CONTENT CENTRIC NETWORKING BASED SECURE DECENTRALIZED DATA MANAGEMENT IN V2G ENVIRONMENT

A Thesis submitted in fulfillment of the requirement for the award of the degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Arzoo Miglani

(Registration No. : 901703016)

Under the guidance of:

Dr. Neeraj Kumar
Dean, Digital Content Transformation (DCT), Professor, CSED

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY,
PATIALA – 147004

May 2024

CERTIFICATE

I, Arzoo Miglani, Regn. No. 901703016, hereby declare that the thesis entitled **"Content Centric Networking based Secure Decentralized Data Management in V2G Environment"** submitted to the Computer Science and Engineering Department at Thapar Institute of Engineering & Technology, Patiala, Punjab, India is an authenticated record of my own work for the award of the degree of "Doctor of Philosophy" under the supervision of Prof. (Dr.) Neeraj Kumar. This report has not been submitted to any other institution for award of any other degree.

Arzoo Miglani
Regn. No. 901703016

Place: Patiala, Punjab (India)
Date: May 16, 2024

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Verified by:

Prof. (Dr.) Neeraj Kumar
(Supervisor)
Professor, CSED
Thapar Institute of Engg. & Technology
Patiala, Punjab (India)

ACKNOWLEDGMENTS

Before discussing my journey of Ph.D., I want to thank the almighty God who gave me the strength and courage to overcome all the obstacles and complete this endeavor. The aim of my life, to be called by the salutation of a 'Doctor', seemed to become a reality, when I got admission in the Doctorate of Philosophy in Thapar Institute of Engineering & Technology. Research initiated with this startup in my life. Without acknowledging the people who supported me throughout this journey, this task would be incomplete. I know words are never enough to express gratitude; I am just delivering the phrase for the acceptance of regard.

Firstly, I would like to express my sincerest thanks to my parents. With their consent, support, and motivation, I thought to accept this biggest challenge in my life. They have been the true source of real inspiration for me. Secondly, I would like to thank my supervisor, Prof. (Dr.) Neeraj Kumar, who has supported me throughout my Ph.D. work with his knowledge and most importantly believing in me. Apart from providing me with excellent supervision, active cooperation, feedback and constant encouragement throughout this journey, he also shared their invaluable experiences with me to succeed in life. I will always remain indebted to him.

I am also grateful to the head of the department, Dr. Shalini Batra, Associate head, Dr. Ajay Kumar, Ph.D. Coordinator, Dr. Sushma Jain, and members of my doctoral committee, Prof. Anil Kumar Verma, Dr. Rajkumar Tekchandani, and Dr. Sudhanshu Tyagi for their constructive suggestions and for ensuring the correct pace of my work. I am also obliged to the Director, Prof. Padmakumar Nair, Dean (RSP), Dr. N. Tejo Praksh, and the management of Thapar Institute of Engineering and Technology, who provided me with all the necessary resources and facilities to complete my work.

The chain of gratitude will definitely be incomplete if I forget to thank my complete family, my father Adv. Mulakh Raj Miglani, my mother Dr. Sheela Miglani, my mother-in-law Mrs. Ruchi Ralhan, my husband Mrigank Ralhan, and my sister-in-law Anubhuti Ralhan for their unconditional support and encouragement in every phase of my life. An extraordinary thanks to my son, Krishank Ralhan who taught me the real meaning of love and for bringing warmth and happiness to my life. Since then, the journey of Ph.D. has been a sweet and bitter ride at times, which leads to a special mention for my mother who stood by me through thick and thin and gave me courage at times when I felt really low. Her constant motivation showed me the silver lining in the dark clouds.

I would also like to pay my sincere regards to all my relatives and cousins Priya Miglani, Piyush Miglani, Vipansha Miglani, Devansh Miglani and Akshay Kochar for their constant

motivation and support. I would also like to express heartfelt thanks to my friends Dr. Shubhani Aggarwal, Vertika Wadhwa, Dr. Aayushi Kundu, Sukhpreet Kaur, Monia Digra, and Harleen Gill who always believed in me and whose blessings have truly played the role of game-changer in my life. A special thanks to my labmates Dr. Simarjeet Kaur, Krishan Kumar, Govind Chimpa and Niaz Wani. They all made this journey more comfortable with words of encouragement, which helped me in finishing my work.

I would also like to thank my colleagues with whom I have traveled this journey of research. Thanks to my seniors, Dr. Ishan Budhiraja, Dr. Anish Jindal, and Dr. Rajat Chaudhary. These people have made my research journey all the more memorable and pleasant. As one cannot mention the names of all well-wishers, friends, and beloved ones, I would like to pay my regards to one and all who supported me during this journey of knowledge.

This endeavor would not have been possible without the generous support from Tata Consultancy Services, who financed my research.

(**Arzoo Miglani**)

ABSTRACT

Due to the increasing popularity of Electric vehicles (EVs) around the globe, vehicle-to-grid (V2G) has emerged as one of the most promising technologies of the modern era in modern smart grid ecosystems. In such systems, timely communications among different entities play a pivotal role in efficient decision-making. However, with an exponential increase in the number of EVs around the globe, it is challenging to assign a unique IP address to each EV for content delivery in V2G environment. Also, with an increase in the number of EVs, the traffic load on the network local area aggregator (LAG), (which aggregates the data between charging stations (CS), EVs and disseminates it among different participating entities) increases manyfold. Most of the existing solutions reported in the literature for the aforementioned problems are based upon traditional Internet Protocol (IP)-based architecture, which may have a performance degradation with an increase in the number of nodes in the network. Also, if the contents (for example, power, voltage profile measurements, and hourly electricity charges) need to be shared with multiple entities, then these contents are transmitted from a central source to each entity instantly, leading to wastage of resources. Also, the current Internet architecture lacks in-network caching, which is pre-requisite to take efficient decisions for fast content delivery. Hence, the model shift from existing TCP/IP-based architecture is required to solve the concerns raised in the current Internet architecture, for example, routing, address assignment, congestion, high data delay, scalability, and security.

In comparison to traditional IP-based networks, content-centric networks (CCN) are widely used for efficient cache-based content sharing for the successful execution of efficient decisions in V2G network scenarios. In CCN, EVs can cache content to serve other nodes as per the requirements. As the same named data is cached at multiple locations in the network, so it can be retrieved from the nearest router, thereby reducing the network traffic and increased throughput. However, due to the selfish nature of EVs, they can only cache the content of their preferences. Also, in CCN, caching nodes separate the content from the content producer, depriving them of direct control over the content; thus, the content producer can't fully trust the caching node to ensure secure data access. Moreover, to get benefits from content-based caching services, it is essential to match the service providers with service requesters optimally during high mobility vehicle scenarios. Most of the reported solutions for the aforementioned problems are centralized, so these may suffer from congestion problems and overhead generation and do not have fault tolerance capabilities. Moreover, the concern of trust for content delivery via caching still

poses threats to the network as content acquisition via CCN communication connects different network nodes, so nodes have concerns about untrusted users connected with them. Also, borderless caching at the network level may introduce multiple security and privacy challenges [1]. Consequently, some nodes may not be willing to participate as providers due to their concerns about privacy leakage or high energy consumption. This situation may create an imbalance in content supply and demand among network nodes. To mitigate the trust and privacy challenges, blockchain emerges as a promising technology that can be used to realize distributed access control. Blockchain ledger can record the hash value of contents to prevent data tampering attacks.

As a solution to reduce the load on LAG, the first approach in this thesis proposes in-network caching for fast content delivery in the V2G network. Particularly, we propose a contract theory based incentive scheme to attract vehicles to participate in content delivery to the recipients. The consortium blockchain network is used to store the reputation value and incentives-related transactions in the ledger of all nodes using blockchain. The performance of the proposed scheme illustrates that it outperforms the existing state-of- the-art schemes with respect to delay incurred, social welfare, throughput, and latency. In the second approach, we model the two-sided preferences of both parties (service providers and requesters) to maximize the cached content sharing CCN communications. Then, we formulated a decentralized matching problem with the joint transmit power of both the content providers and requesters. Finally, a distributed blind matching algorithm (BLMA) is proposed, which is executed by deploying a smart contract on the Ethereum network without involving an intermediate authority. The results obtained show that the proposed scheme is superior in comparison to the existing state-of-the-art solutions with respect to various performance evaluation metrics, including interest satisfaction rate, residual energy, throughput, and latency. Finally, to solve the issue of access control in CCN, we propose a ciphertext-policy attribute-based encryption (CP-ABE) scheme to secure one-to-many data sharing in which blockchain is used for maintaining verifiable access policy records to ensure non-repudiation. In the proposal, instead of one, multiple attribute centers are used for attribute management and secret key distribution; thereby, a single-point failure problem can be mitigated. Security and performance analysis demonstrates that the proposed scheme is superior in comparison to the existing benchmark schemes with respect to metrics such as encryption time, decryption time, key generation time, computational overhead, and storage overhead. To ensure secure content delivery and to achieve trust in the network, we propose the proof of authority (PoA) consensus algorithm among vehicles in the V2G network.

List of Publications

Journal Publications (SCI/SCIE):

1. **Arzoo Miglani**, Neeraj Kumar, *"Blockchain-Based Co-Operative Caching for Secure Content Delivery in CCN-Enabled V2G Networks"*, IEEE Transactions on Vehicular Technology, DOI: 10.1109/TVT.2022.3227291, pp. 1-6, Dec. 2022. (IEEE, IF 6.293-Q1)

2. **Arzoo Miglani**, Neeraj Kumar, "A Blockchain based Matching Game for Content Sharing in Content-Centric Vehicle-to-Grid Network Scenarios." *IEEE Transactions on Intelligent Transportation Systems*,. (IEEE, IF 9.551-Q1)

3. **Arzoo Miglani**, Neeraj Kumar *"A One-to-Many Ciphertext-Policy Attribute-enabled Blockchain Access Control Scheme for CCN-enabled V2G Network Scenarios"*, IEEE Transactions on Vehicular Technology [Under review]

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviations	Definitions
BLMA	Blind matching algorithm
BFT	Byzantine fault tolerance
BS	Base Station
CPS	Cyber physical system
CSs	Charging Stations
CA	Certified Authority
CCN	Content centric networking
DDOS	Distributed denial-of-service
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EV	Electric vehicles
V2G	Vehicle-to-Grid
DNS	Domain name system
DOS	Denial-of-service
DDoS	Delegated proof-of-stake
DR	Demand response
DSO	Distributed system operator
DSRC	Decicated short range communication
DLT	Distributed Ledger Technology
FIB	Forwarding information base
HMM	Hidden Markov model
ICN	Information centric networking
IoT	Internet-of-Things
ISO	International organisation for standardization
LAG	Local area aggregator
MITM	Man-in-the-middle
MTU	Maximum transfer unit
NDN	Named data networking
OBU	On-board unit
OSPF	Open shortest path first
PIT	Pending Interest Table

PBFT	Practical byzantine fault tolerance
PKI	public key infrastructure
PoA	Proof-of-Authority
PoW	Peer-of-work
PoS	Peer-of-stake
P2P	Peer-to-Peer
RSU	Road Side unit
SDN	Software-defined networking
SG	Smart grid
SoC	State-of-Charge
SPoF	Single point of failure
TLS	Transport layer security
TLV	Type length value
TCP	Transmission control protocol
V2G	Vehicle-to-Grid
UAV	Unmanned aerial vehicle
V2X	Vehicle-to-everything
WLAN	Wireless local area network
QoS	Quality of service

Chapter 1

Introduction

Cyber-Physical System (CPS) is a combination of physical and computing powers to provide efficient, adaptable, and reliable services to the end users [2]. Its use cases include automobiles, medical devices, aircraft, smart grid (SG) systems, and robots, etc. However, in recent years, among these sectors, energy management in SG system has drawn the attention of communities around the globe. SG offers a platform that generates, distributes, stores, and transmits energy.

Moreover, since 1990s, researchers have been working to solve the problem of pollution and environmental changes generated by the combustion engine and burning of fossil fuels. In this context, electric vehicles (EVs), which are an integral part of SG CPS, have gained popularity in the last few years due to their ability to provide a green, eco-friendly environment (EVs have less carbon dioxide emission) [3]. With large-capacity batteries, EVs can use any energy source (wind, hydro, etc.) to charge their battery. However, penetration of distributed renewable resources, dynamic-natured uncoordinated EVs, abrupt location change, and high charging time lead to the problem of power fluctuations and distribution imbalance in SG [4]. Using energy storage devices or generators with high running costs is one way to solve this issue. Alternatively, a better option is to deploy a bidirectional energy trading EVs, *i.e.*, Vehicle-to-Grid (V2G) for discharging and Grid-to-Vehicle (G2V) for charging mode [5]. Also, it is reported that a typical vehicle uses a road for only 5% in a day, so they can be easily used to solve the aforementioned problem [6]. V2G technology enables EVs to be an energy storage medium to trade power back to the grid for load balancing in the peak time.

V2G or G2V network enables more intelligence, automation, and innovation to the conventional grid system. EVs can support two types of operations, *i.e.*, charging and discharging. Charging is started by EVs, where power is transmitted from SG to EVs. So, EVs pay SG, whereas discharging is started by SG when the grid gets overloaded during peak time, and for this operation, EVs get the reward. To achieve this, some EVs coordinate to store excess power in an off-peak period that can help provide electricity returned to the system at times of high demand for electricity. Moreover, vehicles can trade their energy to other EVs in a distributed Peer-to-Peer (P2P) manner. Power is transferred from the energy seller to the buyer with the

help of a wired or wireless energy transfer medium. Wireless energy transfer happens using microwaves, laser waves, or millimeter waves. Hence, the V2G applications can manage the demand-supply mismatch level during the peak load [7]. An important application of the V2G system is its usage in the military system as an emergency power supply source.

It is forecasted that one out of three vehicles will be electric by the year 2026 [8]. The largest automobile manufacturers are motivated to build new EV models in the coming ten years [9]. Also, the global EV market is estimated 9.5 million units for the year 2022 and is predicted to rise to 80.7 million units by 2030 [9]. This increase in the sale of EVs demands an enhancement in the current charging infrastructure. Also, onboard sensing and communication infrastructure enable on-road vehicles to connect with surrounding environment. In this context, an efficient communication solution is required to enable the information exchange required for activities, including authentication, billing, smart meter reading, energy scheduling, and coordination [10]. Also, information regarding traffic conditions, state-of-charge (SoC), and waiting time, etc., has to be transferred between EV and charging stations (CSs).

In India, the development of Vehicle-to-Grid (V2G) technology is still in its infancy, but there are promising signs of progress. Various research institutions, government bodies, and private enterprises have initiated experimental projects and studies to explore the feasibility and benefits of V2G implementation. However, widespread adoption remains limited due to several challenges. One of the primary hurdles is the lack of standardized V2G infrastructure and regulatory frameworks. The absence of uniform protocols for communication, charging interfaces, and grid integration inhibits interoperability and scalability. Additionally, concerns regarding battery degradation and warranty implications need to be addressed to gain consumer trust [11]. The following steps should be taken for Wide Adoption of V2G Technology in India.

- Standardization and Regulation: Establishing national standards and regulatory guidelines for V2G technology is imperative. This includes defining protocols for communication, interoperability, and grid interaction to ensure seamless integration across different EV models and charging infrastructure.

- Infrastructure Development: Significant investment in V2G infrastructure is required to support its deployment across the country. This involves the expansion of smart charging networks equipped with V2G capabilities, especially in urban centers, commercial hubs, and along key transportation corridors.

- Policy Support and Incentives: The Indian government can play a pivotal role in promoting V2G adoption through supportive policies and incentives. This may include financial incentives such as subsidies, tax breaks, or grants for V2G-enabled EVs and charging infrastructure.

- Research and Collaboration: Collaboration between industry, academia, and government agencies is vital for advancing V2G technology in India. Continued research and devel-

opment efforts are needed to address technical challenges, optimize V2G systems, and identify opportunities for innovation and scalability.

A typical V2G scenario consists of four main entities: EVs, CSs/ service providers, an aggregator, and a certificate authority (CA). As vehicles are mobile, the SG is responsible for managing vehicle transaction records, access control, and authenticity. For this purpose, SG uses Base Stations (BS) which are called aggregators. Fig. 1.1 represents the existing V2G network architecture with four entities, as discussed below:

- EV: An EV is owned by an individual owner and has a connection to the CS for charging or discharging purposes. An EV has built-in computational software enabling it to interact with the electric grid.
- CSs: It supplies electric power for charging EVs, and they are mostly deployed to a places with heavy parking, for example, shopping malls.
- Local Area Aggregators (LAG)/Energy aggregator: It acts as an intermediator between SG, EV, and CS. A LAG communicates with vehicles and delivers collected information to SG.Also, it communicates with the grid on behalf of all EVs. They perform the tasks of managing charging and discharging schedules, power status monitoring, energy load scheduling, and billing. Calculating cost-effective strategies for charging is another role for the aggregator. Aggregators gather energy from power generators, microgrids, and vehicles to distribute and manage it further.
- CA: CA is a entity that performs third-party auditing. It checks and verifies the billing services or any other transaction records generated by LAG for any vehicle.

One requirement with Vehicle-to-Everything (V2X) communication is data transfer from one point to another quickly without losing data packets. Nevertheless, in order to provide effective EV charging coordination and information transfer, good communication technology needs to be implemented. The choice of communication infrastructure affects the speed, reliability, and security of information transfer. An effective communication infrastructure should avoid undesirable delays, have high throughput, have high reliability (ubiquitous coverage and high quality of service (QoS)), and should provide security services. Direct wired communication based on IEC 61850 and ISO/IEC 15118 is proposed in the literature to support message exchange between EVs and other nodes [12]. Also, high-speed broadband communication is also used in V2G to carry communication signals [13]. Although wired communication supports high security and high data rate but can only be used for static infrastructure (mobility of EVs is not supported). To support V2G communication, dedicated short-range communication protocol is widely used. DSRC (that includes IEEE 802.11p standard) is bidirectional short to medium-range wireless communication protocol that allows a high data transfer rate for the vehicle-to-vehicle or vehicle-to-infrastructure communication [14]. The communication model

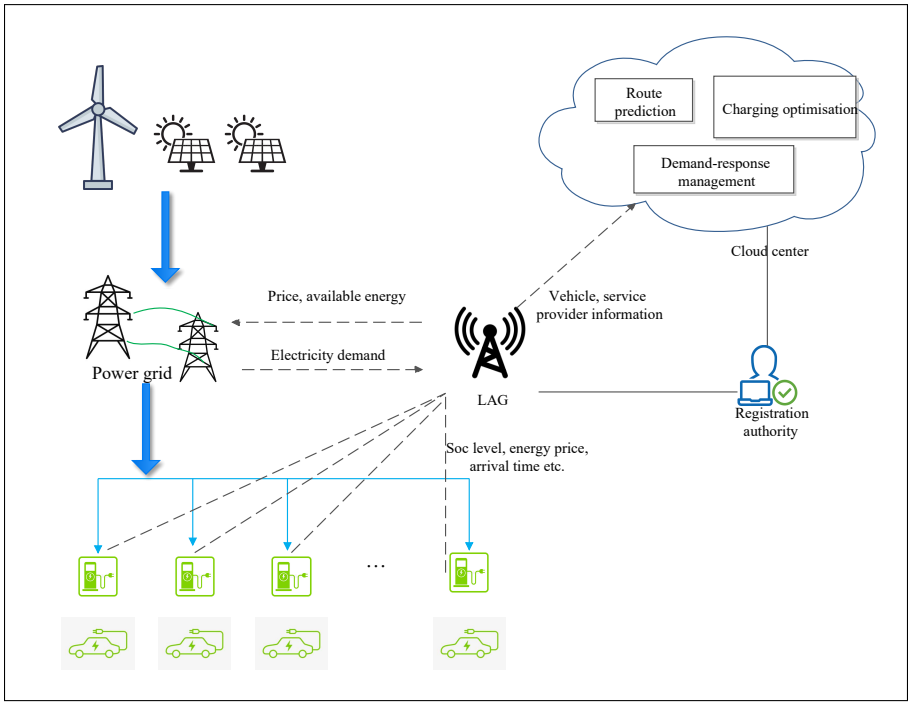

Figure 1.1: Architecture of V2G

of DSRC is based on two nodes: on-board unit (OBU) which is placed inside a vehicle and is mobile; the other is a roadside unit (RSU), which is static and deployed on roadside infrastructure. The network layer of DSRC is based on principals of TCP/UDP and IPV6. DSRC supports low latency, high-speed communication medium, authentication, and reliability, etc. Despite providing advantages, IEEE 802.11p DSRC faces difficulty in the non-line-of-sight process, which reduces the capacity of the communication link between OBU and RSU. So, in a high congestion network, the efficiency of DSRC is deteriorated [15]. Also, these standards depends on Public Key Infrastructure (PKI), that involves centralized CA performing authentication, reliability, interoperability, etc. [16]. LTE cellular network provides ubiquitous coverage, low latency, and high QoS over the DSRC network. However, similar to DSRC, the LTE network provides inefficiency in high-density traffic. In contrast, cellular 5G is capable enough to meet the requirements of V2G as it provides scalability to connect EVs. Another protocol named International Organization for Standardization (ISO) 115118-2: 2014 represents communication for EV supply equipment and EVs. This communication protocol uses the Transport Layer Security (TLS), Transmission Control Protocol (TCP), and Internet Protocol Standard Version 6 [17]. The protocol is based on unilateral authentication which may lead to redirection and impersonation attacks. Moreover, Wi-Fi has limited coverage and capacity, making it unsuitable for vehicular communication. However, these short-range wireless technology can be used for stationary EV charging. With the increasing security threats, V2G communication demands mutual authentication between EVs and respective LAGs or servers. In IEEE 1646 standard documentation, it is reported that data delivery time of wide area network control with TCP/IP ranges from minutes to hours; thus, data may not be delivered to

Table 1.1: Comparison of popular communication technology for V2G.

Communication technology	**Specification**	**Speed**	**Radius**	**Frequency**	**Mobility Support**
Zigbee	IEEE 802.15.4	240 kbps	100m	2.5 GHz	low
LoRA/LoRaWAN	IEEE 802.15 g	28 kbps	15 km+	870-936 MHz	Low
Wi-Fi	IEEE 802.11	100-250 Mbps	100 mts+	20 MHz	Low
DSRC	IEEE 802.11 p	3-27 Mbps	1km	10 MHz	Medium
4G LTE	3GPP	0.2-1 Gbps	30 Kms	600 MHz-2.5 GHz	High
5G	IEEE 802.11 od	1 Gbps	100m-7Km	600 MHz-26 GHz	High

nodes on time. [18]. Although Wireless Local Area Networks (WLAN) and Zigbee technology are enough to manage Home Area Networks (HAN) and home automation, however, broad power line control has to be used for EV charging and automated meter reading. Table 1.1 presents the comparison of existing communication technology for V2G.

Therefore, there lacks an excellent communicational infrastructure for V2G communication. As discussed, V2G is a bidirectional communication between grid and vehicle, which leads to more interactivity. This communication demands delivering data efficiently and with low latency. The conventional host-centric IP network model is still considered to be an effective, widely used, and dominating communication infrastructure, where identification and communication between machines in a V2G network involve the use of IP addresses. Also, the Internet is now content-oriented , i.e., content is shared regardless of the source of content. The IP-based communication model may not be well-suited for content-oriented networks that prioritize data sharing without concern for the specific source serving the request. Some challenges in current communication infrastructures for V2G are highlighted:

- This traditional protocol suite defines a host-centric model in which a connection is first established between two hosts, and then information is exchanged. The host-centric framework indicates that the host in the network should be location-aware. As in the past, the Internet was designed to connect to mainframe computers only. However, connectivity in the V2G network is highly abrupt. Hence, conventional IP based communication is not suited for the V2G scenario due to difficulty in dealing with mobility, network topology change, and continuous disruption.

- In a V2G network, information needs to be multicasted among multiple nodes, for example, the price of electricity per hour. Moreover, producers announce their energy generation profiles to be distributed among multiple nodes. To get the announced decisions, consumers must be in the range of the producer's multicast path. However, maintaining a

huge number of multicast trees is not adaptable for a non-scalable traditional host-centric IP architecture.

- The layer-5 TCP protocol ensures in-order packet delivery and congestion control services which are not required for V2G networks. Also, this protocol leverages 3-way handshaking that brings latency not to be tolerated by delay-sensitive V2G applications.

- In location-based host-centric communication, nodes need to know IP addresses to share information. Such usage of IP addresses may lead to several security vulnerabilities and DoS attacks [19] [20]. Also, IPv4 addressing scheme is being shifted towards IPv6 because of diminishing IP addresses. However, in IPv6, a minimum of 1280 byte Maximum Transmission Unit (MTU) is required, which is not energy efficient for a V2G network.

- Lastly, naming resolution via traditional Domain Name System (DNS) is not enough for most nodes in V2G network

- Moreover, in a V2G system, LAGs are strategically deployed at different locations to facilitate communication between CSs and EVs. Each LAG aggregates all CSs and EVs data and stores it in a local storage. However, with an increasing EV penetration, LAG has to serve more vehicles with enormous mobile traffic, which can lead to high network delays. The increased load on LAG also affects the quality of experience for the receiving nodes. In different time intervals, the same data needs to be transferred between various entities, resulting in a complex data flow with a heavy load on the network infrastructure. To relieve the LAGs load, caching on network nodes is a viable solution. However, the existing IP-based protocol lacks in-network caching to reduce network delay.

Hence, conventional IP-based communication is not befitted for V2G scenario due to difficulty in dealing with mobility, network topology change and continuous disruption. The usage of TCP/IP to improvise content delivery, mobility support, and consistent routing makes V2G more difficult to cope with QoS provisioning. Thus, there is a need to change the communication architecture to suit the requirements of V2G applications. To cope with the aforementioned challenges, a content-centric network (CCN) approach can be used for information sharing in the network. Some of the CCN outcomes include multicast communication, inherent support for multipath routing, provision of mobile service, and inbuilt security, which are not easy for existing IP addresses to provide.

Named Data Networking (NDN)/CCN are among different projects of Information-Centric Networking (ICN), and all these technologies shift location-centric network to data-centric networks. CCN, compared to the host-centric network model, is a location-unaware connectionless communication model that relies on the content of the application for efficient data delivery. The idea behind CCN is that consumers are mainly concerned with information data than original source of data. As compared to host-centric IP-based networking, a packet name

other than a sender and destination IP address is used in CCN communication. Using the namespace of the application eradicates the demand for IP addressing. For V2G implementing CCN, every entity behaves as a CCN router to store content in cache and forward requests to other routers for fast content delivery. When the content is located either from the original publisher or intermediate router, it is forwarded through the reverse path.

It uses a hop-by-hop stateful packet forwarding scheme. CCN packets are of form Type-Length-Value (TLV). It is based on the request-response model, i.e., an Interest packet is generated for making a request, whereas data packets are used to deliver requests. These packets has a name identifier as top field. The data packet is having same name to interest packet, and it is delivered to the destination by its name. A consumer machine initiates data retrieval in CCN by sending an interest packet. In response, original content producer or any other intermediate network machine having the cached content request replies with a matching data packet. With CCN, data achieves security with authentication and integrity directly at the network layer as the producer digitally signs data packets. Producer/Publisher creates data packets followed by naming them in a structured hierarchy. CCN routers use this name to forward data at the network layer. Hence, using CCN can improve traffic congestion control as content retrieval in CCN reduces data access latency and improves security.

CCN content transfer is based on key pairs (public and secret key). Also, CCN-based operations depends on the trusted centralized third party that suffers DoS attacks and privacy issues. Also, a content holder may spread fake information about content availability to requesters, which may affect security of system. Hence, the concern of trust for content delivery via caching still poses threats to the network as content acquisition via CCN communication connects different network nodes, so it may bring trust issues among them because both content requesters and providers have concerns about untrusted users connected with them. Also, borderless caching at the network level may introduce multiple security and privacy challenges [21]. Some nodes may not be willing to participate as providers due to their concerns about privacy leakage or high energy consumption. This situation may create an imbalance in content supply and demand among network nodes. To mitigate the trust and privacy challenges, blockchain emerges as a promising technology which can be used to realize distributed access control. CCN and blockchain are mostly researched independently. Ensuring the dependability of content is crucial for CCN, and blockchain can prove beneficial in achieving this goal. This technology offers a means to deliver reliable content within an untrusted network without the need for a third party. However, some literature work integrates blockchain for CCN in different applications. CCN facilitates the efficient retrieval of content, while blockchain ensures the security of the content. Hence, using blockchain in CCN can improve performance in secure content transmissions.

1.1 Introduction to Content-centric network

1.1.1 Architecture of CCN

Every node of the CCN network has these data structures.

- Content Store: It represents the content store for data packets to satisfy future requests. The content store is managed using management strategies (e.g., Least Recently Used), which make the decision regarding which information to keep. Unlike the buffer of an IP router, the content store can reuse a packet even once forwarded. Caching saves network bandwidth and decreases content retrieval time. Nevertheless, an efficient content replacement policy is required to utilize content stores effectively.

- Forwarding Information Base: FIB contains routing entries that is constructed using a name-dependent protocol (e.g., Open Source Path First for Named-data (OSPFN), or Hyperbolic Routing). It comprises table entries to map name prefixes to the nearest outcoming interfaces. A node takes advantage of longest prefix of name to find the next hop having the required content. The choice of the routing protocol decides how to populate FIB.

- Pending Interest Table (PIT): PIT contains records of all unfulfilled requests, i.e., all those entries of request packets earlier forwarded by router. All entry record contains interest packet name with incoming interfaces. Only the first request packet having same name is forwarded; rest are rejected. Nodes in the network has PIT entries to route without IP address and also to provide multicast support.

1.1.2 Content Naming

Content naming has an important part in CCN-based V2G. For each piece of information created on the network, names in CCN are tailored to offer an extensive addressing range depending on names. Unlike IP addresses, names in CCN are formulated based on application. The naming scheme also enables efficient routing. Different application-related content, such as spatial-temporal, geographical, and safety information can be described uniquely by CCN names. The naming scheme in CCN is hierarchal based, and each component is separated by a "/". For example, suppose a consumer has to generate an interest packet for information about electricity at the current time. In that case, the interest packet is named like $grid\_ID/electricity/price/now$, where $grid\_ID$ is the prefix that specifies the nearest grid ID.

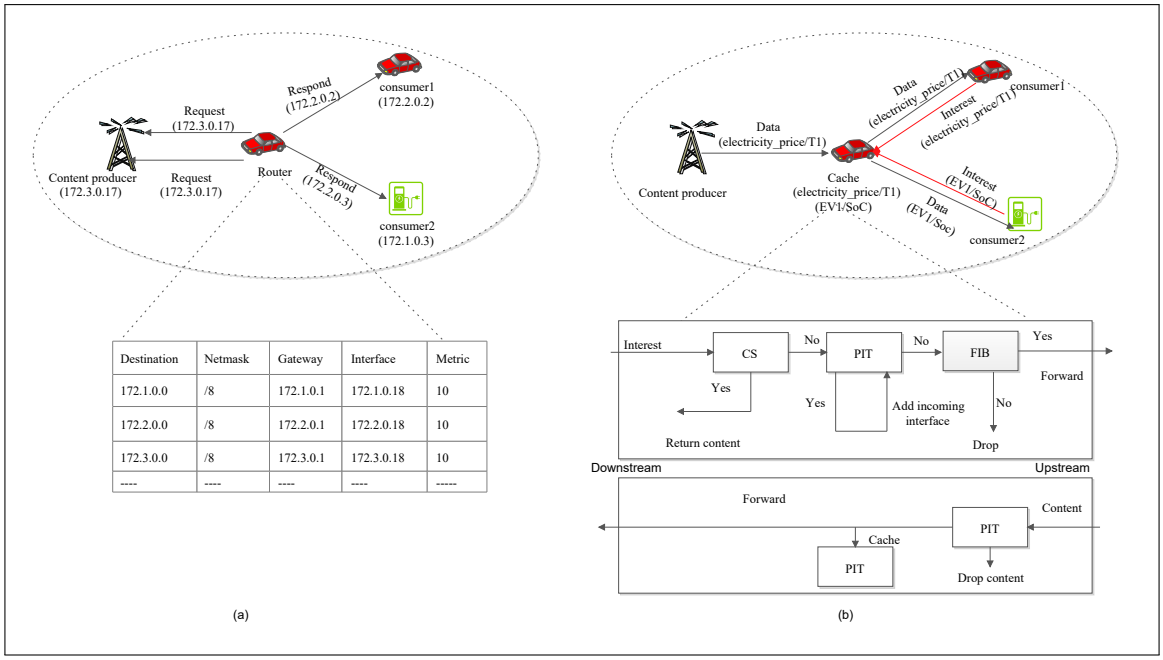

Figure 1.2: a) IP based V2G scenario b) CCN based V2G scenario

1.1.3 Packet forwarding in CCN

1.1.3.1 Interest packet forwarding

- A content requester initiates a request with request packet having unique name. After receiving request packet, the router checks for its name and checks it in its cache. If there in the cache store, the content packet is sent to satisfy request through an incoming port. Otherwise, content name is probed in PIT.

- The router updates the incoming interface to an existing entry in the PIT that already has the same content name, and it discards the request packet in that case. Alternatively, the router adds a fresh entry with the content name and incoming interface to the PIT table if there isn't already one. The PIT is used to combine several Interest requests for content that have the same name from several requesters that have already been forwarded but are yet unfulfilled. The nearby node sends copies of the requested content to every other node connected to the interface mentioned in the relevant PIT entry.

- Lastly, FIB is probed for a routing path, and a face is selected to transfer the request packet to the next router. For mismatch, router broadcasts packets to all outcoming interfaces.

1.1.3.2 Content delivery phase

Upon receiving packet with content, router checks in the PIT, and for a match in PIT, the packet is send to all incoming interfaces to enable multicasting, and the PIT entry is discarded.

Also, depending on caching policy, it is decided to cache content or not. An example of packet processing in CCN is represented in Fig. 1.2

1.1.4 Security and mobility support

In the TCP/based model, the security is dependent on channel security. The server uses cryptographic hashing methods for secure data transfer [22]. In CCN, security is applied on the content only instead of channels. Hence, content transfer or caching content doesn't need communication security. When an EV keeps moving, its IP address changes continuously as of changing network topology. In NDN, EV can re-issue the unsatisfied interest request. Also, the producer of the content digitally signs the chunks using private key, so any receiver node can validate the authenticity.

1.1.5 Inherent multicast support

PIT table can accumulate multiple request packets for the same data packet. Interests having same content name are merged as one entry in PIT table. Hence, redundant transmissions get decreased by forwarding only one interest instead sending multiple requests for the same content.

1.1.6 Robustness

Robustness is the capability to deal with frequent transmission failures without suffering network QoS. Due to instability of wireless links and because of high mobility of EVs, robustness of IP communication is prone to topology changes. With a caching facility, the effect of broken links can be reduced, particularly for popular content as they can be retrieved easily from multiple nodes in the neighboring area. Also, the total hop count for content delivery is decreased. Hence, the robustness of the CCN-based V2G network can be improved with caching.

1.1.7 Challenges in CCN

CCN is still new, and researchers are working to make it a future Internet architecture. To make CCN practical, challenges of content security for efficient caching, and fast data delivery need to be addressed. Next, we will discuss some issues faced by CCN in detail.

1.1.7.1 Lack of incentive mechanism

To relieve LAGs load, each EV actively contributes to the content caching and acts as a potential content source. It causes decrease in network congestion and data access delay with

increased network transactions processing. Placing content copies to EVs is beneficial for promoting LAG load release. However, due to the selfish nature of EVs, they can only cache the content of their preferences. Also, each vehicle wants other nodes in its neighborhood can cache its desired contents. Therefore, there is a strong need to design a strategy to handle the caching content preferences by different content stores. In this context, several proposals in literature have been put forward for making efficient caching content findings based on content popularity [23], [24]. However, popularity-based caching decisions without any incentives are difficult to implement. Hence, to encourage caching and to attain cooperation in caching from EVs, LAG can use an incentive strategy, *i.e.*, EV cache content packets and reply to the requests of other EVs, CSs and utility centers for reducing LAGs load; in return, LAGs pay to EV for cache offloading. Here, EVs or utility centers or CSs with similar interests can retrieve content from caching EVs with the requested content store data instead of requesting LAG. Hence, the in-network caching of CCN successfully minimizes the traffic load and can provide efficient content retrieval in V2G. Nevertheless, there are multiple approaches for designing an incentive-compatible co-operative caching in CCN/NDN, including Stackelberg game, auction theory, and nash bargaining theory. However, all these solutions are based on an iterative mechanism, which has the disadvantages of long convergence time and requires multiple information exchanges between two parties [25]

1.1.7.2 Access control in CCN

As CCN supports in-network caching, the intermediate node can handle a content request in the forwarding path. However, this introduces security concerns that need to be addressed during content delivery, with content access control being of utmost importance.

One of the popular ways to achieve access control is encrypting the content before transferring it to the receiver. Encryption in most of the existing schemes is achieved by using the public keys of receivers; however, this approach is not efficient for one-to-many content transfer [26]. To support security in situations where content is not stored at the content producer's (CP) location, authors in [27] present attribute-based encryption (ABE), which offers sophisticated encryption with fine-grained one-to-many access control. Among one of its types, in CP-ABE-based access control, access control policy (ACP) contains the desired requester attributes (defined by the CP) and is embedded with encrypted content [28]. The attribute authority center (AAC) issues a secret key for each requester corresponding to the requester attributes. However, a single authority center leads to a key escrow problem, as a single authority may apply generated secret key to decipher the ciphertext. Similarly, multi-authority-based CP-ABE schemes, such as in [29], where more than one authority handle disjoint attribute sets independently, still face key escrow problem as an authority has full control over the attribute set. Moreover, crash of any authority results in the loss of the private key of attributes, affecting the system operations. Also, the revocation operation in CP-ABE incurs high computational costs

for operations, including policy updates, re-encryption, key re-generation, and distribution.

The existing access control techniques execute at cache-enabled neighboring node (NN) and involve interaction with the CP during content retrieval [30], [31]. In existing encryption-based access control schemes, key management is difficult with the various nodes in the system. CP has to stay online to allocate keys to new members. Moreover, known access control schemes such as mandatory access control and role-based access control are not applicable to CCN, as these schemes are designed to implement ACP where contents reside with the CP [32]. Also, CP-ABE causes key escrow, heavy computation, and authority collusion problems.

1.1.7.3 Lack of matching model for content producers and requesters.

In CCN, a requester node generates an request packet with desired content name and broadcasts it to the neighborhood. When a data provider receives the request packet with a content name, it transmits it to the requester, and all the in between nodes cache the received data packet. However, with multiple requesters and providers for content, requesters may face competition against the best sharing services, and providers must filter the incoming requests for low energy consumption. After receiving interest packets, if more than one content provider has the content, then all these providers transfer their data to the requesters separately, leading to an unwanted use of transmission resources. Similarly, as a content provider, it may receive multiple requests from requesters in the neighborhood at the same time. However, a service provider may have a limited transmission capacity, so it can transfer data to one receiver at a time. Moreover, it may offer services with minimum overhead [33]. Hence, it needs to select one of the potential consumers to send the data. On the other hand, the requester side may face competition with respect to high reliability and low delay. Also, in terms of the mobility of vehicles, the content duration is limited, leading to a limited transaction time while obtaining content. Hence, a matching model is required to match content requesters and providers for efficient content sharing. Matching theory is a popular mathematical concept that has applications in various domains, including communication networks, college admission programs, and kidney exchange programs [34]. However, most of the existing matching game solutions are centralized, having a single failure point, as they need a coordinator to regulate the matching market [35]. Also, the centralized solutions may lead to congested matching operations, affecting the network's scalability. Moreover, in most of the existing solutions, the decisions of content requesters and providers contradict each other without mutual benefits. The designed matching model should consider the mutual preferences and satisfaction of both content providers and requesters

1.1.7.4 Trust and privacy

With the introduction of a new communication protocol in V2G, it is important to provide secure content delivery to network nodes. Despite serving significant advantages for informa-

tion distribution, CCN has several challenges to provide security in V2G network.

- V2G network benefits from in-caching feature of CCN; however, the concern of trust for content delivery via caching still poses threats to the network. In-network caching feature of CCN allows adversaries to launch various attacks. Also, borderless caching at the network level may introduce multiple security and privacy challenges.
- As the caching-based content delivery does not provide information about the content provider's address, it may result in selfish and unfaithful behaviors.
- The issue of cooperation among nodes in a trusted way still prevails in a large-scale application of CCN like V2G [36], [37].
- In conventional caching mechanisms, nodes rely on a centralized party to verify the transaction of CCN based network, which lead to multiple security threats, including single point of failure, and DDoS. However, most of the current work mainly concentrates on designing an incentive mechanism in CCN-based vehicular networks, however the privacy issues to facilitate secure content delivery are not considered.

For privacy preservation, existing proposals use techniques such as group signature [38], random forwarding [39], ring signatures [40] and network coding [41], but the overhead of these schemes is significant.

1.2 Blockchain Technology and Technical Foundations

Centralized architecture has a central coordination system and every node on the network is connected to this system. Any information sharing in the network has to involve this central coordination system. Nevertheless, there are some disadvantages with a centralized system.

- Single point of failure (SPoF): What if the system or the server crashes? Unfortunately, in case of a crash of this central system, all nodes on the network get disconnected to the network, and all operations get terminated. This situation may lead to the loss of entire information. Therefore, complete dependency on a single server is not efficient.
- Bottleneck: Bottlenecks are common in case of increased traffic.
- Single point of attack: As there is a single central authority, there are chances of a single point of attack. Therefore, this type of architecture can easily suffer a denial-of-service attack.
- Delay: A centralized server is mostly located far from users, so the time to access data increases.

- Higher privacy risk: As centralized architectures involve a third party, so the user is unaware of how the information of users is secured with the third party. The trusted third party may share users' private information with other parties.

To solve the above discussed problems, blockchain emerges as a popular technology that enables decentralization among nodes. Blockchain is a platform that provides support for decentralized and distributed architecture where nodes of the network can share information among themselves. In comparison to a client-server model, blockchain implements a digital P2P network [42]. Blockchain is a growing list of blocks combining cryptography with distributed computing to provide decentralized, transparent, and strong consistency support. In a blockchain network, multiple nodes are connected via the Internet, and every node maintains a local instance of the global sheet. However, these local copies should always be updated as per the global information. In particular, this local copy of data is called a public ledger. A popular example of the public ledger is banking transactions, and the first popular use case of blockchain is the Bitcoin network. Blockchain Bitcoin is a decentralized system for exchanging cryptocurrency and sharing distributed ledger. Many other blockchain cryptocurrency platforms were introduced including Ethereum, leveraging the same public model as Bitcoin, whereas platforms such as Hyperledger, and Ripple are some permissioned blockchains. The distributed applications of blockchain are used in many other sectors including healthcare, IoT, smart grid, etc [43]. Blockchain provides a decentralized common platform for multiple parties who don't trust each other and are involved in information-sharing or rational decision-making processes. This technology provides an effective way of storing transactions securely, transparently, and highly resistantly. The blockchain network makes sure to ensure consistency and maintain synchronization of the document. Anything stored on the blockchain has a transparent nature, and anyone modifying it is accountable for their actions. Moreover, the decentralized nature of this network ensures that a single node on the network can't append invalid blocks to the chain. Before a transaction is appended to the blockchain system, it is verified by all the participants on the system. Before adding a fresh block to network, it is always chained with the previous block using cryptographic hash of the immediately previous block. Therefore, cryptographic linking ensures the integrity of the network [44]. As every block is cryptographically linked to the previous block hash, that is why the name blockchain is defendable. If any block is altered, attackers must modify all subsequent blocks, which is quite difficult.

In particular, blockchain uses the concept of hash functions, Elliptic curve cryptography (ECC), digital signatures and Elliptic Curve Digital Signature Algorithm (ECDSA) to maintain integrity, confidentiality, and non-repudiation of the system. However, a consensus mechanism is used by network participants to achieve mutual agreements on a single state of the network in a distributed environment. Clearly, the consensus mechanism minimizes the risk of fraudulent transactions [45].

Blockchain mining is a process of validating transactions before it is added to the blockchain; miners are the entity for validating and generating a new block in the network. Some special

nodes with some special characteristics (different for every blockchain network) are only regarded as a miner. Further, the mined block is broadcasted in the network to be verified by other nodes before final inclusion in the network. Whenever a new blockchain transaction is made, it is first placed in a transaction pool. Rather than validating a single transaction, miners collect various transactions from the transaction pool to form a candidate block. Hence, a candidate block is referred to as a block that a miner has created, but it is not added to the network. It may so happen that multiple miners can mine a block with exactly the same or some different transactions simultaneously or at a near identical time. However, when two blocks get mined simultaneously, there is a possibility that only one miner's blocks get more blocks on top of it. If multiple valid blocks to the existing chain appear, in that case, only the longest subbranch is accepted and continued further; the blocks that are not accepted are called orphaned blocks, and that path is called forks. In other words, orphan blocks are those blocks that do not have any link to main branch due to missing predecessor. Additionally, if there are two different chains of the same length, then accept the chain that has been broadcasted by more miners. Transactions from these blocks that are not validated are sent back to the transaction pool. In such cases, the efforts of miners go useless as mined blocked becomes unrecorded [46].

1.2.1 Characteristics of blockchain technology

- Decentralization: Blockchain technology does not depend on a centralized system or any governing authority to perform all transactions. Instead, the network is controlled by nodes of the network, making it decentralized. Every node has copy of the shared ledger which is updated. Moreover, it solves the problem of a SPoF [47].

- Better security: Cybersecurity is defined as the capability to prevent and recover from cyber-attacks. Blockchain technology provides better security as there is not any chance of system failure. The use of a cryptography system by blockchain provides protection for users. Another reason for the popularity of blockchain technology is basically its capability to deal with the threat of an individual's privacy. All transactions are verified, and it is quite hard to modify these transactions.

- Immutability: Immutable ledger is the main advantage of the blockchain system. Immutability implies data on the network can't be changed or altered. Blockchain stores permanent records of transactions. After a block is verified and added to the network, it can't be changed or deleted. Moreover, the lack of centralization promotes scalability and robustness. Centralized architecture can be tampered with and requires trust in a third party to maintain integrity.

