resources - ADV WATER RESOUR*, 7397.

Belchior, R., Vasconcelos, A., Guerreiro, S. and Correia, M. (2021a). "A survey on blockchain interoperability: Past, present, and future trends." *ACM Comput. Surv.*, 54(8).

Belchior, R., Vasconcelos, A., Guerreiro, S. and Correia, M. (2021b). "A survey on blockchain interoperability: Past, present, and future trends." .

Ben, E., Brousmiche, K.-L., Levard, H. and Thea, E. (2017). "Blockchain for enterprise: Overview, opportunities and challenges." .

Ben Hamida, E., Leo Brousmiche, K., Levard, H. and Thea, E. (2017). "Blockchain for Enterprise: Overview, Opportunities and Challenges Blockchain for Smart Transactions-IRT SystemX View project Kei-Léo Brousmiche IRT System X Blockchain for Enterprise: Overview, Opportunities and Challenges." .

Bennink, P. (2018). "An analysis of atomic swaps on and between ethereum blockchains using smart contracts." Accessed: 2019-10-02.

Besancon, L., Ferreira da Silva, C. and Ghodous, P. (2019). "Towards blockchain interoperability: Improving video games data exchange." 81–85.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. and Felten, E. (2015). "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies." 2015, 104–121.

Brown, I. and Mues, C. (2012). "An experimental comparison of classification algorithms for imbalanced credit scoring data sets." *Expert Syst. Appl.*, 39, 3446–3453.

Buterin, V. (2014). "Ethereum: A next-generation smart contract and decentralized application platform." Accessed: 2016-08-22.

Buterin, V. (2015). "A next-generation smart contract and decentralized application platform." .

Cachin, C. and Vukolic, M. (2017). "Blockchain consensus protocols in the wild." *ArXiv*, abs/1707.01873.

Cachin, C. and Vukolić, M. (2017). "Blockchain Consensus Protocols in the Wild." .

Castro, M. and Liskov, B. (1999). "Practical byzantine fault tolerance." In *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, OSDI '99, USENIX Association, USA, 173–186.

Castro, M. and Liskov, B. (2002). "Practical byzantine fault tolerance and proactive recovery." *ACM Trans. Comput. Syst.*, 20, 398–461.

Centobelli, P., Cerchione, R., Esposito, E. and Oropallo, E. (2021a). "Surfing blockchain wave, or drowning? shaping the future of distributed ledgers and decentralized technologies." *Technological Forecasting and Social Change*, 165, 120463.

Centobelli, P., Cerchione, R., Vecchio, P. D., Oropallo, E. and Secundo, G. (2021b). "Blockchain technology for bridging trust, traceability and transparency in circular supply chain." *Information and Management*, 103508.

Chawla, N., Behrens, H., Tapp, D., Boscovic, D. and Candan, K. (2019). "Velocity: Scalability improvements in block propagation through rateless erasure coding." In *ICBC 2019 - IEEE International Conference on Blockchain and Cryptocurrency*,

ICBC 2019 - IEEE International Conference on Blockchain and Cryptocurrency, Institute of Electrical and Electronics Engineers Inc., 447–454. Publisher Copyright: © 2019 IEEE.; 1st IEEE International Conference on Blockchain and Cryptocurrency, ICBC 2019 ; Conference date: 14-05-2019 Through 17-05-2019.

Chung, G., Desrosiers, L., Gupta, M., Sutton, A., Venkatadri, K., Wong, O. and Zugic, G. (2019). "Performance tuning and scaling enterprise blockchain applications." .

Croman, K., Decker, C., Eyal, I., Gencer, A. E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., Sirer, E., Song, D. and Wattenhofer, R. (2016). "On scaling decentralized blockchains." volume 9604, 106–125.

Dagher, G. G. and Enderson, T. (2018). "Towards secure interoperability between heterogeneous blockchains using smart contracts." .

De Angelis, S., Aniello, L., Lombardi, F., Margheri, A. and Sassone, V. (2017). "Pbft vs proof-of-authority: applying the cap theorem to permissioned blockchain." .

Demidova, L., Nikulchev, E. and Sokolova, Y. (2016). "The svm classifier based on the modified particle swarm optimization." *International Journal of Advanced Computer Science and Applications*, 7(2).

Ding, D. (2018). "Interchain : A framework to support blockchain interoperability." .

Dinh, T., Liu, R., Zhang, M., Chen, G., Ooi, B. and Wang, J. (2017a). "Untangling blockchain: A data processing view of blockchain systems." *IEEE Transactions on Knowledge and Data Engineering*, PP.