- Anonymity: Blockchain provides anonymity as nodes are known by their public keys on the network. Therefore, the identities of the nodes are kept private.

- Transparency: Any node on the network can audit transactions, and every node has access to the same universal ledger. Every state of data and every updating state is visible to node of the network.
- Redundancy: As a copy of the distributed ledger is stored with every full node on the network, redundancy is inherent to blockchain.
- Efficiency: All transactions are automatically executed via pre-set procedures. Hence, blockchain technology reduces the cost of labor along with improving efficiency.

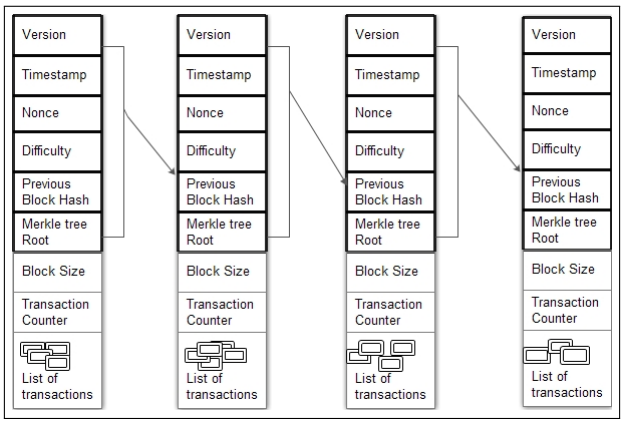

Figure 1.3: Structure of chained blocks

1.2.2 What constitutes a block of blockchain?

The first block in a blockchain is the Genesis block, which don't have a previous block. All network participants should have the same genesis block in order to attain the validness of the blockchain network. The previous hash value for a genesis block is zero. The structure of the blockchain comprises strings of blocks, each holding transactions of data and metadata. Data inside a block contains transactions generated by participants of the network and blocks hold the transactions in a secure way so that they can't be tampered with. A transaction is an atomic event or the smallest building block allowed by a particular protocol. For instance, in Bitcoin, blockchain transactions are user's payments. On the other hand, the metadata contains information regarding the block including parent block hash, timestamp, etc. This informative metadata is used by miners or the other nodes of the network to verify a block or to append a block to the network. The structure of the chained block is represented by Fig. 1.3. The Metadata of the block consists of the following field:

- Version: It tracks protocol upgrades used by blockchain nodes.
- Timestamp: It specifies the creation time of the block.

- Nonce: It is a random number used to solve the Proof-of-Work (PoW) cryptographic puzzle as shown in Eq. 1.1.

$$\begin{aligned} SHA-256(SHA-256(Previousblockhash||Tx_1 \\ ||Tx_2||.....||Tx_n||nonce)) < Difficulty \end{aligned} \tag{1.1}$$

- Difficulty/Target: This is used by the PoW algorithm to solve the mining process. To add a block to the blockchain network, it has to generate a valid hash and difficulty value is used in achieving this task.

- Previous block hash: As mentioned earlier, every n^{th} block of blockchain stores the hash of the previous block, i.e., $(n-1)^{th}$ block. In order to compute the hash of $(n-1)^{th}$ block, all header fields of $(n-1)^{th}$ block are collectively hashed twice.

- Merkle tree: It is basically a tree structure where the nodes at the bottom level contain the hash of the document, and every intermediate node has the hash of the left and a right child. As it is presented in Fig. 1.4, there are 8 transactions, i.e., t_1, t_2,.....,t_8. Leaf nodes of the Merkle tree contain the direct hash of these transactions, and then level 1 has intermediate nodes with the hash value of its left and right child (i.e., obtained hashes are again paired to calculate the hash for the next level). This hash will be recursively calculated until a single root hash is obtained.

1.2.3 Blockchain based V2G system

The introduction of V2G technology marks a significant evolution in conventional power grids, visualizing smart cities powered by a diverse array of resources including microgrids, sensors, smart meters, and electric power substations. However, scaling access to distributed and scalable energy utilities within V2G presents considerable challenges [48]. In this context, decentralization has always been a basis for V2G networks so that all nodes can incorporate and integrate in a dynamic way. Nevertheless, decentralized data management in V2G environments introduces several critical challenges. Ensuring data security and privacy is paramount, given the increased risk of breaches, unauthorized access, and potential data manipulation across multiple nodes. Furthermore, maintaining data consistency and integrity amid dynamic data generation, transmission, and updates poses significant hurdles in a distributed environment. Interoperability and standards also prove essential, with the absence of standardized protocols inhibiting seamless integration and collaboration among stakeholders. Also, scalability becomes a concern when decentralized systems lack resources to meet growing demand, while fault tolerance and resilience are crucial to address potential failures and disruptions.

Emerging blockchain technology holds promise for addressing these challenges and revolutionizing V2G applications. Blockchain's decentralized nature eliminates the need for a central

authority, enabling multiple entities to collaboratively create and maintain a tamper-resistant chain of blocks. This ensures data integrity, transparency, and authenticity, crucial for secure energy transactions within V2G networks. Blockchain finds applications across various V2G domains, including peer-to-peer energy trading, advanced metering infrastructure, demand response management, and power generation [49]. Moreover, blockchain facilitates real-time energy markets, reduces transaction costs, and enhances privacy for V2G entities.

In a V2G-based blockchain environment, entities such as smart meters, microgrids, charging stations, and utility servers are interconnected via a peer-to-peer network. Transactions, including energy trading processes, are validated and stored on the blockchain by network participants, eliminating the need for traditional intermediaries like banks. This streamlined process leads to a real-time energy market, minimizing energy loss and enhancing privacy within V2G. Furthermore, blockchain technology empowers energy distribution agencies with real-time insights into grid operations, enabling optimal dispatching plans. Smart contracts automate and enforce agreements between stakeholders, streamlining transactions and reducing costs. Integrating blockchain into EV charging can further optimize power fluctuations, charging costs, and ensure system security and privacy [50].

1.2.4 Blockchain based CCN network

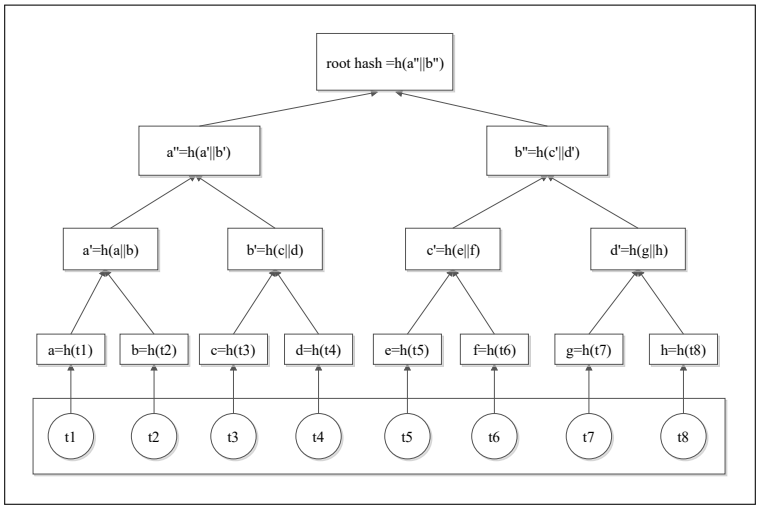

Figure 1.4: Structure of Merkle Tree.

Blockchain in the CCN network can be used for secure content delivery in CCN as network participants maintain trustworthy ledgers in an untrusted environment. Fig. 1.5 represents the applications of a blockchain-based CCN network.

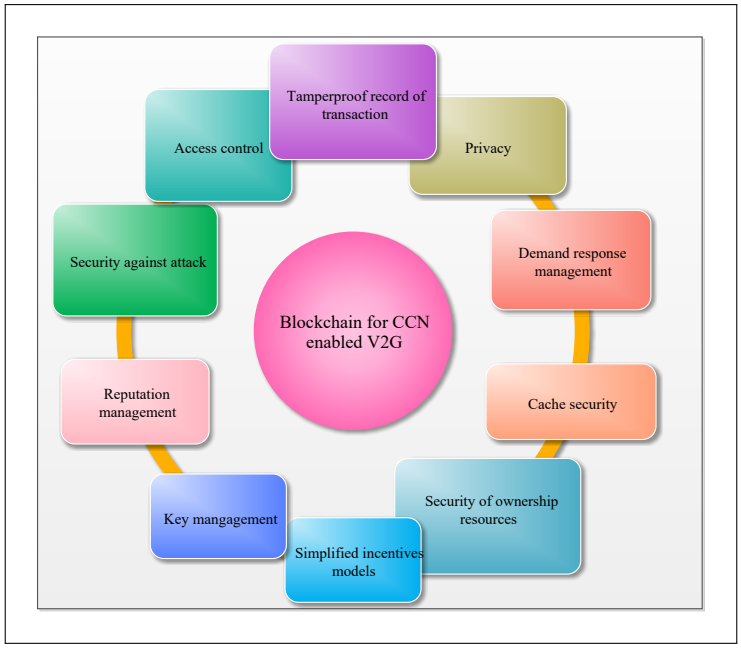

Figure 1.5: Applications of blockchain-based CCN

1.3 Thesis Organization

The whole thesis is organized in six chapters and the summary of each chapter is presented as follows:

Chapter 1: Introduction

This chapter discusses the constraints of the existing IP based Internet architecture and discusses how CCN is better over TCP/IP in terms of data delay, multi casting, mobility and security. Next, we discussed the different characteristics and operations of CCN including its components, architecture, naming, routing, caching, and packet forwarding. Next, we discussed the different challenges faced while its deployment that includes (i) lack of incentive mechanism, (ii) Access control, (iii) Trust and privacy issues, and (iv) lack of matching model for requesters and providers. Finally, we discussed the basics of blockchain technology that is used in this thesis to provide security and privacy to CCN network.

Chapter 2: Literature Review

This chapter discusses the existing work on the content delivery in CCN and and its potential solutions. We discuss the security attacks faced by CCN network along with discussing its available solutions. Next, we provide a survey and taxonomy for access control solution in CCN. In addition, we present the discussion on the available content selection schemes for caching including co-operative caching, probabilistic caching, incentive caching and popularity based caching. Finally, we discuss all blockchain based solutions for providing trust and security in CCN along with providing some open challenges faced while integration of CCN

with blockchain.

Chapter 3: Blockchain-Based Co-operative Caching for Secure Content Delivery
In this chapter, firstly to reduce the load on LAG, we propose in-network caching for fast content delivery in V2G network. Secondly, we propose a contract theory based incentive scheme to attract vehicles to participate in content delivery to the recipients. Thirdly, for secure content delivery and to achieve trust in the network, we propose the proof of authority (PoA) consensus algorithm among vehicles in V2G network. We analyzed the proposed scheme using various performance evaluation metrics compared to the existing state-of-the-art solutions.

Chapter 4: A Blockchain based Matching Game for Content Sharing
In this chapter, we model the two-sided preferences of both parties (service providers and requesters) to maximize the cache content sharing using CCN communications. Then, we formulated a decentralized matching problem with joint transmit power of both the content providers and requesters. The framed matching game involves aspiration level and agreement functions for both the parties. Finally, a distributed blind matching algorithm (BLMA) is also proposed, which is executed by deploying a smart contract on the Ethereum network without involving an intermediate authority. To verify and validate the effectiveness of the proposal, we compare it with three benchmark schemes and evaluated its performance with respect to average individual utility, latency, and energy consumption on the benchmark data sets.

Chapter 5: A One-to-Many Ciphertext-Policy Attribute-enabled Blockchain Access Control Scheme
In this chapter, we propose a ciphertext-policy attribute based encryption (CP-ABE) scheme to secure one-to-many data sharing in which blockchain is used for maintaining verifiable access policy records and to ensure non-repudiation. In the proposal, instead of one, multiple attribute centers are used for attribute management and secret key distribution thereby; a SPoF can be mitigated. In order to generate the attribute secret key and to share master parameters among blockchain nodes, Pedersen (t, n) threshold secret sharing is used. Also, to verify the authentication of secret key shareholders, Bloom filter is used, which provides efficient storage and searching on the blockchain network. Moreover, a content revocation method is also designed if a content producer no longer wants to provide access to content. The proposed scheme is evaluated on benchmark datasets using several performance evaluation parameters. Security analysis demonstrate that the designed scheme is superior in comparison to the existing benchmark schemes when compared with metrics such as: encryption time, decryption time, key generation time, computational overhead, and storage overhead.

Chapter 6: Conclusion and Future Scope
This chapter concludes the research work by discussing the contributions in the proposed

schemes. Also, in this chapter the future directions in the field of CCN is discussed. In the future, we will deploy the proposed solution for large-scale data sets to assess the scalability and efficiency of the proposal. Also, we will evaluate the efficiency of the proposed scheme with an increase in the number of EVs with high mobility.

1.4 Summary

In this chapter, challenges of CCN are discussed. The research contributions focused on following three problems: (i) Incentive mechanism for caching (ii) Matching game for requesters and providers, and (iii) Access control in CCN. In next chapter, existing work on content caching and delivery in CCN with its potential solution has been discussed.

Chapter 2

Literature Review

With the growing popularity of SG and EVs, the V2G network is a topic of interest for both industry and academia. The V2G concept was discovered by Kempton and Lextendre in 1997 [51]. In future, more EVs will be in the market, and these can be used to change the behavior of load curves for demand-side management. Generalized survey articles covering advancement, working, terminologies, and advantages of V2G are presented in [52], [53]. Despite having several benefits, the huge deployment of V2G still posses several challenges. Issues and challenges faced by V2G technology are discussed in [54]. There are significant proposals in the literature to deal with existing challenges. For example, many solutions are proposed to find an optimal charging and discharging schedule [55], [56], [57]. In the work by Karfopoulos et al. (2016), a cooperative mechanism for EV distribution was suggested. This mechanism not only facilitates the effective supervision of charging and discharging activities but also provides V2G regulation services to aid grid operations. Also, Pal et al. (2018) introduced an energy scheduling approach in their work, as documented in [58]. This approach is centered around neighbor connections and delves into energy trading between vehicles and homes (vehicle-to-home) as well as V2G interactions, aiming to lower household electricity expenses. Most of the current works is on EV charging and discharging management [7], [59], [60] while the security and privacy issues in V2G have been ignored. In contrast, the authors in [61], [62] surveyed the advantages, drawbacks, challenges, and opportunities of CCN architecture. Also, authors in [63] investigated the role of CCN in vehicular networks including name resolution, routing, application, and open challenges. Similarly, authors in [64] explored the existing specification, security mechanisms, solutions, and simulation tools used in NDN-based vehicular networks. Next, we will discuss some issues and their solutions that prevails in CCN network.

2.1 Access control in CCN

As the network nodes usually download cached content from neighbor nodes, some security issues should be addressed during content delivery. One such security attribute to be consid-

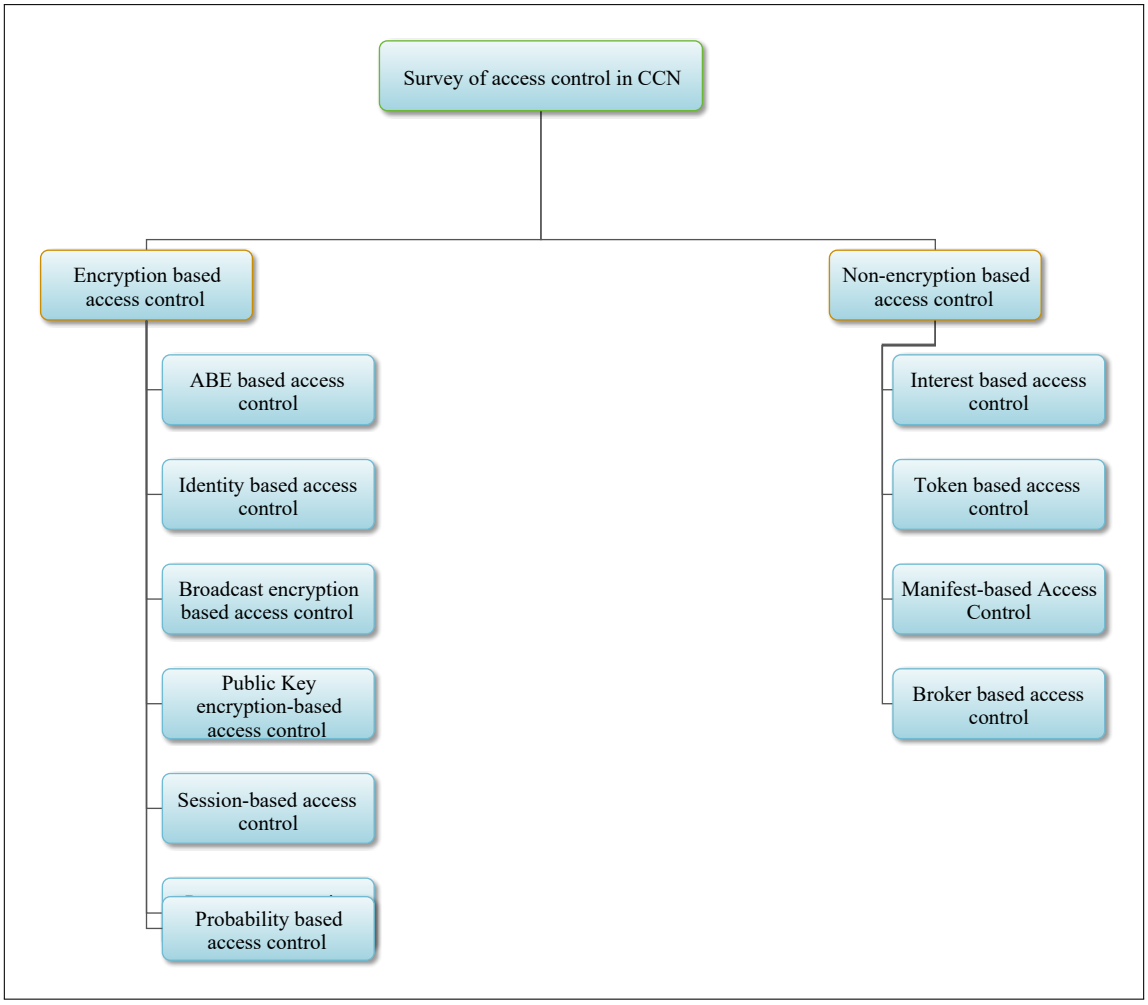

Figure 2.1: Taxonomy of access control in CCN

ered is access control. Utilizing in-network caching services, any request can be satisfied by the nodes in the forwarding path. Once the content is forwarded in the network by the original producer, it can be stored and forwarded by the cached nodes without enquiring whether the requester can access the content or not. Content producers don't possess any control over the caching nodes. Hence, any unauthorized vehicle can fetch the content from the network without having access permissions. An effective access control scheme should guarantee privacy preservation, integrity, data confidentiality, and mobility. In the existing access control scheme, the nodes where the cache hit occurs start the checking process to make an access decision. However, this process involves heavy computational overhead as it is initiated every time a match occurs, and it requires interaction with the content producer or the server. Moreover, frequent computations lead to latency that degrades in-network cache performance. Also, each content chunk has to pass multi authentication check to retrieve the complete file, as in CCN, a content file request is served by multiple routers.

In order to tackle the access control challenge in CCN, much literature work based on encryption has been proposed. However, with traditional symmetric and asymmetric encryp-

tion techniques, if any authorized user key gets locked, the content producer can't protect the confidentiality of data [65]. Also, in the existing Internet scenario, the service request is handled by a centralized content provider that decides whether to grant content access. Also, if a network has a single main server for downloading content, it can't handle the simultaneous access request for content as it may cause overloading and increase data latency. In comparison, in decentralized access control schemes, no extra entity is employed for access control. Network nodes together take responsibility for access control and mutual authentication. To implement decentralized access control schemes, many security techniques are used, for example, PKI, identity-based cryptography, shared secret distribution, etc., but all have high computation costs. A number of recent papers in the literature have studied DDoS attacks, authentication, and privacy issues in CCN. However, only a few efforts have been made to evaluate access control in CCN environment. Next, we will review the pros and cons of access control schemes available in CCN-based proposals. The taxonomy for access control in CCN/NDN is presented in Fig. 2.1.

2.1.1 Encryption based access control

A natural way for access control models is data encryption. In encryption-based access control strategy, the original content generator encrypts the content before forwarding it into network. The requester of the content should have the decryption keys in order to decrypt and have the requested content. As compared to asymmetric key encryption, symmetric key has a low computation cost for encryption. This type of access control generates overhead for content providers, receivers as it demands some data for key generation to be distributed along with content. Moreover, this scheme lacks flexibility if the content is to be modified by an intermediate application, such as media encoding. Next, the discussion delves into subcategories based on the specific encryption methods employed, and the summary of these solutions is discussed in Tables 2.1 and 2.2.

2.1.1.1 Attribute-based access control

Within attribute-based access control, the encryption of content is executed using a predefined set of attributes. Each consumer has a key that is produced from its attributes. The consumer can receive the content if it poses the attributes to decrypt access policy. The key characteristic of ABE is that it can support access control to a group of users. Attribute-based access control further has two types, i.e., key-policy ABE (KP-ABE) and ciphertext policy ABE (CP-ABE). In the former type, secret keys are produced from access policies that specify which ciphertext to be decrypted and how the ciphertext is linked with attribute sets. In later type, the access policy is designed as a tree with an attribute set along with logic gates. All network users obtains a secret key from the registering authority determined by their attributes. Authors in [66] proposed an access control method that can use both KP-ABE or CP-ABE.

However, the client revocation process in this scheme is not described properly. In ABE based access control scheme proposed by [67], a third party is employed to manage the attributes of network participants. Moreover, to manage the distributed attributes at the upper level, an ontology-based solution is presented. This scheme has low computational overhead. However, if the symmetric key gets compromised, the content name has to change with re-keying, which poses a challenge to the fundamental concept of immutable naming in ICN. Authors in [68] also used ABE to implement privacy preservation in ICN. This approach uses a third party to manage the attributes of nodes. The content provider uses attributes to specify access permissions along with generating a random symmetric key to encrypt. However, due to involving a centralized entity, this approach can suffer from SPoF and DDoS attacks. Similarly, authors in [69] used ABE for vehicular networks. This scheme doesn't rely on the continuous availability of any authority and can support high mobility. However, the authors have not discussed how attributes and policies are exchanged. To support access revocation, authors in [70] proposed an access control mechanism. The proposal has a proxy server to control access to the content, which needs to be online and can be a SPoF. To enable privacy-enhanced ICN, authors in [71] proposed a multi-authority ABE to provide access control. The authors suggested using a hash function on the interest packet to preserve information security. The proposal is compatible with heterogeneous mobile environments having trust from various independent sources. However, the overhead generated by hashing of interest packets increases the delay in content retrieval.

In most existing literature work related to CP-ABE, data access control operations are handed over to a third party, like cloud server or attribute management authority. However, they are always assumed to be trusted, which is not practical and might suffer from issues including SPoF, DDoS, privacy exposure, and high trust building costs. Moreover, these proposals use a centralized key generation server for key issuance and management. However, ABE based access control scheme is not scalable for client revocation.

2.1.1.2 Identity-based access control

This type of encryption is also based on principles of PKI. In Identity based encryption (IDE), public keys are generated directly from the identities of users without the certificate, hence having less cost of public key management. In this type of access control, clients are authenticated using their identity. In this context, Hamdane et al. [72] introduced a credential-centric access control framework tailored for NDN. In this proposed system, access privileges are allocated to consumers based on authenticated credentials issued by an Access Control Manager (ACM). The ACM, functioning as an entity managing the private key(s) of the network namespace, establishes access and management regulations. However, the reliance on this central entity raises scalability apprehensions, particularly in expensive network settings. Additionally, the system mandates a dedicated ACM for each namespace, posing an open question

regarding communication between ACMs in the context of distinct namespaces. Also, authors in [73] proposed a fine-grained access control method that enables confidentiality and mobility in NDN. However, this scheme assumes that online shops do not cogitate with each other. Similarly, authors in [74] present an identity-based access control for NetInf. Nevertheless, this scheme has a high communication overhead.

2.1.1.3 Broadcast encryption-based access control

Broadcast encryption-based access control enables the content generator to encrypt the content with a single key for all the consumers, and consumer has their keys for decrypting content. Broadcast encryption-based access control aims to achieve distributed access control for a bunch of users. In a broadcast encryption [75] scheme, the same message is broadcasted to multiple receivers such that only legitimate receivers can decrypt them. The main aim of broadcast encryption is to lessen key management-related transmissions. Authors in [38] have presented a broadcast encryption-based two-layer access control process. Here, a different group of users has different access permissions on content. Authentication is performed at edge routers so that network resources are only utilized by authorized users. However, the authors of [38], only encrypting the content still opens up the chances of DDoS attack in the system. Similarly, authors in [31] designed a broadcast encryption-based access control scheme for power-constrained devices. As an advantage, this scheme does not demand any authorization entity along with providing reduced latency and high availability of content. Unfortunately, with this scheme, the verifier needs to be in an online state. However, this scheme has to bear a minor updating cost for client revocation operation and it has a limitation on the number of revoked users.

2.1.1.4 Public Key encryption-based access control

If a content producer has to encrypt a file, it does not have any idea about the public Key used to encrypt content as producers don't know who can be potential consumers for the content. One solution is to use symmetric encryption having third party server. Traditionally, to implement various security operations, public key infrastructure (PKI) is used in the network. Authors in [72] have used PKI and identity-based cryptography to implement data-oriented access control with a hierarchal tree-assisted content naming. However, this scheme lacks implementation details. Also, as this proposal creates encrypted key chains, so revocation operation leads to overhead for this procedure. Moreover, the entity having only read rights can also write under a protected namespace. In order to overcome the limitation of [72], authors in [76] present a credential-based access control scheme that does not demand any prior knowledge of the network nodes. However, similar to [72], the authors do not discuss the revocation process. Authors in [77] have used mandatory content access control (MCAC) to enforce content flow control in ICN. Here, contents are assigned various labels as per the requirements of content providers. Similarly, Abdallah et al. [30] introduce a decentralized access control scheme by

leveraging a self-certifying naming scheme. Access decisions are made by ICN nodes, and this proposal reduces the number of public messages. This scheme doesn't provide any implementation details and requires the content provider to be online always. Also, Li et al. [78] presents a scheme named "LIVE" that provides signature verification on data with a lightweight algorithm. In particular, integrity verification tokens are allocated to authorized users only. The content producer combined a Merkle hash tree and key vector to generate hash-based signatures. While the proposal offers faster signing compared to traditional RSA methods, nodes are required to synchronize with the content producer to obtain a compatible version of the token. However, PKI is not suitable for CCN networks as operations in PKI generate significant overhead while content forwarding [79].

2.1.1.5 Session-based access control

In this type of access control, a secure session is established between the client and content provider after authentication of the client. Session-based access control in ICN was first proposed by Wang et al. [80]. Here, authors have combined both symmetric and asymmetric encryption along with the concept of dynamic naming for privacy protection of content names. The proposal is represented using an online social network example. However, the content access overhead for this proposal is high as the process has to be executed again and again for each content.

2.1.1.6 Proxy re-encryption based access control

Proxy re-encryption is another well-known method for secure content delivery. Proxy re-encryption is first introduced by Blaze et al. [83] in which a proxy can decipher a ciphertext encrypted with one node public key without revealing their private Key. Authors in [29] have used the proxy re-encryption method to ensure data confidentiality and identity-based signatures are used to achieve anonymous authentication. Also, to ensure privacy, authors have applied encryption based name obfuscation on interest packets. As an advantage, this scheme allows nodes to perform batch verification for signature. However, the decryption time for this scheme is relatively high when compared to the scheme in [38]. Similarly, authors in [81] also used the data re-encryption technique to present a complete file transfer protocol. As compared to other schemes based on proxy re-encryption, this scheme saves storage costs in NDN nodes. However, routers have to implement re-encryption on every content packet, which makes it less efficient. In another approach Wood and Uzun [83] use identity-based proxy re-encryption for access control in CCN. One of the benefits of this scheme is that it minimizes the quantity of shared messages required for both authentication and key retrieval. Also, it has less cryptography cost as here, only the symmetric key has to be encrypted instead of the content. However, if the content provider is unavailable, the content is unavailable to all the requesters. Authors in [82] combined proxy re-encryption and all-or-nothing transformation-

Table 2.1: Comparative analysis of existing surveys articles on Content centric network (encryption based).

Authors	**Year**	**Encryption technique**	**Authentication technique**	**Simulator**	**Metric used**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
[29]	2020	Proxy re-encryption	Identity based signatures	ndnSIM	Computation overhead, communication overhead	✓	×	×	✓	✓	×	×
[38]	2019	Broadcast encryption	Hash chain and group signature	NS3, ndnSIM	Computation overhead, average content retrieval delay, signature verification time	✓	✓	✓	✓	✓	×	✓
[73]	2018	Identity based encryption	Identity based encryption	—	Storage cost, encryption cost, decryption cost	×	×	×	×	×	×	✓
[77]	2015	PKI	—	ndnSIM	Forwarding delay	×	×	×	✓	×	×	×
[31]	2017	Broadcast encryption	Broadcast encryption	ndnSIM	Latency, computation time	×	×	×	×	✓	×	×
[69]	2017	ABE	Signature based	—	Security analysis	×	✓	×	×	✓	×	✓
[81]	2016	Proxy re-encryption	Identity based signature	—	Computation cost	×	×	×	×	×	×	✓
[30]	2016	PKI, random number generation	Digital signature	—	—	×	✓	×	✓	×	×	✓
[82]	2015	proxy re-encryption	—	—	encryption time, decryption time	×	×	×	×	✓	×	×
[70]	2015	CP-ABE	—	GNU/LINUX System	Decryption, encryption time, key generation process, memory usage	×	×	×	✓	✓	×	×
[83]	2014	Proxy re-encryption	Identity based cryptography	ndnSIM	Computation time	×	×	×	×	×	✓	×
[68]	2014	ABE	—	—	Computation consumption and communication overhead	×	×	×	✓	✓	×	✓
[84]	2014	Probability based access control, symmetric key encryption	Bloom filter	PyNDN	Latency, system resource consumption	×	×	×	×	✓	×	×
[72]	2013	PKI	Identity based cryptography	—	—	×	×	×	×	✓	×	×

Note- 1: Anonymity, 2: Replay attack resistance, 3: DoS resistance, 4: Privacy protection, 5: Revocation, 6: Delegation, 7: Integrity, ✓: considered, and ×: not-considered.

based access control for ICN network where nodes need to ask the content producer for the decryption key of the requested data. However, the information related to the revocation of customers is broadcasted to all network routers, which proves costly and inefficient. Never-

Table 2.2: Comparative analysis of existing surveys articles on Content centric network (encryption based).

Authors	Year	Encryption technique	Authentication technique	Simulator	Metric used	1	2	3	4	5	6	7
[71]	2015	CP-ABE	Cryptographic signatures	NRL's CORE 4.3	Encryption time, signature time	×	×	×	✓	×	×	✓
[74]	2015	IBE	Identity based cryptography	—	—	×	×	×	×	×	×	✓
[85]	2015	Proxy re-encryption	Signature based	Java prototype	cache hit rate	×	×	✓	✓	✓	×	✓
[80]	2014	PKI	User name/-password	—	—	×	✓	×	✓	✓	×	×
[78]	2014	PKI	Token based	CCNx	Communication overhead, verification delay	×	×	×	✓	×	×	✓
[66]	2013	ABE	—	NDF 0.3.1 on Ubuntu12.04	Time consumption for data transfer, encryption time, decryption time	×	✓	×	✓	×	×	×

Note- 1: Anonymity, 2: Replay attack resistance, 3: DoS resistance, 4: Privacy protection, 5: Revocation, 6: Delegation, 7: Integrity, ✓: considered, and ×: not-considered.

theless, the usage of resources provided by edge routers affects the scalability of the proposal as in the coming time Internet traffic is going to increase. Another proxy re-encryption-based access control is provided by [85] where the content is broken into fragments and encrypted in two layers by providers. A content provider encrypts the content chunk with a symmetric key that is kept in encrypted chunks. The second layer is used for collusion attack prevention using a key derivation algorithm. Unfortunately, clients have to perform authentication at each content fragment and require the provider to be online always. Also, clients may obtain a set of fragmented chunks each encrypted with a different Key. This requires requesters to download all keys, which increases the overhead of the system. However, as per the authors of [82], proxy re-encryption is unsuitable for CCN/ICN as it involves key management overhead (for maintaining re-encryption key) and communication overhead for routers that act as a proxy.

2.1.1.7 Probability-based access control

In this type of access control scheme, probabilistic data structure bloom filter [86]are placed in the network to check the authorization of client's public key. In this context, Chen et al. [84] presents an encryption combined probability access control method in NDN. Encryption of the content is done with the content generator. Here, bloom filter is used to prevent unauthorized access to encrypted video content. Using bloom filter significantly reduces bandwidth usage for transferring encrypted video content to unauthorized users. However, for experiments, the bloom filter taken by authors is quite high as compared to total users which generates system storage overhead. This scheme has another drawback in that the content provider has to be

online in order to check the credentials of client.

Summary: *Attribute-based access control supports access control for a group of users, but most approaches in the literature involve a centralized entity that affects scalability of the system. Hence, future work may involve proposals that intensify the scalability of attribute-based access control. Also, re-encryption based approaches provide the facility to update access control rules; however, they may suffer from scalability issues as they demand more than two cryptographic operations for authentication and updation. Moreover, broadcast-based access control supports access to a group of nodes, but these approaches have high overhead. Hence, further research is necessary to optimize encryption-based access control in CCN.*

2.1.2 Non-encryption based access control

An access control solution is termed encryption-independent when the establishment of access rules is done without relying on any specific underlying encryption. The reviewed schemes under this category are further categorized to subgroups based on the method of rule definition, and Table 2.3 provides a summary of these classifications.

2.1.2.1 Interest-based access control

Interest-based access control involves attaching information to an request packet. Hence, access control can be achieved for stored content. For example, authors in [87] presented another scheme where access control is executed only by using information from an interest packet. In particular, authors have used name obfuscation and authorized disclosure to achieve interest-based access control. However, with obfuscating content names content can be cached repetitively thus leading to wastage of resources.

2.1.2.2 Token-based access control

In token-based access control policies, a token is released if the access policies match the requester's attributes. Authors in [88] presented a capability-based architecture that achieves access control with tokens in packets. A capability acts like a ticket that depicts the access rights on the content packet. This scheme rejects any packet without the required capabilities, thus providing defense against DoS attacks.

2.1.2.3 Manifest-based Access Control

Manifest-based access control provides a different file specifying access rules, thus decoupling original content from access rules. This kind of access control provides minimum communication overhead. For example, authors in [89] proposed an access control scheme that

is based on CCN manifest feature. Group-based and broadcast access control are used as sample access control schemes to check the flexibility of the proposal. However, the suggestions of using lazy revocation can create overhead for the access control scheme. Also, both of the original generator and requester need to be connected simultaneously.

2.1.2.4 Broker-based access control

Here, a third party is employed as a broker to validate access rules. For example, Fotiou et al. [90] propose a delegation-supported access control scheme where, hosting nodes can check a request for content against an ACP without the access policy itself. The scheme separates the role of each stakeholder. However, the use of rendezvous nodes brings extra communication overhead and response latency along with scalability issues. Also, the use of a trusted third party can lead to a SPoF. In another approach, Sapna Singh [91] proposed a trust-driven scheme to achieve secure access control on data. The scheme includes a broker that registers each new client and uses its credentials, which establish trust between client and broker. The content producer defines an ACP and stores it in the broker database. However, the authors did not discuss revocation and communication overhead. Also, authors in [65] introduce a split data approach for copyright protection in NDN. The large-size data is cached in routers with a unique portion for each authorized user. Any requester can retrieve the large-sized portion from cached routers; however, only authorized users can retrieve the small portion. The scheme is analyzed with respect to data retrieval efficiency and overhead. The weakness of this scheme is that it requires an online provider every time to grant access to a request. Similarly, Zhu et al. [92] proposed a lightweight access control using the notion of content subscription times. This scheme uses three cryptographic techniques, i.e., proxy re-encryption, identity-based, and broadcast-based to encrypt and send it to distribution servers. However, this approach is not scalable.

Summary: *Non-encryption-based access control is efficient and has a simple design but has overhead on the network as it involves a broker and the network relies on the security of the broker. Hence, methods to find security compromises of brokers are required. Token-based access control provides a facility to control access by assigning tokens but managing these tokens is difficult and demands future research. Also, manifest-based access control reduces communication overhead but has scalability issues. Moreover, interest-based access control obfuscates content names but results in redundancy of same content stored at different caches with multiple names. This problem can be solved by using an intelligent cache management strategy that detects multiple copies of the same name.*

Table 2.3: Comparative analysis of existing surveys on Content centric network (non-encryption based).

Authors	Year	Access control technique	Authentication technique	Simulator	Metric used	1	2	3	4	5	6	7
[92]	2020	IBE, proxy re-encryption, and broadcast encryption	Identity based	—	cost performance, communication performance	×	×	×	✓	✓	✓	✓
[88]	2017	Capability based security enforcement	Lightweight one time signature	Thousand LOC in CCNx	Token generation delay, verification delay, data retrieval delay, communication overhead	×	✓	✓	×	×	×	✓
[87]	2015	Name obfuscation	—	—	Computational overhead,storage overhead, bandwidth overhead	×	✓	×	×	✓	×	✓
[65]	2014	Split data based approach	—	—	Data retrieval overhead and time delay	×	×	×	×	×	×	×
[90]	2012	Use of rendezvous nodes	—	Blackadder [93]	Communication overhead	×	✓	×	✓	×	✓	✓

Note- 1: Anonymity, 2: Replay attack resistance, 3: DoS resistance, 4: Privacy protection, 5: Revocation, 6: Delegation, 7: Integrity, ✓: considered, and ×: not-considered.

2.2 Content selection schemes for caching in CCN

Multiple models have been introduced to solve the problem current Internet architecture faces. CCN is considered the most significant as it overcomes the major issues in IP-based scenarios. Caching content at multiple nodes lowers the communication and searching overhead. Each CCN nodes have comparative power to conduct matching process for Interest packet received and cached content. The caching plane of CCN tends to cache content chunks at some nodes, so requesters can fetch content from the nearest neighbor rather than the original source far away. However, one of the issues with CCN caching is the selection of nodes to cache content that can provide content with minimum delay, low congestion, and low bandwidth consumption. Different entities can be selected to cache contents. Choosing entities effectively to reduce access delay is important in this context. Also, once caching nodes are chosen, how to select which content to cache for fast access? Hence, researchers are working to find the optimal place to cache the most efficient content.

The traditional caching mechanism in CCN is based on "Cache everything everywhere." However, this policy creates congestion on caching nodes with limited cache capacity. The aim of content selection for storing is to improve hit ratio of request packets, hence increasing network performance for load, delay, and interest satisfaction. This section will review existing

caching selection strategies proposed in the literature. Fig. 2.2 depicts the taxonomy of content selection scheme for caching in CCN.

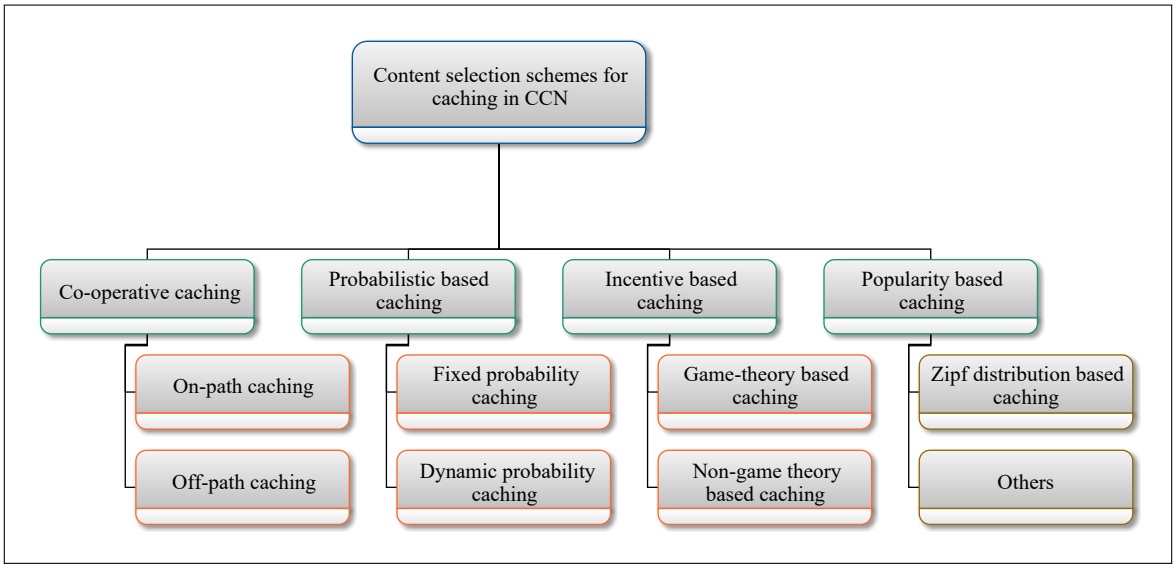

Figure 2.2: Taxonomy of content selection schemes for caching in CCN

2.2.1 Cooperative caching policies

The objective of cooperative caching is to enhance the hit ratio, diminish network traffic, and to decrease delay with cooperation into caching nodes. In cooperative caching, all caching nodes collaborate to fasten the transfer, whereas, in a non-cooperative scheme, each caching node makes content selection decisions independently in the network. In a non-cooperative scheme, the same named content may get stored among multiple nodes in neighborhood, leading to low cache utilization. Hence, non-cooperative caching creates the problem of frequent cache updates and full cache. As per the research in [94], cooperative schemes are significantly more beneficial than non-cooperative schemes and no caching schemes. Cooperative caching schemes are further classified as on-path and off-path caching. In former type caching, content is stored at the node along the reverse path from demander to provider, whereas in later caching, content is stored without considering the data forwarding path. On-path cooperative caching avoids multiple nodes storing the same content chunk at the same location in a particular area, minimizing repetitions. Table 2.4 shows the summary of proposals having co-operative caching policies.

In this context, to tackle the issue of large multimedia file sharing, first, authors in [95] have presented a cache node selection scheme using a minimum-coverage-set to initialize the cache. Next, cooperative caching based on social relationships for ICN network is presented. Simulation results show that the proposal improves content retrieval and playback smoothing. However, it is essential to continuously compute vehicle distribution for ongoing feasibility assessment, which is a difficult task. Similarly, Quan et al. [96] explored social cooperation

among nodes. Using node location and lane data, the proposal finds the social relationship between vehicles and classifies them as partner-assisted and courier assisted. However, the scheme does not suits vehicle with high speed. Also, using the social attributes of nodes can lead to privacy leakage. To solve this problem, Bai et al. [97] used federated learning to obtain privacy of social attributes. Also, deep reinforcement learning is used to achieve optimal caching decisions. Social strength is calculated using interest similarity and social trust. Simulation results validate DRL convergence and performance of the caching scheme. Authors in [24] proposed another cooperative caching scheme using mobility prediction in vehicular CCN. This scheme aims to cache popular content in mobile nodes that repeatedly move around the same areas. Nodes having a long sojourn time in any area can offer more caching services. To capture mobility patterns, the partial matching prediction has been used. The final caching decision is taken using both predicted trajectories and the content's popularity. Here, the road is divided into multiple regions using historical vehicle trajectory.

Combining above-discussed schemes, Yao et al. [98] proposed a caching scheme using social attribute and mobility prediction. However, the proposal has high energy consumption. Here, the social similarity between the requester and provider node is used as a parameter to decide caching node, as nodes with similar interests can likely meet each other in the future. To find the probability of a vehicle reaching to destination, historical trajectory is calculated using a hidden Markov model. Also, authors in [99] proposed that entities will only take part in collaborative caching if they can have some profit from collaboration. Here, the nodes cooperate if the content access cost is decreased by obtaining data from the local cache or neighboring nodes.

Differently, Dua et al. [100] used deleted bloom filter to propose a content cache policy that reduces communication costs and minimizes operational time. In this scheme, vehicles are divided into two categories, i.e., user and helper (based on roles); and content is divided into multiple pieces that are assign to helper nodes based on popularity. However, the authors have ignored the false positives of bloom filter. Also, authors in [101] have proposed cooperative caching using trajectory prediction and consistent hashing. Nodes are classified using their similar future trajectory, and nodes with good bridging centrality are selected to act as cooperative ones. Next, the process of consistent hashing is used to allocate content. To reduce energy consumption, authors in [102] proposed a cooperative caching mechanism using node and content value. Here, the content chunk having a lower content value (low popularity) is replaced in the cache. The proposal is compared with another standalone scheme using parameters cache hit ratio and packet retrieval delays. However, this caching scheme doesn't support the mobility of nodes. In a similar way, to save energy, authors in [103] proposed a cooperative cache where low-powered devices periodically broadcast their contact to the neighbor node. The aim is to save battery, making a few IoT nodes active while others can sleep. However, the scheme in [102] fails to handle high traffic in the network. To tackle this issue, authors in [104] proposed a cooperative congestion control scheme to improvise the QoS of the network. The

proposal also optimizes the multipath forwarding strategy to reduce the utilization of optimal paths. The downstream and routers cooperate to reduce network congestion by setting congestion window size and shifting traffic to alternative paths. Differently, Sellami et al. [105] combined CCN with fog computing to increase the performance of the network. Content popularity and size of content determines where to cache. Also, if the popularity of content is high, it is placed on sensor nodes to avoid delay. In another work, Liu et al. [106] used CCN for high-definition maps required for autonomous driving as this application has rigid requirements of latency. The authors defined joint edge caching and request rescheduling as an integer linear programming problem. The work in [107] presents a multicriteria caching scheme for IoT networks. Here, the devices are clustered, and the head of each cluster acts as the controller. Also, one global cache controller is used to make cache decisions for the whole network. To select caching content three parameters are selected, i.e., freshness, hop count, and size of content. It is shown with simulation results that the proposal has 40% more cache hit.