Dinh, T., Liu, R., Zhang, M., Chen, G., Ooi, B. and Wang, J. (2017b). "Untangling blockchain: A data processing view of blockchain systems." *IEEE Transactions on Knowledge and Data Engineering*, PP.

Dinh, T., Wang, J., Chen, G., Liu, R., Ooi, B. and Tan, K.-L. (2017c). "Blockbench: A framework for analyzing private blockchains." 1085–1100.

Dinh, T. T. A., Liu, R., Zhang, M., Chen, G., Ooi, B. C. and Wang, J. (2018). "Untangling Blockchain: A Data Processing View of Blockchain Systems." *IEEE Transactions on Knowledge and Data Engineering*, 30(7), 1366–1385.

Duboc, L., Rosenblum, D. and Wicks, T. (2006). "A framework for modelling and analysis of software systems scalability." .

Eyal, I., Gencer, A. E., Sirer, E. G. and Van Renesse, R. (2016). "Bitcoin-ng: A scalable blockchain protocol." In *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation*, NSDI'16, USENIX Association, USA, 45–59.

Fan, S., Ghaemi, S., Khazaei, H. and Musilek, P. (2020). "Performance evaluation of blockchain systems: A systematic survey." *IEEE Access*, PP, 1–1.

Fernandez-Delgado, M., Cernadas, E., Barro, S. and Amorim, D. (2014). "Do we need hundreds of classifiers to solve real world classification problems?." *Journal of Machine Learning Research*, 15, 3133–3181.

Foundation, I. (2019). "Icon." .

Gatteschi, V., Lamberti, F., Demartini, C., Pranteda, C. and Santamaría, V. (2018). "Blockchain and smart contracts for insurance: Is the technology mature enough?." *Future Internet*, 10(2).

Gazzah, S. and Essoukri Ben Amara, N. (2006). "Writer identification using modular mlp classifier and genetic algorithm for optimal features selection." In Wang, J., Yi, Z., Zurada, J. M., Lu, B.-L. and Yin, H., editors, *Advances in Neural Networks - ISNN 2006*, Springer Berlin Heidelberg, Berlin, Heidelberg, 271–276.

Gorenflo, C., Lee, S., Golab, L. and Keshav, S. (2019). "Fastfabric: Scaling hyperledger fabric to 20,000 transactions per second." 455–463.

Hardjono, T., Lipton, A. and Pentland, A. (2019). "Toward an interoperability architecture for blockchain autonomous systems." *IEEE Transactions on Engineering Management*, PP, 1–12.

Hill, M. (1990). "What is scalability?." *ACM Sigarch Computer Architecture News*, 18, 18–21.

IBM Corporation (2019a). "hyperledger-fabricdocs Documentation." .

IBM Corporation (2019b). "hyperledger-sawtoothdocs Documentation." .

IBM India. "The difference between public and private blockchain-the difference between public and private blockchain." .

Johnson, S., Robinson, P. and Brainard, J. (2019). "Sidechains and interoperability." .

Jolma, A. and Rizzoli, A.-E. (2003). "A review of interoperability techniques for models, data, and knowledge in environmental software." .

Jones, S., Johnstone, D. and Wilson, R. (2015). "An empirical evaluation of the performance of binary classifiers in the prediction of credit ratings changes." *Journal of Banking and Finance*, 56, 72–85.

Kamilaris, A. (2018). "The Rise of the Blockchain Technology in Agriculture and Food Supply Chain." *ResearchGate*, (September).

Kan, L., Wei, Y., Hafiz Muhammad, A., Siyuan, W., Linchao, G. and Kai, H. (2018a). "A multiple blockchains architecture on inter-blockchain communication." 139–145.

Kan, L., Wei, Y., Muhammad, A. H., Siyuan, W., Linchao, G. and Kai, H. (2018b). "A multiple blockchains architecture on inter-blockchain communication." *2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 139–145.

Kannengiesser, N., Pfister, M., Greulich, M., Lins, S. and Sunyaev, A. (2020). "Bridges between islands: Cross-chain technology for distributed ledger technology." .

Karpenko, L., Izha, M., Onyshko, S., Chunytska, I. and Starodub, D. (2019). "Blockchain as an innovative technology in the strategic management of companies." *Academy of Strategic Management Journal*, 18(Special Issue 1), 1–6.

King, S. and Nadal, S. (2012). "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." .

Koens, T. and Poll, E. (2019). "Assessing interoperability solutions for distributed ledgers." *Pervasive and Mobile Computing*, 59, 101079.