Summary: *To propose cooperative caching, the most recent work includes using historical trajectories to obtain future paths. Also, social similarities are considered to select appropriate caching nodes. Nevertheless, information exchange among vehicles with sufficient resources is required to implement cooperative caching. The information data include driving speed, bandwidth, state-of-charge, and node distribution. This information exchange may affect the privacy of data, so future work may include work on privacy preserving cooperative caching schemes.*

2.2.2 Incentive based caching policies

Every node of the network act as a potential source of content. However, in a network, nodes are selfish and only care about their own preference. Nodes only cache the content they like the most, leading to redundancy and improper storage space utilization. Also, some nodes in the network may not want to participate as content provider in a CCN network because of their privacy concerns or energy consumption. To deal with this issue, incentive-based caching method works as a solution as this caching policy assumes that nodes are driven by profit [109]. CCN network requires cooperative caching to achieve a high-performance network; it is also important to regulate an incentive mechanism active to participate in caching process. Table 2.5 discusses the comparative analysis of incentive based caching proposals

Moreover, Xu et al. [110] proposed another incentive scheme using Stackelberg game model. However, nodes' mobility is not considered, making them unsuitable for mobile communication. Authors have concluded that the order of prices of content chunks follows the reverse order of popularity. Similarly, authors in [111] proposed a popularity incentive caching scheme where base station rewards vehicles that perform cache offloading and content sharing with others. Authors have used Stackelberg game theory for interaction among rational utility. Also, Chen et al. [112] presents an incentive mechanism where nodes that cache content or

Table 2.4: Comparative analysis of cooperative caching proposals

Reference	Year	Application	Cooperative caching basis	Comparison metric	Implementation platform	Merits	Demerits
[104]	2023	Generic	Adjusting size of congestion window	Throughput	ndnSIM	High throughput, low latency, no packet loss	Mobility of nodes not considered
[97]	2023	D2D	Social relationship	Mean absolute error, mean absolute percentage error	—	Improved overall offloading, reduced average delay	Collaborative computation offloading is ignored
[108]	2022	Mobile vehicles	Popularity of nodes	Average latency, average hit ratio, average energy savings	OMNET++	No extra communication heads	Scalability of network is not evaluated
[101]	2022	VCCN	Mobility prediction and consistent hashing	Cache hit ratio, access latency, hop count, average delivery ratio, communication cost	ONE	Lower cache redundancy, improve delivery ratio	Cache hit ratio decreases with increasing speed
[105]	2022	IoT	Popularity and content size	Energy efficiency, user satisfaction rate, cache hit, delay	CCNx contiki framework	Reduced bandwidth consumption, energy efficient	Not suitable for big data size content
[102]	2021	IoT	Network topology, content value	Energy consumption, Average delay, cache efficiency	TOSSIM	Reduced enery consumption, data retrieval delay	Node mobility is not considered
[106]	2020	Autonomous vehicle	Incentive	Latency	—	Minimixed retrieval latency	Mobility factor not considered
[98]	2017	V2V	Social interactions	Average access delay, success ratio, average hop count, average storage usage	ONE	Higher cache hit ratio, low access delay	High energy consumption
[107]	2019	IoT	Freshness, hop count, content size	Cache hit rate	Omnet++	Higher cache hits, lower hop count, faster retrieval	Cache replacement is not considered
[100]	2019	VCCN	Relative distance and velocity between nodes	Time elepsed between request and receipt of packet, cache hit ratio	ndnSim	Efficient cache retrieval	False positives of bloom filter are ignored
[24]	2017	VCCN	Mobility pattern	Success ratio, latency, storage overhead	ONE	Higher success ratio, lower latency	Mobility prediction error is not discussed.
[103]	2017	IoT	sleep/activity ratio	Data availability, energy consumption	RIOT extending the CCN-lite library	Reduced energy consumption, improved hit ratio	Mobility of nodes not considered
[95]	2015	VANETs	Socialized relations	Waiting time, utilization of play buffer	ndnSim	Minimized playback freezing ratio	Vehicle distribution needs continuous updating
[96]	2014	VANETs	Socialized relations	Start-up delay, playback freezing ratio	ndnSim	Larger downloading bandwidth, reduced latency	High energy consumption
[99]	2013	Generic	Content access patterns	Hit ratio	PlanetLab	Reduced access cost	Intentional misbehavior of nodes cannot be detected

ONE: Opportunistic network environment

share it with other obtains a reward from the base station. Here, also Stackelberg game theory is used to characterize the base station and user terminal, and an iterative gradient algorithm is used to achieve equilibrium. Simulation results depict that proposal is better than no incentive scheme. However, in this scheme, mobility of nodes is not considered. Considering mobility information, authors in [113] proposed an incentive mechanism using caching storage require-

ments and taking congestion among different providers.

Also, authors in [114] presented a incentive mechanism along with a cache replacement policy for content in NDN. An incentive mechanism is proposed using auction theory, which uses frequency-based content size checking to refrain bidders from bidding along with sending content sizes different from required. Simulation results states that the proposed scheme improves both parties' cache hit ratio along with utilities. Similarly, authors in [115] presents another scheme for paid content using a reverse auction. The reverse auction mechanism aims to maximize the payment of ISP for retrieving the content. Also, authors in [116] introduced an incentive enabled edge caching for SDN supported IoV. Moreover, Xu et al. [110] used economic relations within ICN nodes and proposed an efficient incentive enabled cooperative caching scheme. The proposed framework is characterized as multiple-choice knapsack problem and is solved using suboptimal caching scheme. Authors have concluded that with increasing cache capacity, more content chunks can be cooperatively cached. However, no actual implementation of the proposal is given.

Differently, to maximize social welfare, Agyapong and Sirbu [117] proposed a work that evaluates economic incentives for a variety of network participants and the metrics that affect the incentives in an ICN network. Also, authors in [118] proposed a caching and pricing mechanism for ICN networks with using content popularity. The interactions with using content popularity among entities are modeled using game theory. In this proposal, the cost of caching is different for different popularity of the content, whereas the provider cost for each unit of content is fixed for all types. It is depicted using numerical results that intermediate nodes only cache content with more popularity, and less popular content is only served by the original provider. However, malicious behaviors of nodes are not considered in this work. Also, as rational entities control the network nodes, they can misbehave, which can affect network performance. Similarly, authors in [119] proposed a robust scheme for incentivizing that detects malicious behavior. This work has used a reputation-based incentive scheme in which the decision of taking or giving content to other nodes is taken by considering others' vehicle reputation. The proposal works in three phases: reputation calculation and evaluation phase, trust phase, and decision phase. To classify vehicles based on their behavior, Bayesian classification is used.

Summary: *The interaction between two parties is mostly characterized by game theory and non-game theory based proposals have less work. Additionally, it has been observed that there is a scarcity of proposals addressing the practical implementation of CCN in vehicular networks.*

2.2.3 Content popularity based caching policies

The popularity of content is reflected in the frequency of content requests originating from the requester side. Given that content is stored at various locations, content popularity policies

Table 2.5: Comparative analysis of incentive based caching proposals

Reference	Year	Application	Incentive model	Comparison metric	Implementation platform	Merits	Demerits
[111]	2020	VNDN	Stackelberg game model	Delay saved, BS burden saved, utility, impact of cache capacity	Numerical simulations	Reduced delay cost, considered mobility	Multi-hop forwarding not considered
[116]	2019	IoV	Stackelberg game	Players utility	MATLAB	Improved QoE	Local SDN controllers are not discussed
[114]	2018	Generic NDN	Auction theory	Cache hit ratio, zipf parameter	ndnSIM 2.1	Minimized delays, increased cache hit ratio	Complexity of this scheme is high
[119]	2018	Vehicular network	Reputation based	Detection time of misbehaving vehicle, robustness against false accusation	ONE	Reduced latency	Reputation system takes more time to converge
[110]	2017	Generic ICN	Stackelberg game model	Impact of content popularity, impact of content distribution on incentive policy	—	Efficient content sharing	Not suitable for mobile communication
[113]	2017	Small cell network	Stackelberg game	Total storage usage, average revenue	—	Efficient cache storage allocation	Impact of update frequency of popular content is ignored
[118]	2017	Generic ICN	Game theory	—	Caching threshold index vs zipf parameter	Improved caching ratio	Impact of malicious users are not considered
[120]	2017	Generic ICN	Sub-optimal caching scheme	Total caching capacity	—	Content retrieval cost saving	No actual implementation of proposals
[115]	2017	CCN	Reverse auction	Providers monthly profit	—	Content retrieval woth reduced delay	Mobility not considered
[112]	2016	D2D	Stackelberg game theory	Serving cost, effect of Zipf parameter	—	Higher average utility of user terminal	Mobility of nodes not considered

involve caching popular content in the store. This strategy aims to reduce content retrieval delay and raises the cache hit ratio. The content having higher popularity is cached first than low popular data. Table 2.6 represents the comparison of available popularity based caching policies.

In this context, authors in [121] proposed to cache only popular content that saves resource consumption and provides high caching performance. Here, each node notices the total number of request for each name and record it into a "popularity table." To prevent flooding of same content, the popularity count is reset. Also, authors in [122] present a caching proposal aggregating arrival rate and content request paradigm for vehicle parking systems. Considering these factors, the presence for content duplicacy is analyzed. Simulation results depict that the proposal reduces data access latency and improves hit rates. Similarly, authors in [123] present a popularity based caching scheme named "PopCC" by using the hidden markov model (HMM) to calculate content popularity. Service request ratio,number of requests, and priority of the content are the parameters used by HMM to calculate popularity.

Huang et al. [124] clustered nodes having similar mobility patterns in a group. In each group, the head is selected as caching node with priority. Next, caching node selection is called by RSU using mobility trajectory. A secondary header is selected as caching node if the expiration time between head node and LAG is greater than pre-fixed value. Also, based on request

frequency, the popularity of content is decided. Moreover, Wei. et al. [125] introduced a cache management strategy for vedio streaming in VCCN. Caching potential value is calculated by nodal geographic relation supported by video's popularity. Simulation results prove that the proposal improves the quality of experience and cache hit ratio. Also, authors in [126] used content popularity and cache distance to place content objects into the network. Simulation results reveals that the proposal achieves an better cache hit ratio and increased delivery rates.

Differently, Zhao et al. [127] proposed a dynamic probability approach by measuring community similarity and vehicles privacy. The caching decisions are made according to the popularity of the content and community similarity. In the data packet, a new field named "Interval" is added, which depicts the hop number and content that will be cached every other number of counts. The "Interval" value is related to content popularity that is calculated using request frequency. Also, authors in [128] proposed a dynamic approach that keeps on adjusting the content popularity threshold. If storage is available, the proposed scheme always caches content; otherwise, only popular content is cached in case of storage constraints. In a similar way, authors in [129] use a variable size popularity window to probe for popular content. Along with popularity, the content message navigates distance with reference from the prior on-path router that stores cached content. The threshold value decreases with increased distance navigated and content popularity. The proposal is evaluated based on QoS delivered for various cache sizes.

Summary: *Most popular content schema have slow convergence of hit rates, and the hitting rate is unstable for different cache sizes. However, to implement popularity-based caching, an accurate calculation of popularity is dependent on various factors such as- previous content request count, statistics of request pattern, etc. The proposed schemes should reduce the requirement for extra coordination and streamline cache management. Future work should work on improving cache replacement schemes that consider past event information and current cache insertion decision to design better replacements.*

2.2.4 Probabilistic based caching schemes

An important function of these caching selection algorithms is to cache a copy of content chunks near requesters to minimize the cost of further requests. A caching policy depicts whether to store a content chunk in its cache. However, an empty cache store gets full by caching such items. With an increase in node demands, the content in the network will be huge; however, having infinite space is not possible. Hence, a cache replacement policy is needed in such a scenario. Instead of following the always caching scheme (universal caching strategy), the entities in probabilistic caching caches content chunks using some pre-defined probability. The content chunks are cached using a pre-calculated probability in the data forwarding path. Universal caching is a special case of probabilistic caching where the probability of caching content in the data forwarding path is always one. The research in [130], [131], [132] depicts

Table 2.6: Comparative analysis of popularity based caching proposals

Reference	Year	Application	Probability calculation model	Comparison metric	Implementation platform	Merits	Demerits
[129]	2021	CCN	Zipf distribution	Hit ratio, average network delay, hop count	—	Communication overhead is low	Storage overhead is high
[123]	2019	Vehicular CCN	Hidden markov model	Success ratio, access latency, average hop count, storage overhead	ONE	Mitigate bandwidth pressure, reducing data access latency	Randomness of vehicle movements not considered
[124]	2019	Vehicular NDN	Zipf distribution	Average access delay, sserver request ratio	ndnSIM, SUMO	Improved user QoE	Computation overhead is high
[126]	2017	Vehicular network	Zipf distribution	Cache efficiency	OMNet++, SUMO	Improved resource utilization, increase cache efficiency, reduced network load	Cache replacement not discussed
[127]	2017	V2V	Zipf-Mandelbrot disribution	Average access time, cache hit ratio	OMNet++	Reduced cache hit distance, reduced average time delay, increased cache hit ratio	Cache replacement overhead is high
[125]	2016	vedio streaming	Zipf like	Average freeze time, average frame ratio, average bit rate	NS3	Improved cache hit ratio	Social cooperation among vehicles is ignored
[128]	2014	CCN	Pareto distribution	Cache hit ratio	OPNET modeler 1.0	Fast convergence speed	cooperative sharing among nodes is ignored
[121]	2013	CCN	MZipf distribution	Cache hit ratio, diversity, stretch, ratio of cached element	ccnSim	Improves caching performance, decrease resource consumption	Additional storage space is required

that the performance of probabilistic caching is better over universal caching when compared with respect to hit rate and hop distance. Probabilistic caching has further two parts, i.e., fixed probabilistic caching and dynamic probabilistic caching. In former type, a pre-fixed probabilistic value is set to store the requester's interested content in the delivery path. In latter type, caching-related decisions are dependent on cache capacity, traffic data, the popularity of the content, etc. Table 2.7 depicts the comparative analysis of probabilistic based caching.

In this context, authors in [133] studied the properties of probabilistic caching scheme along with know replacement methods, Least recently used (LRU), Least frequently used (LFU), and random replacement. The authors have concluded that the proposed probabilistic scheme works best with LRU policy. Also, the results suggest that the value probability should be decreased to distribute various content in the CCN network efficiently and to use in-network caching services better. In [134], authors have proposed a popularity-based probabilistic caching where it is decided whether to reuse content or not depending on the popularity of content and stores content having different possibilities. Also, Psaras et al. [132] introduced a scheme named ProbCache that considers distance content source and requesters and stores chunks at nodes closer to the requester with high probability, hence decreasing transfer delay. Using the amount of traffic it has to deal with in time; each intermediate router calculates the approximate count of copies of an incoming content chunk that the forwarding path can accommodate. Also, in [135] nodes

cache passing content with some probability depending on the content's popularity and content placement benefits. Simulation results depict that the proposal outperforms the proposal in [132] regarding cache hit ratio, data delay, and link bandwidth savings. It is also shown that the LRU replacement policy works well with the proposal.

Differently, authors in [136] proposed a hop-based probabilistic caching that aims to push content that is popular to a new edge. Simulation results show that the proposal is superior in terms of cache efficiency when compared with LRU, LFU, and ProbCache. However, for huge content, caching performance decreases. Also, in an attempt to maximize hit ratio, authors in [137] proposed greedy caching that decides the caching at each node based on content popularity. Using the request stream of directly connected nodes along with the miss stream of downstream nodes, the proposal caches the top most popular content of a node. In another scheme, [138], the probability of caching is decided by the combination of popularity and priority of content. The content priority is calculated from the quality of the content. Simulation results depict that the proposed scheme performs well in terms of cache hit ratio and load reduction. Similarly, authors in [139] used software-defined networking to make caching decisions efficiently. Here a controller is used to capture the network's topology and link state information. Matching node importance and popularity calculates the probability of caching content.

Above discussed schemes have not considered the mobility of nodes. In this context, Deng et al. [140] proposed another distributed probabilistic caching where decisions are taken by each node without any dependency. The caching decisions are taken for all nodes receiving content packets in some range rather than just nodes on the forwarding path. Probabilistic caching is designed using the mobility of nodes and dynamic topology. Simulation results depict that the proposal performs better over the universal cache strategy. However, the authors have not discussed any cache replacement policy.

Summary: *The probabilistic caching strategy caches content with pre-defined probabilities in different application scenes. Nevertheless, signal obstacles, and mobility of vehicles, lead to determining a pre-defined threshold with a Gordian node.*

2.3 Content sharing in CCN: a content requester and provider pairing

In CCN, a requester node generates an request packet having wanted content name and broadcasts it to the neighborhood. When a data provider gets the Interest packet with a content name, it transmits it back to the requester, and all the nodes on return path stores the received data packet. However, with multiple requesters and providers for content, requesters may face competition against the best sharing services, and providers must filter the incoming requests for low energy consumption. Also, in terms of the mobility of vehicles, the content duration is limited, leading to a limited transaction time while obtaining content. Hence, a matching model

Table 2.7: Comparative analysis of probabilistic based caching proposals

Reference	Year	Application	Replacement	Probability determination	Comparison metric	Implementation platform	Merits	Demerits
[139]	2016	CCN	LRU	Popularity and matching node importance	Cache hit ratio, request delay, content hop acquisition	—	Improved cache hit ratio and reduced request delay	co-operative caching is ignored
[137]	2018	Generic ICN	LRU	Popularity based	Latency, Hit ratio	Icarus	Low computation overhead	caching decisions are not optimal
[135]	2018	Generic ICN	LRU	Content popularity, content placement benefit	Hit ratio, access latency, communication overhead, computation overhead, link bandwidth saving	ndnSIM	Gives best results with small size cache	Mobility factor not considered
[134]	2017	Generic NDN	LRU	Popilarity	Cache utilization	ndnSIM	Efficient redundant traffic eliminator, better cache utilization	Cofficient choosing is not discussed
[140]	2016	VANETS NDN	—	mobility of nodes, dynamic topology	Hop count, hit ratio, delay	ndnSIM, SUMO	Improves cache utilization	No cache replacement policy is discussed
[138]	2016	Generic CCN	LRU	Popularity and priority of content	Cache hit percentage, server load reduction	—	offers best normalized information value	Impact of malicious entity is not considered
[141]	2015	IoT	LRU	Defer Time	Retrieval delay	ndnSIM	Reduced traffic volume, saves device energy resources	Computational overhead is high
[133]	2014	CCN	LRU, LFU, Random replacement	Random	Cache hit ratio, round trip hop distance	ndnSIM	Improves network performance over universal caching	Malicious behaviour of nodes not considered
[136]	2013	Generic ICN	LRU	Popularity	Hit ratio	C++	Improved hit ratio over LRU	Doesnot work well for huge content
[132]	2012	Generic ICN	LRU	Distance between nodes	Hop reduction ratio, server hit saving	—	Network traffic redundancy is reduced	Mobility of nodes not considered

ONE: Opportunistic network environment

is required to match content requesters and providers for an efficient content sharing. Matching theory is a popular mathematical concept having applications in various domains, including communication networks, college admission programs, and kidney exchange programs [34]. However, most of the existing matching game solutions are centralized, having a single failure point, as they need a coordinator to regulate the matching market. Also, the centralized solutions may lead to congested matching operations, which affects the scalability of the network. Moreover, in most of the existing solutions, the decisions of content requesters and providers contradict each other without mutual benefits. The designed matching model should consider the mutual preferences and satisfaction of both content providers and requesters. In this context, matching theory is a powerful concept in game theory to provide co-operation among untrusted parties.

Multiple techniques exist for finding a matching result in linear assignment games with various assumptions, [142], [143]. In particular, the aim of the assignment game is to obtain optimal matching for every player, and the allocation should have pairwise stability. For example, Nax and Pradelski [142] proposed a scheme to achieve a pairwise stable matching in a linear assignment game. The market participants have limited information about other players' utility. Players have aspiration levels to improve in each iteration based on their experience. This dynamic solution converges to pairwise stability in finite time. Also, Kim *et al.* [144] de-

signed a one-to-many matching solution for the parking assignment problem. In the proposal, the formulated equation is expressed as a mixed integer linear programming issue, and altering direction method of multipliers is used to obtain a near-optimal solution. Also, authors in [145] discusssed dynamics of multiple seller and buyer for spectrum trading. Similarly, the authors in [146] proposed a many-to-one matching game for resource allocation. Moreover, authors in [147] proposed a one-to-one matching about multiple providers and demanders, and the used matching theory exploits both sides' physical and social networking characteristics. However, both in [146], and [147], a pairing model is used based on a binary variable. In contrast, in our model, transit power controlling is achieved by a continuous variable to find a joint solution for the satisfaction of both sides. Also, these schemes demand learning about the preference of the other side of players. Authors in [148] designed a solution to assign secondary users to orthogonal primary user channels. The proposal's complexity is low; however, compared to our scheme, the proposal relies on heavy information exchange. With a similar motive to maximize content sharing, Jiang *et al.* [149], explores the maximal matching for sender-receiver pair in a decentralized way. They formulated the problem for selecting partners for communication as maximum weighted matching issue between sender and receiver, taking transfer rates as the weights assigned to each party. However, they have not considered any other physical and social metric for describing the utilities. Also, the work in [150] proposed a hypergraph-inspired three-dimensional matching issue among content demander-providers, and reuse of cellular user resources for device-to-device (D2D) link. However, here it is assumed that either of provider or demander is ready to take the decision from the opposite player side even if it can perform better by declining it. Therefore, the mutual preferences of both content providers and requesters are ignored. So, the resulting decisions may contradict different parties without mutual profits. BLMA was first proposed by authors in [151] to use its applications in cognitive radio.

All the above-discussed matching schemes assign a centralized coordinator to perform matching among different parties, so they may face a SPoF and lack privacy protection [152]. In this context, authors in [153], designed a decentralized co-ordination scheme for scheduling mobile users to the efficient virtual machine for computational task completion by leveraging blockchain technology. Nevertheless, this scheme lacks implementation details regarding blockchain network specification. Also, authors in [154] used matching theory to match the client requirements with the servers supply in vehicular named data networks. They have used a double layer blockchain for content transmission security. Different from our scheme, the authors have modeled the trading system as a one-to-many matching problem. Also, in [154], reliability and delay are the factors for requesters, whereas consumed energy and received vehicle coin are the factors which define provider's utility. However, here reliability is calculated from the reputation value and transmission rate. Due to the mobility of EVs, a requester can't download a whole content for a content duration. Successful delivery of content is required for high-reliability requirements which is ignored in this work.

2.4 Trust and security in CCN: Blockchain based solutions

2.4.1 Security challenges in CCN

In CCN, content delivery is a challenging issue as there are various security concerns in CCN including access control, confidentiality of data, integrity attacks, cache-related attacks, and routing related attacks, which are discussed as follows:

- **Interest flooding:** It may so happen that an adversary sends successive Interest packets to flood the network with extra packets. Hence, the interest transmission rate should be properly tuned so that traffic can be controlled as per available resources. Also, the attacker sends non-existing content to PIT, which degrades network performance. The packet delivery ratio decreases with increase in interest flooding attackers. Notably, an interest flooding attack is easy to be implemented, and its effects are devastating.
  To deal with interest flooding attacks, various techniques have been proposed. For example, minimum PIT size can be determined by setting a limit on the router total bandwidth and PIT expiration timeout. Some other options are throttling (rate limiting) [155], cumulative entropy that determines the request abnormal distribution [156], using Long Short-Term Memory (LSTM) by short-term memory. Also, PIT less routing (replaces stateful forwarding plane to a stateless plane) is another solution to mitigate interest flooding attack [157]. Approach in [158] suggests to embed signatures of content producer owner, to authenticate owner and to put check on frequency of request packets from host. However, in V2G, where EVs are dynamic and mobile, rate limit is difficult to manage. A malicious user may refrain sending to a particular host till rate limit is achieved, further it may shift to other area and start sending packets to another host.

- **Cache poisoning attack:** Cache poisoning implies that adversary nodes inject fake content into routers. To mitigate cache poisoning attacks, CCN provides in-built security by verifying signatures on content packets. However, signature verification at each packet is a computationally expensive process. Also, verification is accurate only if the verifier trusts the public key used for verification.
  To deal with poisoning attack, many techniques such as- random packet verification [159], reputation calculation [160], [161], self certifying names [162], public key digest information [163], popularity based probabilistic caching [159] has been proposed in literature.

- **Cache pollution attack:** A malicious user may store unpopular content to occupy content store storage. This attack comes under cache misappropriation attack. Cache pollution attack decreases average packet delivery ratio whereas the average delay in network

increases. As a solution, authors in [164] proposed an approach CacheShield that reduces cache pollution attack by storing popular content. However, in this scheme verification overhead leads to increase in computation task. Similarly, Zhou et al. [165] proposed a cache pollution prevention scheme with deep reinforcement learning. Here, authors designed an appropriate action room for agents to communicate with entities and designed a incentive function to give timely feedback to agents.

- **Content access control:** In CCN, content is stored in untrusted intermediate nodes, implementing access control at individual router is not practical [166]. Hence, most literature work in CCN enforces access control with encryption process. To provide access to content chunk, content provider shoud share the decryption key with valid users. For this any encryption based access control scheme, including proxy reencryption, CP-ABE, and KP-ABE can be implemented.

- **Privacy challenge in CCN**: Executing a border-less caching at network layer may give rise to various privacy and copyright threats. CCN allows any node to store content and serves it to other nodes as well. Any intruder can pose as trusted node and send unlimited interest packet to launch DoS attacks. Nodes should trust provider that may have an illegal identity. In IP based communication, any node can track user activity by monitoring IP address of requester. In contrast, CCN uses content name instead of IP address and its difficult to track requesters. Also, unauthorized access of data revealing location of vehicles is a huge privacy concern. Hence, network should validate requested content name that allows only legitimate packet to be cached. Hence, CCN familiarizes network nodes with various privacy interruptions in that network where blockchain can act as way for discussed security threats. To solve privacy issues, blockchain technology is proposed in literature to increase trust among nodes in data access control system. However, it is important to achieve a global consensus among content producer and content forwarders in the network. The cryptographic primitives used by blockchain technology can keep nodes data more private. In this context, authors in [163], [167], [168] uses blockchain technology to preserve privacy in the network.

- **Selfish attack**: It may so happen that because of limitation of computational and storage resources, providers may get selfish and don't forward interest or data packet [169]. Selfish attack further results in DoS attack. To deal with such attacks, incentivizing nodes for serving request has been proposed. Authors in [118], and [119] proposed work that incentivize nodes for providing data to nodes in the network.

To deal with above discussed attacks, blockchain technology has been introduced in literature. CCN and blockchain have been subject of separate research efforts. However, in recent year, there are some research work that proposed blockchain over CCN instead of IP to provide better hierarchical access. Blockchain technology was initially meant for TCP/IP. In TCP/IP model,

when a node wants to send content to multiple other nodes, it must send the data to each node individually, that leads to unnecessary data transmission overhead. As discussed, CCN provides in-network caching which can improve the fundamental broadcasting in blockchain [170]. The content centric approach of CCN supports effective record distribution and efficient coordination of blocks in blockchain. In CCN, content is distributed among multiple nodes rather node rather than placing it on a specific location. The integration of blockchain and CCN removes the concept of light weight (SPV client) and full node hence treating every nodes impartially. This will eliminate the security threats in existing distributed system as SPV client are dependent on full nodes for data verification. Consequently, the integration of blockchain technology with CCN provides efficient data distribution.

Other way we can use blockchain in CCN network for secure content delivery, as network participants maintain trustworthy ledgers in an untrusted environment. Next, we will discuss the significance of blockchain in CCN network in details. Table 2.8 and 2.9 represents the comparison of blockchain based CCN proposals. The classification of the application of blockchain in CCN is represented in Fig. 2.3.

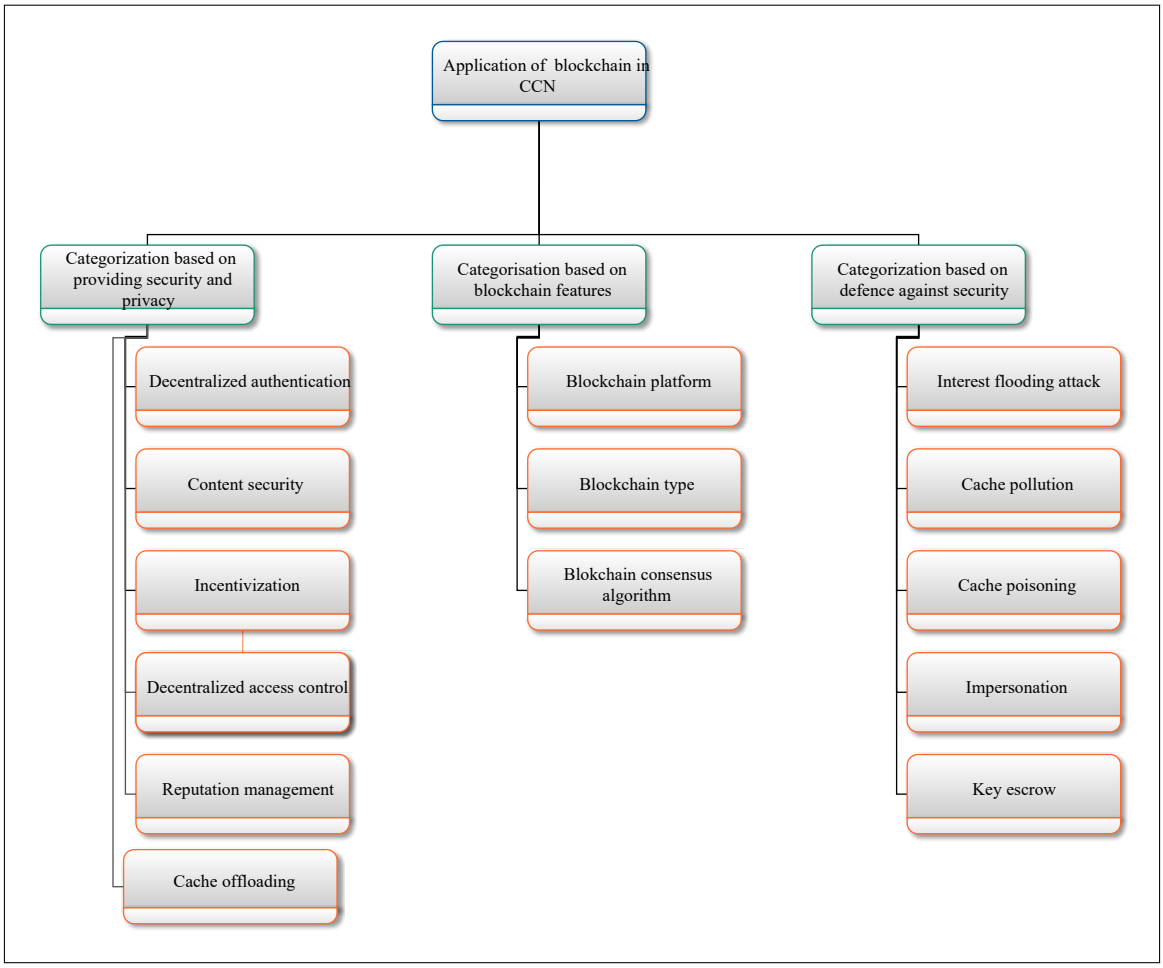

Figure 2.3: Applications of blockchain in CCN

2.4.2 Reputation management

Nevertheless, clear advantages provided by CCN, it also introduces new security challenges because of in-network caching supported by it. As a in-built security mechanism, each content provider has to sign the content generated by it. However as compared to signature generation, signature verification is treated as a large cryptographic process [40]. In addition to the high expenses associated with signature validation, routers would to retrieve different public key certificates to establish trust for public key responsible for verifying a content signature. This leads to a substantial increase in overhead. Here, reputation-based trust schemes are used as an alternate method to handle attacks specific to CCN network. Within reputation mechanism, individuals provide ratings to providers of services or resources, with these ratings serving as assessments of the quality of their direct communication. Trust is calculated by aggregating ratings based on individual experiences within the local context, while also considering the feedback provided by other entities within the network. The motivation of using reputation value is to overcome the insecurity of content store and to develop trust among network providers and consumers. However, in the existing systems [171], reputation management is accomplished with a centralized framework. This demands presence of a third party responsible for collecting the ratings and calculating the ultimate trust scores. On the other side, there exist decentralized approaches in which each node independently computes ratings and share them among nodes via a peer-to-peer dissemination protocol. This approach eliminates the concern of a SPoF. However, it introduces the requirement that individual nodes conducting computations and offering reputation data must be trustworthy. To address these challenges, blockchain technology can be combined into CCN as a remedy. Blockchain provides trustworthiness typically associated with a central entity (e.g., by recording ratings in an immutable and verifiable ledger) and the advantages of distributed reputation value dissemination.

In blockchain based reputation management category, the decentralized blockchain network is used to store reputation value of vehicle node and content store of CCN network. The reputation value are updated based on served content from content store. After verification of content, content requester updates blockchain ledgers about the quality of served content. As the data stored on blockchain network is immutable, the reputation value stored in blocks can't be changed by malicious user. For example, authors in [172] proposed a method to store the content requesters feedback so that unmanned aerial vehicle (UAV) network nodes reach a consensus. In a same way the proposal in [154] used reputation evaluation scheme for a reliable content delivery. The reputation value represents the QoS of different provider which is calculated based on previous interactions with the provider. In [154] only the local opinions are considered for computing reputation value whereas in [173], the vehicles not only considers local opinions but also take recommendation from other vehicle nodes. Authors have used smart contract for secure data storage and a traditional subjective logic model is used to design a reputation based data sharing.

In all above discussed solutions, generation of new block is performed after transmission process. In contrast in [174], generation of new block to blockchain has no effect on forwarding process. Authors suggested a reputation supported blockchain mechanism to safe content caching in named network. The reputation scores are stored as transaction, further recorded in blockchain system after forwarding process completion. Here, PoW consensus is used to verify and store transaction in system. There are two types of reputation values, i.e., node based reputation and content based reputation. This scheme also provides a method to compute average expunge time of existing PIT entries, that can prevent DoS attack. The entities queries blockchain network to investigate trust of served node or content store. After verifying the content, the requester will create a new block informing blockchain its decisions. If the content received is not valid, the reputation value is decreased and updated on blockchain network. The result section depicted that only reliable content is cached and served to other nodes. However, the authors didn't mentioned the implementation details of blockchain network. Also, in case of wrong content received by node, it keeps on sending interest request for the same content, which leads to increase in redundancy.

Specifically, to detect content poisoning attack, authors in [175], present a scheme that enables caching nodes to decide content acceptance using rule based reputation system. The decentralized blockchain platform is used to store reputation value in immutable way. Different from pull based scheme, this scheme follows push based content dissemination. Hence, the reputation value is disseminated using the push based method.

2.4.3 Decentralized Authentication

A adversary node may spread fake messages in network that causes congestions. Hence, individuals engaged in the Internet of Vehicles (IoV) should not only be concerned with the source of content request but also with the originator of content packet. The data packet must originate from a verified producer and remain unaltered by any other producers. So, the content package should be generated by authenticated content producer (CP). A content consumer checks the authentication of CP by using CP's public key for content verification. In order to authenticate the content generator identity and ensure data integrity, the generator should digitally sign the content. This process effectively and securely associates the name with the content. Consequently, consumers and routers can then validate the signature, enabling them to identify the data source and thereby ensuring trust in the received data packets. However, the digital signature is insignificant if the public key received is incorrect. Hence, key management and authenticity of keys is of great importance. Existing CCN is based on hierarchical key model where root key acting as trust anchor generates digital signatures for domain secret key. Further, the domain's key signs the public key of nodes in the domain and finally the node key digitally signs the public key of its application. To check the authenticity of public key, secret key chain is retrieved by using the key name. However, the root key is prone to SPoF

and in cross domain key verification, one domain can't verify the authenticity of other without having a trust anchor. For verifying a content packet, a trust schema is used to recursively verify each signing key till it finds a trust anchor. Also, this kind of verification leads to significant latency which is not acceptable in delay sensitive application. Many recent study have used blockchain based key management for having mutual trust between different domain. By leveraging blockchain, the need for centralized party is removed hence reducing the overhead of recursive verification. Blockchain technology guarantees data traceability and deters unauthorized alterations by employing decentralized data storage and consensus mechanisms. This ensures the integrity nodes in the network. Next, we will discuss the work related to blockchain based authentication schemes in CCN.

To reduce delay incurred by signature verification, authors in [176] used blockchain for providing authentication services in CCN. Here, the blockchain network will store the certificate of each vehicle producer. The content requester probes for the certificate linked with received data. Similarly, Shi et al. [177] used the smart contract facility of blockchain to provide authenticity of public key in an untrusted decentralized network. Privacy is achieved using an optimized zero knowledge proof. Public key certificate generation and management is achieved using blockchain. Miner nodes after verification registers public key certificate along with identity proof on blockchain network. Also, authors in [178] used blockchain to provide decentralized authentication for information centric 5G networks. The robustness of the proposal is tested against attacks including replay, impersonation, cache pollution and false reputation. With a similar motive, authors in [179] present blockchain enabled key management strategy aiming to address the lack of trust issue among trust anchors. The proposal refrains from establishing a root key and relies on each size to use blockchain as trust anchor, which ensures authenticity of subsequent layer keys through hash storage. The blockchain operates as decentralized system which uses cryptographic methods for message verification among nodes. The failure of any single node does not impact the full system's availability, thus mitigating the risk of SPoF. However, this proposal doesn't provide any deployment or implementation details. Also, blockchain network store public keys of all nodes to have a long chain length, which increases the query time. Also, this scheme does not ensure mapping in content name to public key. To resolve this issue , Liu et al. [180] proposed a blockchain based key authentication and management for vehicular NDN by taking advantage of decentralized and non-tampering characteristics of blockchain. Authors proposed a flat hierarchical structure for key management to reduce the number of signature and authentication verification. The proposed scheme provides mappings from content name to public key. Also, the number of domain public keys are kept smaller than number of content producer's public key so to have a short blockchain length. Also, the authors in [167] used blockchain for key management in CCN.

In traditional scheme, public parameters of system are generally stored with a central authority. To solve this problem, Li and Ma [181] proposed to use blockchain for storing public parameters in NDN. Also, authors proposed a hierarchical structure to map content name to CP

public key. Also, to save space, all producers in the same area share the same public parameter. The requesters request for public parameters from blockchain network when they request content from content producer. Next, requesters compose public key of producer from parameters for authentication and integrity verification. In another work [182], blockchain is used to share system parameters and content specific secret while using a hierarchical IBE (HIBE) to design a access control scheme. In HIBE, identity of a node is used as public key and private key is generated by a special node called private key generator. To resolve the key escrow problem faced by using a private key generator, blockchain is used to share system parameter. CP also registers name of content produced by it in blockchain. While retrieving the content, requesters first download the system parameters of CP from blockchain network.

In V2G network, vehicles are continuously moving from one place to another, however,the above discussed scheme doesn't discussed mobility of vehicles. In this context, Conti et al. [183], proposed a lightweight blockchain based CP authentication schemes that addresses mobility management in ICN. The proposal involves using mobile device SIM to register key pair of nodes. The security analysis represent that the proposal perform better over other authentication schemes as it mitigates DoS, prefix hijacking and other network attacks. To avoid the overhead of consensus algorithms, the scheme uses a time based consensus algorithm where block creator is randomly selected from all miners. However, this scheme does not discussed key management. Differently, authors in [184] proposed a new consensus algorithm Proof-of-Time, to authenticate content. To implement Proof-of-Time, each content packet is associated with a credibility score that represents the time duration for which a publisher can sign content. Blockchain network is employed to ensure that content owner (CO) can generate only one Proof-of-Time at one time. The proof-of-time of any content contains hash of content, CO signature and previous block hash. Result section depicts that the proposed consensus process is improvised in terms of communication complexity and reliability as compared to PBFT and stellar consensus protocol.

2.4.4 Content security

Effective content sharing will enhance the security of CCN network. However, malicious entities may disseminate fake information in the network, which may effect the charging discharging decision in network.

In this context, Li et al. [195] proposed trust enabled blockchain tracing method for content delivery in network. Blockchain here is used to prevent cache pollution attack by recording hashes of discovered messages. With similar motive, authors in [194] proposed a blockchain enabled solution attacks for content life cycle protection in ICN. This proposal contains five types of different transactions and four types of smart contract to provide authentication, scaling and regulation in network. In this framework, a novel node category called the Data Dam Blockchain Node (DDBN) has been incorporated. This node type serves the dual purpose of

managing local registrations and regulating the data flow. Additionally, it facilitates the interaction between the blockchain and ICN. An examination of security and performance reveals that the proposed framework effectively fulfills the design criteria necessary to safeguard the data's lifecycle. Moreover, Chen et al. [154] proposed a model having double layer blockchain for information trading in VNDN. Here, the leader node that verifies the block transaction is awarded with vehicle coins. The blockchain ledgers will store the transaction of content exchange. To save computational resources of PoW, the center opted for block verification will only act as miner node. Also, authors in [189] uses blockchain technology in NDN to achieve decentralization, integrity, accountability and transaction verification. Similary in order to save computation overhead of consensus algorithms, authors in [193] suggested to use PoS to choose the miner node in blockchain based NDN proposal. Also, authors discussed about the challenges faced by the integration of blockchain in NDN, such as- miner selection, optimization of consensus, keys management ans cross platform authentication. However, overhead of computation and communication cost of NDN network are not discussed without which the efficiency of system cannot ne verified.

Some literature work used consensus algorithm of blockchain to check the validity of content. For example, to deal with the scalability issue of blockchain, Doku et al. [190] proposed the combination of NDN and blockchain in Internet of Battlefield things. In this proposed method, nodes having similar data are assigned to same interest group. Here, a node only adds content to its ledgers if content similarity store is greater than any of the content in its ledger. Also, authors proposed a consensus algorithm called proof of common interest to achieve validity of data. Differently, authors in [192] used blockchain for prevention of cache pollution attack and compared their output with a non-blockchain based network. Blockchain in this scheme is used for attaining privacy traceability in network. Authors concluded that cache pollution attack has more impact on small size network as compared over large scale ICN network. Experimental results are analyzed to compute time needed in Hyperledger fabric to update the ledger after any changes are made to the network.

2.4.5 Decentralized access control

Within CCN, the CP looses control over generated data. Also, as data is cached among multiple nodes, the CCN network faces access control issues. Existing access control proposals introduce SPoF and congestion due by involving external party. In the proposal [197], blockchain is used to store access policies in ledgers in an immutable way. The content sharing in the scheme has low overhead, reduced network delay and bottleneck. However this work has high computation time because of involving complex encryption/decryption operation. To enable fast encryption/decryption XoR based encoding/decoding scheme is used by [196] which saves time of CP. Decryption information is kept on blockchain network instead of being stored at CP, so even if the CP is offline, the requesters can download the content. With help of using

blockchain traceability functionality, CP can detect leaks via unique sequence defending entity's right. Similarly, Lyu et al. [191] discussed a blockchain supported access control framework enabling a CP to share data security. A blockchain supported access token is realized for CU satisfying the condition of ACP. The access token can further be granted, shared, delegated and revoked. To reduce verification overhead of access token, cuckoo filter is used.

2.4.6 Incentivize nodes

One security vulnerability in CCN causing security attacks, arises from the self-centered behavior exhibited by caching nodes. In this context, particularly within infrastructure-free networks, hosts play the role of forwarding interest and data packets. However, having computational and storage constraints of caching nodes, they may behave selfishly and don't forward interest or data packets, which can result in DoS attacks. In such scenarios, if a packet is directed towards an attacker, they can intercept the packet's name and filter some content component that they don't want to serve, thereby preventing further forwarding. Also, an attacker might opt not to provide any forwarding assistance to others due to their self-centered disposition. As studied by authors in [199], the selfish nature of nodes in a network can result in decrease in average delivery ratio by 50%. Lack of incentive mechanism in ICN based network lead to nodes caching interest packet of their own choice. Consequently, the hit rate of this node decreases which results in wastage of transmission resources and increasing delay of network. An incentive mechanism in system activates nodes to establish a collaborative caching agreement which improves hit ratio.

Authors in [185] used blockchain consensus mechanism for incentivize nodes in ICN collaborative caching. The nodes with high caching ability are rewarded more. The rewarding mechanism in ICN motivates other nodes for caching. An ICN overlay is designed to validate the efficiency of collaborative caching. Similarly, Zhu et al. [199] suggested to incentivize nodes for content packet forwarding. The collaborative game theory is used by nodes to motivate for caching meanwhile minimizing the delay cost. Here, incentives are also provided to miner nodes for transaction verification. In a similar way, authors in [200] proposed to incentivize nodes proportional to content validated by them. The blockchain framework manges the mobile content generation and retrieval in Web3 applications leveraging ICN architecture. The scheme uses machine learning for nodes to construct a fog network with use of distributed IPFS for content storage.

2.4.7 Routing table update

In CCN based scheme, location of content is not required, content is retrieved by its name having identifier in the network. Using the identifier, contents are cached in routers so that a node can retrieve content from nearest neighbor. Nevertheless, under any system failure, routing table information is required for smooth functioning. The available routing procedure

(including open shortest path first, link state routing) sends link state advertisement to publish the existence of content name prefixes that are cached in routers. In case of router failure, the content information regarding content can't be updated till the routers get back to function again.
To solve this issue, authors in [187] proposed link state routing where blockchain creates routing table. First, each network routers check the connection information of other routers and creates a transaction that creates all link connection. Using information from blockchain, each router design the similar topology in network that further constructs a routing table. Compared to existing solutions, the presence of content and their identifier are synchronized even in failure as the data is synced in system in an decentralized routing architecture.

2.4.8 Blockchain enabled cache offloading

Routers in CCN offload content to nearby nodes and nodes participating in the process are incentivized. Hence, a trusted and cooperative mechanism is required for a optimal cache offloading scheme between nodes. Authors in [188] proposed a blockchain supported cache offloading mechanism for VR/AR services in 6G. The system uses a proof-of-cache offloading (PoCo) as verification mechanism to verify content and resource transaction in network. The nodes with high cache hit and improved hit rate can become block verifier.