Kokoris-Kogias, E., Jovanovic, P., Gasser, L., Gailly, N., Syta, E. and Ford, B. (2018). "Omniledger: A secure, scale-out, decentralized ledger via sharding." In *2018 IEEE Symposium on Security and Privacy (SP)*, 583–598.

Krstić, M. and Krstić, L. (2020). "Hyperledger frameworks with a special focus on hyperledger fabric." *Vojnotehnicki glasnik*, 68, 639–663.

Kshetri, N. and Voas, J. (2018). "Blockchain in developing countries." *IT Professional*, 20, 11–14.

Kuo, T.-T., eui Kim, H. and Ohno-Machado, L. (2017). "Blockchain distributed ledger technologies for biomedical and health care applications." *Journal of the American Medical Informatics Association : JAMIA*, 24, 1211 – 1220.

Kwon, J. and Buchman, E. (2016). "Cosmos: A network of distributed ledgers." *URL https://cosmos. network/whitepaper*.

Lamport, L., Shostak, R. and Pease, M. (2002). "The byzantine generals problem." *ACM Trans. Program. Lang. Syst.*, 4.

Lessmann, S., Baesens, B., Seow, H. V. and Thomas, L. C. (2015). "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research." *Eur. J. Oper. Res.*, 247, 124–136.

Li, W., Sforzin, A., Fedorov, S. and Karame, G. O. (2017). "Towards scalable and private industrial blockchains." In *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, BCC '17, Association for Computing Machinery, New York, NY, USA, 9–14.

Liao, Y. and Vemuri, V. (2002). "Use of k-nearest neighbor classifier for intrusion detection11an earlier version of this paper is to appear in the proceedings of the 11th usenix security symposium, san francisco, ca, august 2002." *Computers and Security*, 21(5), 439–448.

Lim, T. S., Loh, W.-Y. and Shih, Y.-S. (2000). "A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms." *Machine Learning*, 40, 203–228.

linux foundation project, T. (2017). "Hyperledger quilt." Accessed: 2019-09-22.

Lombrozo, E., Lau, J. and Wuille, P. (2015a). "Bip141: Segregated witness (consensus layer)." *GitHub, December*, 21.

Lombrozo, E., Lau, J. and Wuille, P. (2015b). "Segregated witness (consensus layer)." *Bitcoin Core Develop. Team, Tech. Rep. BIP*, 141.

Louie, T. (2017). "Wanchain." .

Luu, L., Narayanan, V., Zheng, C., Baweja, K., Gilbert, S. and Saxena, P. (2016). "A secure sharding protocol for open blockchains." In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, Association for Computing Machinery, New York, NY, USA, 17–30.

Macià, N. and Bernadó-Mansilla, E. (2014). "Towards uci+: A mindful repository design." *Information Sciences*, 261, 237–262.

McCorry, P., Möser, M., Shahandashti, S. and Hao, F. (2016). "Towards bitcoin payment networks." volume 9722, 57–76.

Miller, A. K. and LaViola, J. J. (2014). "Byzantine consensus from moderately-hard puzzles : A model for bitcoin." .

Mockapetris, P. and Dunlap, K. (2001). "Development of the domain name system." *ACM SIGCOMM Computer Communication Review*, 25.

Mukherjee, S. and Sharma, N. (2012). "Intrusion detection using naive bayes classifier with feature reduction." *Procedia Technology*, 4, 119–128. 2nd International Conference on Computer, Communication, Control and Information Technology(C3IT-2012) on February 25 - 26, 2012.

Nakamoto, S. (2009). "Bitcoin: A peer-to-peer electronic cash system." *Cryptography Mailing list at https://metzdowd.com*.

Nathan, S., Thakkar, P. and Vishwanathan, B. (2018). "Performance benchmarking and optimizing hyperledger fabric blockchain platform." .

Nguyen, Q. (2016). "Blockchain - a financial technology for future sustainable development." *2016 3rd International Conference on Green Technology and Sustainable Development (GTSD)*, 51–54.

Ongaro, D. and Ousterhout, J. (2014). "In search of an understandable consensus algorithm." .

overline (2017). "Block collider." Accessed: 2019-12-22.

Panda, S. S., Mohanta, B. K., Satapathy, U., Jena, D., Gountia, D. and Patra, T. K. (2019). "Study of blockchain based decentralized consensus algorithms." *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*, 908–913.

Pervez, H., Muneeb, M., Irfan, M. U. and Haq, I. U. (2018). "A comparative analysis of dag-based blockchain architectures." In *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, 27–34.

Pongnumkul, S., Siripanpornchana, C. and Thajchayapong, S. (2017). "Performance analysis of private blockchain platforms in varying workloads." 1–6.