2.5 Research Gaps

After conducting a thorough analysis of the aforementioned existing proposals, we have identified certain research gaps that require further investigation.

2.5.1 Decentralized access control system for CCN enabled V2G

- Most of the reported solutions for the access control grants are centralized, so these may suffer with congestion problems, overhead generation, and do not have fault tolerance capabilities.
- The existing access control techniques executes at cache-enabled neighboring node (NN) and involve interaction with the CP during content retrieval. However, this verification process at the caching node lowers forwarding services.
- Moreover, traditional access control schemes such as mandatory access control (MAC) and role-based access control (RBAC) are not applicable to CCN, as these schemes are designed to implement ACP where contents reside with the CP. Also, CP-ABE causes key escrow, heavy computation, and authority collusion problems.

- CP-ABE can facilitate data confidentiality and access control for CCN network, but it can't provide transaction transparency, as there is no information about malignant nodes accessing the content ciphertext.
- Existing CP-ABE based solutions use Pedersen (t,n) secret sharing protocol for key generation and authentication of secret shareholders using signature verification technique has high storage and computation overhead, causing delays in network.

2.5.2 Lack of incentive mechanism for caching services

- In recent years, various caching strategies have been proposed to make cache decisions by considering cache popularity. However, the popularity-based caching without an incentive mechanism and non-game based approach is difficult to implement in reality.
- Nevertheless, there are multiple approaches for designing an incentive-compatible cooperative caching in CCN/NDN, including Stackelberg game, auction theory, and nash bargaining theory. However, all these solutions are based on an iterative mechanism, which has the disadvantages of long convergence time and requires multiple information exchanges between two parties.
- Also, most of the current work mainly concentrates on designing an incentive mechanism in CCN-based vehicular networks, while the security and privacy issues to facilitate secure content delivery is not considered.
- None of the existing approaches have focused on implementing decentralization in incentivized cooperative caching for CCN-enabled V2G networks.

2.5.3 Decentralized matching algorithm for content requesters and producers

- However, most of the existing matching game solutions are centralized, having a single failure point, as they need a coordinator to regulate the matching market. Also, the centralized solutions may lead to congested matching operations, which affects the scalability of the network.
- The existing schemes have not considered the trustworthiness among EVs, charging stations, and local area aggregators (LAG).
- In existing solutions, it is assumed that either the provider or demander is ready to take the decision from the opposite player side, even if it can perform better by declining it. Therefore, the mutual preferences of both content providers and requesters are ignored. So, the resulting decisions may contradict different parties without mutual profits.

- In the existing works, reliability is calculated based on reputation value and transmission rate. Due to the mobility of EVs, requesters cannot download entire content within its duration. Hence, successful content delivery becomes crucial for meeting high-reliability requirements, which is ignored in previous works.

2.6 Objectives of the Research Work

The following objectives are suggested in the proposal following a review of previous proposals and the identification of research gaps.

1. To design a decentralized access control mechanism for information management in the V2G environment.
2. To design a secure and decentralized content management scheme among nodes in the V2G environment.
3. To develop a CCN based communication network for a V2G network.
4. To conduct performance evaluation comparison between CCN and host centric based V2G scenario.

2.7 Summary

In this chapter, we reviewed the existing work done in field of CCN. Firstly,we reviewed the pros and cons of access control schemes available in CCN-based proposals. Also, we provided the taxonomy for access control in CCN/NDN. In next section we reviewed the existing caching selection strategies proposed in the literature. The aim of content selection for caching is to improve the hit ratio of request packets, hence increasing network performance in terms of load, delay, and interest satisfaction. We also reviewed the matching model available in literature required to match content requesters and providers for an efficient content sharing in CCN/NDN. Finally, we discussed the significance of blockchain in CCN network in details along with providing the classification of the application of blockchain in CCN. The blockchain in CCN network for secure content delivery, as network participants maintain trustworthy ledgers in an untrusted environment. The next chapter discusses a in-network cooperative caching scheme for fast content delivery.

Table 2.8: Comparative analysis of blockchain based CCN proposals

Reference	Year	Blockchain usage	Application	Attack prevention	Simulator	Comparison matrix	Blockchain network	Consensus mechanism	Merits	Demerits
[181]	2023	Decentralized authentication	NDN	Impersonation	ndnSIM	Signatute time, data verification time	Hyperledger fabric	PBFT	Reduced response delay time	Privacy preservation not considered
[175]	2023	Reputation management	VNDN	Content poisoning	MATLAB	Block integrity verification	Public	PoW	Integrity is achieved 100%	Not performed on NDN testbed
[176]	2023	Decentralized authentication	VNDN	Impersonation, blockchain tampering attack, MITM, DoS	AVISPA, security protocol animator	Verification of authenticity		PoW	Fast authentication	certificate revocation not discussed
[185]	2023	Incentivize nodes	ICN IoT	DoS	Socket programming, Python 2.7	Hit ratio	—	DPoS	Improved link quality	Computational complexity is high
[177]	2022	Key management	ICN	Replay	FISCO BCOS	Certificate security and communication security	Public	PBFT	High scalability	High storage demands
[178]	2022	Decentralized authentication	ICN 5G	Replay, cache pollution, DoS, impersonation	—	Delay in authentication	—	—	Delay savings	No proof of concept implementation
[186]	2021	Key management	VNDN	Data tampering, impersonation, SPoF	—	Energy consumption , delay	Consortium blockchain	PBFT	Reduced number of signature verification, reduced time delay of key acquisition	Privacy preservation not considered
[187]	2021	Routing table update	CCN	—	Cefore	Entry update and deletion in router	—	PoS	Improved fault tolerance	Route selection system not flexible
[174]	2020	Reputation management	VNDN	DoS, content poisoning, interest flooding	ndnSIM	Forwarding process, storage time, cache utilization, overhead of PIT entries				Implementation details not discussed
[188]	2020	Cache offloading	Vr/AR in 6G	—	MATLAB	Cache utilization ratio, cache hit ratio, throughout	Permissioned	PoCo	Consensus mechanism has low resource overhead	Mobility of nodes not considered
[189]	2020	Content security	Vehicle to everything NDN	DoS, impersonation	NS-3	Communication overhead, probability of propagating false information	—	PoW	Trust among nodes	Blockchain environment implementation details missing
[190]	2020	Content security	IoT Battlefield things	Packet flooding	PyNDN	Packet loss rate	—	Proof-of-common interest	Better packet transmission	Blockchain network implementation details missing
[191]	2020	Tokan management	ICN	Content poisoning, MITM, DoS	Web3, Node.js	Computation cost, communication cost	Ethereum	PoS	Reduced computation cost	Transaction verification efficency of blockchain not discussed
[180]	2020	Key management	NDN	Impersonitation attack, data tampering attack	Microsoft visual studio	Signature generation time, signature verification time	—	—	provides packet integrity	Blockchain network details missing
[184]	2020	Decentralized authentication	NDN	Content poisoning	—	Communication complexity, scalability and fault tolerance	—	proof-of-time	Resilient and fault tolerance	Content revocation not discussed
[154]	2019	Data transmission security	VNDN	SPoF privacy leakage	Statistical experiments	Social welfare, matching rate, requesters utility	Ethereum	Pow	Secure information interaction	Implementation details missing

Table 2.9: Comparative analysis of blockchain based CCN proposals

Reference	Year	Blockchain usage	Application	Attack prevention	Simulator	Comparison matrix	Blockchain network	Consensus mechanism	Merits	Demerits
[192]	2019	Privacy preservation	ICN	Cache pollution	Go Language, Python 2.6	Query delay, update delay	Hyperledger	PBFT	Less update delay	Experiments conducted using few network nodes
[172]	2019	Reputation management	UAV ICN	Internal attack, DoS	ndnSIM	Latency	Permissioned	—	Lower system overhead	blockchain is not integrated with ndnSIM for simulation
[193]	2019	Secure content dissemination	IoV NDN	—	—	—	—	PoS/PoW	Content security	Implementation of proposal not discussed
[194]	2019	content security	ICN	Data misuse, impersonation	—	Data retrieval delay	Ethereum	—	Data delay is low in large network size	Details of simulation setting not provided
[183]	2019	Decentralized authentication	ICN	DoS, prefix hijacking, replay	—	Delay, throughput, storage cost	Permissioned	Time based	Mobility of vehicles is considered	Key management not discussed
[195]	2019	content security	ICN	DoS, interest flooding, cache pollution	Python 3.6, SQLite	Throughput of blockchain network, time cost, bandwidth occupation	Public	—	High throughput of blockchain transactions	High bandwidth consumption
[196]	2018	Decentralized access control	ICN	Collusion, impersonation	—	Security analysis only	—	—	Access control process has low overhead	No implementation details provided
[197]	2018	Decentralized access control	CCN	Dos , external attack	—	Computation and communication cost	—	DPoS	Privacy efficient	Revocation operation not discussed
[179]	2018	Key management	NDN	—	—	Verification efficiency	—	PBFT	High verification efficiency	Does not ensure mapping in content name to public key
[198]	2018	Key management	NDN	DoS	Node.js	Validation of functions	—	PoS	can resist attacks that compromise less than 50% of miner	Communication not implemented via NDN
[182]	2016	Key management	NDN	Chosen ciphertext and key escrow	charm crypto library	blockchain query time	Namecoin	PoW	free from key escrow problem	Does not support variable size system parameter storage

Chapter 3

Blockchain-Based Co-operative Caching for Secure Content Delivery

3.1 Introduction

The traditional IP-based V2G network has issues in balancing the demands of ever-increasing traffic and low-delay transmissions. In V2G network, all queries related to charging operation are carried out by LAG, causing it to be overloaded, which in turn effects the quality of experience to the other nodes. To relieve LAGs load, mobile edge computing has been proposed in the literature in which data from LAG can be cached on EV nodes [201]. However, to make caching possible on EV nodes, the characteristics of V2G network, *i.e.*, limited bandwidth, fast mobility, heterogeneous nature, and always changing topology should considered. These characteristics make it difficult for existing IP-based transmission solutions to support caching. The inherent in-network caching feature provided by CCN motivates us to use it as a communication protocol in V2G network. However, attacker nodes in the network may send fake messages from their cache store, resulting in deteriorating network performance. A reputation mechanism may supports nodes to provide reliable content during sharing. Also, each EV hopes that nodes in its neighborhood stores their favorite contents due to selfish concerns. However, the popularity based caching depriving an incentive mechanism and non-game based approach is difficult to implement in reality. Compared to non-game theory based approach, the game based approaches are efficient for V2G networks. Also, a reward mechanism can encourage nodes to cache a variety of content from LAG. On the other hand, placing content in the cache of EVs leads to LAG load release. Moreover, the cooperation of multiple CCN nodes' in a trusted and efficient way leads a challenge to security in the V2G network. Hence, it's essential to introduce an efficient incentive mechanism and privacy protection mechanism so that more nodes can participate in caching to share data while preserving privacy and achieving trust during the content sharing.

3.2 Contributions

Following are the research findings of this chapter.

- We propose a CCN approach for data retrieval in the V2G network. Before retrieving content from the content provider, each node evaluates its reputation score to mitigate content poisoning attacks. The consortium blockchain network is used to record the reputation value and incentives-related transactions in the ledger of all nodes using blockchain for which PoA algorithm is used to achieve the consensus among network nodes.
- We then propose a reward mechanism for EVs to cache the content in order to reduce network traffic on LAG in the network. We use contract theory to incentivize EVs to participate in content sharing and to model the interaction among LAG and EVs. LAG works as a contract designer and EVs as followers, which share content with other nodes to obtain a reward according to their contribution.
- Finally, we evaluated the system performance through extensive simulations. The numerical results illustrated that the proposed incentive scheme outperforms the existing proposals using parameters social welfare, delay saved, latency and throughput.

Chapter 3 is organized as follows. An overview of the System model and incentive mechanism using contract theory is presented in Section 3.3. Section 3.4 discussed the optimal contract designing strategy along with the detailed consensus algorithm. The experimental results are discussed in Section 3.5

3.3 System model

3.3.1 Network model

As illustrated in Fig. 3.1, we have considered an EV charging management scenario comprising multiple charging stations (CSs), a local area aggregator (LAG), a fleet of electric vehicles (EVs), and a blockchain network.

CSs are mainly deployed in zones with a high concentration of EVs, such as shopping malls and parking places. To achieve an efficient charging with minimized waiting time and load balancing across multiple CSs, the information regarding CS conditions is disseminated to EVs for taking their charging decisions. For this, we have used LAGs to share the information between CSs and EVs. CS publishes its information, such as queuing time, location, the current state of charge (SoC), and number of charging slots, to a nearby LAG. Also, each EV disseminates its information, including SoC, location, and driving pattern, to the nearby LAG. Each LAG manages multiple CS and coordinates activities of a group of EVs in a charging slot. Upon receiving a charging request from an EV, LAG sends its latest stored information to the EV.

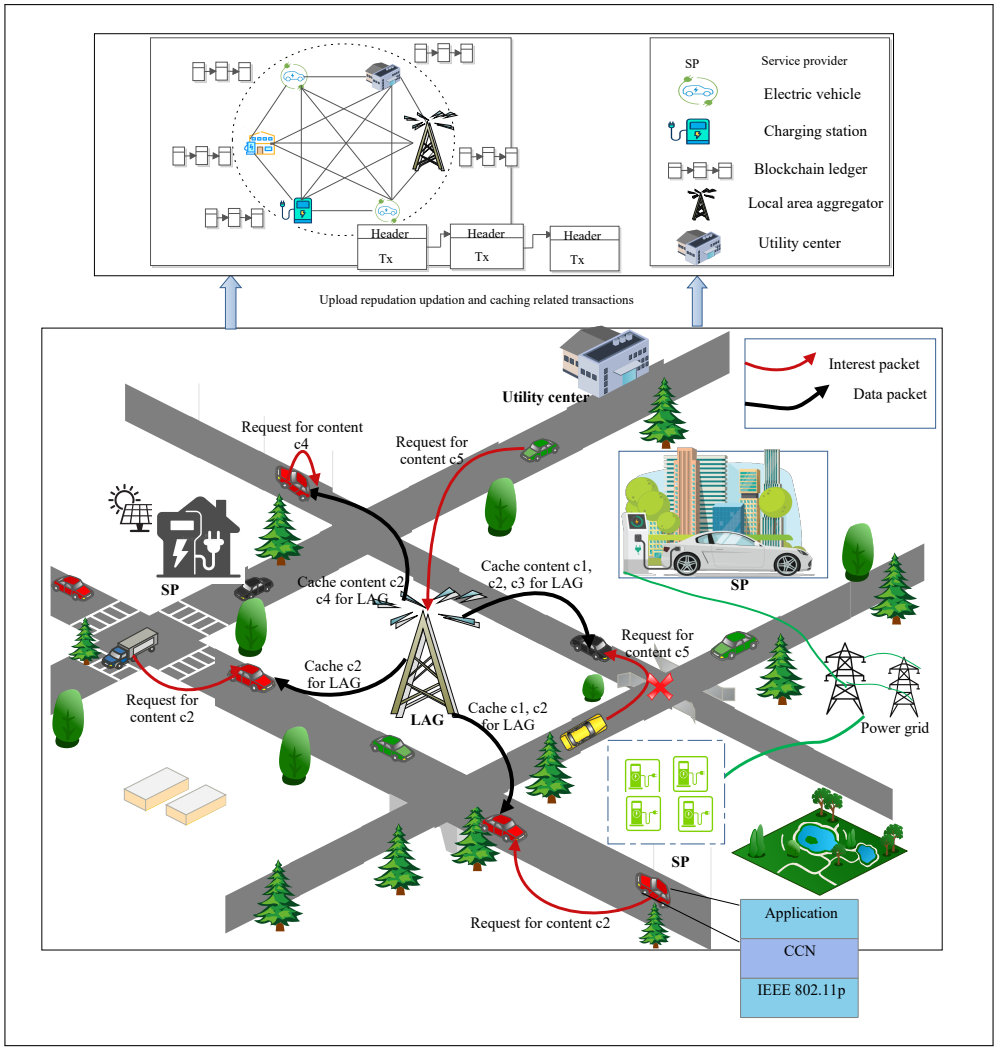

Figure 3.1: System model

In the next time interval, once new information from CS or EV has been received by LAG, it replaces the obsolete information of the past. Also, LAG sends the aggregated information from EV and CS to the utility center for demand response management.

To reduce the traffic load between an increasing number of EVs and LAG and to enable fast data transmission, LAG caches the copies of data received from CSs, EVs, and utility center in the content store of some EVs which are geographically closer to the LAG. After caching the content, EVs can meet their content requirement while serving other entities as well. However, both entities are assumed to be selfish and intend to maximize their profit. Moreover, to tackle the trust problem for content delivery in a content-centric-based network, the reputation value of the content provider is first checked in the proposed scheme. Intuitively, a well designed incentive mechanism should provide rewards to EVs based on the contribution of content sharing. Users with high battery percentage, high cache capacity, and high reputation value are more likely to contribute to content sharing. However, LAG does not have prior knowledge about battery percentage and of caching capacity each EV can contribute. This information

asymmetry between LAG and EV cost too much for LAG to give incentives to the EVs. Therefore, the optimal strategy for LAG is to create an incentive mechanism to diminish the impact of information asymmetry. Moreover, EVs contributing more for content sharing should be inecntivized more. Thus, we used contract theory for developing the incentive mechanism. However, the functions of EVs and LAG are updated in every new time slot to handle variations in cache popularity and vehicle mobility.

The Blockchain network ensures the traceability of various transactions among nodes in V2G network. All caching-related transactions and reputation updation transactions are packed in blocks and stored in the blockchain after getting verified by a consensus mechanism process. Each network transaction is encrypted with the sender's digital signature of the transaction.

Table 3.1: Symbols and notations used

Variable	**Notation**
N	Total number of EVs
$\mathscr{N}$	Set of EVs
$N_{\mathrm{i}} \in \mathscr{N}$	set of neighboring EVs to EV i
$\mathscr{C}$	set of content
C	Total number of content chunks
s_c	size of cth content chunk
c_i	cache size of ith EV
$d_{l,i}$	unit transmission delay cost between LAG and EV i
$d_{nb,i}$	unit transmission delay cost between two EVs
p_i^c	probability of vehicle i requesting content c
x_i^c	Caching status of content c by EV i
$Rep_i(t)$	The reputation value of EV i in time slot t
r	The unit reward paid by LAG for caching files
ζ	The parameter of Zipf distribution
θ_i	Type of EV
λ_i	Probability that a EV i belongs to type-i

3.3.2 Communication model

We have used CCN as the communication model, designed for point-to-point connections between nodes and focus on content dissemination and retrieval [202]. CCN communication is characterized by the receiver-driven transport protocol (query and pull) based on two packet types, *i.e.*, Interest and data packet. A requester sends its request by broadcasting an Interest packet on all available connectivity interfaces, whereas any node with the content can be a

provider. Moreover, data packets are sent to the receiver with reverse path of the request, and packet-level caching is transparently performed at every network node. In particular, content is divided into content chunks identified and requested by consumers through explicit pre-chunk requests. A name-based routing protocol (OSPF) guarantees that requests are correctly sent toward the data source [203]. Each passing node records pending queries to deliver content back to the requester and caches it in the least recently used managed cache.

We consider a single cell V2G network consisting of multiple CS, a LAG and a set of N EVs denoted as $\mathscr{N}$=$1,2,...,N$, where each EV i has a storage capacity of c_{i}, $i \in \mathscr{N}$. Let $N_{\mathrm{i}} \subseteq \mathscr{N}$ denotes the set of neighboring EVs to EV i. Each EV can communicate and share content with neighboring EVs using CCN architecture. All EVs are divided into N different types sorted in an ascending order: type-1, type-2,...., type-i, ...type-N. To attract more EVs with high battery level, high reputation value, and high content sharing capacity, we define a parameter θ, about content sharing willingness as the type of EV, which is denoted as:

$$\theta_i = min[1, \frac{SoC_i}{100} + Rep_i + \frac{c_i}{c_{max}}] \tag{3.1}$$

where, SoC_{i} is the remaining battery level of an EV i, Rep_{i} is the current reputation value of EV i (discussed in next section), c_{max} is the maximum cache capacity of the network. θ_{i} denotes the type of EV and follows:

$$\theta_1 < ..... < \theta_i < ..... < \theta_N, i \in [1, ...N] \tag{3.2}$$

A higher θ means EV has more remaining battery with high reputation value, and it has more contribution towards secure data sharing, thus deserves more rewards. Without loss of generality, we assume that θ_{i} suppose that a uniform distribution with probability density function f(θ) = 1 , $\theta \in$ [0, 1]. For a type-i EV, LAG rewards it with reward r_i according to its caching contribution. Under information symmetry, LAG construct a reward contract based on EV contribution towards caching, denoted by:$(S, r(S))$, where, S represents the total amount of content the EV shares with other neighboring nodes and r depicts the corresponding incentive provided by LAG. It is clear that $r(S)$ is strictly increasing function of C. Instead of offering same contract to all EVs, LAG provides different contract bundles according to each EV type θ. Thus, the contract offered by LAG for type-i EV is written as $(S_{\mathrm{i}}, r_{\mathrm{i}})$. The LAG does not have any idea about the type of EV, however, it knows the probability that a UE belongs to type-i, which is presented by λ_{i}, with $\sum_{i=1}^{N} \lambda_{\mathrm{i}} = 1$.

A content library $\mathscr{C}$ with C files is considered denoted as $\mathscr{C}$= $1,2,....,C$, and s_{c} denotes the size of content chunk c, $c \in \mathscr{C}$. We denote by $d_{\mathrm{l,i}}$, the unit transmission delay cost between EV i and LAG and $d_{\mathrm{nb,i}}$ between EV i and its neighboring EV nb, $\forall nb \in N_i$. Table 3.1 presents the list of symbols and notations used.

3.3.3 Attack model

In-network caching is an advantageous characteristic of CCN that helps content distribution efficiently. However, due to the usage of wireless connectivity, it allows adversaries to launch attacks. This chapter addresses cache poisoning attack, *i.e.*, a intruder content provider injects fake information into content files for received interests. In particular, reactive and proactive attacks of content poisoning are considered. In reactive attack, a malicious node injects fake content into the network without predicting the requests of other nodes. On the other hand, in proactive attack, first, from compromised nodes, a malicious node anticipates a set of interest packets for a valid content packet to inject malicious information into the content store. The content file with manipulated information passes with the reverse path of corresponding interest to the content requesters, and some intermediate nodes cache the poisonous content. Next, the compromised nodes replies to later requests with fake content causing more significant damage unintentionally.

3.3.4 Assumptions

- Each node in the network has a global positioning system (GPS) for location tracking.
- Each EV can transfer data with its neighbor EV in the wireless communication range.
- We assumed a heterogeneous request model, where the preferences of EVs are different; hence the popularity of one content varies from one EV to another. We assume that the average demand during any time interval of set $\mathscr{C}$ of content is known in advance.

3.3.5 Service model

- Consider EV i requesting for content c, the EV i first check in its cache store. If the requested content is there in the local cache store of EV i the request is satisfied with no delay.
- However, for an uncached content request, EV i communicates with neighborer EVs in N_i. Here, we assume that the set N_i is sorted in an increasing order of the transmission delay cost $((d_{\mathrm{nb,i}})_{\mathrm{nb}\in\ \mathrm{N_i}^+})$ to EV i ($N_{\mathrm{i}}{}^{+} = N_{\mathrm{i}} U i$). Let $(i)_{\mathrm{nb}}$ denote the index of EV with ith lowest delay cost to EV nb. If the requested content is not cached locally, EV i communicates with neighboring EV according to their order in N_{i}. In this case, PIT of EV i is inquired. If there is a entry in the PIT, it is updated. Otherwise, the FIB is inquired. If there is content named item in FIB, request packet is sent to the destination interface in N_i as per the defined rules. If atleast one neighbor EV in N_{i} caches the requested content, contents are retrieved from the content store. It continues untill a copy of content is found in local cache of neighboring EV $nb \in N_{\mathrm{i}}$. In this case, the transmission delay of $s_{\mathrm{c}}.d_{\mathrm{nb,i}}$ incurs.

- If no copy of the requested content is available at the local cache of neighboring EV, the EV has to request and obtain the content from LAG. It results in transmission delay of $s_c.d_{l,i}$. We assume that $d_{l,i} > d_{nb,i}$ for any EV i.

3.3.6 Content request generation model

The content request pattern of different EVs varies with their location, SoC, and purpose of the query. For example, some EVs are only interested in knowing the hourly electricity charges, while others might be interested in knowing the driving pattern of other EVs so that EVs could be properly assigned to CSs located along their travel routes. Let p_i^c denotes the probability of EV i requesting content c, where $i \in \mathscr{N}$ and $c \in \mathscr{C}$. Zipf distribution is adopted to model the probability of content. The average probability of requesting popular chunk c by EV i is :

$$p_i^c = \frac{\left[\sum_{c=1}^{\mathscr{C}} \frac{1}{r_c^\zeta}\right]^{-1}}{r_c^\zeta} \tag{3.3}$$

where, $\zeta \in [0,1]$ is the parameter of Zipf distribution. r_c is the ranking of content c in order of their popularity. The popularity of each type of content is reflected by the probability of requests for it.

3.3.7 Cache placement model

Caching the entire content of each file may result in a low successful communication rate and low hit ratio. We consider a caching model where the whole content of data is separated into chunks. Let x_i^c is the portion of content c cached by EV i, where, $c \in \mathscr{C}$ and $i \in \mathscr{N}$, which ranges in $(0,1]$. The caching decision of all EVs can be illustrated by a matrix X with size $\mathscr{C} * \mathscr{N}$ as:

$$X = (x_i^c)_{c \in \mathscr{C}, i \in \mathscr{N}} \in [0,1]^{\mathscr{C} * \mathscr{N}} \tag{3.4}$$

and it should satisfy constraint,

$$\sum_{c=1}^{\mathscr{C}} x_i^c.s_c \leq c_i \tag{3.5}$$

3.3.8 Reputation calculation

However, some selfish EVs may send wrong content into the system and restrict legitimate EVs from the valid content source. To deal with content poisoning attacks, CCN attach a digital signature in each content chunk, and nodes perform signature verification. In the proposed scheme, before extracting content from the content store of its neighbor in (N_i), the reputation value of the content provider is checked. Reputation is defined as rating an entity's trustworthiness by others based on its current and previous behaviors. Algorithm 1 defines the process

to reputation-based content delivery. To compute reputation value of EV i, each neighbor node ($nb \in N_i$) evaluates the data providing activities of this EV i, by validating the received data packets from EV i. A high reputation value signifies a lower probability of a content provider expecting a cache poisoning attack. The verification process is conducted through digital signatures. Let N_{vf}^{i} denotes the number of signature verification failure of received data packets from EV i. Then, the reputation value of the neighbor EV i computed by EV nb is given by:

$$Rep_i^{nb} = \frac{1}{2^{N_{vf}^{i}}} \tag{3.6}$$

If a reliable sharing happens between the node pair, the provider's reputation toward the consumer is increased. More verification failures tends to lower reputation value to depict that neighbor is less trustable. Next, calculate the aggregated reputation value of vehicle i from the reputation values calculated by all neighbors nb by using:

$$Agg\_rep_i^{nb} = \frac{\sum_{nb \in N_i} Rep_i^{nb}}{|N_i|} \tag{3.7}$$

However, the final reputation value of EV n is given by :

$$Rep_i(t+1) = \beta.Rep_i(t) + (1-\beta).Agg\_rep_i^{nb} \tag{3.8}$$

where, $Rep_i(t)$ denotes the reputation value of EV i in time slot t, β ($0 \leq \beta \leq 1$) is the weight factor to determine whether to give more preference to a node's previous behavior. Each node updates the evaluation value after computing the new evaluation in the current time slot to evaluate its neighbor nodes more efficiently. The proposed scheme caches legitimate content in the content store of other nodes. Let rep_o be the threshold of a reputation for accepting data packets. EV nb will not accept data packets from EV i if its reputation value is lower than rep_o. The updated reputation value of the nodes is stored as a transaction in the blockchain network.

Algorithm 1 Reputation calculation algorithm

Input: Content packets, $Rep_i(t-1)$
Output: Rep_i

1: $Rep_i^{nb} \longleftarrow$ Compute Eq.3.6
2: $Agg\_rep_i^{nb} \longleftarrow$ Compute Eq. 3.7
3: $Rep_i(t) \longleftarrow$ Compute Eq. 3.8
4: **if** $(Rep_i(t) \geq rep_o)$ **then**
5: Accept content packet provided by EV i
6: **else**
7: Drop content packet from EV i
8: **end if**

3.3.9 EVs model

The reward an EV receives from LAG is proportional to the total amount of content served to its neighbors. EV has to optimize its caching strategy by maximizing the total amount of

content shared to obtain more rewards. The proportion of content c that EV i shares with its neighboring vehicles nb is denoted as $S^c_{(nb,i)}$ and is given by:

$$S^c_{(nb,i)}(X) = min\{x^c_i, max\{0, 1 - \sum_{j=1}^{[i]_{nb}-1} x^c_{(j)_{nb}}\}\} \tag{3.9}$$

where, $i \in \mathscr{N}$ and $c \in \mathscr{C}$.
Thus, the total amount of contents shared by EV i to its neighbor nb is computed as:

$$S_{(nb,i)}(X) = \sum_{c\in\mathscr{C}} \sum_{nb\in N_i} p^c_{nb}.s_c.S^c_{(nb,i)}(X), i \in \mathscr{N} \tag{3.10}$$

For convenience, we denote $S_{(nb,i)}(X)$ as S_i.

When an EV joins for co-operative caching and content sharing, the cost incurred by the EV is the total amount of content it shares with other neighbors, and in return, LAG offers a incentive to drive EV to cooperate. Thus the utility of EV i is defined as the difference between the received reward and the cost (in terms of content sharing) incurred when it selects the contract $(S_\mathrm{i}, r_\mathrm{i})$.which can be described as:

$$U_{VH}(i) = \theta_i V(r_i) - \varepsilon S_i, i \in \mathscr{N} \tag{3.11}$$

where, $V(r_i)$ is the evaluation function of reward r_i that EV i obtains and ε is the EVs unit caching resource cost for sharing. Given the utility function in Eq. 3.11, EV chooses the bundle that minimizes its payoff. Here, we define $V(r_i) = \delta ln(1 + r_\mathrm{i})$ where δ >0 is a conversion parameter. As $V'(r_i)$>0 and $V''(r_i)$<0, $V(r_i)$ is a strictly concave function of r. Here, we set, ε=1 and δ=2.

3.3.10 LAG model

The utility function of LAG when EV i selects contract $(S_\mathrm{i}, r_\mathrm{i})$ and is defined as the difference between obtained caching resources and the reward paid to EV, as follows:

$$U_{LAG}(i) = S_i - \omega r_i \tag{3.12}$$

where, $\omega > 0$ is the LAGs unit reward cost. For co-operative caching to be beneficial for LAG, it is clear from Eq. 3.12 that $S_i - \omega r_i \geq 0$. As there are N types of EVs each with probability λ_i, the expected utility of LAG is defined as follows:

$$U_{LAG} = \sum_{i-1}^{N} \lambda_i (S_i - \omega r_i), i \in \mathscr{N} \tag{3.13}$$

3.3.11 Social welfare

The system's overall performance is examined by social welfare, which is the combination of LAG and EV utilities. The value of EVs and value of EV types equates N, the number of EVs belonging to each type is 1. Assuming that the distribution of EV type is uniform, then summing up Eqs. 3.11 and 3.12 from 1 to N we have,

$$S = \sum_{i=1}^{N} [U_{LAG}(i) + U_{EV}(i)], i \in \mathscr{N} \tag{3.14}$$

3.4 The Proposed scheme

In this section, we solve the optimization problem by jointly designing caching strategy and optimal reward contract. Then, we discuss the role of consortium blockchain in CCN-enabled V2G network as follows.

3.4.1 Conditions for contract feasibility

To ensure that the EV participates actively in content sharing, here are the several constraints which must be satisfied for the designed contract.

Definition 1: *(Individual rationality (IR))*: The contract that EV selects should ensure that $U_{EV}(i)$ is nonnegative, *i.e.*,

$$U_{EV}(i) = \theta_i V(r_i) - \varepsilon S_i \geq 0, i \in \mathscr{N} \tag{3.15}$$

To motivate an EV participation, the received reward must compensate the expense of caching resources. If $U_{EV}(i) < 0$, EV of type i does not share contents with other EVs.

Definition 2: *(Incentive compatibility (IC))*: To achieve maximum utility, EVs must prefer the contract designed specifically for their own types, *i.e.*,

$$\theta_i V(r_i) - \varepsilon S_i \geq \theta_i V(r_j) - \varepsilon S_j, i, j \in \mathscr{N}, i \neq j \tag{3.16}$$

The IC constraint depicts that the maximum utility of a type-i EV is achieved only when (S_i, r_i) contract will be selected.

Apart from IC and IR constraints, several more conditions must be satisfied:

Lemma 1: For any feasible contract (S_i, r_i), $r_i > r_j$, if and only if $\theta_i > \theta_j$ and $r_i = r_j$, if and only if $\theta_i = \theta_j$.

Proof: We prove this lemma using the IC constraint. First we prove the sufficiency, if, $\theta_i > \theta_j$ then $r_i > r_j$

According to IC constraint in Eq. 3.16, we have

$$\theta_i V(r_i) - \varepsilon S_i \geq \theta_i V(r_j) - \varepsilon S_j \tag{3.17}$$

$$\theta_j V(r_j) - \varepsilon S_j \geq \theta_j V(r_i) - \varepsilon S_i \tag{3.18}$$

where, i, $j \in \mathcal{N}$. Adding the two inequalities together, we get,

$$\theta_i V(r_i) + \theta_j V(r_j) \geq \theta_i V(r_j) + \theta_j V(r_i) \tag{3.19}$$

After merging similar items, we get

$$(\theta_i - \theta_j) V(r_i) \geq (\theta_i - \theta_j) V(r_j) \tag{3.20}$$

As $\theta_i > \theta_j$, so $\theta_i - \theta_j > 0$. Dividing both sides by $\theta_i - \theta_j$, we get $V(r_i) \geq V(r_j)$. As $V(r_i)$ is strictly increasing function of r_i, we can conclude that $r_\mathrm{i} > r_\mathrm{j}$.
Next, we prove if $r_\mathrm{i} > r_\mathrm{j}$, then $\theta_i > \theta_j$. Similarly, from IC constraints we get:

$$\theta_i (V(r_i) - V(r_j)) \geq \theta_j (V(r_i) - V(r_j)) \tag{3.21}$$

Clearly, $V(r_i) - V(r_j) > 0$ as $r_\mathrm{i} > r_\mathrm{j}$ and $V(r_i)$ is strictly increasing function of r_i. Then dividing both side of inequality by $(V(r_i) - V(r_j))$, we can derive that $\theta_i > \theta_j$.
Thus, an EV of high type should receive more rewards than an EV of low type. If two EVs receive the same rewards, they must belong to the same type.

Definition 3: *(Monotonicity)*: For any feasible contract of EV type-i, (S_i, r_i), the reward r_i follows:

$$0 \leq r_1 \leq ..... \leq r_i \leq .... \leq r_N, i \in \mathcal{N} \tag{3.22}$$

Monotonicity implies that an EV of the higher type, *i.e.*, EV with more battery, high reputation, and more cache capacity deserves more rewards for a feasible contract. Since $r(S)$ increases as S, the following proposition can be made from Definition 3.

Proposition 1: As a strictly increasing function of r, the contents shared by EV satisfy the following constraints

$$0 \leq S_1 \leq ..... \leq S_i \leq .... \leq S_N, i \in \mathcal{N} \tag{3.23}$$

3.4.2 Optimization problem formulation

Next, we formulated the system optimization problem given the contract feasibility constraints. Under the information asymmetry, only information at LAG is probability λ_i with which a particular EV might belong to θ_i. The objective of LAG is to maximize the utility function by constructing the optimal contract. Therefore, the maximization problem is formu-

lated as:

$$\begin{aligned}
&\max_{(X,r)} \sum_{i=1}^{N} \lambda_i [S_i(X) - \omega r_i] \\
&\text{s.t.} \quad C1 : x_i \in [0,1]^{1*C} \\
&C2 : \sum_{c=1}^{\mathscr{C}} x_i^c . s_c \leq c_i \\
&C3 : \theta_i V(r_i) - S_i \geq 0 \\
&C4 : \theta_i V(r_i) - S_i \geq \theta_i V(r_j) - S_j \\
&C5 : 0 \leq r_1 \leq ..... \leq r_i \leq .... \leq r_N \\
&i, j \in \mathscr{N}, i \neq j
\end{aligned} \tag{3.24}$$

where, $C1$ and $C2$ are cache state and cache size constraints respectively of N EVs, $C3$, $C4$ and $C5$ are IR, IC and monotonicity constraints respectively (Taking ε=1).

3.4.3 Optimal cache strategy

All EVs optimize their caching strategy by sharing maximum content with neighboring EVs to get maximum utility. We design a novel caching algorithm using content popularity to achieve the optimal caching strategy for each EV. The details of the optimal caching strategy are described in Algorithm 2. The input parameters are total EV N, total content C, site of content c denoted by s_{c} and total cache space of each EV, *i.e.*, c_{i}. Firstly, we start with the cache state of each EV as zero. Then, using the popularity matrix of caching content in neighbors, calculate the cache percentage that each EV stores from all the contents having distribution as popularity matrix (P_{nb}^{c}) is the randomly generated popularity matrix representing the neighboring vehicle request heterogeneity for content c at time slot t). After computing cache state of all EVs, calculate S_{i} using Eqs. 3.9 and 3.10.

Algorithm 2 Optimal caching strategy based on content popularity

Input: N, C, s_{c}, c_{i}
Output: X, S_{i}

1: Initialize caching state: $x_i^c \leftarrow \frac{P_{nb}^{c} * c_{\mathrm{i}}}{s_{\mathrm{c}}} \; \forall \, i \in \mathscr{N}, c \in \mathscr{C}$
2: Obtain optimal caching strategy, X, calculate S_{i} by using eq 3.9, 3.10.
3: Return X and S_{i}

3.4.4 Optimal reward contract

After obtaining the optimal caching strategy, the contract $(S_{\mathrm{i}}(X), r_{\mathrm{i}})$ can be rewritten as $(S_{\mathrm{i}}, r_{\mathrm{i}})$. Then, the optimization problem in Eq. 3.24 can be converted into the problem as

follows:

$$\begin{aligned}
&\max_{(r)} \sum_{i=1}^{N} \lambda_i [S_i - \omega r_i] \\
&s.t. \quad C1 : \theta_i V(r_i) - S_i \geq 0 \\
&C2 : \theta_i V(r_i) - S_i \geq \theta_i V(r_j) - S_j \\
&C3 : 0 \leq r_1 \leq ..... \leq r_i \leq .... \leq r_N \\
&i, j \in \mathscr{N}, i \neq j
\end{aligned} \tag{3.25}$$

Notably, this is not a convex optimization problem; however, we can take the following measures for a solution: *Step 1) Reduce IR constraints*: From 3.25 ($C1$), it could be observed that there are total of N IR constraints. Also, from Eq. 3.2, we know that $\theta_1 < ..... < \theta_i < ..... < \theta_N$. By using IC constraints, we have,

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_1) - S_1 \geq \theta_1 V(r_1) - C_1, \tag{3.26}$$

Thus, if the IR constraints of type-1 EV is satisfied, the other constraints will automatically hold.

Step 2) Reduce IC constraints: The IC constraints are made up of downward IC (DIC) and upward IC (UIC). The IC constraints between type-i and type-j, $j \in \{1, 2, ...i-1\}$ EV are called DIC. In particular, the IC constraints between type-i and type-$(i-1)$ EV is called local DIC (LDIC). Similarly, UIC represents constraints between type-i and type-j, $j \in \{i+1, ..., N\}$ EV and LUIC represents constraints between type-i and type-$i+1$. First, we prove that all DICs can be reduced.

Proof: As there are total of N EVs, there exist total of $N(N-1)$ IC constraints. Considering three types of EVs that follows: $\theta_{i-1} < \theta_i < \theta_{i+1}$, we have two following LDICs:

$$\theta_{i+1} V(r_{i+1}) - S_{i+1} \geq \theta_{i+1} V(r_i) - S_i \tag{3.27}$$

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_{i-1}) - S_{i-1} \tag{3.28}$$

Combining Eq. 3.28 with Lemma 1, where we have depicted that $r_i \geq r_j$ whenever $\theta_i \geq \theta_j$, the second inequality becomes:

$$\theta_{i+1}[V(r_i) - V(r_{i-1})] \geq \theta_i [V(r_i) - V(r_{i-1})] \geq S_i - S_{i-1}, \tag{3.29}$$

$$\theta_{i+1} V(r_{i+1}) - S_{i+1} \geq \theta_{i+1} V(r_i) - S_i \geq \theta_{i+1} V(r_{i-1}) - S_{i-1} \tag{3.30}$$

Thus, we have

$$\theta_{i+1} V(r_{i+1}) - S_{i+1} \geq \theta_{i+1} V(r_{i-1}) - S_{i-1} \tag{3.31}$$

Therefore, if for type-i EV the LDIC holds, the *IC* with respect to type-$(i-1)$ EV holds as well. This process can be extended downward from type-$(i-1)$ to type-1 EVs to prove that all

DICs hold,

$$\begin{aligned}\theta_{i+1}V(r_{i+1}) - S_{i+1} \geq & \theta_{i+1}V(r_{i-1}) - S_{i-1} \\ & \geq ... \\ & \geq \theta_{i+1}V(r_1) - S_1 \\ & i = 1, 2, ..., N-1\end{aligned} \tag{3.32}$$

Thus, we have proved that with LDIC constraints, all the DICs hold, that is:

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_j) - S_j, N \geq i \geq j \geq 1 \tag{3.33}$$

Next, we prove that all UICs will be held if LUIC is satisfied. *Proof:* From IC constraints:

$$\theta_{i-1}V(r_{i-1}) - S_{i-1} \geq \theta_{i-1}V(r_i) - S_i \tag{3.34}$$

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_{i+1}) - S_{i+1} \tag{3.35}$$

Combing Eq. 3.35 with Lemma 1, we get

$$S_{i+1} - S_i \geq \theta_i[V(r_{i+1}) - V(r_i)] \geq \theta_{i-1}[V(r_{i+1}) - V(r_i)] \tag{3.36}$$

$$\theta_{i-1}V(r_{i-1}) - S_{i-1} \geq \theta_{i-1}V(r_i) - S_i \geq \theta_{i-1}V(r_{i+1}) - S_{i+1}, \tag{3.37}$$

Thus, we have,

$$\theta_{i-1}V(r_{i-1}) - S_{i-1} \geq \theta_{i-1}V(r_{i+1}) - S_{i+1} \tag{3.38}$$

Therefore, if for a type-$(i-1)$ EV, the IC with respect to type-i EV holds, then all UICs are also satisfied. The same process can be extended upwards from type-$i+1$ to type-N EVs to prove that all the UICs hold,

$$\begin{aligned}\theta_{i-1}V(r_{i-1}) - S_{i-1} \geq & \theta_{i-1}V(r_{i+1}) - S_{i+1} \\ & \geq .... \\ & \geq \theta_{i-1}V(r_N) - S_N \\ & i = 2, ..., N.\end{aligned} \tag{3.39}$$

Thus, we have completed the proof that with the LUIC constraints, all UICs hold, *i.e.*,

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_j) - S_j, 1 \leq i < j \leq N \tag{3.40}$$

Indeed, with the monotonicity condition r_{i-1}<r_i, the LDIC:

$$\theta_i V(r_i) - S_i \geq \theta_i V(r_{i-1}) - S_{i-1} \tag{3.41}$$

can imply that the LUIC:

$$\theta_{i-1}V(r_i) - S_i \leq \theta_{i-1}V(r_{i-1}) - S_{i-1} \tag{3.42}$$

can be fulfilled and can be diminished. Thus, we proved that with all LDIC, all UICs are reduced. Therefore, the optimization problem reduces to:

$$\begin{aligned} &\max_{(r)} \sum_{i=1}^{N} \lambda_i [S_i - \omega r_i] \\ &s.t. \quad \theta_i V(r_i) - S_i = 0 \\ &\theta_i V(r_i) - S_i = \theta_i V(r_{i-1}) - S_{i-1} \\ &0 \leq r_1 \leq ..... \leq r_i \leq .... \leq r_N \\ &i = 1, 2, ..., N \end{aligned} \tag{3.43}$$

The above-formulated contract optimization is under convex optimization problem and is solved by the Langrangian multiplier method. To solve this maximization problem, we first frame and solve the relaxed problem by ignoring the monotonicity condition and then solve it using the standard procedure of the Langrangian multiplier. Next, we check whether the obtained solution to the relaxed problem follows the monotonicity constraint. The Lagrangian function is constructed as follows:

$$\begin{aligned} L(r_i, \gamma_1, \gamma_2) = & \\ &\sum_{i=1}^{N} \lambda_i [S_i - \omega r_i] + \gamma_1 [\theta_1 V(r_1) - S_1] + \\ &\gamma_2 [\theta_i V(r_i) - S_i - \theta_i V(r_{i-1}) + S_{i-1}] \\ &= F + \gamma_1 [\theta_1 V(r_1) - S_1] - \gamma_2 [S_i - S_{i-1}], \\ &i = 1, 2, ..., N \end{aligned} \tag{3.44}$$

where, $F = \sum_{i=1}^{N} \lambda_i [S_i - \omega r_i] + \gamma_2 \theta_i [V(r_i) - V(r_{i-1})]$, $V(r_i)$=$\delta \ln(1 + r_i)$, δ=2
Let

$$\frac{\partial L}{\partial r_i} = \frac{\partial F}{\partial r_i} = -\omega + \frac{\gamma_2 \theta_i \delta}{1 + r_i} = 0, \tag{3.45}$$

$$\frac{\partial L}{\partial \gamma_1} = \theta_1 V(r_1) - S_1 = 0, \tag{3.46}$$

we get

$$r_i = \frac{\gamma_2 \delta \theta_i}{\omega \lambda_i} - 1 \tag{3.47}$$

$$\theta_1 \delta ln(1 + r_1) - S_1 = 0, \tag{3.48}$$

thus,

$$r_i = \frac{\theta_i e^{\frac{S_1}{\delta\theta_1}}}{\theta_1} - 1, i \in \mathcal{N} \tag{3.49}$$

The obtained solution satisfies the monotonicity constraint, and hence it is an optimal solution. We denote the optimal solution as:

$$r_i^* = \frac{\theta_i e^{\frac{S_1}{\delta\theta_1}}}{\theta_1} - 1, i \in \mathcal{N} \tag{3.50}$$

3.4.5 Computational complexity analysis

As there are N pairs of optimization variables, the computational complexity of constraint transformation is $O(N)$. After constraint transformation is completed for each type of θ_i, the computational complexity to solve eq. 3.43 is $O(N)$. Hence, the overall computational complexity of the proposed scheme is $O(N^2)$.

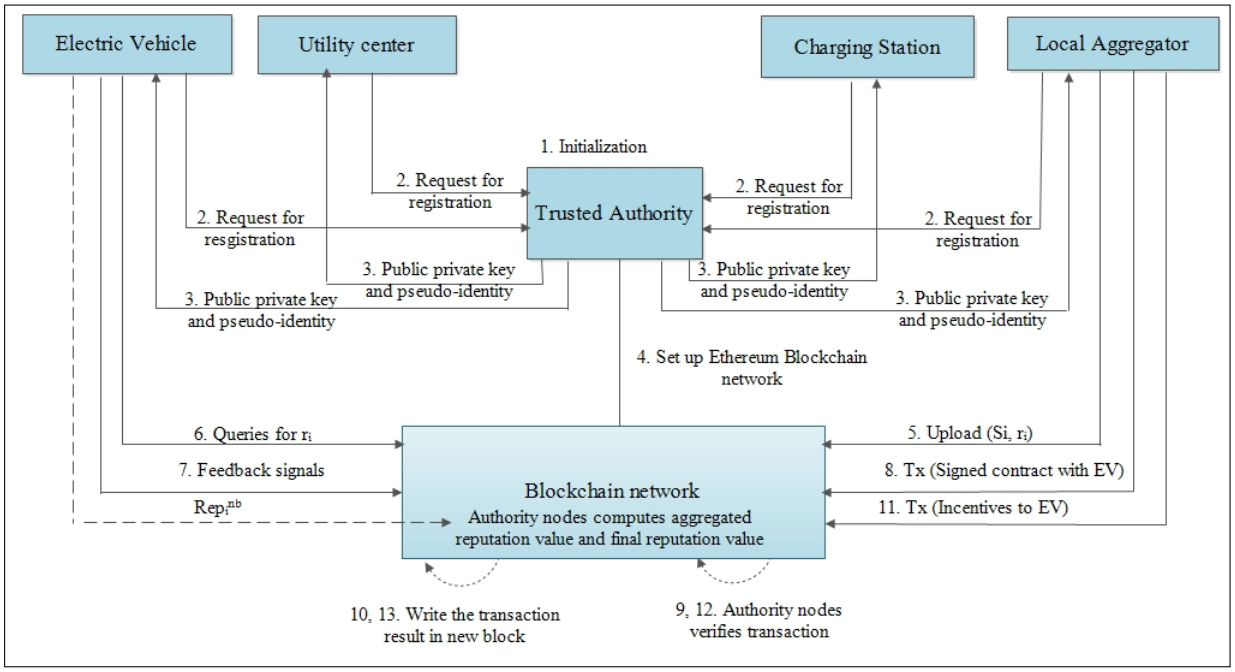

Figure 3.2: Flow of information among different entities in the proposed blockchain network

3.4.6 Consortium blockchain network

In the beginning, nodes send a request to a trusted authority (TA) to get an account address and a pair of public and private keys. Every entity on the network has a unique ID (account address).

After designing the reward contract for a type-i EV, LAG upload the contract to the blockchain system. First, EV queries the blockchain system for the current unit reward price for caching

Algorithm 3 PoA Consensus algorithm

Input: $\mathcal{N}$, $\mathbb{C}$: EV's and CS's respectively, LAG, UC, $Tx's$
Output: Valid block

procedure AUTHORITY NODE_SELECTION($\mathcal{N}$, $\mathbb{C}$, LAG, UC)
    /* For the nodes in $\mathbb{C}$ */
    Set threshold value: PP^{REQ}, M^{REQ},
    for (i=1, i≤size($\mathbb{C}$), i++) **do**
        Enquire $PP_{\text{i}}{}^{\text{AVL}}$, $M_{\text{i}}{}^{\text{AVL}}$
        if $PP_{\text{i}}{}^{\text{AVL}} \geq PP^{\text{REQ}}$ && $M_{\text{i}}{}^{\text{AVL}} \geq M^{\text{REQ}}$ **then**
            Put i in list $\mathbb{L}$
        end if
    end for
    /* For the nodes in $\mathcal{N}$ */
    Set threshold value: $PP^{\text{REQ\_EV}}$, $M^{\text{REQ\_EV}}$, $Rep^{\text{REQ\_EV}}$
    for (n=1, n≤size($\mathcal{N}$), n++) **do**
        Enquire $PP_{\text{n}}{}^{\text{AVL}}$, $M_{\text{n}}{}^{\text{AVL}}$, Rep_n
        if $PP_{\text{n}}{}^{\text{AVL}} \geq PP^{\text{REQ\_EV}}$ && $M_{\text{n}}{}^{\text{AVL}} \geq M^{\text{REQ\_EV}}$ && $Rep_n \geq Rep^{\text{REQ\_EV}}$ **then**
            Put n in list $\mathbb{L}$
        end if
    end for
    Put LAG and UC in list $\mathbb{L}$
    /* Compute total number of nodes in the network*/
    M= size($\mathbb{C}$) + size($\mathcal{N}$) +2
    /*Compute number of authority nodes*/
    $\mathbb{AN} = (M - \frac{M}{2} + 1)$
    Randomly select $\mathbb{AN}$ from list $\mathbb{L}$
end procedure
AN^{first} in round robin from $\mathbb{AN}$ collects $Tx's$
for all $Tx's$ in block **do**
    Calculate MRH
end for
AN^{first} signs, generate block and broadcast to secondary ANs for verification
Each secondary AN receive the block and check its correctness
if the new block is legal **then**
    audit and send the result including <h, d, s>to each other, where, h is height, d is hash of the block, and s is the signature of this node
    if the received audit result >$\frac{AN-1}{3}$ + 1 **then**
        Each node compare its work with others and send positive message to the leader <h, d, s>
        if the leader receive positive message from more than $\frac{2(AN-1)}{3}$ +1 **then**
            Append the block to the global chain and broadcast to all nodes
        else
            The leader request to nodes who sent negative message to check their audit once again
        end if
    else
        Ignore
    end if
    if time expires and no consensus is reached **then**
        Abandon this consensus
    end if
else
    Discard the new block
end if

services according to its type before caching any data. Then, by analyzing the contracts, EV sends feedback signals to signify its willingness to participate as per the estimated utility. After receiving the feedback from EVs, LAG signs the contract with EVs in a transaction which accepts it, and this transaction is stored in the blockchain network for audit. EVs share their content with neighboring nodes according to the content-sharing requirements in the signed contract. Finally, EVs obtain the corresponding incentives from the contract publisher (LAG).

Similar to a caching-related transactions, reputation related transaction is used to record reputation value of a content provider EV by a content receiver. The blockchain network ensures accurate reputation calculation thus, it improves the detection of cache poisoning attacks. All reputation values are recorded in the blockchain network to remain unforgeable. After verifying content from the content provider, the content receiver updates its reputation opinion for content providers (Rep_i^{nb}). These reputation opinions with digital signatures are recorded as transactions and uploaded to the verifier of the blockchain network. The authority node computes the aggregated reputation value using reputation value from the past, and final reputation value of the node is computed. The authority node puts the reputation value of the node into a data block and adds the block to the blockchain network after block verification and executing PoA consensus algorithm. Finally, a content requester can choose a reliable provider having high reputation with the help of blockchain network. The whole process of a blockchain network is depicted in Fig. 3.2.

In a blockchain network, all recorded transactions are verifiable and available to each participants of the system. After successful transaction validation, block is appended to the blockchain network. A linked list structure of the block makes reference of a block to its parent block via a cryptographic hash link to achieve immutability of recorded transactions. The consensus process ensures that all participants mutually agree for transaction validation and synchronization in a digital ledger. We use Proof-of-Authority (PoA) consensus algorithm to validate network transactions. The hash is calculated through SHA-256 cryptographic algorithm.

3.4.6.1 Miner node selection

ANs are responsible for collecting the transactions, creating and adding the block onto the blockchain. PoA enables a fair distribution of authorities for block creation among selected ANs. It relies on round-robin where a primary authority proposes a block in each round. The other selected ANs validate the proposed block and add that block to the blockchain network. If the primary authority proposes an invalid block, then the other validators call for voting. If the majority of the votes go against the authority, the validator is declared malicious. Unlike PoW, PoA is lightweight and has higher throughput. Also, as there is no mining competition, PoA supports better CPU utilization.

PoA chooses ANs from EVs and CSs, LAG, and utility center (UC) as follows. Let there are a total of M nodes, then a total of $M - \frac{M}{2} + 1$ are selected as ANs. For CS, their processing power (in GHz), and available memory (in Kb) are taken into account. The parameters of a CS in $\mathbb{C}$, *i.e.*, memory available (M^{AVL}), and processing power (PP^{AVL}) are compared with required processing power (PP^{REQ}), and memory (M^{REQ}). Then, the shortlisted nodes are added to the list $\mathbb{L}$. For EVs in $\mathscr{N}$, their available memory, processing power, and reputation value are matched against the required processing power ($PP^{\text{REQ\_EV}}$), required memory ($M^{\text{REQ\_EV}}$), and required reputation value ($Rep^{\text{REQ\_EV}}$). It is assumed that there is only one LAG and one utility center in a specific region, having enough computing capabilities. So, these are added to the set of authority nodes in each round.

In each round of the consensus process, the primary AN collect transactions, packs them into a block, and broadcasts to secondary AN with its signature for verification and validation. The secondary ANs check the block's validity and broadcast their audit results with digital signatures to other secondary AN for combined supervision. After receiving the audit outcomes, each secondary AN differentiate its observation with others. If the transaction is valid and the signature is valid, they vote for the block and reply to primary AN. The reply message has the audit outcome, signature, and outcome results. The primary AN analyzes the received reply packets from secondary ANs. If a valid vote is received from $\frac{3}{4}$ of the secondary AN, the block is appended to the global chain.

3.4.6.2 Block creation and validation

After miner node is selected, next, blocks are generated and validated before adding to blockchain network. AN^{first} first generates a block by solving a problem and starts the verification by transferring the proof to rest authority nodes for auditing. If the output obtained from AN^{first} are in consensus, the leader AN appends the block to network; otherwise block is discarded. The block creation process is explained as follows.

- First, AN^{first} computes the transaction received and combines it into a blockchain transaction T_L.
- Next, AN^{first} computes Merkle root hash (MRH) of the transactions. The previous block hash (H_{pre}) is taken from blockchain and block header (BH) is generated from MRH, H_{pre}, random number (rn), and timestamp (ts).
- A fresh message is formed by appending padding bits to generate a fixed length message. Further, these bits are fed to a SHA-256 algorithm to generate a 256-bit message digest.
- To add block in blockchain, majority of the authority nodes have to be approve about the block value. AN^{first} takes the lead and broadcasts the obtained results to other authority nodes.

3.4.6.3 Communication cost

The communication cost is calculated with message bits transferred. AN^{first} extracts 160 bits of H_{pre} and MRH is also of 160 bits. So, BH comprises of [160+160+32+32] 384 bits (assuming that rn and ts both takes 32 bits each). Now, 128 padding bits are attached to generate a 512-bit long message which is further given to SHA-256 that generates 256-bit message digest. The final value takes 256 bits. The verification bit is of 1 bit. So, to communicate the data for block validation to other authority nodes, the overall cost is [256+1] 257 bits.

3.4.6.4 Computation cost

While forming and validating a block, the operations used are addition, hashing, and append operations. Let's assume that the time consumed for these operations is 1 ms, 2.5 ms, and 0.5 ms, respectively. The validation process involves 5 append operations, $n-1$ additions operations (for n attributes in Merkle tree), and 3 hash operations. Hence, the computation time for validating a block with 100 transactions is [5*0.5+ 99*1+ 3*2.5] 109ms.

3.5 Performance evaluation

3.5.1 Numerical settings

We have used ndnSIM (NS3-based simulator) to analyze the performance of co-operative caching for V2G. Also, we have used Python to evaluate the traces of files generated by ndnSIM. Simulation results have been performed on Intel Core 5 Duo CPU at 2.4 GHz, with 8 GB DDR3 SDRAM. We assume N=100 and evaluated the simulations results with ten types of EV (according to their remaining battery, reputation value, and high cache capacity), wherein the probability of an EV belonging to a certain type is $\frac{1}{10}$ =0.1. The different type of EVs has a fixed cache capacity between 20Mbit-50Mbit. To verify the efficiency of the proposed model, the two-way multi-lane road network having constant (average) vehicle moving speed and a LAG with a coverage radius of 300m is considered. We have evaluated the performance of the proposed model based on a real dataset available at [204]. We have taken a network that consists of 8 compromised EVs for reputation calculation. The compromised EVs provide poisonous cached contents to other nodes. The initial reputation of each EV is set as 0.6. SHA-256 is used as a secure hashing algorithm. All the results are computed by taking an average of 100 runs. The block size limit is set as 1 MB and the average block generation time is 10 s. Parameters list is given in Table 3.2.

Table 3.2: List of parameters

Parameters	Value
Number of vehicles	10
Number of contents	50-80
Content size	5 MB-15 MB
Zipf parameter	(0.5,1.2]
Average speed of EVs	10m/s-30m/s
Coverage radius of LAG	300m
Cache capacity	20M-50M
ω	0.015
δ	2
ε	1
Block size limit	1 MB
Average block generation time	10 s
Consensus algorithm	PoA

3.5.2 Contract feasibility

3.5.2.1 Incentive compatibility

Fig. 3.3 represent the utilities of EVs with type-2, type-4, type-6 and type-8. It has been observed that each type of EV obtains the maximum utility while choosing the contract item designed for its type, which also indicates IC constraint.

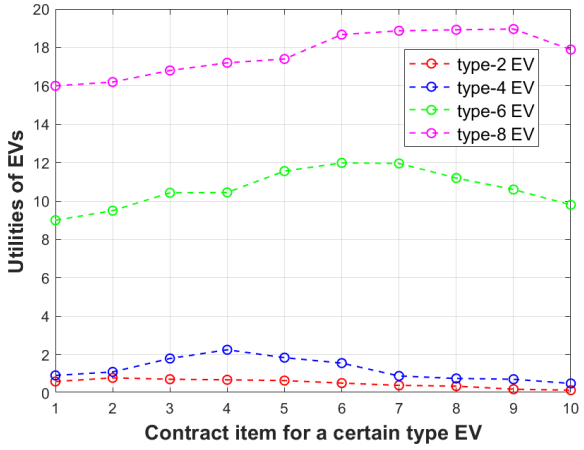

Figure 3.3: Utilities of EVs under different contract items

3.5.2.2 Monotonicity

Fig. 3.4 shows the reward of different type of EVs provided by LAG. We compared the result of the proposed scheme with an optimal contract under no information asymmetry (where LAG is unaware of the types of EVs). It has been observed from Fig. 3.4 that the reward

provided to EVs is a strictly increasing function of EV type as the more content an EV serves to others, the more reward it deserves for a feasible contract. Also, it can be depicted that EVs achieve a high reward when there is information symmetry in the system as LAG has full knowledge of EV types.

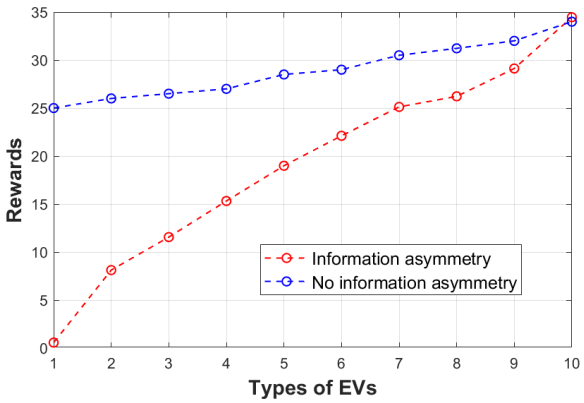

Figure 3.4: Reward provided by LAG to different type EVs

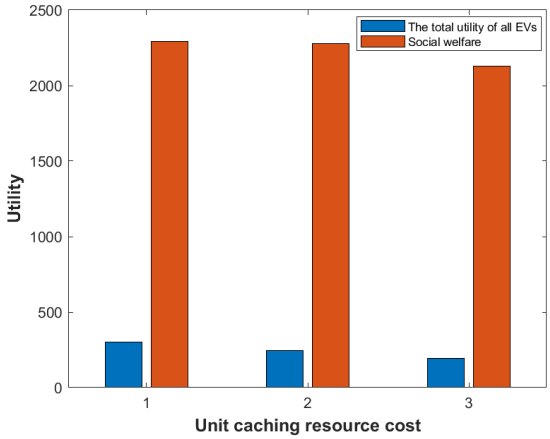

Figure 3.5: Utility comparison under different unit caching resource cost

3.5.3 Impact of unit caching resource cost parameter

In the proposed scheme, there exist overhead, including maintenance cost and network resource cost in caching resources for sharing. Fig. 3.5 shows that the utility of EVs decreases with the increase in unit caching resource cost (ε). The total cost of the content sharing increases with an increase in value of ε, which decreases the utility value for an EV. Similarly, the social welfare of the proposal decreases with an increase in value of ε, the utility value for an EV decrease, which further reduces the value of social welfare.

3.5.4 Impact of cache capacity

We compared the performance of our scheme with a NDN based cache strategy (PICS) [23] and a non-incentive scheme (P-VCCN) [205] (in terms of content request delay saved with varying cache capacity and EVs density) which are described as follows:

- *Probability based caching in VCCN (P-VCCN)*: Authors in [205] proposed a cache strategy for vehicular CCN that models the data cache probability based on nodes movement, cache occupancy, content popularity, and similarity. To define the level of each attribute, weight vector calculations are performed. However, for our proposal, vehicle location is constant for an interval, so we neglected the weight of the node's movement for simulation. Notably, the authors in this scheme have not considered the case of co-operative caching.

- *Popularity incentive caching for Vehicular NDN (PICS)*: Authors in [23] define a popularity incentive caching where base station incentivize vehicles who execute cache sharing with other nodes. The Stackelberg game with rational utilities is used to model the interaction between base station and vehicles. PICS considers the request for other neighboring nodes while making cache decisions. Moreover, to increase the diversity of content caching, content popularity differences between different vehicles are considered.

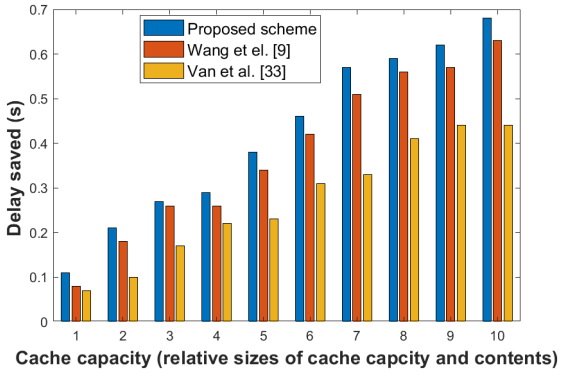

Figure 3.6: Average delay saved when EVs cache capacity varies

Fig. 3.6 shows the impact of cache capacity on the delay saved for fixed EV density. The delay saved for all three schemes increases with respect to EV cache capacity. With the increase in cache capacity, EVs can cache more content units, which leads to shorter content request delays. In simulations, the reduced content request delay is defined as:

$$
\begin{aligned}
D_{saved} = \sum_{c \in \mathscr{C}} p_i^c . s_c . (d_{l,i} - \\
\sum_{nb=1}^{|N_i|} [\prod_{m=0}^{nb-1} (1 - x_m^c)] x_{nb}^c (d_{nb,i}) - \\
[\prod_{m=0}^{|N_i|} (1 - x_m^c)] (d_{l,i}))
\end{aligned}
\tag{3.51}
$$

In Eq. 3.51, if content c is cached in local cache of EV i then $x_0^c = 1$ in last two terms which results in 0 for them. In this case, revenue saved is $s_c.(d_{l,i})$. Otherwise, either one of the last two term becomes nonzero. If $\hat{nb}$ EV in N_i caches a copy of content and any other EV with a transmission delay lower than $\hat{nb}$ fails to cache this content (*i.e.*, $x_{\hat{nb}}^c = 1$ and $x_{nb}^c = 0$), $\forall\, \hat{nb}$, $nb \in N_n$ and $nb_{(\mathrm{l})\mathrm{n}} < \hat{nb}_{(\mathrm{l})\mathrm{n}}$. We have then the term $\prod_{m=0}^{nb-1}(1-x_m^c)x_{nb}^c$ equals to one for $nb = \hat{nb}$ and 0 for any other neighbor EV $nb \in N_n$ while the term $\prod_{m=0}^{|N_i|}(1-x_m^c)=0$. If no copy of the content c can be found in either the EV i or any of the neighboring EV then the term $\prod_{m=0}^{|N_i|}(1-x_m^c)=1$, while $\sum_{nb=1}^{|N_i|}[\prod_{m=0}^{nb-1}(1-x_m^c)x_{nb}^c]=0$.

It has been observed from Fig. 3.6 that the proposal outperforms the other two schemes. It is because that the proposed scheme has considered the request probability for each EV while making cache decisions, and the formula for computing delay saved considers the distributive cooperative caching. However, in PICS, the delay saved is defined as $\sum_{c\in\mathscr{C}}\sum_{n\in\mathscr{N}} x_{cj}p_{cj}D$ where, D is the transmission delay between BS and vehicle. PICS also considers the request probability of other vehicles, but while computing delay saved case of cooperative caching is not reflected in the given formula. In contrast to these two schemes, in P-VCCN, global content popularity is considered.

Also, Fig. 3.7 depicts the comparison of the proposed scheme with PICS and P-VCCN with varying EVs density. It has been observed from Fig. 3.7 that as the number of EVs increases, there is an increase in opportunistic caching percentage, which leads to shorter content request delay.

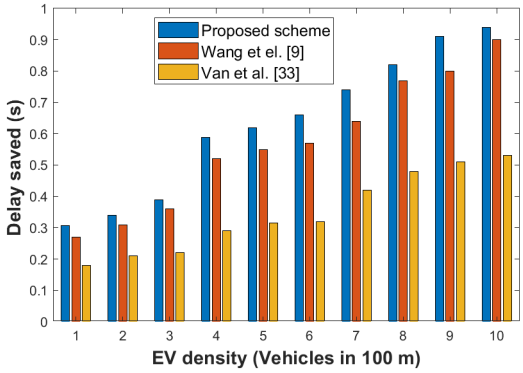

Figure 3.7: Average delay saved when EVs density varies

3.5.5 Impact of total number of types of EVs

We also compared the profit of a LAG from the contract model and the Stackelberg game models from [206]. The contract theory model is superior than that of the Stackelberg game model. Fig. 3.8 depicts that the profit of LAG enhances with the total number of verifier types. More EV types bring more EVs and contract item choices for an EV with high secure content sharing willingness. Also, Fig. 3.9 depicts that the value of social welfare obtained by the contract theory model and the Stackelberg game model increases with the increase in the total

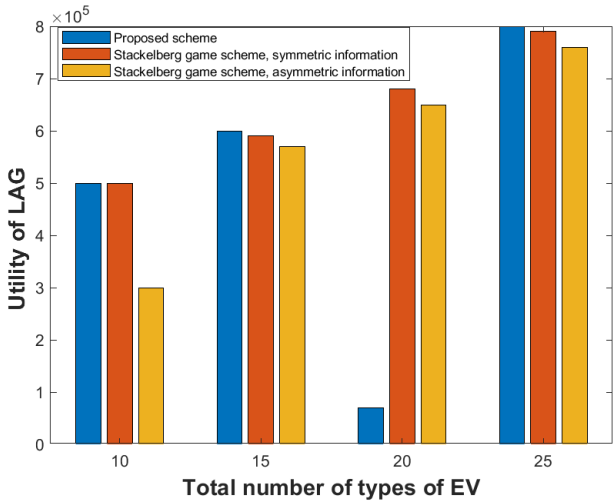

Figure 3.8: Utility of LAG under total number of EV types

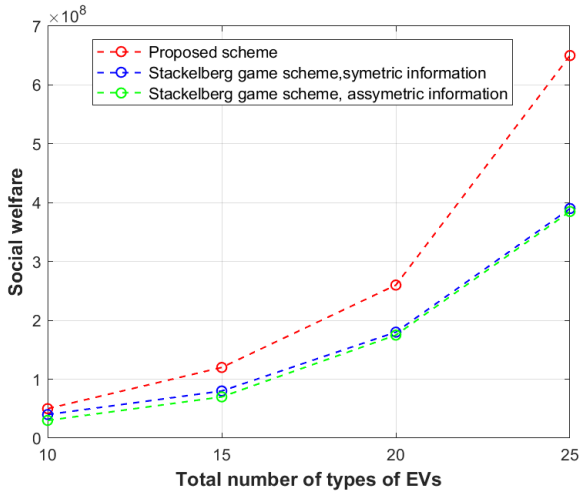

Figure 3.9: Social welfare of different incentive mechanisms

number of verifier types. However, the social welfare of the contract theory model is higher than the Stackelberg game models because in a monopoly market, the LAG (acts as monopolist) provides limited contract items to get more benefits from EVs. The proposed contract provides profit to the utility of LAG. In contrast, in the Stackelberg game model, rational EVs can optimize their individual utilities, which leads to less profit for LAG. Therefore, LAG obtains higher profit over the Stackelberg game models. Also, it can be analyzed that the Stackelberg game model with symmetric information has better performance over the asymmetric information case as the LAG in the Stackelberg game with asymmetric information can optimize its profit better because of having some information about the actions of EVs.

3.5.6 Throughput and latency

Throughput is defined as the number of transactions processed by network nodes per second (tps). On the other hand, latency is the time lag between any transaction sent to the blockchain

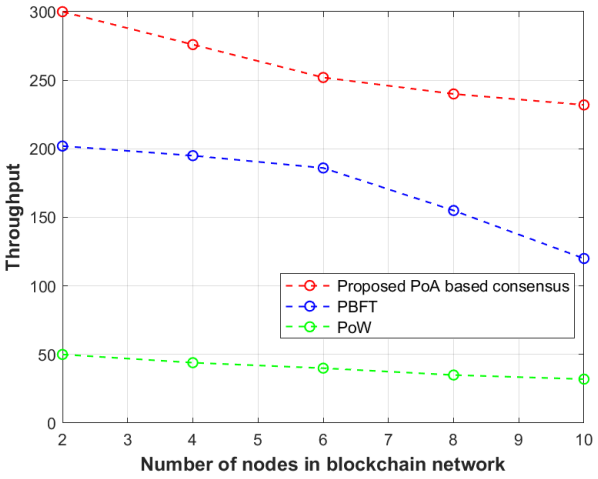

Figure 3.10: Throughput vs. number of nodes

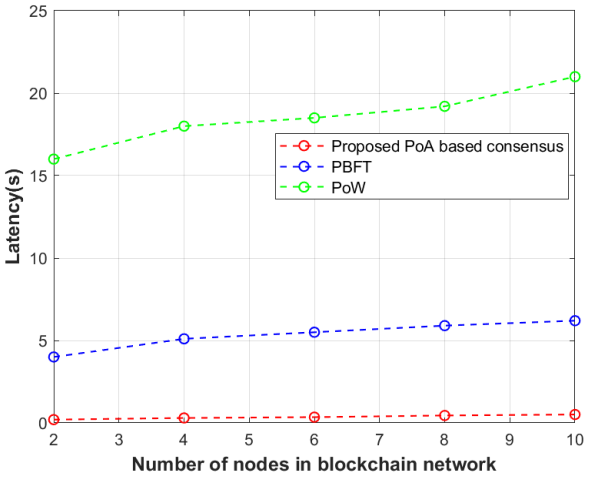

Figure 3.11: Latency vs. number of nodes

network and the response received by network nodes. Simulation results in Fig. 3.10 reveal that the proposed PoA based consensus scheme provides superior throughput than Proof-of-Work (PoW) and Practical Byzantine Fault Tolerance (PBFT) consensus algorithm. Unlike PoW, PoA doesn't require high computational resources and possesses a great speed of validating transactions. Also, different from PBFT, PoA exchanges fewer messages hence providing better performance. Moreover, compared to PoW, PBFT shows reduced energy consumption usage as in PBFT miners don't have to solve PoW hashing algorithms for each block.

Fig. 3.11 shows that the transaction processing latency of the proposed PoA based consensus algorithm is reduced by 90.5% and 50.9% compared to PoW and PBFT, respectively.

3.6 Summary

This chapter proposes a secure content delivery framework for CCN based V2G network by amalgamating consortium blockchain and contract theoretical modeling. First, we propose

a consortium blockchain-based co-operative caching mechanism for content delivery which secures the caching-related transactions by taking advantage of security decentralization and trust. Next, under information asymmetry, we have designed an reward method for in-network caching scheme based on the framework of contract theory. We formulate and jointly design a caching strategy and reward contract optimization problem to maximize the social welfare. The social welfare maximization problem was solved using standard procedure of Lagrangian function. We evaluated the performance of the proposal by varying values of different weight parameters to analyze the impacts on the utility of EVs, LAG and social welfare. The simulation results show that the proposed consensus scheme improves throughput and latency of the blockchain based V2G network. Further, it is demonstrated that the proposed scheme is incentive compatible and outperforms other schemes for social welfare. Also, the scheme ensures resilience from cache poisoning attacks in content delivery. The next chapter discusses a matching game for content requesters and providers.

Chapter 4

A Blockchain based Matching Game for Content Sharing

4.1 Introduction

In CCN, a requester node generates an request packet having the desired content name and send it to the network. When a data provider receives the request packet with a content name, it transmits it back to the requester, and all the in between nodes cache the received data packet. However, with multiple requesters and providers for content, requesters may face competition against the best sharing services, and providers must filter the incoming requests for low energy consumption. Also, in terms of the mobility of vehicles, the content duration is limited, leading to a limited transaction time while obtaining content. Hence, a matching model is required to match content requesters and providers for an efficient content sharing. Matching theory is a popular mathematical concept having applications in various domains, including communication networks, college admission programs, and kidney exchange programs [34]. However, most of the existing matching game solutions are centralized, having a single failure point, as they need a coordinator to regulate the matching market [207]. Also, the centralized solutions may lead to congested matching operations, which harms the scalability of the system. To solve this issue, decentralized matching algorithms with lower complexity and limited information exchanges are required. Market decentralization implies that we don't need a centralized party to evaluate the convergence for stability. In this context, we formulate a universal matching problem which incorporates a cardinal utility market, taking into account market decentralization. In particular, we have used a blind matching algorithm [151] to produce a match between content requesters and providers. In the proposed matching game, pairing decisions are made with a holistic perspective, promoting mutual benefits for all parties involved. We demonstrate that the proposed model encompasses linear and generalized assignment game frameworks.

Moreover, as content acquisition via CCN communication connects different network nodes, so it may bring trust issues among them, as both content requesters and providers have concerns

about untrusted users connected with them. Also, some nodes may not be willing to participate as providers due to their concerns about privacy leakage or high energy consumption. This situation may create an imbalance in content supply and demand among network nodes. To address these challenges, we have used blockchain technology to record the content related transactions, which are written into blocks after being verified by the consensus algorithm.

4.2 Contributions

The following contributions of this work are presented.

- We propose a decentralized matching scheme for secure content sharing in a V2G network. In particular, we implemented a blind matching algorithm (BLMA) to solve the formulated matching problems on the blockchain network by designing a smart contract to match content providers and requesters. We then provide a theoretical analysis on the stability, convergence, and complexity of the proposed scheme.
- We evaluated the matching preferences of content requesters and providers, which jointly consider latency, reliability for requesters and energy consumption, social reciprocity index of content providers. We model the joint issue of the content requester and provider pairing for content sharing in the CCN network as a generic combinatorial matching issue. Then, we formulated a matching game for solving the NP-hard problem with each node having agreement functions and aspiration levels.
- Finally, we analyze the blockchain-based decentralized matching solution using simulations. The results obtained show that the designed scheme has superior performance in comparison to its counterparts with respect to various performance evaluation metrics.

The rest of the chapter is structured as follows. Section 4.3 presents the network and blockchain-based content sharing model. Also, this section models the preferences of content providers and requesters. Section 4.4 introduces the matching problem, followed by section 4.5, which describes the proposed BLMA algorithm. The implementation analysis and complexity of the proposed algorithm are discussed in Section 4.6. Section 4.7 discusses the smart contract implementation of requester-provider matching on the blockchain. We subsequently present and discuss the performance evaluation results in Section 4.8.

4.3 System model

The model for CCN-based data sharing among content requesters and providers in a V2G network is represented in Fig. 4.1. We consider a V2G network within the coverage of a LAG, which acts as an aggregator used to share information between network entities. A utility

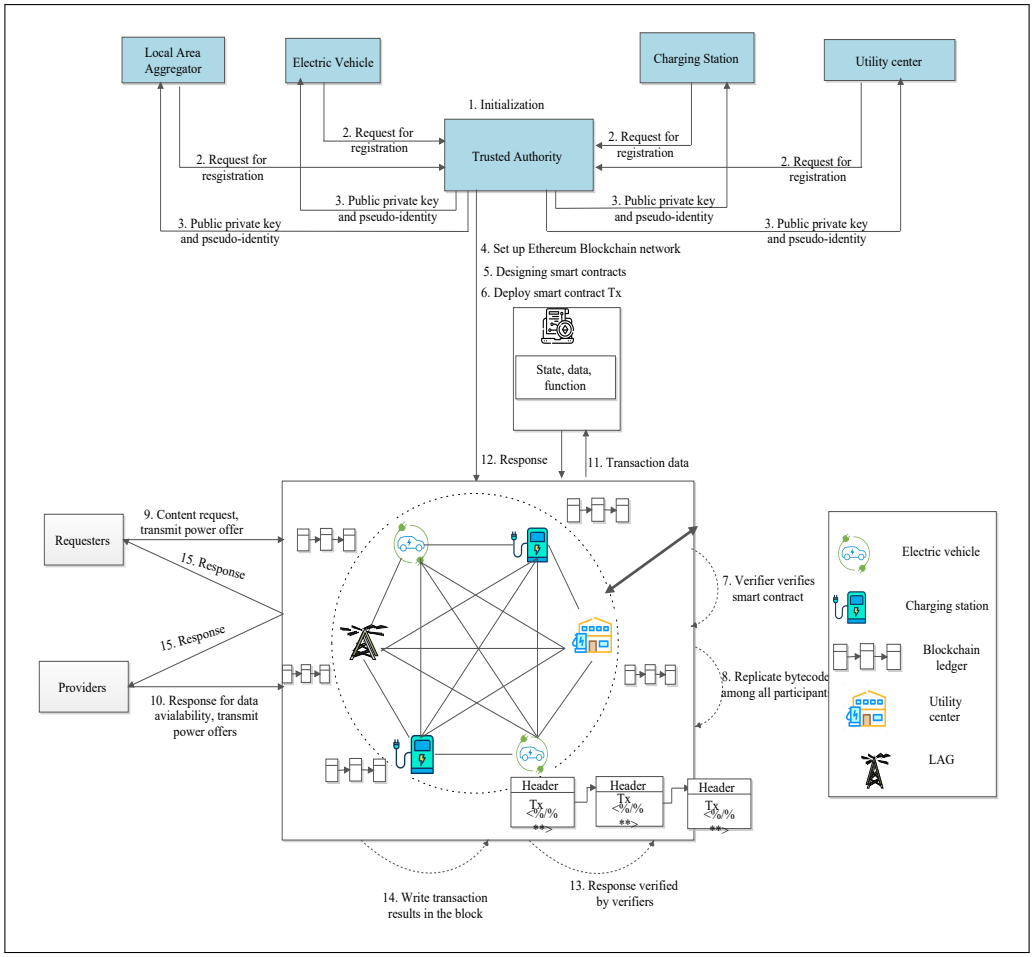

Figure 4.1: System model.

center is deployed to manage the billing activities in the network. Charging stations (CSs) are placed in areas having high strength of EVs, for example- shopping mall. To perform various operations in network, the data related to CS state is forwarded to EVs for charging decisions. Similarly, each EV needs to disseminate its information to other nodes, including SoC, location, and driving pattern. To minimize traffic volume among increasing EVs and to achieve fast availability of data, some geographically closer network nodes caches the content collected from CSs, and EVs in their content store. After storing the content, these network nodes can satisfy their content demands along with serving content to other network nodes. As depicted in Fig. 4.1, all entities are first registered on the blockchain network and receive private and public key pairs. A centralized authority (TA) is responsible for deploying matching smart contracts on the network, and the smart contract executes the matching operation between providers and requesters. All transactions related to matching operations are stored on the blockchain after verification by authority (verifier) nodes.

4.3.1 Network scenario

We used the concepts of CCN communication for data transmission between the content requester and content provider because of its strength in achieving high throughput in a dynamic network [208]. Each node with limited storage can pre-cache and spread popular contents to other nodes via CCN communication. As a security measure, each content packet in CCN based communication is signed with the signature of CP, allowing the content requester to verify the content before receiving it. We consider co-operative caching as it provides advantages by cooperation within neighboring nodes (as the nodes cache content that is not cached in the neighboring node), resulting in higher data availability and decreased latency [161]. For a time interval, LAG divides all nodes of V2G under its coverage into two disjoint set of players (for each content $c_{\mathrm{m}} \in C$), *i.e.*, content requesters (Rs) and content providers (Ps). We use $R = \{R_1, ..., R_{\mathrm{x}}, ..., R_{\#_{\mathrm{R}}}\}$, $P = \{P_1, ..., P_{\mathrm{y}}, ..., P_{\#_{\mathrm{P}}}\}$ and $C = \{c_1, ..., c_{\mathrm{m}}, ..., c_{\mathrm{Mc}}\}$, $|R| = \#_{\mathrm{R}}$, $|P| = \#_{\mathrm{P}}$ and $|C| = Mc$. The size of each content $c_{\mathrm{m}} \in C$ is denoted by s_{m}. However, each content in content library has diverse popularity [146].

A content requester will request data by diffusing Interest packets to its connected interfaces. The smart contract deployed on blockchain collects the requests and content availability at different providers and makes matchings. For new content creation, each node issues the content it has by declaring its metadata (content name, location, and signature) and broadcasts the packet to its connected neighbor's interface. Receiving the content packet with a hop count value of 1, the recipient updates its FIB with its signature, location, and hop count. Next, the immediate connected neighbor broadcasts the packet to its connected neighbors. After having a packet with a hop count value of two, receivers record the updated content information to its FIB, only if the received nonce value didn't exist previously. For designing a CCN network, the name-based routing protocol, *i.e.*, open shortest path first (OSPF) is considered to route requests correctly to data repository [203]. After receiving an request packet, node i first checks its content store, and if the requested content is found, the request is satisfied with no delay. For an unsatisfied request, the node i that received the request communicates with its neighboring nodes. In this case, the PIT of the node i is probed, and for a corresponding match, the entry is updated. Otherwise, FIB is probed, and the interest packet is forwarded to the destination interface according to the defined routing policy. If any one of the neighboring nodes caches the content, content is retrieved from the content store. This process is continued until a copy of the content is there in the content store of the neighboring node of node i. If the copy of the requested content is not found in the store of the neighboring EV, the requester has to obtain the content from the original producer. Upon receiving a data packet, the node probes its PIT, and the content is forwarded to all incoming interfaces. Also, based on the popularity of the content, it is decided whether to cache the content or not. The zipf distribution [209] is used to

characterize the popularity of content c_m by node i and is defined as follows:

$$p_i^m = \frac{\left[\sum_{m=1}^{Mc} \frac{1}{r_m^{\chi}}\right]^{-1}}{r_m^{\chi}} \tag{4.1}$$

where, χ is the distribution parameter and r_{m} represents the rank of m in order of their popularity.

4.3.2 Blockchain based content sharing

The proposed model is based on blockchain for secure content sharing. Also, there are other privacy preservation techniques available for V2G, including encryption, anonymity, signature-based, and perturbation-based (such as-differential privacy and local differential privacy). However, the usage of differential privacy may lead to a trade-off between correctness and privacy as noise addition to data leads to loss of precision in data [210]. Also, the encryption and signature-based techniques increase the computation and communication overhead of V2G network, whereas anonymized data can cause linking attacks [198]. In contrast, our deployed blockchain provides a decentralized matching between content requesters and providers for data exchange in V2G environment. Blockchain replaces the traditional task assignments platform with decentralized ledgers backed up by miners [211]. In particular, requesters and providers communicate directly with the blockchain network rather communicating among themselves. The decentralized blockchain network enables requesters and providers to exchange data by executing matching mechanisms. We used game theory inspired matching theory to model the interaction and to execute data negotiation among requesters and providers.

To establish a blockchain network, each vehicle should register for a unique wallet and generate their account keys. We use an elliptic curve digital signature algorithm for identity setup. The legitimate node (i) compromises a public key, a secret key, and a certificate denoted as: $PK_{\mathrm{i}}, SK_{\mathrm{i}}, cert_{\mathrm{i}}$. The public key represents the sender address of the transaction and verifies the digital signatures of a transaction. In contrast, a private key signs a transaction. The proposed model is implemented on the Ethereum blockchain network. In this model, all nodes in the network are required to pay transaction fees before participating in the matching process. It effectively prevents various attacks, including DoS, Sybil, and false reporting attacks [212].

To enable matching, first content requester $x \in R$, diffuses their Interest packet for content request to the blockchain network with the content name. The content request message of node x, $x \in R$ includes the required content c_{m} (in its naming convectional form), current loc loc_{x}, public key PK_{x}, certificate $cert_{\mathrm{x}}$, timestamp t_{s} and this request message is signed with private key SK_{x} and which is described as

$$request_{\mathrm{R_x}} = \{c_{\mathrm{m}} || loc_x || PK_x || cert_x || t_s\}_{SK_x}. \tag{4.2}$$

After receiving all interest packets, if a provider (either an intermediary node with cached content or the original provider) wants to transfer content, it replies to the blockchain network (details are discussed in section 4.7). The matching task is performed by deployed smart contracts to automate service transactions with the requests and responses submitted by requesters and providers to the blockchain network. The contract first accepts the requests and responses from requesters and providers, and then executes the matching algorithm.

A transaction triggers the predefined method execution defined in the contract, which changes its state. The bytecode generated from source code of the contract is kept with each node of the network. The verifier in blockchain network also executes the predefined methods, and the matching outputs are stored as a transaction. Next, the completed transactions are stored into a cryptographically tamper-proof blockchain database called a block. Also, the blocks are chained in a chronological order by using cryptographic techniques. Each block is generated by a specific verifier by following a consensus process. However, it is complex for the nodes in the system to generate a block by finding the hash value using a random nonce value, as it requires high computational costs. So, we have used Proof-of-authority (PoA) [161] as a consensus mechanism for experiments. However, PoA consensus algorithm can't guarantee fair ordering since it lacks restrictions on node behaviors and identity authentication [213]. To address this issue, we have taken specific measures during implementation to ensure the correct matching of authority node identity and difficulty during block verification, especially in scenarios where the primary authority is not present. Moreover, we have used anonymous or hidden identities (using verifiable random function [214]) for authority nodes to reduce the risk from external attackers. Also, we have introduced randomness in authority node rotation step so that random block producer cannot be predicted. For traceability, each block consists of some fields generated using cryptographic concepts such as- Merkle root hash and the previous block hash. After the transaction verification process, the blockchain network records the transaction and responds with a message to the participating entities.

4.3.3 Matching preferences for content requesters and providers

To conduct matching, requesters and providers have different matching preferences. Requesters prioritize low latency and high reliability, while providers prioritize consumed energy and social reciprocity.

The latency from P_{y} to R_{x}, considering R_{x}'s focus on packet delay, which is given as follows:

$$L_{y,x} = \frac{s_{m_x}}{LR_{y,x}} \tag{4.3}$$

where, m_x is the content c_m requested most by requester, *i.e.*, m_x=arg $\max_{c_m \in C}\{p_x^m\}$ and its size is denoted by s_{m_x}. $LR_{y,x}$ is the link rate between R_x and Py given as:

$$LR_{y,x} = log_2(1 + \frac{t_{y,x}H_{y,x}}{\sigma_x^2 + I_x}) \tag{4.4}$$

where, $t_{y,x}$ is the transmit power of P_y for transmitting content to R_x, $H_{y,x}$ is the channel gains from P_y to R_x, σ^2 is the variance of noise at requesters side, and I_x is the intra cell interference received by R_x.