Qasse, I., Abu Talib, M. and Nasir, Q. (2019a). "Inter blockchain communication: A survey." 1–6.

Qasse, I. A., Abu Talib, M. and Nasir, Q. (2019b). "Inter blockchain communication: A survey." In *Proceedings of the ArabWIC 6th Annual International Conference Re-*

search Track, ArabWIC 2019, Association for Computing Machinery, New York, NY, USA.

Qingyi, Z., Loke, S., Trujillo-Rasua, R., Jiang, F. and Xiang, Y. (2019). "Applications of distributed ledger technologies to the internet of things: A survey." *ACM Computing Surveys*, 52, 1–34.

Raynal, M. (2010). *Communication and Agreement Abstractions for Fault-Tolerant Asynchronous Distributed Systems*, volume 1.

Rohrer, E. and Tschorsch, F. (2019). "Kadcast: A structured approach to broadcast in blockchain networks." In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, AFT '19, Association for Computing Machinery, New York, NY, USA, 199–213.

Saad, M. and Mohaisen, A. (2018). "Towards characterizing blockchain-based cryptocurrencies for highly-accurate predictions." *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 704–709.

Safavian, S. and Landgrebe, D. (1991). "A survey of decision tree classifier methodology." *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3), 660–674.

Schäffer, M., Angelo, M. D. and Salzer, G. (2019). "Performance and scalability of private ethereum blockchains." .

Shala, B., Trick, U., Lehmann, A., Ghita, B. and Shiaeles, S. (2019). "Novel trust consensus protocol and blockchain-based trust evaluation system for m2m application services." *Internet of Things*, 7, 100058.

Shanahan, J. and Dai, L. (2015). "Large scale distributed data science using apache spark." 2323–2324.

Silvano, W. F. and Marcelino, R. (2020). "Iota tangle: A cryptocurrency to communicate internet-of-things data." *Future Generation Computer Systems*, 112, 307–319.

Siris, V. A., Nikander, P., Voulgaris, S., Fotiou, N., Lagutin, D. and Polyzos, G. C. (2019). "Interledger approaches." *IEEE Access*, 7, 89948–89966.

Sompolinsky, Y., Lewenberg, Y. and Zohar, A. (2016). "Spectre: A fast and scalable cryptocurrency protocol." `https://ia.cr/2016/1159`.

Sompolinsky, Y. and Zohar, A. (2015). "Secure high-rate transaction processing in bitcoin." In Böhme, R. and Okamoto, T., editors, *Financial Cryptography and Data Security*, Springer Berlin Heidelberg, Berlin, Heidelberg, 507–527.

Sompolinsky, Y. and Zohar, A. (2018). "Phantom: A scalable blockdag protocol." *IACR Cryptol. ePrint Arch.*, 2018, 104.

Sousa, J., Bessani, A. and Vukolic, M. (2018). "A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform." *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 51–58.

Spoke, M. (2019). "Aion." .

Sukhwani, H., Martínez, J. M., Chang, X., Trivedi, K. S. and Rindos, A. (2017). "Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric)." *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*, 253–255.

Sukhwani, H., Wang, N., Trivedi, K. S. and Rindos, A. (2018). "Performance modeling of hyperledger fabric (permissioned blockchain network)." 1–8.

Sun, F. and Duan, P. (2014). "Solving byzantine problems in synchronized systems using bitcoin." .

Swathi, P., Modi, C. and Patel, D. (2019). "Preventing sybil attack in blockchain using distributed behavior monitoring of miners." 1–6.

Szabo, N. (2018). "Smart contracts : Building blocks for digital markets." .

Sztorc, P. (2015). "Drivechain - the simple two way peg." Accessed: 2018-10-12.

Tech, Z. (2017). "Anlink blockchain network." Accessed: 2019-05-02.

Thomas, S. and Schwartz, E. (2016). "A protocol for interledger payments." .

T.Swanson (2015). "Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems." .

Uddin, M., Stranieri, A., Gondal, I. and Balasubramanian, V. (2021). "A survey on the adoption of blockchain in iot: Challenges and solutions." *Blockchain: Research and Applications*, 100006.

unknown (2020). "Mining." 51–54.

Velankar, S. K., Valecha, S. and Maji, S. (2018). "Bitcoin price prediction using machine learning." *2018 20th International Conference on Advanced Communication Technology (ICACT)*, 144–147.

Vernadat, F. (2007). "Interoperable enterprise systems: Principles, concepts, and methods." *Annual Reviews in Control*, 31, 137–145.

Vo, H., Wang, Z., Karunamoorthy, D., Wagner, J., Abebe, E. and Mohania, M. (2018). "Internet of blockchains: Techniques and challenges ahead." 1574–1581.