Moreover, requesters aim for high-reliability performance, measured by successful delivery probability [215]. Due to the random mobility of EVs in the V2G network, requesters and providers may lose contact while sharing content. To account for connection quality, we introduce contact duration, which represents the time span from initial contact to when they are within transmission range. Successful delivery probability is used to ensure that requesters and providers are in contact for the required duration to successfully deliver content. The contact duration distribution is modeled using a gamma distribution. Assume $En_{y,x}$ is the frequency R_x and P_y encounter each other, and CD_n be the contact duration of each n_{th} encounter. The estimated content duration length is given as:

$$D_{y,x} = \frac{\sum_n CD_n}{En_{y,x}} \tag{4.5}$$

and to reflect fluctuations, its variance is given as follows:

$$V_{y,x} = \frac{\sum_n (CD_n - D_{y,x})^2}{En_{y,x}}. \tag{4.6}$$

Let $\mathscr{Z} \sim \Gamma(\alpha, \theta)$ is gamma distributed encounter duration, with probability density function is given as:

$$f(z;\alpha,\theta) = \frac{z^{\alpha-1}e^{\frac{-z}{\theta}}}{(\theta^\alpha \Gamma(\alpha))} \tag{4.7}$$

where, $\alpha = \frac{D_{y,x}^2}{V_{y,x}}$, $\theta = \frac{V_{y,x}}{D_{y,x}}$. Also, the success probability for P_y transferring the requested content m_x to R_x for a qualified content duration is given as:

$$\begin{aligned} P_{y,x}^{rel} =& 1 - \int_0^{\frac{s_{m_x}}{LR_{y,x}}} f(z;\alpha,\theta)\mathrm{d}z = \\ & 1 - \frac{\gamma\big(\alpha, \frac{s_{m_x}}{LR_{y,x}\theta}\big)}{\Gamma(\alpha)} \end{aligned} \tag{4.8}$$

where, $\frac{s_{m_x}}{LR_{y,x}}$ is the minimal contact duration for transmitting content successfully, and $\gamma\big(\alpha, \frac{s_{m_x}}{LR_{y,x}\theta}\big) = \int_0^{\frac{s_{m_x}}{LR_{y,x}\theta}} z^{\alpha-1}e^{-z}\mathrm{d}z$.

For a provider, its matching preference includes remaining energy and social reciprocity

index. The remaining energy of P_y after serving data to R_x is expressed as the difference of available energy and the consumed energy in transmitting data to R_x :

$$E_{y,x}^{rem} = E_y^{avl} - t_{y,x} L_{y,x}(t_{y,x}) \tag{4.9}$$

where, E_y^{avl} represents the available energy of P_y and time required for transmitting a content c_m is $\frac{s_{m_x}}{LR_{y,x}}$. Then, consumed energy of P_y to transmit this content is $\frac{t_{y,x} s_{m_x}}{LR_{y,x}}$.

When computing the social reciprocity index $RI_{y,x}$, the main focus is on compensating the requester who previously contributed content to P_y. $RI_{y,x}$ is expressed as the proportion of the combined contribution of R_x and P_y relative to their mutual contribution.

$$RI_{y,x} = \frac{e_{x,y}}{e_{x,y} + e_{y,x}} \tag{4.10}$$

where, $e_{x,y}$ is the number of times R_x has earlier provided data to P_y and $e_{y,x}$ is vice versa. In particular, $e_{y,x}$ computes the give and take balance between them. The higher the value of $RI_{y,x}$, the higher is the compensation from the provider (P_y), which indicates that P_y is more willing to offer its data to R_x.

4.4 Problem formulation

Let G represent $\#_P \times \#_R$ decision matrix for content requesters towards content providers, $(y,x) - th$ element $g_{y,x} \in \{0,1\}$ represents the decision of requester R_x for P_y. The term $g_{y,x} = 1$, implies that R_x choses to receive the desired content from P_y via CCN communication; otherwise, its value is zero. Each requester looks forward to match with a content provider which can provide maximum content with low latency and high reliability.

The individual utility of R_x while receiving its demanded content from P_y is formulated as follows:

$$k_{y,x} = \frac{\lambda_{y,m_x} p_{m_x}^x w_x P_{y,x}^{rel}}{L_{y,x}(t_{y,x})} \tag{4.11}$$

where, $\lambda_{y,m_x} = 1$ represents provider P_y is having content c_m wanted most by R_x; otherwise $\lambda_{y,m_x} = 0$. Also, w_x represents constant to represent the degree of attention for latency and reliability performance. If its value is greater than one, R_x prefers reliability factor more; else, focus is more on latency factor. Hence, the individual utility represents that R_x wants the content from provider having the lowest latency and high reliability. For each R_x, the matching problem for all providers in P is given as follows:

$$\max_{G,T} \sum_{P_y \in P} \sum_{R_x \in R} g_{y,x} k_{y,x}(t_{y,x}) \tag{4.12}$$

$$\text{s.t} \sum_{P_y \in P} g_{y,x} P^{rel}_{y,x} \geq P^{rel}_{x,thr} \tag{4.13}$$

$$g_{y,x} = \{0,1\} \forall P_y \in P, R_x \in R \tag{4.14}$$

$$\sum_{P_y \in P} g_{y,x} \leq 1, R_x \in R. \tag{4.15}$$

where, T is the matrix of size $\#_P \times \#_R$, having transmit power of providers for requesters, and (y,x) element corresponds to $t_{y,x}$ that represents the transmit power of provider P_y for R_x. The formulated problem finds the best match for R_x to optimize their transfer rate with a given transmit power. The constraint in Eq. 4.13 depicts that successful delivery probability for a pair P_y and R_x should be higher over threshold for successful delivery probability of R_x ($P^{rel}_{x,thr}$). The constraints in Eqs. 4.14 and 4.15 represent that a requester can obtain its data from a single provider at a time.

Similarly, let B ($\#_P \times \#_R$) represent the pairing decision matrix of content providers for requesters where $(y,x) - th$ element $b_{y,x} \in \{0,1\}$ represents the decision of providers P_y towards requesters R_y. If $b_{y,x} = 1$, P_y wants to provide content to R_x; otherwise, $b_{y,x} = 0$. However, while taking a decision to provide data, following points should be considered:

- The content demanded by the requester should be available in its content store.
- Due to the selfish nature, it is concerned on cost acquired by providing content to R_x, in terms of remaining energy.
- To promote effective cooperation, social reciprocity index is considered before selecting the requester.

So, the individual utility gained by P_y by providing data to R_x can be defined as follows:

$$f_{y,x} = \lambda_{y,m_x} . E^{rem}_{y,x} RI_{y,x}. \tag{4.16}$$

Hence, the joint requester-provider matching problem for each providers P_y is computed as follows:

$$\max_{B,T} = \sum_{P_y \in P} \sum_{R_x \in R} b_{x,y} f_{y,x}(t_{y,x}) \tag{4.17}$$

$$\text{s.t } 0 \leq t_{y,x} \leq t_{y,max} \tag{4.18}$$

$$b_{x,y} = \{0,1\} \, \forall P_y \in P, \forall R_x \in R \tag{4.19}$$

$$\sum_{R_x \in R} b_{y,x} \leq 1 \, \forall P_y \in P. \tag{4.20}$$

The above optimization problem determines transmit power of P_y for energy efficiency and to achieve high social reciprocity index while having the basic provisions of R_x. If the social reciprocity index of requester is low, then the resulting utility gain of provider in comparison to the requester is less as it depends only on the remaining energy of the requester. It depicts the

lack of incentives for social cooperation. The constraint in Eq. 4.18 is with respect to transmit power constraint, *i.e.*, $t_{y,x}$ should not excel maximum transmit power of $P_y, (t_{y,max})$. Also, the constraints in Eqs. 4.19, 4.20 ensure that each provider can accommodate a single requester at a time.

Hence, our goal is to obtain a solution for both optimization problems in Eq. 4.12 and Eq. 4.17.

4.5 Matching Game Modeling

In this section, we formulate the joint provider-requester matching problem as a one-to-one matching game. In such a game model, two finite and disjoint player sets specify their preferences, inspired from utilities, to catch the activity between two sides. Here, the content providers and requesters act as players.

While designing the matching game, we need to consider the constraints mentioned in Eqs. 4.14, 4.15, 4.19, and 4.20. Each requester $R_x \in R$ has records of utility function $k_{y,x}(t_{y,x})$ (Eq. 4.11) and each provider has records of $f_{y,x}(t_{y,x})$ (Eq. 4.16). Hence, the transmit power given by providers can serve as proposals; and, $k_{y,x}(t_{y,x})$ depicts how much R_x considers $t_{y,x}$ as its compatible offer to P_y. Similarly, $f_{y,x}(t_{y,x})$ represents how much P_y considers $t_{y,x}$ as its own compatible offer to R_x. Formally, the notions of matching algorithm are defined as [216]:

Definition 1: A matching ζ is described as a function from $P \cup R \longleftrightarrow P \cup R \cup \phi$, *i.e.*, ζ: $P \cup R \cup \phi$ such that for any $P_y \in P$ and $R_x \in R$:

- $\zeta(P_y) \in R \cup \phi$;
- $\zeta(R_x) \in P \cup \phi$;
- $R_x = \zeta(P_y) \leftrightarrow P_y = \zeta(R_x)$.

P_y and R_x are only matched in a matching ζ if $R_x = \zeta(P_y)$ and $P_y = \zeta(R_x)$ are both equivalent. If $\zeta(s) = \phi$, then player $s \in P \cup R$ is said to be single.

In matrix form, the matching ζ is defined as a matrix Ξ of size $\#_P \times \#_R$ having elements in $\{0,1\}$ following :

$$\Xi(P_y, R_x) = \begin{cases} 1, & \zeta(P_y) = R_x \\ 0, & \text{else.} \end{cases} \tag{4.21}$$

Let $\mathscr{M}$ be the set of possible matching matrices inspired from ζ. To achieve decentralization, we use the agreement function and aspiration levels. The preference of players is described by aspiration levels to define the abstraction of potential utility to be delivered from the matching game. Let u_y and v_x be the aspiration levels of P_y and R_x, respectively then players P_y and R_x want to be matched if their matched results can provide at least u_y and v_x utilities, respectively.

Agreement function is used to decide if the the current aspiration level is agreeable, and the

resulting matching is feasible or not. An agreement function in mathematical form is given as in [217].

Definition 2: An agreement function for a content provider and content requester is a mapping:

$$Ag_{y,x} : R_+ \times R_+ \longrightarrow \{0,1\} \tag{4.22}$$

following:

- if $Ag_{y,x}(u_y, v_x) = 0, Ag_{y,x}(u'_y, v'_x) = 0$ for all $u'_y \geq u_y$ and $v'_x \geq v_x$;
- there exist a Λ such that $Ag_{y,x}(u_y, v_x) = 0$ if $u_y \geq \Lambda$ or $v_x \geq \Lambda$.

If $Ag_{y,x}(u_y, v_x) = 1$, it indicates that u_y and v_x are agreeable; otherwise, if $Ag_{y,x}(u_y, v_x) = 0$, they are not agreeable. The first point establishes a monotonicity property, suggesting that if u_y and v_x are not agreeable, then increasing the aspiration levels are also not agreeable. The second point depicts an unbounded property for accepted aspiration levels.

Eqs. 4.18 and 4.20 represent that all content providers have constraints on the maximum transmit power, whereas all requesters have minimum successful delivery probability constraints. Hence, we introduce:

$$\begin{aligned} S_{y,x}(u_y, v_x) = &\{t_{y,x} | 0 \leq t_{y,x} \leq t_{y,max}; P^{rel}_{y,x}(t_{y,x}) \geq P^{rel}_{x,thr} \\ &k_{y,x}(t_{y,x}) \geq v_x; f_{y,x}(t_{y,x}) \geq u_y\}. \end{aligned} \tag{4.23}$$

$S_{y,x}$ describes all exchanges of $t_{y,x}$ from P_y to R_x in return for $t_{y,x}$ from R_x to P_y, to satisfy their own aspiration levels. So, we have:

$$Ag_{y,x}(u_y, v_x) = \begin{cases} 1, & S_{y,x}(u_y, v_x) \neq \phi. \\ 0, & \text{otherwise.} \end{cases} \tag{4.24}$$

If $Ag_{y,x}(u_y, v_x) = 1$, u_y, v_x are agreeable, and corresponding to that R_x and P_y can be matched.

Definition 3: For $\kappa > 0$, the match ζ and aspiration levels u_y, v_x make a κ-pairwise stable solution if :

- For all (P_y, R_x) such that $\zeta(P_y) = R_x, Ag_{y,x}(u_y, v_x) = 1$;
- For all $(P_y, R_x), Ag_{y,x}(u_y + \kappa, v_x + \kappa) = 0$;
- For all $s \in P \cup R$ with $\zeta(s) = \phi, as_s = 0$, as_s is aspiration level of s.

In Point 1, it is indicated that for a matched requester and provider, u_y and v_x should be agreeable. Point 2 states that there should not exist any pair of agents with agreeable κ-improvement aspiration levels. Point 3 implies that individual player must have an aspiration level of zero for a κ-pairwise stable solution.

4.6 Stable matching between content providers and requesters

4.6.1 Algorithm explanation

To find a κ-pairwise stable matching output to the above discussed problem, we have used BLMA [217]. Let $t^{P}_{y,x}$ be the power offers of P_y and $t^{R}_{y,x}$ is the power offer of R_x. The time intervals are represented as $j = 0,1,2,....j_{max}$, j_{max} represents the maximum number of steps for Algorithm 4, *i.e.*, after j_{max} rounds, the algorithm will stop.

To achieve information decentralization, the algorithm generates two vectors of stable aspiration levels, each for the content provider and content requester. BLMA enables both parties to learn about their agreeable aspiration levels. Algorithm 4 represents the pseudocode for the proposed algorithm.

- The aspiration levels $a(j)$ and $b(j)$, along with matching output $\Xi^{(j)}$, changes with $j = 0,1,2,...., j_{max}$ or to the point of stability.
- The algorithm randomly activates pairs of content providers and requesters *i.e.*, at state j, randomly activates players (P_y, R_x).
- Next, these players reveal their power offers to each party. However, the proposal should match their current aspiration levels. At this stage, these aspiration levels are increased or atleast remain unchanged. If this power proposal at-least makes κ improvements to the aspiration levels of requesters, providers and their updated preference value, *i.e.*, $u_y(j)+\kappa$ and $v_x(j)+\kappa$ are agreeable, $Ag_{y,x}(u_y(j)+\kappa, v_x(j)+\kappa) = 1$, then players P_y and R_x are matched with a probability Ω and they cut-off their existing matchings. In our case, the probability is computed by a random function that generates output Ω, where random function is an independent and identically distributed sample for a uniform random variable in $[0,1]$. Next, $u_y(j+1)$ and $v_x(j+1)$ and $\Xi^{(j+1)}$ are updated accordingly.
- If $u_y(j)+\kappa$ and $v_x(j)+\kappa$ are not acceptable, *i.e.,* $Ag_{y,x}(u_y(j)+\kappa, v_x(j)+\kappa) = 0$, then the previous matching result does not change, *i.e.*, $\Xi^{(j+1)} = \Xi^{(j)}$ and $u_y(j+1)$ and $v_x(j+1)$ are updated to $[u_y(j)-\Delta]^{+}$ and $[v_x(j)-\Delta]^{+}$.
- If a player remains single, their aspiration level is decreased by Δ and they wait for the next actionable opportunity.

BLMA is blind, means that match between player P_y and R_x are results of bilateral arbitration which are only dependent on the $Ag_{y,x}(u_y, v_x)$. The negotiation process is achieved by random outcomes from $RAND[0,1] \geq \Omega$. As the result can be randomized, it is tough to end in deterministic conclusion from a proposal being refused. Hence, BLMA for this proposal is non-deterministic.

Moreover, as $\frac{\partial k_{y,x}(t_{y,x})}{\partial t_{y,x}} > 0$ (i.e., function's derivative is positive in its domain), so $k_{y,x}(t_{y,x})$ is

monotonically increasing function of $t_{y,x}$. Also, the derivative of $f_{y,x}(t_{y,x})$ guarantees $\frac{\partial f_{y,x}(t_{y,x})}{\partial t_{y,x}} \leq 0$, which makes it monotonically decreasing function of $t_{y,x}$. It implies that interests of players on both side of matching game are opposite. If $\lfloor f_{y,x}(t^R_{y,x})\rfloor_\Delta \geq u_y + \kappa$ and $\lfloor k_{y,x}(t^P_{y,x})\rfloor_\Delta \geq v_x + \kappa$, then any point on connecting line, $t^R_{y,x}$ and $t^P_{y,x}$ must be acceptable. Thus, we can compute the new transmit power as represented in line 11 of Algorithm 4. As $\frac{\partial P^{rel}_{y,x}}{\partial t_{y,x}} \geq 0$ and $t^R_{y,x}$ computed in line 3 must satisfy constraint in Eq. 4.13, the resulting $t_{y,x}$ which is not smaller than $t^R_{y,x}$ must also satisfy constraint in Eq. 4.13. Similarly, $t^P_{y,x}$ computed in line 4 must follow constraint in Eq. 4.18. Hence, the resulting $t_{y,x}$ is in the set $S_{y,x}$ in Eq. 4.23.

Algorithm 4 Joint provider-requester algorithm using BLMA.

data: $\kappa > \Delta > 0$ and $\Omega \in (0,1]$
Initialize: $j = 0$; $\Xi^{(0)} = 0$; $T(0)$.

1: **while** $(j \leq j_{max})$ **do**
2: Randomly initialize a pair of provider and requester, i.e., $(P_y, R_x) \in P \times R$;
3: Compute $t^R_{y,x}(j)$ aiming to $v_x(j) + \kappa = k_{y,x}(t^R_{y,x}(j))$, following constraint in Eq. 4.13;
4: Compute $t^P_{y,x}(j)$ aiming to $u_y(j) + \kappa = f_{y,x}(t^P_{y,x}(j))$, following constraint in Eq. 4.18.
5: **if** $\lfloor f_{y,x}(t^R_{y,x}(j))\rfloor_\Delta \geq u_y(j) + \kappa$ and $\lfloor k_{y,x}(t^P_{y,x}(j))\rfloor_\Delta \geq v_x(j) + \kappa$ **then**
6: $Ag_{y,x}(u_y(j) + \kappa, v_x(j) + \kappa) = 1$;
7: **if** RAND[0,1] $\geq \Omega$ **then**
8: $\Xi^{(j+1)}(P_{y'}, R_x) \leftarrow 0 \; \forall \; P_{y'} \in P$;
9: $\Xi^{(j+1)}(P_y, R_{x'}) \leftarrow 0 \; \forall \; R_{x'} \in R$;
10: $\Xi^{(j+1)}(P_y, R_x) \leftarrow 1$;
11: Determine $t_{y,x}(j+1)$ by selecting a value randomly in $[t^R_{y,x}(j), t^P_{y,x}(j)]$;
12: Update $u_y(j+1) \leftarrow \lfloor f_{y,x}(t_{y,x}(j+1))\rfloor_\Delta$;
13: Update $v_x(j+1) \leftarrow \lfloor k_{y,x}(t_{y,x}(j+1))\rfloor_\Delta$;
14: **end if**
15: **else**
16: $Ag_{y,x}(u_y(j) + \kappa, v_x(j) + \kappa) = 0$;
17: $\Xi^{(j+1)} = \Xi^{(j)}$.
18: **if** $\sum_{R_x \in R} \Xi^{(j)}(P_y, R_x) = 0$ **then**
19: $u_y(j+1) = [u_y(j) - \Delta]^+$;
20: **end if**
21: **if** $\sum_{P_y \in P} \Xi^{(j)}(P_y, R_x) = 0$ **then**
22: $v_x(j+1) = [v_x(j) - \Delta]^+$.
23: **end if**
24: **end if**
25: $j = j + 1$.
26: **end while**

4.6.2 Analysis

We define ψ to illustrate the state of Algorithm 1 which is a triplet $\psi \in R_+^{\#_P} \times R_+^{\#_R} \times \mathscr{M}$, an integration of aspiration levels (of the content provider and requester) and the matching matrix. Every execution of the algorithm 4 outcomes in an updated state. A state is called reachable, if ψ is realized in a finite rounds. Also, if (P_y, R_x) satisfies, $R_x = \zeta(P_y), Ag_{y,x}(u_y, v_x) = 1$, ψ is agreeable.

4.6.2.1 Convergence performance

The state ψ is called pre-stable if,

1. $\forall \; (P_y, R_x)$ such that $\zeta(P_y) = R_x, Ag_{y,x}(u_y, v_x) = 1$;

2. $\forall\ (P_y, R_x), Ag_{y,x}(u_y + \kappa, v_x + \kappa) = 0.$

As compared to κ-pairwise stable matching, pre-stable state can contain single players having non-zero aspiration level.

For any $\psi = (u, v, \Xi)$, let $\rho(\psi)$ represent the set of all singles players having non-zero aspiration level, expressed as:

$$\rho(\psi) = \{s \in P \cup R \cup | \zeta(s) = \phi, as_s > 0\}. \tag{4.25}$$

The relationship between a pre-stable state $\psi = (u, v, \Xi)$ with aspiration level as_{s*} and a state $\psi' = (u', v', \Xi')$ which is not prestable can be defined as follows:

$$\Xi' = \Xi \tag{4.26}$$

$$as_s^{'} = \begin{cases} as_s, & s \neq s^* \\ [as_s - \Delta]^+, & s = s^*. \end{cases} \tag{4.27}$$

The state ψ is prestable as well as tight if (1) there are no κ-improvement matches for existing aspiration levels (2) there exists an κ-improvement agreeable match after a player in $\rho(\psi)$ decrements its aspiration level by Δ.

Now, we define some propositions that will be used later to prove a theorem

Preposition 1: From any reachable state ψ, there is a finite order of allowable transitions to a prestable state ψ'.

Preposition 2: There exists a finite sequence of allowable transitions to a state ψ' from any prestable state ψ that is either prestable and tight or κ-pairwise.

Preposition 3: From any prestable state and tight state ψ having $|\rho(\psi)| \neq 0$, there exists a finite order of allowable transitions to a prestable state ψ' having $|\rho(\psi')| < |\rho(\psi)|$, *i.e.*, a stringent number decrement in the strength of single agents having non-zero aspiration level.

The combined effect of Propositions 1-3 is designed to transform any reachable state into a κ-pairwise stable state. Consequently, from any reachable state, there exists a finite sequence of permissible transitions. This characterization leads to definition of Theorem 1 as follows:

Theorem 1: Given $k_{y,x}(t_{y,x})$, $f_{y,x}(t_{y,x})$ with opposing directions in $t_{y,x}$ and $\kappa = W\Delta > 0$ for $\{W | W > 1, W \in \mathbb{Z}_+\}$, the algorithm 1 will converge to an κ-pairwise stable solution having probability one.

Proof: As per lines 5 and 6 in Algorithm 1, the activated pair of content provider P_y and requester R_x having aspiration levels u_y and v_x respectively, have a match if and only if $\lfloor f_{y,x}(t^R_{y,x}) \rfloor_\Delta \geq u_y + \kappa = k_{y,x}(t^P_{y,x})$ and $\lfloor k_{y,x}(t^P_{y,x}) \rfloor_\Delta \geq v_x + \kappa = f_{y,x}(t^R_{y,x})$. As already discussed, $f_{y,x}(t_{y,x})$ is decreasing in $t_{y,x}$ whereas $k_{y,x}(t_{y,x})$ is monotonically increasing with $t_{y,x}$. Then any point between $t^R_{y,x}$ and $t^P_{y,x}$ is considered agreeable, which triggers the update of aspiration levels as indicated in line 11 of Algorithm 1. As $\kappa > 0$, aspiration levels can be updated by $u_y^+ \geq u_y + \kappa > u_y$ and $v_x^+ \geq v_x + \kappa > v_x$, respectively. Due to the fact that aspiration levels

are bounded, this procedure must end within finite time steps. Therefore, by proposition 1, the Algorithm in 1 can output in prestable state by using a finite order of allowable transitions.

Also, Proposition 2 can be derived by examining the implementation in lines 18-23 of Algorithm 1. For a prestable and tight state ψ, any requester $R_x \in \rho(\psi)$, having an aspiration level v_x is activated currently with any content provider P_y with an aspiration level u_y. In such a case, there is no match. Once R_x decrements its aspiration level by Δ, an agreement is made because R_x is single and aspiration levels are tight. Then, for $\kappa > \Delta > 0$, P_y and R_x would make a match with updated aspiration levels $u_y^+ \geq u_y + \kappa > u_y$ and $v_x^+ \geq v_x - \Delta + \kappa \geq v_x$. At this time, again, a prestable state is reached. Also, we leverage the tightening process described in Proposition 2 and recursively apply Proposition 3 to achieve an κ-pairwise stable state comprehensively.

As $\kappa = W\Delta$ and the flooring process of the aspiration level update phase in lines 12 and 13 of Algorithm 1, we have $u_y, v_x \in \{0, \Delta, 2\Delta, ..., \Lambda\}$. Accumulating the limitation on aspiration levels to discontinuous grid of Δ steps having requirements of κ-improvements agreement, it is guaranteed that, even if the proposed algorithm advances and the aspiration levels become prestabilized, there is still a possibility (with probability $\Omega > 0$) for certain potential providers and requesters to form matches among the available combinations.

Considering the aforementioned factors, there is a probability of finite steps of allowable transitions to a κ-pairwise stable state from any reachable state. Due to the finite number of elements in sets P, R and the boundless property of agreement functions, we can have a finite rounds for the proposed scheme, *i.e.*, j_{max}, after which the probability of attaining an κ-pairwise stable state from any reachable state ψ is at-least q(κ, Δ)>0. Therefore, neither $q(\kappa, \Delta)$ nor j_{max} is dependent on ψ, *i.e.*, probability of not breaking off after j_{max} steps is at most $(1 - q(\kappa, \Delta))$. In a similar way, probability of not breaking off after $g\, j_{max}$ iterations is at most $(1 - q(\kappa, \Delta))^g$. Let, V_g denote the event of not stopping after $g\, j_{max}$ rounds. Nevertheless, the summation $\sum_{g=1}^{+\infty} V_g$ is finite, the designed algorithm 4 reaches to an κ-pairwise stable solution having probability one as per the Borel-Cantelli lemma described in [218].

Moreover, the obtained κ-pairwise stable solution is guaranteed to be unique. This conclusion is derived from definition of κ-pairwise stable matching in Definition 3. It can be proved from the idea of lattice theorem provided in [219] for the one-to-one result. By iteratively applying lemma 3 in [220] on two-sided stable matching outcome, the optimal solution for both parties can be obtained (Details of proof are omitted).

4.6.2.2 Complexity

The complexity is analyzed based on its convergence rate. The work in [221], describes the most related results to the proposed scheme. The proposal in [221] is a linear assignment matching game similar to our model. Players match, break-up, and rematch by a random variable in search of better opportunities. The decisions of players vary as per an exogenous random variable. The best bound for the rate of convergence in [221] is $O(I^4)$, where I repre-

sents the maximum count of nodes in either party of the market. Hence, we can remark that the convergence in simulation outcomes does not exceed $O((max\{\#_P, \#_R\})^4)$.

4.7 Smart contract based matching

We propose an Ethereum based smart contract approach to execute the designed matching algorithm among content requesters and providers on the blockchain platform. The smart contract enables requesters and providers on the Ethereum network to interact with it by submitting requests and responses along with their requirements. A smart contract is defined as a computer program consisting of functions (executable code that can be reused) and data (its state) [222]. The functions defined in a contract can only be executed by specific roles in a contract.

The smart contract first receives and records action data from requesters and providers of the Ethereum network and then automatically runs the matching algorithm. The detailed process of smart contract execution is discussed as follows:

- A newly designed contract is deployed on blockchain by TA.
- The requesters diffuse the interest packet, while the providers advertise the availability of data.
- The designed contract runs the defined algorithm to form a requester-provider pair.
- Requesters, providers approve the transaction of content transfer based on matching results.
- The smart contract agrees the market clearance.

Fig. 4.2 represents the state transitions involved in smart matching contract executed for requesters and providers. A requester or a provider has 5 states, *i.e.*, offline, online, engaged, change, and finished. A shift in state happens by running code defined in the smart contract. The primary functions of the contract are defined as follows:

- *register*: This function is used by requesters and providers who want to participate in the matching process for their actions to be performed. The requesters, providers remain in the "offline" state before calling *register* and it allows participants to send their interest requests or response packets to the network. In order to encourage nodes to follow smart contract rules reliably, requesters or providers have to pay the refundable amount after they start interacting with smart contract on blockchain. After executing this function, the requesters or providers will be shifted from the "offline" to the "online" state.
- *match*: Once requesters and providers transit to "online" state, the function *match* is called. The contract first analyzes the authenticity of participants and integrity of the

transaction's matching data. After a successful check, smart contract executes Algorithm 4 to find matching results. Meanwhile, the matched requesters and providers are shifted from "online" to "engaged" state after getting the results, while the unmatched requesters and providers stay in "online" state.

- *change*: In the "engaged" state, if a requester or a provider determines to change its location, leading to a change in content preference or availability, the function *change* is executed to represent the change in content preference. As a result, the deposit is not re-compensated; however, the participant is eliminated from the current matching process and it has to again specify its content preference or content availability. Next, the participants have to call the *match* function to again take part in matching process as per their current state.

- *confirm*: If a requester is mapped with a provider, it starts sending requested content to the requester with agreeable transmit power. When requesters receive the requested data from their corresponding providers, they send confirmation packets to the contract. Similarly, the provider sends confirmation to the Ethereum contract once it is done sending. The function *confirm* is called by the confirmed requesters and providers to shift from "engaged" to "finished" state.

- *return*: This function is called to shift from "finished" to back to "offline" state. Now, if a requester, after finishing the transfer, has updated data in its cache and wants to be a provider for some other requesters (or same provider wants to be the provider for other requests), then they again perform the registration process to shift from "offline" to "online" state.

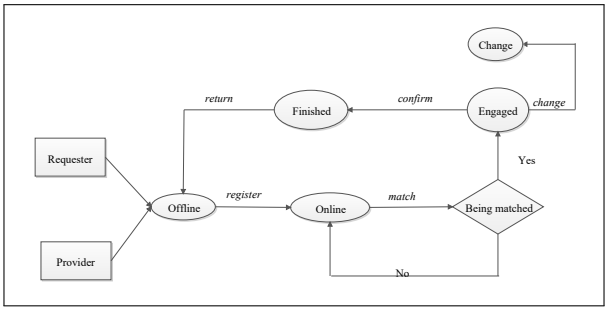

Figure 4.2: State transition diagram for smart contract implementation.

4.8 Performance evaluation

4.8.1 Numerical settings

For proposal evaluation, we choose a road segment with four-lane bi-directional traffic flow, having a cell served by a LAG within a radius of 500 m. Content requesters and potential providers are distributed randomly in a cell area. The smart contract is designed using the Solidity language, and ndnSIM is used to evaluate the V2G CCN network performance. The smart contract is implemented according to the states described in Fig. 4.2. To participate in the matching process, nodes in the network (either requester or provider) first call the *register* function and transit from "offline" to "online" state. Next, the matching process is executed according to Algorithm 4, and after the content transfer is done, the state is changed back to "offline" state by calling *return* function. Also, Python is used to evaluate the files from ndnSIM. To perform simulation results, we used Intel Core 5 Duo CPU at 2.4 GHz, with 16 GB DDR3 SDRAM. The performance of the proposed model has been evaluated on a real dataset accessible at [204]. The data set has power and energy-related information for private homes, EVs, weather data, etc., to test, develop and validate. We use the mobility traces of EVs gathered from the GPS coordinates of 300 EVs over a span of 30 days. The system bandwidth is set as 10 MHz. The transmit power of content providers is between 23 dBm to 30 dBm. Each node has additive white Gaussian noise independent values around 10^{-12} W. We take Δ=0.05 and κ=3Δ. We assume that each P caches part of the content during off-peak time, and later Rs can demand the contents from Ps. There are multiple service providers having a negative social relationship, *i.e.*, they have a lower level of social closeness with certain requesters. Also, since nodes in the network randomly encounter each other, so to compute statically average values, outcomes are achieved by applying 1000 Monte Carlo iterations, and the simulation duration for each iteration is around 350 s. Table 4.1 gives the parameters used in the simulations.

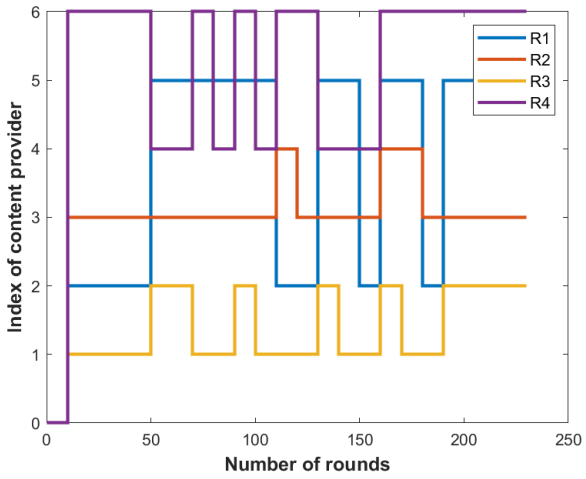

Figure 4.3: Stable matching for content requesters.

Table 4.1: List of parameters.

Parameters	Value
# contents	50-80
Content size	5 MB-15 MB
Zipf parameter	(0.5,1.2]
Coverage radius of LAG	500m
Cache capacity	20M-50M
Average speed of EVs	10m/s-30m/s
Transmission power of Ps	23dBm-30dBm
System bandwidth	10^6 Hz
Block size limit	1 MB
Average block generation time	10 s
α	1
θ	10
κ	0.15
Δ	0.05
Consensus algorithm	PoA
# Monte Carlo iterations	1000
W	3
$h_{y,x}$	[10,20]
$b_{y,x}$	{0,1}
$g_{y,x}$	{0,1}
w_x	1

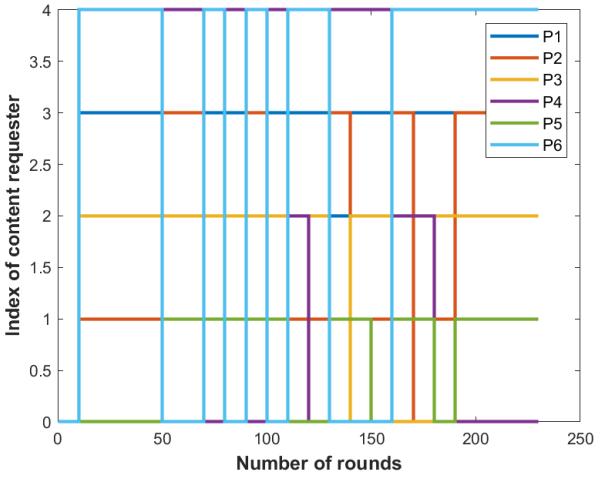

Figure 4.4: Stable matching for content providers.

4.8.2 Results and discussion

We compared the result of the proposal with the following proposals.

- **Social metric based resource allocation**: The scheme in [223] proposed a one-to-one matching game for optimizing resource allocation in D2D communications. In this approach, both parties rank each other based on utilities computed from social layer metrics. The simulation section depicts that socially aware offloading allows better traffic over conventional context-aware approach. Compared to the proposed scheme, the pairing strategy is modeled as a binary variable. Also, authors have mainly focused on exploiting social metrics for resource allocation, while neglecting physical metrics.

- **Physical-social aware content sharing**: Authors in [147] proposed a physical-social aware matching game to achieve a win-win situation. Here, a distributed algorithm using

Dinkelbach iteration is used, and similar to the proposed scheme, power control strategies are used for matching. As compared to this scheme, we have considered vehicle mobility, and we assume that a requester cannot have full content transfer while having one contact duration.

- **Random matching**: In a random matching scheme, pairs of provider and requester are matched randomly, considering providers have the requested content.
- **Contention based resource management**: Authors in [224], proposed a distributed resource block selection using sequential spatial adaptive play. However, this kind of scheme needs to specify a particular order on user's decision-making, which creates unilateral dominance. Compared to the scheme in [224], in the proposed scheme there is limited information exchange.

The above discussed schemes are compared using following parameters:

- Residual energy ratio: It depicts the remaining energy of providers after providing requested content to requesters.
- Average latency: It represents the delay in receiving the content after the request has been generated.
- Average individual utility: Represents the utility gain of content providers by providing content to requesters.
- Throughput: Throughput is defined as the total transactions processed by blockchain network nodes per second (tps).
- Latency: It is defined as the time difference between the transaction initiated by the sender to the network and after processing the final output received by it.

4.8.2.1 Stable matching states

Fig. 4.3, 4.4 refers the plot depicting the number of rounds till stable matching results are achieved using BLMA for six content providers and four requesters. Fig. 4.3 shows the index number of content providers paired to their most relevant content requester, and Fig. 4.4 depicts the index of the content requester paired to most relevant provider. As the result in our case is unbalanced, P1 and P4 don't have any matches, so they remain single with a zero aspiration level. R1, R2, R3 and R4 obtain their desired content chunk from P5, P3, P2, and P6, respectively. It have been observed that the proposed algorithm converges at around 180 iterations. The matching matrix remains the same after this point. In case of $\#_P$=20 and $\#_R$=15, algorithm converges at around 650 iterations. In other case of unbalanced matching, if the number of requesters is more than number of providers, extra requesters remain single

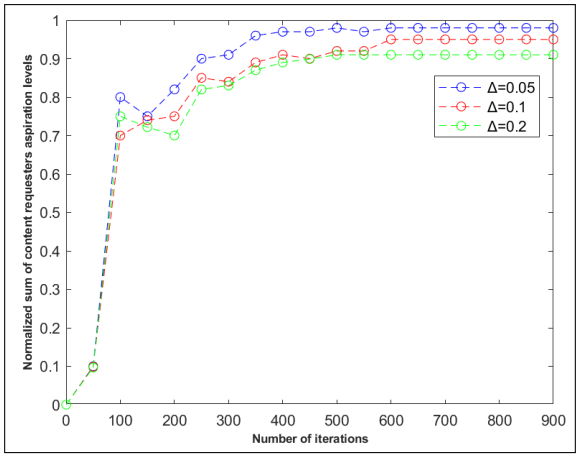

Figure 4.5: Impact of different value of Δ.

and compete in subsequent rounds. However, aspiration level of all single players must be zero for stability to occur. Thus, it can be verified that the rate of convergence does not exceed $O((max\{\#_P, \#_R\})^4)$.

4.8.2.2 Impact of different value of Δ

Fig. 4.5 represents the impact of various values of Δ on the time of convergence with $\#_P$=20 and $\#_R$=15 (κ=3Δ). Clearly, irrespective of Δ and κ values, the algorithm converges to stable outcomes. All requesters get paired with a provider; however, five providers among 20 remain single. Moreover, with an increase in the number of content requesters and providers, the rate of convergence decreases compared to the case where $\#_P$=6 and $\#_R$=4, as the number of rounds to obtain a stable solution increases. However, with an increase in Δ value, the time of convergence for the proposed scheme improves, while the normalized sum of the requester's aspiration level reduces. Nevertheless, it is better to adaptively choose the value of Δ with the suitability of the proposed algorithm to achieve a good trade-off between both the factors.

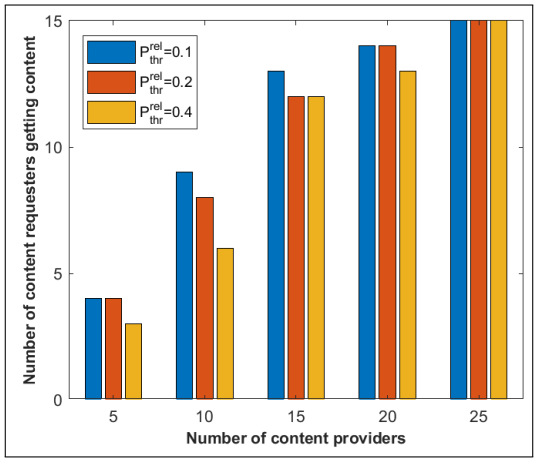

Figure 4.6: Impact of different value of P_{thr}^{rel}.

4.8.2.3 Impact of different values of threshold for successful delivery probability

Fig. 4.6 depicts the graph for the number of requesters vs. the number of providers by varying threshold values for successful delivery probability ($\#_R$=15). Clearly, with an increase in content providers, probability of requesters getting the content also increases. Also, if the value of the threshold for successful delivery probability gets higher, the probability of providers that provide the content to requesters gets reduced as requesters specify high standards for data delivery on providers. However, with an increase in the number of content requesters and providers, the number of rounds required to obtain a stable solution also increases because this creates more options for providers and requesters to decide which is the best match to get a stable solution.

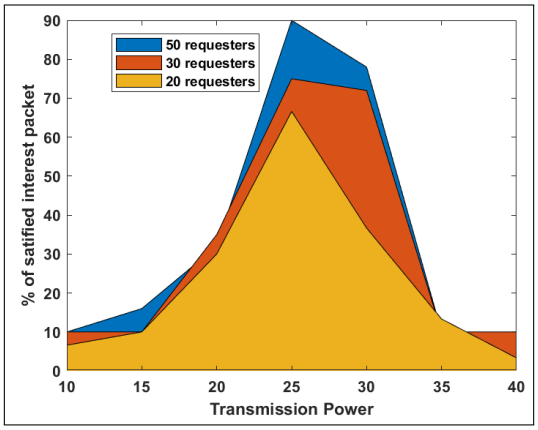

Figure 4.7: Interest satisfaction rate, $\#_P$=50.

4.8.2.4 Interest satisfaction rate

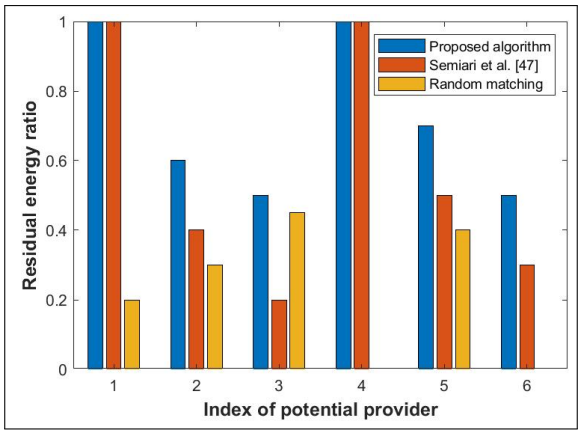

Figure 4.8: Residual energy ratio of different content providers, $\#_P$=6, $\#_R = 4$.

The system shows 80% matching rate when the transmit power is 25dbm. The number of providers in this case is 50. As the transmission rate is increased after this point, providers

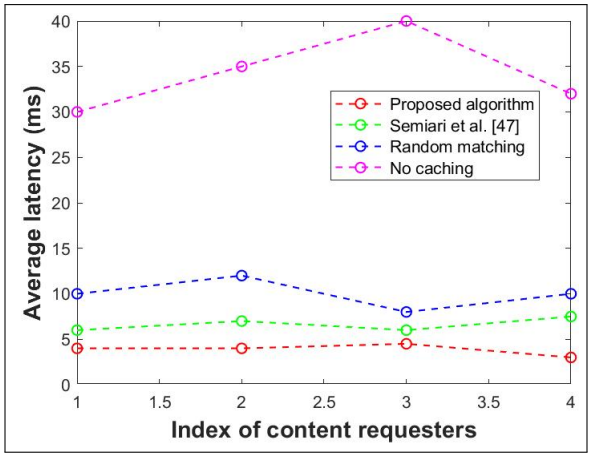

Figure 4.9: Average data latency, $\#_P$=6, $\#_R = 4$.

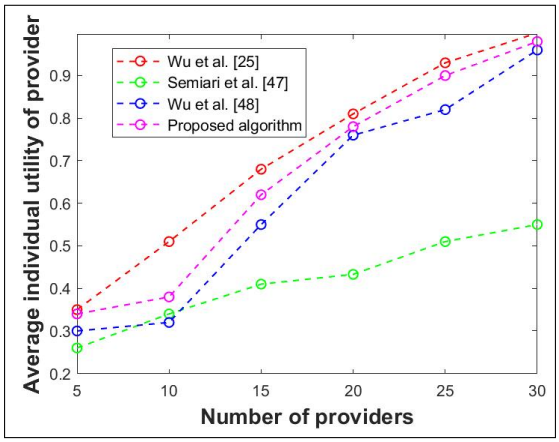

Figure 4.10: Normalized individual utilities of providers by varying number of providers based on various implementations, $\#_R = 15$.

prefer to remain single following the maximum transmit power constraints and to save energy. Also, if the transmit power is quite low, requesters prefer to remain single as they aim to achieve low latency and high reliability, which demands higher transmit power (Refer to Fig. 4.7). Consequently, it is important to reach an optimal transmission power that maximizes network throughput. Moreover, an increasing number of requester nodes intuitively decreases the distance between nodes (due to random distribution); hence the number of unsatisfied interest packets decreases.

4.8.2.5 Performance improvement of proposed matching algorithm

To validate the proposed matching algorithm, we evaluated the data latency along with the residual energy ratio and correlated it with existing proposals.

Residual energy ratio: Fig. 4.8 represents the residual energy ratio of different content providers in the proposed algorithm, the scheme in [223] and the random matching (for $\#_P$=6, $\#_R = 4$). For P1 and P4, their consumed energy is negligible as they do not take part in content transfer. P2, P3, P5, and P6 send their content from the content store to match corresponding

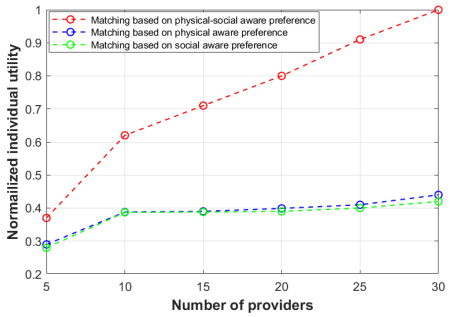

Figure 4.11: Comparison of normalized individual utilities of content providers for different matching basis from perspective of content provider by varying number of requesters and providers $\#_R$ = 10.

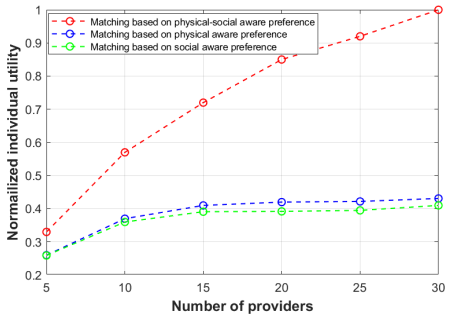

Figure 4.12: Comparison of normalized individual utilities of content providers for different matching basis from perspective of content provider by varying number of requesters and providers $\#_R$ = 15.

requesters; thus, their residual energy is decreased according to the transmit power and the distance from the receiver. We considered the case of the scheme in [223], that maximizes the individual utility of requesters, *i.e.*, it mainly concentrates on optimizing the requesters at the cost of content providers. On the other hand, random matching behaves differently as it matches pairs randomly without considering any parameter. Similarly, in the case of $\#_P$=20, $\#_R = 15$, consumed energy for five single providers is negligible, and the rest fifteen providers (paired with fifteen requesters) residual energy ratio is in range of $\sim 0.55 - \sim 0.65$.

The average latency: Clearly, from Fig. 4.9, it is noticeable that both of the proposed scheme and scheme in [223], achieve lower data latency when compared to random matching as it matches randomly and is less stable. Moreover, no caching method has the worst performance as the content requesters can only receive content from cellular networks leading to high latency. Also, compared to [223], the designed scheme considered the case of co-operative caching, so it achieves less delay over the latter scheme. In the case of $\#_P$=20, $\#_R = 15$, the average data latency for all content requesters lies in range 4.7ms-6ms.

4.8.2.6 Individual utility

Fig. 4.10 shows the average utilities received by content providers with different number of providers. We have compared the average individual utilities of content providers in the

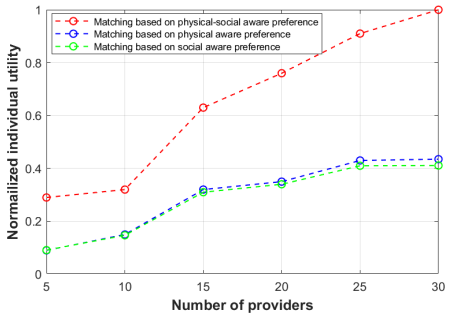

Figure 4.13: Comparison of normalized individual utilities of content providers for different matching basis from perspective of content provider by varying number of requesters and providers $\#_R = 20$.

proposed scheme to the schemes in [147], [223] and, [224]. In [147], content providers and requesters rank each other based on proposed preferences. We have observed that our scheme's normalized average individual utilities of content providers are lower over that of the scheme in [147], and the difference in the two algorithms is approximately 4.2%. Compared to the proposal in [147], in the proposed algorithm, players randomly select other players (whereas in [147], providers rank all requesters based on proposed preferences), so we average our outcomes over 1000 times of the proposed scheme to check for various means which can lead to convergence. However, the proposal's performance in [223] is less than our proposal as it only concentrates on optimizing the requesters at the cost of content providers. Compared to [223], the proposed matching algorithm considers the requirements of all requesters and providers to obtain a win-win scenario. Moreover, the utilities in case of exhaustive search [224] are nearly equivalent to our proposed scheme. The exhaustive search, obtained by brute force, leads to high delay, which degrades its performance a bit. Also, with an increase in the number of providers, the average individual utility of the provider increases in our proposed algorithm, as requesters get more opportunities to receive the content from the nearby content store.

Fig. 4.11, 4.12, 4.13 compares the normalized individual utility of content providers using the proposed algorithm for three different matching basis. The different matching basis from the perspective of content provider are defined as: i) considering both physical-social preference (taking both remaining energy and social reciprocity index). ii) considering only physical preference (taking only remaining energy into account). iii) considering only social preference (taking only social reciprocity index into account). No matter how many requesters and providers are there, matching with physical-social preference is better over the two other matching basis because it considers both consumed energy and the reciprocity index of providers as its preference to make the matching decision. Moreover, it has been observed that the matching for physical preference only doesn't have incentives for co-operation. In contrast, the matching for social preference incurs extra overhead as it does not consider remaining energy. Hence, the normalized individual utility is lower compared to matching based on both physical and social preferences. Also, for different $\#_R$, the performance increases with the number of con-

tent providers increases, as it considers both remaining energy and social reciprocity index of content providers, stimulating effective co-operation. Also, it can be implied that more content providers provide more opportunities for requesters to obtain more sharing services.

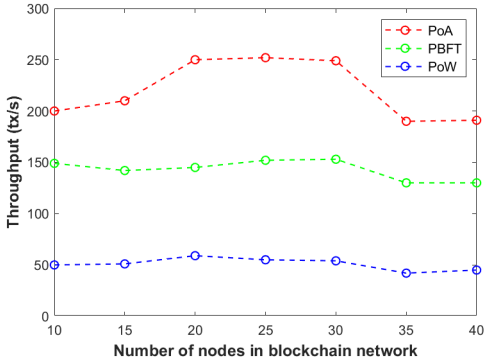

Figure 4.14: Throughput versus number of network nodes.

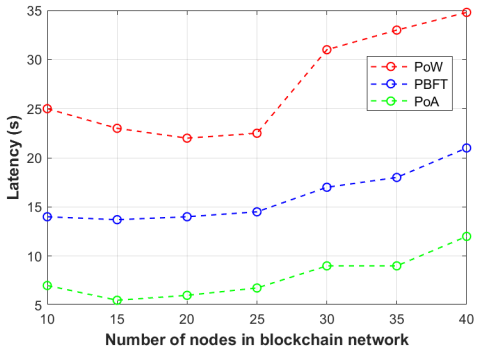

Figure 4.15: Latency versus number of network nodes.

4.8.2.7 Throughput and Latency of Ethereum network

For simulations, we have compared the poposed PoA based consensus process with Proof-of-Work (PoW) and Practical Byzantine Fault Tolerance (PBFT). As the number of nodes in the network is limited, no packet loss is recorded in this scenario. However, for connectivity among large number of nodes, the network may face packet loss because of congestion. Simulation outcomes in Fig. 4.14, 4.15 depict that the PoA-based consensus mechanism supports better throughput and reduced latency respectively in comparison to PoW and PBFT consensus process. Compared to PoW, PoA doesn't demand high computational resources and has a high speed for verifying transactions. Moreover, unlike PBFT, PoA transmits less information in form of packets, thus has better performance. Also, different from PoW, PBFT is energy efficient as its validators do not compute PoW hashing puzzles for each block. Notably, as per the authors of [225], a block size of 1 MB or less is suggestible for a blockchain system. Tests to measure transaction confirmation time and throughput are carried out using a block size of 1 MB while varying the number of nodes in the block creation process. Initially, with an increase

in the number of network nodes, transaction confirmation time decreases (as the network size grows, there are more participants for verification) while resulting in higher throughput. However, if the number of nodes are around 40, the large amount of matching information may lead to high network delay; thus, leading to decrease in network throughput.

4.9 Summary

In this chapter, a one-to-one provider-requester matching game for a CCN-based secure content sharing, having multiple requesters and providers for content is proposed. To achieve an efficient match, transmitting power offers from both sides is considered. In particular, a decentralized algorithm, BLMA, to achieve stable matching and distributed transmit power control scheme with this matching is designed. BLMA pairs players in this two-sided market without mentioning the negotiation mechanism among network nodes. However, multiple degrees of freedom (for controlling utility value) occupies a large search space for stable solutions. Nevertheless, this problem can be suppressed with a proper design of the negotiation phase. Also, an Ethereum-based smart contract to execute the proposed matching algorithm on a decentralized platform that performs two-sided matching without an intermediary party is designed. Simulation results depict that the proposed decentralized matching algorithm can improve system efficiency and converge to a stable state. The next chapter discusses about a one-to-many ciphertext policy based access control scheme.

Chapter 5

A One-to-Many Ciphertext-Policy Attribute-enabled Blockchain Access Control Scheme

5.1 Introduction

As CCN supports the concept of caching, any caching node can act a content request in the forwarding path. However, this introduces security concerns which need to be addressed during content delivery, with content access control being of utmost importance. The existing access control techniques execute at cache enabled neighboring node (NN) and involve interaction with the CP during content retrieval [30], [31]. However, this verification process at the caching node lowers forwarding services. In existing encryption-based access control schemes, key management is difficult with numerous nodes in the system. CP has to stay online to allocate keys to new members. Moreover, traditional access control schemes such as mandatory access control (MAC) and role-based access control (RBAC) are not applicable to CCN, as these schemes are designed to implement ACP where contents reside with the CP [32]. Also, CP-ABE causes key escrow, heavy computation, and authority collusion problems. In response, the proposed scheme integrates CP-ABE with the Pedersen (t,n) secret sharing protocol [226] for attribute secret key generation. It ensures cooperation among t AACs and effectively resolves the key escrow problem. However, a drawback of this scheme is the absence of any identity authentication of secret key shareholders, thus limiting its applicability in secret sharing. Nevertheless, authentication of secret shareholders using signature verification technique has high storage and computation overhead, causing delays in network nodes.

Moreover, in-network caching enables caching nodes to directly respond, making it easy for them to acquire information about requested content and the requester. For privacy preservation, existing proposals use techniques such as group signature [38], random forwarding [39], ring signatures [40] and network coding [41], but the overhead of these schemes is significant.

Moreover, only some schemes discusses content revocation, which is important for the deployment of CCN in the V2G network. Motivated by the issues discussed above, we present a blockchain-based efficient one-to-many fine-grained access control scheme for the V2G network using CP-ABE and secret sharing protocol. To enhance security, the proposal employs a Bloom Filter to verify the authenticity of secret shareholders.

5.1.1 Contributions

- We propose using CP-ABE and (t,n) threshold secret sharing to achieve one-to-many and robust access control, where multiple AACs manage the whole attribute set. The CP defines the access policy, and only receivers with attributes that satisfy ACP can only decrypt the ciphertext.
- We propose to use blockchain technology to record the ACP defined by the CP. The blockchain participants make use of Pedersen secret sharing to generate master public parameters and to manage the complete lifecycle of users' attribute secret key generation, which is essential for content decryption. Also, a bloom filter is stored on a blockchain network for identity authentication of secret key shareholders.
- We present a security and performance analysis of our proposal with respect to data confidentiality, integrity, anonymity, non-repudiation, collusion attack, encryption time, decryption time, key generation time, and storage overhead.
- We present a content revocation scheme for instances where a CP no longer wishes to cache content in NN.

The rest of the chapter is structured as follows. Section 5.2 introduces preliminaries and background including the definitions of the bilinear map, bloom filter, access structure, Pedersen's (t,n) Secret Sharing Protocol, and CP-ABE. The system model and security assumptions of the proposed scheme are presented in Section 5.3. Section 5.4 defines the details of the proposed scheme with descriptions of data revocation and the security analysis. The performance analysis of the proposed scheme is presented in Section 5.5.

5.2 Preliminaries and background

5.2.1 Bilinear mappings

Let G_1 and G_2 are two cyclic groups having prime order p and g be the generator of G_1. Let $e : G_1 \times G_1 \rightarrow G_2$ follow a bilinear mapping having properties:

1. Bilinearity: $\forall a,b \in Z_p$ and $g,h \in G_1$, $e(g^a,h^b) = e(g,h)^{ab}$.

2. Computability: $\forall g,h \in G_1$, $e(g,h)$ can be easily computed

3. Non-degeneracy: There exist $g,h \in G_1$ such that $e(g,h) = 1$.

5.2.2 Access Structure

Let $T = T_1, T_2, ..., T_n$ be a set of parties. A collection $\mathscr{A} \subseteq 2^{\{t_1,t_2,.....,t_n\}}$ is considered monotone if $P \in \mathscr{A}$ and $P \subseteq Q$, then $Q \in \mathscr{A}$ $\forall P,Q$. The non-empty subset $\mathscr{A}$ of $\{T_1, T_2, ..., T_n\}$ is access structure, i.e., $\mathscr{A} \subseteq 2^T \setminus \{\phi\}$. The attribute set in $\mathscr{A}$ are defined as authorized set; otherwise, it is called unauthorized set.

Observing the conclusion in [217], an LSSS access structure can be utilized for representing the access policy $\mathscr{A}$. From the approach discussed in [227], any monotonically behaving boolean formula can be transformed into an LSSS representation.

5.2.3 Pedersen's (t,n) Secret Sharing Protocol

Secret-sharing protocols are used to share secrets between groups of parties, each of which has partial information regarding secret. The complete secret can only be reconstructed if information from some participants is combined. Pedersen secret sharing protocol was proposed to avoid any node being the bottleneck point, as this proposal eliminates the need for any central trusted third party. Lets say there are n total participants in network, represented as: $P = \{P_1, P_2, ..., P_n\}$ and t $(t \leq n)$ is the threshold value. Each participant (shareholder) in this protocol participates in generating master parameters as follows:

- Master secret generation: Each participant choses random secret called subsecret, $\theta_i \in Z_p$ from which the master secret can be calculated as: $\theta = \sum_{i=1}^{n} \theta_i$.

- Sub-share generation: Each participant shares this subsecret with others using the Shamir secret sharing protocol. Each P_i chooses a random $(t-1)$ degree polynomial $f_i(x)$, such that $\theta_i = f_i(0)$, next it computes n subshare $s_{ij} = f_i(x_j)$ for $j = 1,2,...,n$, P_i sends each s_{ij} to other participants secretly. Also, P_i computes $s_{ii} = f_i(x_i)$.

- Master share generation: After having n subshare s_{ij}, P_i computes master share as: $\theta_i = \sum_{j=1}^{n} s_{ji} = \sum_{j=1}^{n} f_j(x_i)$.

- Master secret reconstruction: More than t participants collaborate with their master shares to construct master secret $\theta = \sum_{j=1}^{n} f_j(0)$ using Langrange interpolating formula.

5.2.4 Ciphertext-policy attribute-based encryption (CP-ABE)

CP- ABE is used to implement one-to-many fine-grained access control, which is helpful in solving the security problems associated with sharing content. Most CP-ABE scheme consists

of three basic entities- the authority, the CP, and the Content user (CU). The authority publishes parameters and generates a attribute key for CU. A noticeable feature of CP-ABE is that it provides CP to directly control access based on access policies to ensure flexible and fine-grained access control for CCN-based content services. CP constructs the ACP, and CU is assigned a secret key associated with its own attributes, and they can only decrypt the content if their attribute satisfies the attributes defined in ACP. However, most of the current CP-ABE based scheme in literature uses cloud network to store ACP which leads to a SPoF and privacy disclosure problem. CP-ABE scheme compromise of four algorithms:

- Setup: It takes a security parameter λ and outputs a public parameter (PP) and a master key MK.
- Key generation: It takes MK and $\mathscr{S}$ (attribute set) and outputs a secret key SK.
- Encrypt: It takes PP, content c and access structure A and outputs a ciphertext CT after encrypting content c.
- Decrypt: It takes SK and CT and outputs the original content c otherwise return symbol $\perp$.

5.2.5 One-way hash function

A hash function receives an input message x and generates a fixed-size hash output $H(x)$. For the same message x, the hash function always generates the same hash value [228].

5.2.6 Bloom Filter

A *bloom filter* is used to check element membership in a probabilistic set representation [229]. It is a bit array of predefined size with all its bits initialized to zero. Due to the limited space of the filter, the output of a hash function on a non-existing member can result in bit positions that are already set, leading to false positives. If we set a sufficiently large array size, the probability of hash collision decreases [230]. However, the query results of a bloom filter results in false positives but not false negatives. The error probability is dependent on the number of nodes (n), the number of hash functions (k), and the size of the bloom filter (m). However, if we take array size large, the chances of hash collision decrease, and the error probability is reduced. For k hash functions, the false positive probability is described by equation 5.1.

$$P = \left(1 - e^{\frac{-kn}{m}}\right)^k \tag{5.1}$$

Fig. 5.1 illustrates a bloom filter with k=3. In this example, the bloom filter is filled with the IDs of nodes registered on the network by trusted authority. While registering a new node,

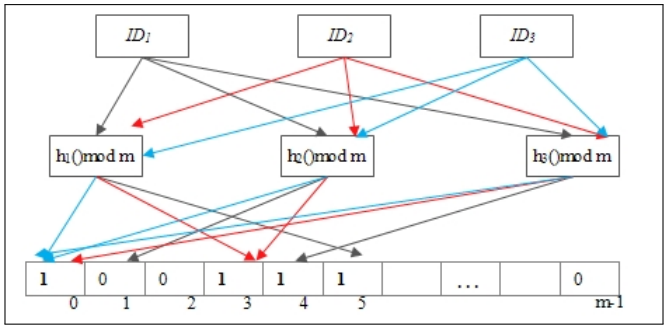

Figure 5.1: Bloom filter illustration

each node ID is hashed as per the chosen algorithm and the number of hash functions. Next, each bit array position corresponding to the hash functions is set to one. The mapping value will not increase if more than one hash value refers to the same position. To verify if the attribute set belongs to the node, each node *ID*, specified by a requested node, is passed through *k* hash functions to get *k* index values. If any of the bits corresponding to these positions are zero, then the node is not registered on the network, and it is illegal; otherwise, if all the mapping values do not contain zero, the node may be registered on the network. However, false positives are possible if a hash collision happens during the hashing operation [231].

5.3 System model and problem definition

As depicted in Fig. 5.2, the proposed scheme consists of 6 entities: *content producer, neighboring nodes, content user, blockchain network, trusted authority, and attribute authority center*. The description of each entity is described as follows.

- **Trusted authority (TA):** TA plays an important role in the proposed scheme. It initializes the whole system in the beginning. It registers all AACs and nodes in the system, providing them pseudonyms to prevent anonymity. Meanwhile, TA also generates public keys for all attributes in the universal set. Moreover, TA also constructs a bloom filter with verified IDs for each node and publishes the *bloom filter* to the blockchain network. It also publishes security parameters necessary for generating secret keys and transferring content from one entity to another. The process of master key sharing and generation of attribute secret key doesn't involve TA, which avoids the situation of performance bottleneck.

- **Content producer (CP):** CP is a utility center responsible for generating bills or a Local Area Aggregator (LAG), which gathers various data like meter readings from charging stations (CSs) and EVs. Initially, CP performs encryption with CP-ABE scheme. It encrypts the content with a symmetric key and defines ACP using an attribute set. It also encrypts symmetric key (denoted as CT) under a defined policy, using an attribute public

key generated by a trusted authority. Then, it caches the generated CT within neighboring nodes (NNs), and the related CT information gets recorded on the blockchain network in the form of a transaction. NN verifies whether the CT received by them is correct or not by checking the hash value of CT. If the computed value by NN matches, then CP compiles the specified ACPs, the hash of the original content, and the corresponding NN signature into a transaction on the blockchain network.

- **Neighboring nodes (NN):** NN receives and caches the CT, and then computes the signature of CT. The computed signature value is sent back to CP. NN does not perform access control for CP.

- **Content User (CU):** CU can be an EV or a CS seeking to access content within the network. Before retrieving content, CU first applies for attribute secret key from AACs. Then, it generates an interest packet for the desired content, and it can only recover the requested content if they have an authorized key. In CCN architecture, we use the content names of messages for content transfer and forwarding. While CU can easily download CT from NN, decryption of the CT is only feasible if CU's attribute set matches the corresponding ACP recorded on the blockchain. After the successful decryption of CT, CU can verify the integrity of content by using a hashing mechanism.

- **Attribute Authority Center (AAC):** It is responsible for managing node attributes and key generation. In this scheme, all AACs jointly supervise the complete attribute set, with each node possessing full control over any attribute; also, they co-ordinate to share the master key. Each AAC uses Pedersen's (t,n) sharing protocol to produce a master secret parameter. Hence, no communication among AAC is required in the generation of attribute keys. Clearly, the value of the master secret key is unknown to any of the authority nodes, while everyone has its secret share of the master key. However, the assumption that all AACs are honest in an untrustworthy network is difficult to consider. Any AAC can check the authentication of sub-shareholders by using the bloom filter that is generated by TA and stored on the blockchain network. This identity verification helps prevent a Man-in-Middle (MITM) attack.

- **Blockchain network:** In this scheme, we propose a consortium blockchain model (Ethereum-based) in which each node in the network is identified by a unique ID, represented as a wallet address, and associated with specific attributes. All transactions' relevant operations are recorded on the blockchain network, and their accuracy is confirmed by authority members using a consensus algorithm. Transactions are first accumulated in the data pool and then logged on the chain. AAC, CP, and CU each generate their key pair and send the public keys to be stored on the blockchain network. Also, global parameters are stored in the ledgers of the blockchain network. Moreover, a *bloom filter* is stored on the blockchain network by TA, so the authentication process is carried out without any

third-party involvement. The verified transactions are recorded in a tamperproof database called blocks. Related to content, its hash value, corresponding ACPs, and signature of NN that caches the CT are recorded in a blockchain block. These blocks are chained in chronological order using hashing. For traceability of the transaction, each block contains a Merkle root hash and the previous block hash [232]. However, it is complex for network nodes to output a block by computing the hash value from a random nonce value. So, we consider using PoA as a consensus mechanism for experimental purposes.

5.3.1 Communication model

We have used CCN as the communication model, which is specifically made for point-to-point communications between nodes and emphasizes content distribution and retrieval [202]. CCN communication is characterized by its receiver-driven transport protocol, relying on two types of packets: *Interest and data packets*. In this setup, a requester initiates its request by broadcasting an Interest packet to all available connectivity interfaces, while any node possessing the requested content can serve as a provider. Moreover, data packets are dispatched back to the receiver through the reverse path of the initial request, and packet-level caching is automatically carried out at every network node. Contents are divided into a series of content chunks which are uniquely identified and requested by consumers through explicit pre-chunk requests. At each intermediary node, pending queries are logged to facilitate the delivery of the requested content back to the requester.

5.3.2 Security model

In this security model, there are two entities: challenger C and adversary $\mathbb{A}$ In our proposed scheme BOMAC, which is supported by blockchain technology having multiple AACs, it is assumed that adversary $\mathbb{A}$ can't compromise greater than t number of blockchain nodes to obtain secret sharing information. The CT is generated by the challenger and is encrypted under the access structure $\mathbb{M}$. C hides the details of secret key generation from $\mathbb{A}$. $\mathbb{A}$ can query decryption key of attribute set S that is not satisfied under $\mathbb{M}$. The formal security game is described as follows:

- Initialization: C runs the initialization phase to provide system parameters.
- Query Phase-I: $\mathbb{A}$ submits attribute key queries for attribute set S; however, $\mathbb{A}$ can't satisfy challenge access structure.
- Challenge: $\mathbb{A}$ submits two messages M_0 and M_1 of same length to C. Also, challenger gains structure $\mathbb{M}$ from $\mathbb{A}$. Next C chooses a message M_α randomly and encrypts under $\mathbb{M}$. The generated CT is sent to $\mathbb{A}$.

- Query Phasr-II: $\mathbb{A}$ repeats query phase I to ask for more attribute secret keys from S as long as S does not satisfy $\mathbb{M}$.
- Guess: A outputs α' as a guess of α. The advantage of $\mathbb{A}$ in the game is : $Pr[\alpha' = \alpha] - \frac{1}{2}$.

5.3.3 Security Assumptions

To maintain security in the system, the following security assumptions are made:

- **Assumption 1**: It is assumed that the caching node may play with the integrity of content or leak the content to another node. Also, the caching node may collude with the attacker node to gain information about encrypted content.
- **Assumption 2**: The TA or AAC are considered fully trusted nodes, but a malicious node can compromise them. In the worst-case scenario, even if these get compromised, malicious nodes would still be unable to manipulate ACP or access content with unauthorized access (unless more than 51% of the blockchain nodes agree).
- **Assumption 3:** CUs are not always honest and may try to gain unauthorized access to data. CU may collude with other nodes to get unauthorized access; however, they will not collude with more than t blockchain members to get illegal access permissions.
- **Assumption 4:** We also assume that if a node's attribute does not satisfy the access policy, it is regarded as an illegal node (may be registered or unregistered).
- **Assumption 5:** Blockchain members are assumed to be trusted; however, less than 50% of them can be hacked by malicious nodes.
- **Assumption 6:** Content producers are fully trusted.

.

5.3.4 Design Goals

- **Integrity:** To ensure data integrity, the proposed scheme should guarantee that content reaches the requester without any modifications
- **Fine-grained access control**: The CP designs the ACP for their generated content; they should allow or revoke node access on a fine-grained level by changing the access attributes.
- **Confidentiality:** To ensure content confidentiality, the proposed scheme should ensure that only nodes having attributes satisfying the access policy can decrypt the ciphertext.

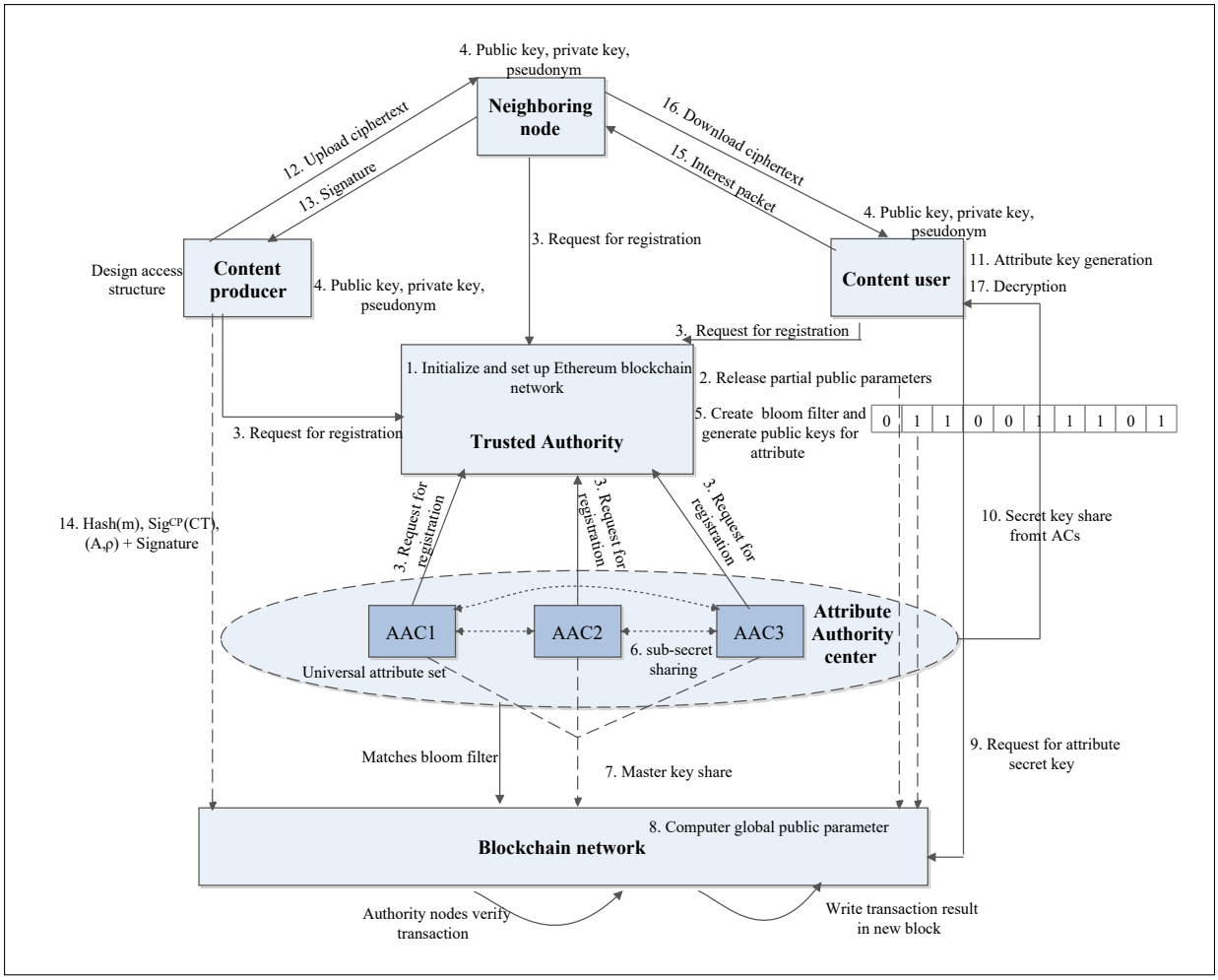

Figure 5.2: The Schematic overview of the proposed scheme

- **Collusion attack resistant:** Two or more nodes with attribute sets that don't match the ACP may collude illegally to decrypt the CT. The proposed scheme should be capable of defending against collusion attacks.
- **Privacy preservation:** The data collected by smart meters may disclose the node's behavior. While delivering content, no node should be able to link the real identity with the location or any other information provided by CP or NN.

5.4 The proposed Scheme

5.4.1 Overview

The proposed scheme is described in Fig. 5.2 in which TA is responsible for registration of nodes and release of public parameters on blockchain network. TA creates bloom filter with registered nodes IDs; along with generating public keys for attributes. Both bloom filter and public keys attributes are stored in blockchain network. AACs share sub-secret to generate

master key share. The master key is reconstructed by using g^{θ} and $e(g,g)^{\theta}$, which is a discrete log problem to compute master key θ from g^{θ} and $e(g,g)^{\theta}$. Hence, the problem of reuse of the master key among different AACs can be solved.

Next, the attribute secret key is reconstructed using t secret shares from AACs. CP encrypts the content and sends it to NN, where NN computes the hash of the content. If the value from NN is equivalent to the hash of CT, then CP packs the ACP, hash of the content and NN signature as a transaction.

5.4.2 System setup

TA generates public parameters (PP) and on the network. Then,TA chooses two cyclic groups (multiplicative), G_1, and G_2, of prime order p, and g is the generator of G_1. Next, it selects a hash function such that $H_1 : (0,1)^*$ and also chooses random values $\phi \in Z_p$, sets it as master key and computes g^{ϕ} to be released as public parameter. The master secret key only remains with TA and doesn't need to be revealed to other nodes. Moreover, TA randomly generates public keys of all attributes in the system $Att_j, (j = 1,2,.....,N) = p_1, p_2, ..., p_N \in G_1$. Also, TA computes key pair (s_{TA}, v_{TA}) for signature and verification where v_{TA} is public and published on blockchain networks.

5.4.3 Registration

The registration algorithm is run by TA such that it registers network nodes on the blockchain network and assigns a unique global identity (*nid*) to each legal user based on its wallet address. Also, while registration, AAC also registers themselves in this phase, and a global ID ($acid \in Z_p$) is allocated to them. TA generates a bloom filter using IDs of registered users in the network, as depicted in Fig. 5.1. Moreover, it publishes k has functions for BF membership checking on bloom filter. Let n represent the number of AACs in the system.

5.4.4 Node's key generation

For each node having registration in the system, it selects a random number $s \in z_p^*$ and assigns s_{nid} as the node's secret key (SK_{nid}) and computes the public key for decryption (PK_{nid}) as:

$$SK_{Nid} = s \tag{5.2}$$

$$PK_{nid} = \{g^{\frac{1}{s_{nid} \cdot \phi}}, g^{\frac{1}{s_{nid}}}\} \tag{5.3}$$

5.4.5 Attribute Authority center setup

Suppose there are a total of n AACs in the system, and the blockchain network decides the threshold value t that represents the number of AACs that take part in attribute secret key

generation. All AACs co-operatively use Pedersen's(t,n) secret sharing as follows.

1. Each *AAC* ($AAC_{acid_i}, i = 1,2,....,n$) independently selects a sub-secret $\theta_i \in Z_p$, then the master secret is calculated as: $\theta = \sum_{i=1}^{n} \theta_i$. The value θ should be kept secret from any node alone. Each AAC_i executes Shamir secret sharing by selecting a random $t-1$ polynomial $f_i(x)$ such that $\theta_i = f_i(0)$.

2. AAC_{acid_i} computes sub shares $s_{ij} = f_i(acid_j)$ for $j = 1,2,...,n$ and sends s_{ij} to AAC_j via a secret channel.

3. After receiving sub shares from other $n-1$ AAC, AAC_{acid_i} calculates its master share as $ms_i = \sum_{j=1}^{n} s_{ji}$, then each AAC_{acid_1} publishes $e(g,g)^{ms_i}$ as its relevant public key share, PK_{acid_i}, on blockchain network.

$$PK_{acid_i} = e(g,g)^{ms_i} \tag{5.4}$$

   Before collecting secret shares from different AACs, each AAC verifies the identity of other AACs. To check the authentication of AAC sending sub-share, the bloom filter is used by receiving AAC.

5.4.6 Global public parameter computation

This phase is executed by the blockchain network. To compute global public parameter, blockchain chooses t out of n AAC public key shares, $PK_{acid_i}, i = 1,2,...t$. Then, the public parameter is reconstructed as follows.

$$\begin{aligned} e(g,g)^{\theta} &= e(g,g)^{\sum_{i=1}^{t}(ms_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i})} \\ &= \prod_{i=1}^{t} e(g,g)^{ms_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} \\ &= \prod_{i=1}^{t} PK_{acid_i}^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} \end{aligned} \tag{5.5}$$

Finally, the complete list of public parameters is as follows:

$$\begin{aligned} PP = &[p, G_1, G_2, g, g^{\phi}, e(g,g)^{\theta}, n, t, \\ &H_1, p_1, p_2, ..., p_N, h_1, h_2, ..., h_k] \end{aligned} \tag{5.6}$$

5.4.7 Attribute secret key generation

This algorithm is run by CU and selected t AAC on the blockchain network. If there are less than t AACs available, the attribute secret key can't be generated. To obtain a secret share from $AAC_{acid_i}, i = 1,2,.....,t$, CU nid_j first sends a transaction request of attribute secret key

generation to the blockchain network. Then, selected t AACs, AAC_{acid_i} verifies nid_j signatures over request. Each AAC_{acid_i} chooses $\Phi_i \in Z_p$ and generates secret key share as:

$$M_i = g^{ms_i}.g^{\phi\Phi_i}, L_i = g^{\Phi_i}, \forall Att_i \in S_{att_i} : M_{Att_i} = p_{Att_i}^{\Phi_i} \tag{5.7}$$

After getting t secret shares from t AACs, node's attribute secret key is generated as follows.

$$\begin{aligned} &M = \prod_{i=1}^{t} M_i^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}}, L = \prod_{i=1}^{t} L_i^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}}, \\ &\forall Att \in S_{att_i} : M_{Att} = \prod_{i=1}^{t} K_{Att_i}^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} \end{aligned} \tag{5.8}$$

Further,

$$\begin{aligned} M &= \prod_{i=1}^{t} (g^{ms_i}.g^{\phi\Phi_i})^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} \\ &= \prod_{i=1}^{t} \left((g^{ms_i})^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} . (g^{\phi\Phi_i})^{\prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}} \right) \\ &= g^{\sum_{i=1}^{t} \left(ms_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right)} . g^{\sum_{i=1}^{t} \left(\phi\Phi_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right)}, \\ &= g^{\theta}.g^{\phi . \sum_{i=1}^{t} \left(\Phi_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right)}, \\ L &= g^{\sum_{i=1}^{t} \left(\Phi_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right)}, \\ \forall Att &\in S_{att_i} : M_{Att} = p_{att}^{\sum_{i=1}^{t} \left(\Phi_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right)} \end{aligned} \tag{5.9}$$

Lets introduce v as:

$$v = \sum_{i=1}^{t} \left(\Phi_i \prod_{j=1,j\neq i}^{t} \frac{acid_j}{acid_j - acid_i} \right) \tag{5.10}$$

Hence, the node's secret key is :

$$M = g^{\theta} g^{\phi . v}, L = g^{v}, \forall Att \in S_{att_i} : M_{Att} = p_{Att}^{v} \tag{5.11}$$

5.4.8 Encryption

This algorithm is executed by CP independently which generates access structure (A) for their content. A CU can only access content C if user attribute set S satisfies the corresponding ACP described in A. Firstly, the CP randomly chooses a number $K \in G_2$ to use as its symmetric key and encrypts the content C using any encryption algorithm. Let's denote the obtained encrypted data as $E_K(C)$. Next, CP encrypts K with the CP-ABE scheme under access structure A (generated by CP itself). Meanwhile, CP turns access structure to LSSS access structure [227], expressed as $(\mathscr{A}, \rho)$ where, $\mathscr{A}$ is $k * l$ matrix, k represents the scale of the attribute set related to an access policy, and l is a variable related to monotonic boolean formula. ρ is a mapping func-

tion that maps specific attributes to rows of $\mathscr{A}$ defined as $\rho(j) = Att_1, Att_2, ..., Att_N$. Next, CP chooses d as the encryption exponent and a set of randomly chosen number $s_2, s_3, ..., s_l$ to share encryption exponent to form a random vector $\bar{v} = (d, s_2, s_3, ..., s_l \in Z_p)$. Next, CP computes $\lambda_j = \mathscr{A}_j \bar{v}^\intercal$ for $j = 1, 2, ..., k$. Further, CP randomly chooses $\{w_1, w_2, ..., w_k\} \in Z_p$ and extracts the public key generated by TA to obtain CT as follows.

$$\begin{pmatrix} C = Ke(g,g)^{\theta d}, C' = g^d \\ \forall j \in [1,k], C_j = (g^{\phi})^{\lambda_j}.p_{\rho(j)^{-w_j}}, D_j = g^{w_j} \end{pmatrix} \tag{5.12}$$

CP sends this CT to random NNs as $E_K(C)$ and $C, C', \{C_j, D_j\}_{j=1 to k}$ together.

This algorithm outputs CT, which is cached randomly in NNs.

5.4.9 $Signature_{NN}$

This algorithm is executed by NN, which has cached the content C generated by CP. After receiving CT from CP, NN computes $H_1(CT)$ and produces $Sig_{NN}(CT) = [H_1(CT)]_{SK_{nid}^{NN}}$. Finally, NN returns $Sig_{NN}(CT)$ to CP.

5.4.10 Transaction generation

This algorithm is run by CP. After receiving $Sig_{NN}(CT)$ from NN, CP verifies $[Sig_{NN}(CT)]_{PK_{nid}^{NN}} = H_1(CT)$ holds. If the hash value doesn't match, an error report is sent to the blockchain network; otherwise, CP generates a transaction that packs ACP $(\mathscr{A}, \rho)$, hash value of content, and NN signatures. Note that a new transaction is generated for each NN caching the content. Then, CP produces the hash value of the transaction using its secret key, i.e., $Sig_{CP}(transaction) = [H_1(transaction)]_{PK_{nid}^{CP}}$. Finally, CP binds the transaction and signature and broadcasts to other blockchain nodes labeled with the content name.

5.4.11 Interest request

Whenever a CU wants to have content, it first downloads the transaction of the corresponding content name from the blockchain network. Next, CU maps its attribute to $\mathscr{A}'s$ corresponding row using function ρ. If the attribute set of nodes matches with attributes specified in the access policy, it broadcasts the interest packet having a content name to NNs.

5.4.12 Data packet response

As soon as NN receives this interest packet, it checks its content store. If a match occurs at any NN, then CU downloads CT from NN and decrypts it. CU can only decrypt CT if and only if its attribute set meets the access structure specified in CT. Lets suppose for a node N, $\mathscr{A}_N$ be submatrix of $\mathscr{A}$ and each row of $\mathscr{A}_N$ specifies the attributes in node, N attribute set S_N.

Say $I \subset \{1,2,....,k\}$ represents $\{j : \rho_j \in S_N\}$ and $\mathscr{A}_j$ represents the jth row of $\mathscr{A}$.

If S_N satisfies access structure $(\mathscr{A},\rho)$, the vector $\bar{e} = (1,0,....,0)$ is within the matrix $\mathscr{A}_N$, which depicts a suitable parameter $\{\chi_j = Z_p\}_{j\in I}$ can be computed so that it satisfy $\bar{(}e) = (\chi_1,\chi_2,.....,\chi_{|I|})\mathscr{A}_N$.

This value $\{\chi_j\}_{j\in I}$ can be used by a node to discover hidden secret encryption exponent d as:

$$\begin{aligned} d =&(1,0,....,0)(d.s_1,s_2,....s_n)^T \\ =&\bar{e}.\bar{v}^T = (\chi_1,\chi_2,.....,\chi_{|I|})\mathscr{A}_N.\bar{v}^T \\ =&(\chi_1,\chi_2,.....,\chi_{|I|}).\bar{\lambda_I}^T = \sum_{j\in I}\chi_j.\lambda_j \end{aligned} \tag{5.13}$$

where, $\bar{\lambda_I}$ is sub vector of $(\lambda_1,\lambda_2,.....,\lambda_k)$.

Next, using $\{\chi_j\}_{j\in I}$, nodes further calculate:

$$\begin{aligned} C_N =&\frac{e(C',M)}{\prod_{j\in I}(e(C_j,L).e(D_j,M_{\rho(j)}))^{\chi_j}} \\ =&\frac{e(g^d.g^{\theta}.g^{\phi.v})}{\prod_{j\in I}(e((g^{\Phi})^{\lambda_j}.p_{\rho(j)}^{-\omega_j},g^v)e(g^{\omega_j},p_{\rho(j)}^v)))^{\chi_j}} \\ =&\frac{e(g.g)^{\theta d}.e(g,g)^{\phi d.v}}{\prod_{j\in I}e(g,g)^{v\phi\lambda_j.\chi_j}} = e(g,g)^{\theta d} \end{aligned} \tag{5.14}$$

Finally, nodes can compute symmetric keys as:

$$K = \frac{C}{C_N} = \frac{C}{e(g,g)^{\theta d}} \tag{5.15}$$

CU obtains the content c by using the symmetric key.

5.4.13 Verify$_{\text{content}}$

This algorithm is executed by CU which has obtained message C. To check the integrity of received content, it computes the hash of c and checks if this value matches $H_1(C)$ on the blockchain network. If there is a match, then it sets $z = 1$, which shows that there is no modifications in message c; otherwise, CU sets $z = 0$, which shows C is not the same as the original content that was uploaded and an error report is sent to CP and blockchain network.

5.4.14 Data revocation

If CP needs to revoke the cached content in NNs, then following algorithms are executed.

- $Signature_{CP}(CT)$: This algorithm is called by CP. If CP wants to revoke content in NN, then it calculates $H_1(CT)$ and then outputs $Sig_{CP}(CT) = [H_1(CT)]_{SK_{nid}^{CP}}$. Then, CP sends

$Sig_{CP}(CT)$ to NN.

- $Verify_{NN}$: This algorithm is executed by NN. After receiving $Sig_{CP}(CT)$ from CP, it removes the corresponding CT from the cache store. Then, it computes $H_1(Sig_{CP}(CT))$ and outputs $Sig_{NN}(Sig_{CP}(CT)) = [H_1(Sig_{CP}(CT))]_{SK_{nid}^{NN}}$. Lastly, it returns $Sig_{NN}(Sig_{CP}(CT))$ to CP.

- Transaction generation: This algorithm is executed by CP. After recieving $Sig_{NN}(Sig_{CP}(CT))$ from NN, CP verifies if $[Sig_{NN}(Sig_{CP}(CT))]_{PK_{nid}^{NN}} = H_1(Sig_{CP}(CT))$, it generates a transaction depicting that NN has send $Sig_{NN}(Sig_{CP}(CT))$ to CP. Then, CP computes the hash of the transaction and signs it with its secret key, i.e., $Sig_{CP(transaction)} = [H_1(transaction)]_{SK_{Nid}^{CP}}$. Then, CU packs the transaction along with the signature and broadcasts it to nodes for verification of the transaction. When a CU searches for content and finds the transaction record that depicts content is revoked, it does not send an access request to NN.

5.4.15 Security Analysis

- Node collusion attack resistant: When some malicious nodes collude with other nodes, they may share their secret keys, but they can't achieve additional privileges beyond their own. The reason is existence of random element $v = \sum_{i=1}^{t} \left(\Phi_i \prod_{j=1, j\neq i}^{t} \frac{acid_j}{acid_j - acid_i}\right)$, in the process of attribute secret keys. This element ensures that each component of the same attribute in different nodes' attribute keys is distinct. Hence, they can't upgrade their privileges by aggregating their secret keys. Also, in the proposed scheme, TA is a fully trusted node, and it can't collude with NN to generate content.

- Soundness and completeness: In the proposed scheme, soundness implies that attackers can't take advantage of the network if they compromise less than t AACs. On the other hand, completeness implies that as long as greater than t AACs maintain their functionality, the network can work properly. The establishment of soundness and completeness is effectively ensured using (t,n) threshold secret sharing in [226] to distribute master key among AACs, which relies on Shamir (t,n) threshold secret sharing [233] as a foundational component, characterized by following attributes:

  1. having knowledge of t or more secret components allows for secret computation easily.
  2. having knowledge of $t-1$ or less secret components renders the secret entirely undermined

  Hence, based on these two properties, the proposed scheme ensures the following:

  1. The master secret key of AAC can be easily reconstructed with t or more master key shares from AACs

2. Fewer than t master key shares from AACs do not provide any insight into the master key. Hence, the attribute secret key can't be produced until successful master key reconstruction occurs.

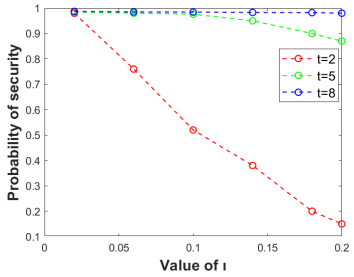

Figure 5.3: Probability of security

- Security against attacked AAC: The assurance of soundness property guarantees that an attacker has to compromise a greater number of AACs (more than t) to attack the network. Let's assume ι represents the probability of an attacker acquiring a secret key share from an AAC. So, the security of the network can be evaluated by the following probability:

$$\sum_{i=0}^{t=1} \binom{i}{n} \iota^i (1-\iota)^{n-i} \tag{5.16}$$

It is clear from this equation that the probability of the proposed scheme representing this attack is proportional to the value of t having any number of AACs. However, a large number of t will bring extra computational and communicational overhead rather than increasing security. To exemplify, let's take $\iota = 0.2$, and with 15 AACs, the system security probability is 0.87 for $t = 5$, whereas for $t = 8$, it is 0.98 as depicted in Fig. 5.3. Hence, it can be concluded that the proposed scheme is secure against the discussed attack, having an appropriate value of t.

- Security of access policy: To prove the security of BOMAC, we define the following theorem:

Theorem 1: If the decisional q-BDHE assumption hold, adversary $\mathbb{A}$ can't break BOMAC with challenge matrix having size $l^* \times n^* (l^*, n^* \leq q)$, using polynomial algorithm.
Proof: Let's consider a scenario where we have an adversary $\mathbb{A}$, possessing a significant advantage $\varepsilon = Adv\mathbb{A}$ in the selective security game outlined in Section 5.3 for our construction. Furthermore, let's assume that this adversary $\mathbb{A}$ selects a challenge matrix $\mathbb{M}$, with both dimensions not exceeding q. Within the security game, $\mathbb{A}$ has the capability to make inquiries about secret keys, provided that these keys cannot be directly utilized to decipher the ciphertext. The challenger, in response, hides the specifics of AAC secret key shares and generates a comprehensive secret key for adversary $\mathbb{A}$. Hence, the security game can be considered the same as the single authority scenario. For details of proof, refer to [234].