Wang, H., Cen, Y. and Li, X. (2017). "Blockchain router: A cross-chain communication protocol." 94–97.

Wang, H., Zheng, Z., Xie, S., Dai, H. N. and Chen, X. (2018). "Blockchain challenges and opportunities: a survey." *International Journal of Web and Grid Services*, 14(4), 352.

Windeatt, T. (2006). "Accuracy/diversity and ensemble mlp classifier design." *IEEE Transactions on Neural Networks*, 17(5), 1194–1211.

Wolpert, D. H. (1996). "The lack of a priori distinctions between learning algorithms." *Neural Comput.*, 8(7), 1341–1390.

Wood, C. (2017). "Ark." .

Wood, D. D. (2014). "Ethereum: A secure decentralised generalised transaction ledger." .

wood, G. (2016). "Polkadot: Vision for a heterogeneous multi-chain framework." .

Xu, Z., Han, S. and Chen, L. (2018). "Cub, a consensus unit-based storage scheme for blockchain system." *2018 IEEE 34th International Conference on Data Engineering (ICDE)*, 173–184.

Yapa, C., de Alwis, C. and Liyanage, M. (2021). "Can blockchain strengthen the energy internet?." *Network*, 1(2), 95–115.

Yasaweerasinghelage, R. M. R., Staples, M. and Weber, I. (2017). "Predicting latency of blockchain-based systems using architectural modelling and simulation." .

Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K. (2016). "Where is current research on blockchain technology?—a systematic review." *PLOS ONE*, 11.

Zamani, M., Movahedi, M. and Raykova, M. (2018). "Rapidchain: Scaling blockchain via full sharding." In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, Association for Computing Machinery, New York, NY, USA, 931–948.

Zheng, Z., Xie, S., Dai, H.-N., Chen, X. and Wang, H. (2017). "An overview of blockchain technology: Architecture, consensus, and future trends." .

Zhou, Q., Huang, H. and Zheng, Z. (2020). "Solutions to scalability of blockchain: A survey." *IEEE Access*, PP.

Zoican, S., Vochin, M., Zoican, R. and Galatchi, D. (2018). "Blockchain and consensus algorithms in internet of things." .

PUBLICATIONS

JOURNAL PAPERS

1. Swathi P, and M Venkatesan.” Scalability Improvement and Analysis of Permissioned Blockchain.” ICT EXPRESS JOURNAL (2021),ISSN 2405-9595, https://doi.org/10.1016/j.icte.2021.08.015.- (IMPACT FACTOR : 4.317)

2. Swathi P, and M Venkatesan.” Interoperability in Permissioned- Blockchain” , *SUSTAINABILITY: EDGE ARTIFICIAL INTELLIGENCE IN FUTURE SUSTAINABLE COMPUTING SYSTEMS (2021)*, https://doi.org/10.3390/su132011132- (IMPACT FACTOR : 3.251)

3. Swathi P, and M Venkatesan. (2021). Transition among consensus in blockchain, *Connection Science* (Paper submitted on September 5th, 2021 and status is “Under Review”)

BOOKCHAPTERS

1. Swathi P, and M Venkatesan: ’A deep dive into Hyperledger’ (Healthcare Technologies, 2020), ’Blockchain and Machine Learning for e-Healthcare Systems’, Chap. 4, pp. 85-107, DOI: 10.1049, https://digital-library.theiet.org/content/books/10.1049/pbhe029e_ch4

CONFERENCE PAPERS

1. Swathi P, and M Venkatesan.”A Survey on Blockchain Technology.” National conference on Blockchain and Smart Contract Technologies (BSCT-2019),NIT Trichy, 2019, PP.8-11.

2. Swathi P, and M Venkatesan.”Is Data Science and Blockchain a Perfect Match.”

6th International Conference on Information and Communication Technology for Competitive Strategies (ICTCS-2021)- ACCEPTED

BIODATA

Name:	Swathi P
Date of Birth:	2nd June 1993
Gender:	Female
Marital Status:	Married
Father's Name:	Dileep Kumar P
Mother's Name:	Jyothishkumari K
Address:	Achutham, Punathumkandi Olavanna (P.O) Calicut Kerala-673019
E-mail:	swathip2693@gmail.com
Mobile:	+91 8289956533
Qualification:	B.Tech in Information Technology and Engineering (Government Engineering College , Barton hill, Thiruvananthapuram, Kerala, India) M.Tech in Computer Science & Engineering – Blockchain (National Institute of Technology, Goa, India)
Areas of Interest:	Blockchain, Data Science