- Data Confidentiality: Only the CU whose attributes match the attributes specified in ACP can access the data. Also, if an NN is curious about plain text, it can't obtain it, as CT can only be decrypted if the node has a secret key, i.e., it is only with CU. Moreover, the blockchain network only records the hash value of content instead of the actual content. So, the proposed scheme satisfies data confidentiality. Also, NN is not responsible for AAC master secret sharing and attribute secret key generation.

- Content Integrity: The proposed scheme records the hash value of the content on the blockchain network to avoid data tampering attacks. Once the transactional data in the block is stored to the blockchain, it has a hash of the previous block, and its hash value is stored in the next block [235]. Hence, so it is difficult for attackers to change data without high computational resources.

- No trusted third party: To achieve secure access control, existing work in the literature uses an intermediary node, which leads to high trust-building costs, DoS attack, single point of failure, and privacy disclosure. In the proposed scheme, we use blockchain networks to record ACPs that realize self-certification and non-repudiation rather than using a single entity to manage the ACP. CUs don't need a third party to send their access request; instead, they can themselves verify through blockchain whether their attribute satisfies ACP or not. Moreover, Pedersen (t,n) threshold secret sharing used in this scheme does not need any third party to share secrets among nodes.

- Privacy protection: Each node in the blockchain network is assigned a pseudonymous ID that protects its privacy and account security. Moreover, each node use a public key for sending access requests and for generating an attribute secret key rather than using its true identity, which prevents unauthenticated users from capturing the private information of nodes.

- Transparency: Any ACP-related data is not saved on a single node and is transparent to every node on the network. So, any node on the network can verify the corresponding transaction data.

5.5 Performance evaluation

5.5.1 Simulation environment

This section numerically evaluates the performance of the proposed scheme. For performance evaluation, we consider a road segment with a four-lane bi-directional traffic flow, having a cell served by a LAG within a radius of 500 m. The performance of the proposed scheme has been accessed on a real dataset available at [204]. It contains power and energy-related details for residential properties, EVs, weather data, and more, which serve as a means for

testing, enhancing, and validating the model. We used the mobility records of 300 EVs, collected through GPS coordinates, over a period of 30 days. Ethereum network is chosen as a blockchain platform for the simulation environment. We used *Ethereum Go* client on a Intel core i7-9700k@2.20 GHz with Ubuntu 16.04 64 bit environment. We performed simulations on an NS-3-based NDN simulator called *ndnSIM*, and a neighbor knowledge-based scheme is used as location-based forwarding scheme. We used LIFO as a cache replacement policy and a randomly generated topology is used by the system. To ensure the involvement of all attributes in the decryption, the access structure is constructed from "AND" gates. Table 5.1 gives the list of parameters used in the simulations, and the notations for performance analysis are depicted in Table 5.2. We design the simulation using the network model discussed in Section III. Simulation results are averaged over 50 simulations.

Table 5.1: List of parameters.

Parameters	**Value**
# contents	50-80
Content size	$1*10^5 - 5*10^5$ Bytes
Coverage radius of LAG	500m
Cache capacity	20M-50M
Average speed of EVs	10m/s-30m/s
System bandwidth	10^6 Hz
Block size limit	1 MB
Average block generation time	10 s
Consensus algorithm	PoA
k	5
m	256 bits

Table 5.2: Notations for performance analysis

Notation	**Meaning**
N	Total number of nodes in network
N_{AA}	Total number of AACs in network
μ	Total number of attributes in system
μ_c	Total number of attributes in a CT
μ_N	Average number of attributes owned by a node
t	threshold value
N_{AA_t}	t number of AACs randomly selected for key generation
N_{AA_s}	number of AACs that manage the attribute set

5.5.2 Results and discussion

In this section, we evaluate the performance of the proposed scheme with [236], [237], [238]. All these schemes support access control using CP-ABE; however [236] and [238] are

blockchain based and [237] is non-blockchain based. The experiments are carried out to compute the time incurred for public parameter generation, key generation, encryption, decryption, and blockchain transaction verification. In any cryptosystem, these parameters are evaluated to analyze the efficiency of the proposal. A high value for these parameters indicates that the cryptosystem's performance is slower. Moreover, we analyzed the proposed scheme's resilience from a probabilistic perspective. Also, we have analyzed the false positive probability of the *bloom filter* to check the possibility of AAC being wrongly authenticated by the bloom filter.

5.5.2.1 Setup Algorithm execution time

For generating the public parameters, each AAC calculates θ_i and broadcasts to other $n-1$ AACs. Then, each AAC computes ms_i and $e(g,g)^{ms_i}$, and broadcasts the value to $n-1$ AACs. Finally, each AAC calculates global public key $e(g,g)^{\theta}$ with the cooperation of other AACs and packs this result into a transaction using the consensus algorithm described in [239]. For experiments, we have considered that AACs act as authority nodes in our PoA consensus mechanism. After verification, this value is published in the blockchain network. The value of t in our experiments is chosen to satisfy the condition $t < 0.6 * n$. For example, if the total number of authority nodes is 40, the threshold is set to 24. The value of t is selected to follow the blockchain assumption, which states that controlling the majority of resources in a network should be difficult. From Table 5.3, it can be concluded that the time of computing global public parameter $e(g,g)^{\theta}$ is proportional to the number of authority nodes.

Compared to the proposed scheme, in [236], [237], the setup phase includes the generation of public keys and private keys for all network entities, whereas, in [238] along with generating key pair, the system generates two default patients with a global identity. Hence, compared to the proposed scheme, the setup algorithm time is less in the discussed schemes as the proposed scheme as it involves the generation of global public parameters. These public parameters are used for decentralized key generation to enhance the system's security, which is not supported by any other scheme. The simulation results depicting the above comparison are represented in Table 5.3. The value of t is only applicable in the proposed scheme as the other three schemes are not based on threshold secret sharing. As the system involves multiple AACs, the parameter generation is scattered among all authorities, and setup overhead of AACs is reduced. As indicated in Table 5.3, with an increase in the number of total nodes, the parameter generation time increases, but even in the case of (15,25), the proposed algorithm is still efficient. In this experiment, the waiting time for block generation has been ignored.

Table 5.3: Setup algorithm execution time (ms)

n	***t***	**[236]**	**[237]**	**[238]**	**BOMAC**
15	9	158	155	170	280
20	12	185	183	196	333
25	15	210	199	220	425
30	18	266	255	301	556
35	21	295	290	325	692
40	24	376	379	410	736

5.5.2.2 Impact on Attribute secret key generation time

In the proposed scheme, CU submits a signed request for attribute key generation to the blockchain network. It calculates attribute secret key after receiving key shares from t different AACs. Hence, time generation for the attribute secret key is proportional to both the number of AACs and CU attributes, as depicted in Fig. 5.4 and 5.5. The proposed scheme takes less time when the number of attributes of a CU is ten compared to when there are twenty attributes. This is because a secret key is generated using attribute secret keys, which increases the key generation time when there are more attributes.

In [236], a single AAC processes key generation so it takes comparatively less time when compared to other schemes. Also, in [238] and [237], the key generation process requires the participation of all AACs in the system which results in longer key generation times compared to the proposed scheme, where only t out of n AACs are used.

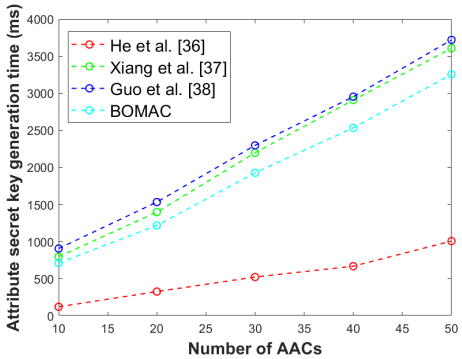

Figure 5.4: Attribute secret key generation time, Number of attributes = 10

5.5.2.3 Impact on Storage overhead

The size of the attribute secret key in the proposed scheme is linearly proportional to an average number of attributes owned by a node and the presence of t AACs in its generation, whereas compared to the proposed scheme, in [236] size of attribute secret key is less (specifically, $(\mu_N + 4)|p|$, where $|p|$ is the size of elements in G) as it only depends on the node's attribute set. Moreover, the size of the attribute secret key in [237] and [238] is dependent on

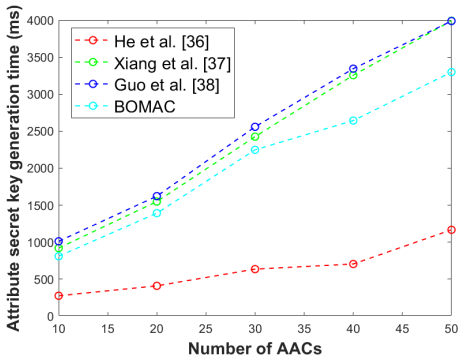

Figure 5.5: Attribute secret key generation time, Number of attributes = 20

the total number of AAC nodes, whereas, in the proposed scheme, a subset of AACs is selected for attribute secret key generation. Also, it has been observed from Table 5.4 that in all discussed schemes, CT size is linear to the number of attributes in the access policy. Authors in [236] proposed multi-user collaborative decryption, allowing private keys of other nodes to be used for CT decryption. Let μ_{collab} be the average number of collaborative attributes, and N_{AA_s} be the number of AACs that manage the attribute set. Hence, the size of CT in [236] is comparatively higher than the proposed scheme. In both [237] and [238], CT is generated by some numbers of AACs by considering attributes defined in an access policy.

In the proposed scheme, public parameters publish the public keys associated with each attribute and the hash functions for bloom filter, whereas in [238], public parameters contain a separate version of attributes for each node present in the system. However, in [237], the attribute authority management module stores authority public keys of all AACs and public keys of all attributes, which adds extra storage overhead to all systems. However, the storage overhead for CP remians constant in this case.

Table 5.4: Storage overhead

Scheme	**Size of attribute secret key**	**Size of CT**	**Size of public parameters**
[236]	$O(\mu_N)$	$O(\mu_c \times \mu_{collab})$	$O(1)$
[237]	$O(N_{AA} \times \mu_N)$	$O(N_{AA_s} \times \mu_c)$	$O(1)$
[238]	$O(N_{AA} \times \mu_N)$	$O(N_{AA} \times \mu_c)$	$O(N \times \mu)$
Our	$O(N_{AA_k} \times \mu_N)$	$O(\mu_c)$	$O(\mu + k)$

Table 5.5: Computation overhead

Scheme	Key generation	Encryption	Decryption
[236]	$O(\mu_N)$	$O(\mu_c \times \mu_{collab})$	$O(\mu_{c,\gamma} + tr_\gamma)$
[237]	$O(N_{AA} \times \mu_N)$	$O(\mu_c \times \mu_{AA_s})$	$O(1)$
[238]	$O(N_{AA} \times \mu_N)$	$O(N_{AA} \times \mu_c)$	$O(N_{AA} \times \mu_N)$
Our	$O(N_{AA_t} \times \mu_N)$	$O(\mu_c)$	$O(\mu_N)$

5.5.2.4 Computation overhead

Table 5.5 illustrates how the computational costs of the discussed schemes vary with the node's attribute set, number of attributes in access policy, and number of AACs. The introduction of t AACs increases the computation on the node for a key generation; it computes the attribute secret key for decryption after receiving secret shares from AACs. In contrast, both [238] and [237] involve all AACs in attribute key generation which leads to higher computational costs than the proposed scheme, which involves only t AACs. Also, the scheme in [236] involves only one TA and AAC for key generation, and depends on the number of node attributes which results in the least computation overhead compared to other schemes.

We have also analyzed the comparison of encryption times on CP and decryption time on CU with a fixed number of 20 AACs.In the case of decryption, [237] outperforms other schemes by outsourcing decryption to caching nodes, resulting in a constant time complexity of $O(1)$. The proposed scheme's decryption cost depends on the total number of attributes of a node, resulting in a complexity of $O(\mu_N)$. However, in [238], the computation for encryption and decryption are linearly dependent on the number of authorities. The decryption cost in [236] is tied to the rows in the LSSS matrix related to an attribute set ($\mu_{c,\gamma}$ is the number of required attributes for user group and tr_γ is the number of required collaboration attributes).

Comparing encryption costs, both our proposed scheme and [237] exhibit lower costs compared to [238] and [236]. The proposed scheme's encryption depends only on the number of attributes in the CT. However, in [237], the CP needs to send a set of managing attribute authority and attribute set to a TA, which further generates provisional CT to send to the CP. This process involves heavy pairing computations delegated to AACs rather than CP, resulting in higher encryption overhead. In [236], encryption is linear with the number of attributes and the number of collaborative attributes, whereas in [238], all authorities process the attributes in the access policy to generate CT. Fig. 5.6 represents the comparison of encryption times of different schemes concerning plaintext bytes, and Fig. 5.7 represents the decryption time of various schemes. For simulation, the total number of attributes is set as 20, and the number of collaborative attributes is set to 5 for the scheme in [236].

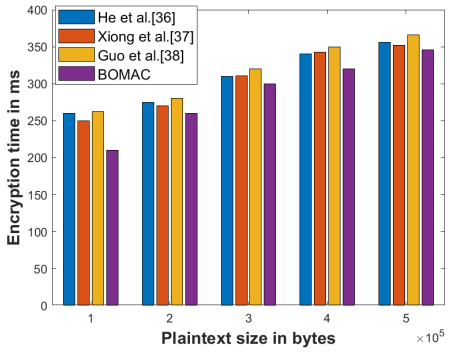

Figure 5.6: Comparison of encryption times of different schemes concerning plaintext bytes.

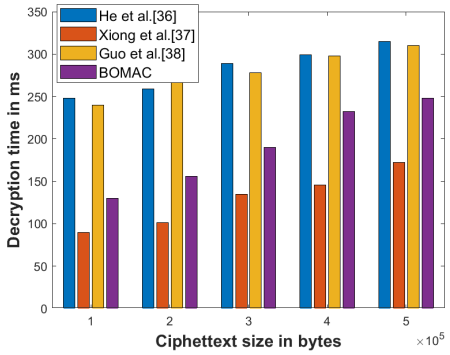

Figure 5.7: Comparison of decryption times of different schemes concerning ciphertext bytes

5.5.2.5 Impact on False positive probability

We computed the false positive probability using equation 5.1 while varying the number of hash functions (k) and the number of users (n) and keeping the filter size constant, i.e., m= 256. We have observed that the false positive rate is low when n is set to 15 or 20 but increases as the number of users in the network rises. Also, for a large value of n, m=256 does not impact the performance of the designed scheme. To accommodate an increased number of nodes in the network, a larger array size for the bloom filter is used to avoid false positives. Increasing the number of hash functions in the bloom filter has two effects: it sets more bits, increasing the risk of false positives, and it increases the chances of high false positives. The graph in Fig. 5.8 depicts that increasing k up to a point reduces the false positive rate, but increasing k beyond this point increases the error rate. In the proposed scheme, an optimal choice of k is between 4 and 6; we have used k=5, n=15-20, and m=256, which results in the probability of false positive as 0.000783736. The false positive probability is very small and can be ignored. Moreover, if a malicious AAC is to be falsely authenticated, the access control needs to reject the requesting user from accessing content to ensure security.

5.5.2.6 Impact on Throughput and Latency of Ethereum network

The simulation results, depicted in Figure 5.9 and 5.10, clearly demonstrate that the PoA-based consensus algorithm outperforms both Proof-of-Work (PoW) and PBFT in terms of

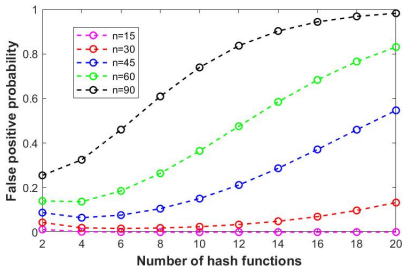

Figure 5.8: Evolution of false positive probability according to hash function and number of elements

throughput enhancement and reduced latency. Compared to PoW, PoA has more efficient usage of computational resources and fast transaction verification. Unlike PBFT, PoA transmits a low volume of information in packets, so it has superior performance compared to its counterparts. Also, unlike PoW, PoA is energy-efficient as its verifiers do not engage in PoW hashing puzzles for each block as only the selected authority computes the difficulty value. It has been observed by authors in [225], that a block size of 1 MB or less is best for an optimal performance within a blockchain system. The proposed scheme evaluates transaction confirmation time and throughput using a block size of 1 MB. Fig. 5.10 shows that the transaction processing latency of the proposed PoA-based consensus algorithm is reduced by 90.5% and 50.9% over PoW and PBFT, respectively.

Initially, with an increase in the count of network nodes, transaction confirmation time has a decline as the growing network size leads to more participants available for verification. This simultaneous decrease in confirmation time corresponds to an increase in throughput. However, when the number of nodes approaches around 40, many AACs increase the secret key generation time and system parameter generation time. This scenario leads to significant network delays, which results in decrease in network throughput when the number of nodes in the network is around 40.

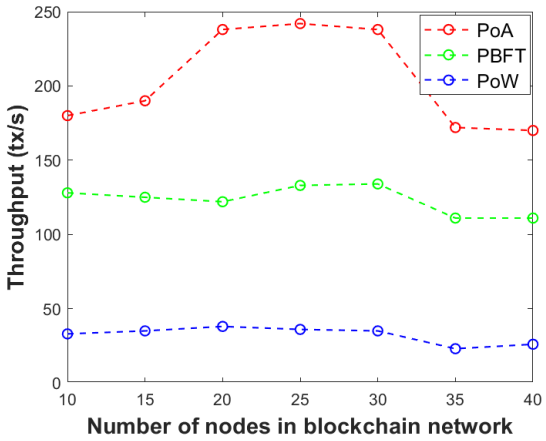

Figure 5.9: Throughput versus number of network nodes

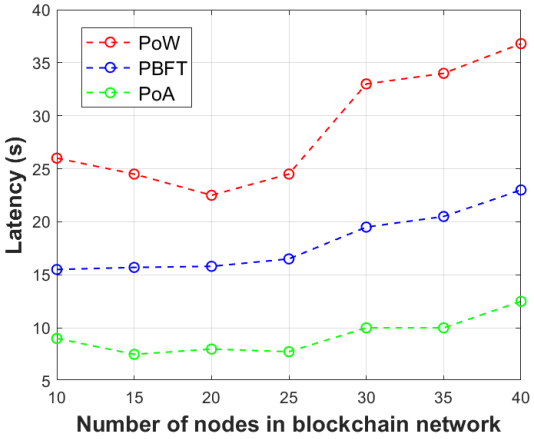

Figure 5.10: Latency versus number of network nodes.

5.5.2.7 Impact on Robustness

As discussed in Section 5.4.15, the completeness property ensures the proper functionality of the proposed scheme with a minimum number of t AACs. It implies that the system continue to function without disruption in case of failure or offline status of no more than $n-t$ AACs. Let's denote the probability of an AAC experiencing a crash as κ. So, in a single authority system, the probability of operating without disruption is $1-\kappa$. In the proposed scheme, the network maintains its normal functioning if at least t AACs are in a functional state. Hence, the probability of the system functioning normally is as follows:

$$\sum_{i=0}^{t=1} \binom{i}{n} (1-\kappa)^i \kappa^{n-i} \tag{5.17}$$

Fig. 5.11 and 5.12 represents the probability for varying number of AACs and t threshold value. It is clear from Fig. 5.12 that the proposed scheme is resilient against AAC crashes due to the presence of redundant authorities. Although we have assumed that $t < 0.6 * n$, even if there is 50% probability of AAC crash, we can efficiently configure the proposed scheme by choosing an appropriate value of t, for example for $n = 30$ and $t = 10$, a reliable operation is ensured. In scenarios involving compromised AAs, we have observed that the choice of t represents a trade-off between security and robustness. However, our analysis reveals that finding suitable values for (t, n) can be challenging. For instance, setting the total number of authorities to 15 and choosing t as 5 proves to be a viable approach. In this scenario, even if the adversary compromises authorities with a probability of 0.2, and there's a high chance of authorities experiencing crashes (50 percent probability), the system can still maintain both security and robustness with a high probability.

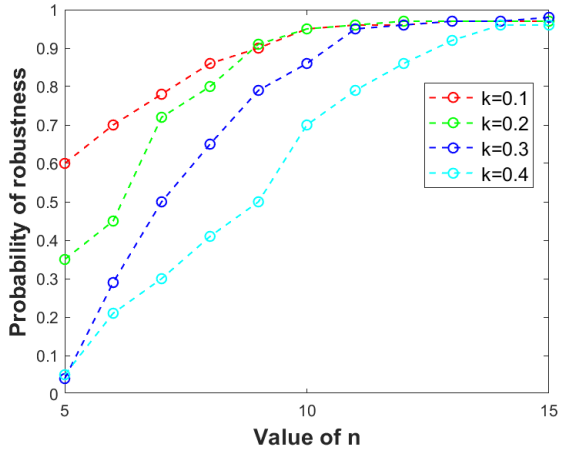

Figure 5.11: Probability of resilience against AAC crash, t=5

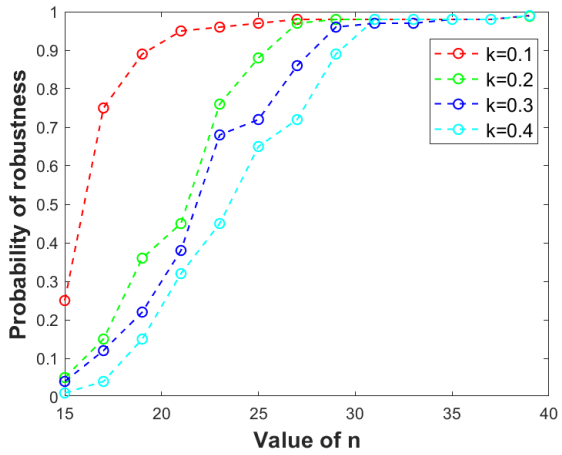

Figure 5.12: Probability of resilience against AAC crash, t=15

5.6 Summary

In this chapter, a blockchain-based traditional CP-ABE scheme. Pedersen (t,n) threshold secret sharing is used to share the master key and to generate attribute secret keys is proposed. The authenticity of secret shareholders is ensured by using a *bloom filter*, which significantly reduces both storage requirements and time complexity. The proposed scheme records access policies on the blockchain network to implement fine-grained access control. In order to prevent caching nodes from returning the incorrect result, a hash-based verification is performed. Also, the proposed scheme implements content revocation when a CP no longer desires to cache content in NN. The performance analysis demonstrates that the proposed scheme is robust and secure, as it guarantees that a secret key can't be generated when less than t AACs are present in the system. The next chapter concludes the thesis along with providing future research direction.

Chapter 6

Conclusion and Future Scope

Most of the existing solutions reported in literature for V2G are based upon traditional IP which may have a performance degradation with an increase in the number of nodes in the network. Also, it lacks in-network caching, which is pre-requisite to take efficient decisions for fast content delivery. In comparison to traditional IP-based networks, CCN are widely used for efficient cache-based content sharing. In this work, we propose a blockchain-based traditional CP-ABE scheme. Pedersen (t, n) threshold secret sharing is used to share the master key and to generate attribute secret keys for secure content delivery in the CCN network. Also, we proposed a secure content delivery framework for CCN based V2G network by combining consortium blockchain and contract theoretical modeling. The proposal ensures resilience from cache poisoning attacks in content delivery. Lastly, we propose a one-to-one provider-requester matching game for a CCN-based secure content sharing, having multiple requesters and providers for content. Moreover, we designed the proof of authority (PoA) consensus algorithm among entities in V2G network. We evaluated the proposed schemes using various performance evaluation metrics compared to the existing state-of-the-art solutions. The performance of the proposed schemes illustrates that they outperforms the existing state of- the-art schemes with respect to delay incurred, encryption time, computation overhead, throughput, and latency.

In future, we will deploy the proposed solution for large-scale data sets to assess the scalability and efficiency of the proposal. Also, we will evaluate the efficiency of the proposed scheme with an increase in the number of EVs with high mobility. Also, in future, we will use IOTA Tangle as distributed ledger technology instead of a blockchain. Rather than using a linear linked list (where transactions are being grouped into blocks), Tangle uses a Directed Acyclic Graph for maintaining ledgers. Therefore, transactions on Tangle can be issued simultaneously, synchronously, and continuously. Also, Tangle has no mining or transaction fee concepts that will reduce computation times and make it faster.

Moreover, future work will focus on the practical implementation of the proposed decentralized security mechanism on a V2G testbed. This implementation will serve to validate the applicability and feasibility of the mechanism in real-world scenarios, providing valuable in-

sights into its performance and effectiveness under realistic conditions. This integration will involve several key steps, including adapting the algorithm to the testbed environment, configuring the testbed to support the mechanism, and conducting extensive testing and evaluation. By leveraging a V2G testbed, we aim to simulate various V2G communication scenarios and assess the mechanism's ability to secure communication channels, authenticate vehicles, and ensure data integrity in dynamic and interconnected environments.

Bibliography

[1] D. Sadhya and S. K. Singh, "Privacy preservation for soft biometrics based multimodal recognition system," *Computers & Security*, vol. 58, pp. 160–179, 2016.

[2] S. A. Seshia, S. Hu, W. Li, and Q. Zhu, "Design automation of cyber-physical systems: Challenges, advances, and opportunities," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 9, pp. 1421–1434, 2017.

[3] C. Samaras and K. Meisterling, "Life cycle assessment of greenhouse gas emissions from plug-in hybrid vehicles: implications for policy," 2008.

[4] E. Zio and G. Sansavini, "Vulnerability of smart grids with variable generation and consumption: A system of systems perspective," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 3, pp. 477–487, 2013.

[5] W. Kempton and J. Tomić, "Vehicle-to-grid power implementation: From stabilizing the grid to supporting large-scale renewable energy," *Journal of power sources*, vol. 144, no. 1, pp. 280–294, 2005.

[6] Electric Vehicles Charging towards a bright future, Available: https://www.avendus.com/india/ev-report, [Accessed: July. 1, 2023].

[7] Z. Zhou, C. Sun, R. Shi, Z. Chang, S. Zhou, and Y. Li, "Robust energy scheduling in vehicle-to-grid networks," *IEEE Network*, vol. 31, no. 2, pp. 30–37, 2017.

[8] Electric Vehicles Technology deep dive, Available: https://www.iea.org/reports/electric-vehicles, [Accessed: March. 1, 2021].

[9] Open vs. Closed Charging Stations: Advantages and Disadvantages, Available: https://www.iea.org/reports/electric-vehicles, [Accessed: Nov. 10 2020].

[10] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic iot networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 269–282, 2017.

[11] A. Miglani and N. Kumar, "A blockchain based matching game for content sharing in content-centric vehicle-to-grid network scenarios," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–17, 2023.

[12] S. Rinaldi, M. Pasetti, E. Sisinni, F. Bonafini, P. Ferrari, M. Rizzi, and A. Flammini, "On the mobile communication requirements for the demand-side management of electric vehicles," *Energies*, vol. 11, no. 5, p. 1220, 2018.

[13] W. Su, H. Eichi, W. Zeng, and M.-Y. Chow, "A survey on the electrification of transportation in a smart grid environment," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 1, pp. 1–10, 2011.

[14] N. Saxena, S. Grijalva, V. Chukwuka, and A. V. Vasilakos, "Network security and privacy challenges in smart vehicle-to-grid," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 88–98, 2017.

[15] G. Naik, B. Choudhury, and J.-M. Park, "Ieee 802.11 bd & 5g nr v2x: Evolution of radio access technologies for v2x communications," *IEEE access*, vol. 7, pp. 70 169–70 184, 2019.

[16] T. Weil, "Vpki hits the highway: Secure communication for the connected vehicle program," *IT Professional*, no. 1, pp. 59–63, 2017.

[17] ISO 15118-2:2014, Available: https://www.iso.org/standard/55366.html, [Accessed: Nov. 1, 2018].

[18] Y.-J. Kim, M. Thottan, V. Kolesnikov, and W. Lee, "A secure decentralized data-centric information infrastructure for smart grid," *IEEE Communications Magazine*, vol. 48, no. 11, 2010.

[19] A. K. Das, S. Zeadally, and D. He, "Taxonomy and analysis of security protocols for internet of things," *Future Generation Computer Systems*, vol. 89, pp. 110–125, 2018.

[20] A. Mustapha, R. Khatoun, S. Zeadally, F. Chbib, A. Fadlallah, W. Fahs, and A. El Attar, "Detecting ddos attacks using adversarial neural network," *Computers & Security*, vol. 127, p. 103117, 2023.

[21] S. H. Mekala, Z. Baig, A. Anwar, and S. Zeadally, "Cybersecurity for industrial iot (iiot): Threats, countermeasures, challenges and future directions," *Computer Communications*, 2023.

[22] S. Panwar, M. Kumar, and S. Sharma, "Digital image steganography using modified lsb and aes cryptography," in *4th International Conference on Internet of Things and Connected Technologies (ICIoTCT), 2019: Internet of Things and Connected Technologies*. Springer, 2020, pp. 366–375.

[23] C. Wang, C. Chen, Q. Pei, N. Lv, and H. Song, "Popularity incentive caching for vehicular named data networking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 3640–3653, 2022.

[24] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 5435–5444, 2017.

[25] Z. Han, D. Niyato, W. Saad, T. Başar, and A. Hjørungnes, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge university press, 2012.

[26] K. Fan, Q. Pan, K. Zhang, Y. Bai, S. Sun, H. Li, and Y. Yang, "A secure and verifiable data sharing scheme based on blockchain in vehicular social networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5826–5835, 2020.

[27] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology–EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005. Proceedings 24*. Springer, 2005, pp. 457–473.

[28] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE transactions on parallel and distributed systems*, vol. 24, no. 1, pp. 131–143, 2012.

[29] S. Jiang, J. Liu, L. Wang, Y. Zhou, and Y. Fang, "Esac: An efficient and secure access control scheme in vehicular named data networking," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 10 252–10 263, 2020.

[30] E. G. AbdAllah, M. Zulkernine, and H. S. Hassanein, "Dacpi: A decentralized access control protocol for information centric networking," in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[31] S. Misra, R. Tourani, F. Natividad, T. Mick, N. E. Majd, and H. Huang, "Accconf: An access control framework for leveraging in-network cached data in the icn-enabled wireless edge," *IEEE transactions on dependable and secure computing*, vol. 16, no. 1, pp. 5–17, 2017.

[32] S. Fugkeaw, "Secure data sharing with efficient key update for industrial cloud-based access control," *IEEE Transactions on Services Computing*, 2021.

[33] A. Miglani, N. Kumar, A. Kishore, and N. Mohammad, "Uav-enabled edge computing and blockchain based secure charging station selection for energy trading in v2g environment," in *Proceedings of the 5th International ACM Mobicom Workshop on*

Drone Assisted Wireless Communications for 5G and Beyond, ser. DroneCom '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 103–108. [Online]. Available: https://doi.org/10.1145/3555661.3560872

[34] A. E. Roth and M. Sotomayor, "Two-sided matching," *Handbook of game theory with economic applications*, vol. 1, pp. 485–541, 1992.

[35] Y. Gu, W. Saad, M. Bennis, M. Debbah, and Z. Han, "Matching theory for future wireless networks: Fundamentals and applications," *IEEE Communications Magazine*, vol. 53, no. 5, pp. 52–59, 2015.

[36] T. Alladi, V. Chamola, N. Sahu, V. Venkatesh, A. Goyal, and M. Guizani, "A comprehensive survey on the applications of blockchain for securing vehicular networks," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2022.

[37] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues and challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1508–1532, 2019.

[38] K. Xue, P. He, X. Zhang, Q. Xia, D. S. Wei, H. Yue, and F. Wu, "A secure, efficient, and accountable edge-based access control framework for information centric networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 3, pp. 1220–1233, 2019.

[39] K. Fan, Y. Gong, C. Liang, H. Li, and Y. Yang, "Lightweight and ultralightweight rfid mutual authentication protocol with cache in the reader for iot in 5g," *Security and Communication Networks*, vol. 9, no. 16, pp. 3095–3104, 2016.

[40] Y. Yu, Y. Li, X. Du, R. Chen, and B. Yang, "Content protection in named data networking: Challenges and potential solutions," *IEEE Communications Magazine*, vol. 56, no. 11, pp. 82–87, 2018.

[41] R. Boussaha, Y. Challal, and A. Bouabdallah, "Authenticated network coding for software-defined named data networking," in *2018 IEEE 32nd international conference on advanced information networking and applications (AINA)*. IEEE, 2018, pp. 1115–1122.

[42] A. Miglani and N. Kumar, "Blockchain management and machine learning adaptation for iot environment in 5g and beyond networks: A systematic review," *Computer Communications*, vol. 178, pp. 37–63, 2021.

[43] B. Bera, D. Chattaraj, and A. K. Das, "Designing secure blockchain-based access control scheme in iot-enabled internet of drones deployment," *Computer Communications*, vol. 153, pp. 229–249, 2020.

[44] M. Khari, A. K. Garg, A. H. Gandomi, R. Gupta, R. Patan, and B. Balusamy, "Securing data in internet of things (iot) using cryptography and steganography techniques," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 73–80, 2019.

[45] A. Miglani, N. Kumar, V. Chamola, and S. Zeadally, "Blockchain for internet of energy management: Review, solutions, and challenges," *Computer Communications*, vol. 151, pp. 395–418, 2020.

[46] K. Yu, L. Tan, M. Aloqaily, H. Yang, and Y. Jararweh, "Blockchain-enhanced data sharing with traceable and direct revocation in iiot," *IEEE transactions on industrial informatics*, vol. 17, no. 11, pp. 7669–7678, 2021.

[47] B. Hammi, S. Zeadally, Y. C. E. Adja, M. Del Giudice, and J. Nebhen, "Blockchain-based solution for detecting and preventing fake check scams," *IEEE Transactions on Engineering Management*, vol. 69, no. 6, pp. 3710–3725, 2021.

[48] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE power and energy magazine*, vol. 7, no. 2, pp. 52–62, 2009.

[49] K. Mannaro, A. Pinna, and M. Marchesi, "Crypto-trading: Blockchain-oriented energy market," in *2017 AEIT International Annual Conference*. IEEE, 2017, pp. 1–5.

[50] C. Liu, K. K. Chai, X. Zhang, E. T. Lau, and Y. Chen, "Adaptive blockchain-based electric vehicle participation scheme in smart grid platform," *IEEE access*, vol. 6, pp. 25 657–25 665, 2018.

[51] W. Kempton and S. E. Letendre, "Electric vehicles as a new power source for electric utilities," *Transportation research. Part D, Transport and environment*, vol. 2, no. 3, pp. 157–175, 1997.

[52] Y. Zhou and X. Li, "Vehicle to grid technology: A review," in *2015 34th Chinese Control Conference (CCC)*, 2015, pp. 9031–9036.

[53] M. Yilmaz and P. T. Krein, "Review of the impact of vehicle-to-grid technologies on distribution systems and utility interfaces," *IEEE Transactions on Power Electronics*, vol. 28, no. 12, pp. 5673–5689, 2013.

[54] R. Deivanayagam, "Vehicle-to-grid technology: Concept, status, and challenges," *The Journal of Undergraduate Research at the University of Illinois at Chicago*, vol. 10, no. 1, 2017.

[55] K. Czechowski, "Assessment of profitability of electric vehicle-to-grid considering battery degradation," 2015.

[56] Y. He, B. Venkatesh, and L. Guan, "Optimal scheduling for charging and discharging of electric vehicles," *IEEE transactions on smart grid*, vol. 3, no. 3, pp. 1095–1105, 2012.

[57] K. Mets, T. Verschueren, F. De Turck, and C. Develder, "Exploiting v2g to optimize residential energy consumption with electrical vehicle (dis) charging," in *Smart Grid Modeling and Simulation (SGMS), 2011 IEEE First International Workshop on*. IEEE, 2011, pp. 7–12.

[58] S. Pal and R. Kumar, "Electric vehicle scheduling strategy in residential demand response programs with neighbor connection," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 980–988, 2018.

[59] E. L. Karfopoulos, K. A. Panourgias, and N. D. Hatziargyriou, "Distributed coordination of electric vehicles providing v2g regulation services," *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 2834–2846, 2016.

[60] T. Zhang, W. Chen, Z. Han, and Z. Cao, "Charging scheduling of electric vehicles with local renewable energy under uncertain electric vehicle arrival and grid power price," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2600–2612, 2013.

[61] X. Qiao, H. Wang, W. Tan, A. V. Vasilakos, J. Chen, and M. B. Blake, "A survey of applications research on content-centric networking," *China Communications*, vol. 16, no. 9, pp. 122–140, 2019.

[62] M. Amadeo, C. Campolo, A. Molinaro, and G. Ruggeri, "Content-centric wireless networking: A survey," *Computer Networks*, vol. 72, pp. 1–13, 2014.

[63] S. H. Bouk, S. H. Ahmed, and D. Kim, "Vehicular content centric network (vccn) a survey and research challenges," in *Proceedings of the 30th annual ACM symposium on applied computing*, 2015, pp. 695–700.

[64] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, and A. Ksentini, "Named data networking in vehicular ad hoc networks: State-of-the-art and challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 320–351, 2019.

[65] X. Tan, Z. Zhou, C. Zou, Y. Niu, and X. Chen, "Copyright protection in named data networking," in *2014 Sixth International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2014, pp. 1–6.

[66] M. Ion, J. Zhang, and E. M. Schooler, "Toward content-centric privacy in icn: Attribute-based encryption and routing," in *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, 2013, pp. 39–40.

[67] B. Li, D. Huang, Z. Wang, and Y. Zhu, "Attribute-based access control for icn naming scheme," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 2, pp. 194–206, 2016.

[68] B. Li, Z. Wang, D. Huang, and Y. Zhu, "Toward privacy-preserving content access control for information centric networking," 2014.

[69] D. Grewe, K. P. Rao, S. Schildt, M. Wagner, D. Schoop, and H. Frey, "Encircle: Encryption-based access control for information-centric connected vehicles," in *2017 8th International Conference on the Network of the Future (NOF)*. IEEE, 2017, pp. 114–119.

[70] R. S. Da Silva and S. D. Zorzo, "An access control mechanism to ensure privacy in named data networking using attribute-based encryption with immediate revocation of privileges," in *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*. IEEE, 2015, pp. 128–133.

[71] M. Raykova, H. Lakhani, H. Kazmi, and A. Gehani, "Decentralized authorization and privacy-enhanced routing for information-centric networks," in *Proceedings of the 31st annual computer security applications conference*, 2015, pp. 31–40.

[72] B. Hamdane, A. Serhrouchni, and S. G. El Fatmi, "Access control enforcement in named data networking," in *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*. IEEE, 2013, pp. 576–581.

[73] Y.-F. Tseng, C.-I. Fan, and C.-Y. Wu, "Fgac-ndn: Fine-grained access control for named data networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 143–152, 2018.

[74] M. Aiash and J. Loo, "A formally verified access control mechanism for information centric networks," in *2015 12th International Joint Conference on e-Business and Telecommunications (ICETE)*, vol. 4. IEEE, 2015, pp. 377–383.

[75] A. Fiat and M. Naor, "Broadcast encryption," in *Advances in Cryptology—CRYPTO'93: 13th Annual International Cryptology Conference Santa Barbara, California, USA August 22–26, 1993 Proceedings 13*. Springer, 1994, pp. 480–491.

[76] B. Hamdane and S. G. El Fatmi, "A credential and encryption based access control solution for named data networking," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 1234–1237.

[77] Q. Li, R. Sandhu, X. Zhang, and M. Xu, "Mandatory content access control for privacy protection in information centric networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 5, pp. 494–506, 2015.

[78] Q. Li, X. Zhang, Q. Zheng, R. Sandhu, and X. Fu, "Live: Lightweight integrity verification and content access control for named data networking," pp. 308–320, 2014.

[79] P. Gasti, G. Tsudik, E. Uzun, and L. Zhang, "Dos and ddos in named data networking," in *2013 22nd International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2013, pp. 1–7.

[80] Y. Wang, M. Xu, Z. Feng, Q. Li, and Q. Li, "Session-based access control in information-centric networks: Design and analyses," in *2014 IEEE 33rd international performance computing and communications conference (IPCCC)*. IEEE, 2014, pp. 1–8.

[81] C.-I. Fan, I.-T. Chen, C.-K. Cheng, J.-J. Huang, and W.-T. Chen, "Ftp-ndn: File transfer protocol based on re-encryption for named data network supporting nondesignated receivers," *IEEE Systems Journal*, vol. 12, no. 1, pp. 473–484, 2016.

[82] Q. Zheng, G. Wang, R. Ravindran, and A. Azgin, "Achieving secure and scalable data access control in information-centric networking," in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 5367–5373.

[83] C. A. Wood and E. Uzun, "Flexible end-to-end content security in ccn," in *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*. IEEE, 2014, pp. 858–865.

[84] T. Chen, K. Lei, and K. Xu, "An encryption and probability based access control model for named data networking," in *2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2014, pp. 1–8.

[85] M. Mangili, F. Martignon, and S. Paraboschi, "A cache-aware mechanism to enforce confidentiality, trackability and access policy evolution in content-centric networks," *Computer Networks*, vol. 76, pp. 126–145, 2015.

[86] N. Kumar and A. Miglani, *Probabilistic Data Structures for Blockchain-Based Internet of Things Applications*. CRC Press, 2021.

[87] C. Ghali, M. A. Schlosberg, G. Tsudik, and C. A. Wood, "Interest-based access control for content centric networks," in *Proceedings of the 2nd ACM Conference on Information-Centric Networking*, 2015, pp. 147–156.

[88] Q. Li, P. P. Lee, P. Zhang, P. Su, L. He, and K. Ren, "Capability-based security enforcement in named data networking," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2719–2730, 2017.

[89] J. Kuriharay, E. Uzun, and C. A. Wood, "An encryption-based access control framework for content-centric networking," in *2015 IFIP networking conference (IFIP networking)*. IEEE, 2015, pp. 1–9.

[90] N. Fotiou, G. F. Marias, and G. C. Polyzos, "Access control enforcement delegation for information-centric networking architectures," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*, 2012, pp. 85–90.

[91] S. Singh, "A trust based approach for secure access control in information centric network," *International Journal of Information and Network Security*, vol. 1, no. 2, p. 97, 2012.

[92] L. Zhu, N. M. Lwamo, K. Sharif, C. Xu, X. Du, M. Guizani, and F. Li, "T-cam: Time-based content access control mechanism for icn subscription systems," *Future generation computer systems*, vol. 106, pp. 607–621, 2020.

[93] fp7-pursuit, Available: https://www.fp7-pursuit.eu/, [Accessed: March. 1, 2020].

[94] Y. A. M. G. Alnagar, "Non-cooperative and cooperative caching schemes for vehicular networks," 2020.

[95] L. C. Liu, D. Xie, S. Wang, and Z. Zhang, "Ccn-based cooperative caching in vanet," in *2015 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, 2015, pp. 198–203.

[96] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, "Social cooperation for information-centric multimedia streaming in highway vanets," in *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*. IEEE, 2014, pp. 1–6.

[97] Y. Bai, D. Wang, G. Huang, and B. Song, "A deep reinforcement learning-based social-aware cooperative caching scheme in d2d communication networks," *IEEE Internet of Things Journal*, 2023.

[98] L. Yao, Y. Wang, X. Wang, and W. Guowei, "Cooperative caching in vehicular content centric network based on social attributes and mobility," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 391–402, 2019.

[99] X. Hu, C. Papadopoulos, J. Gong, and D. Massey, "Not so cooperative caching in named data networking," in *2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 2263–2268.

[100] A. Dua, M. Shishodia, N. Kumar, G. S. Aujla, and N. Kumar, "Bloom filter based efficient caching scheme for content distribution in vehicular networks," pp. 1–6, 2019.

[101] L. Yao, X. Xu, J. Deng, G. Wu, and Z. Li, "A cooperative caching scheme for vccn with mobility prediction and consistent hashing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20 230–20 242, 2022.

[102] S. Anamalamudi, M. S. Alkatheiri, E. Al Solami, and A. R. Sangi, "Cooperative caching scheme for machine-to-machine information-centric iot networks," *IEEE Canadian Journal of Electrical and Computer Engineering*, vol. 44, no. 2, pp. 228–237, 2021.

[103] O. Hahm, E. Baccelli, T. C. Schmidt, M. Wählisch, C. Adjih, and L. Massoulié, "Low-power internet of things with ndn & cooperative caching," in *Proceedings of the 4th ACM Conference on Information-Centric Networking*, ser. ICN '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 98–108. [Online]. Available: https://doi.org/10.1145/3125719.3125732

[104] Z. Li, X. Shen, H. Xun, Y. Miao, W. Zhang, P. Luo, and K. Liu, "Coopcon: Cooperative hybrid congestion control scheme for named data networking," *IEEE Transactions on Network and Service Management*, 2023.

[105] Y. Sellami, G. Jaber, A. Lounis, H. Lakhlef, and A. Bouabdallah, "A cooperative caching scheme in fog/sensor nodes for ccn," in *2022 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2022, pp. 481–486.

[106] J. Liu, C. Zhang, Y. Wang, L. Wei, and J. Liu, "Cooperative caching in a content-centric network for high-definition map delivery," in *2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*. IEEE, 2020, pp. 96–101.

[107] S. Khodaparas, S. Yousefi, and A. Benslimane, "A multi criteria cooperative caching scheme for internet of things," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[108] J. Iqbal, Z. u. Abideen, N. Ali, S. H. Khan, A. Rahim, A. Zahir, S. A. H. Mohsan, and M. H. Alsharif, "An energy efficient local popularity based cooperative caching for mobile information centric networks," *Sustainability*, vol. 14, no. 20, p. 13135, 2022.

[109] Z. Hu, Z. Zheng, T. Wang, L. Song, and X. Li, "Game theoretic approaches for wireless proactive caching," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 37–43, 2016.

[110] Y. Xu, Y. Li, S. Ci, T. Lin, and F. Chen, "Distributed caching via rewarding: An incentive caching model for icn," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.

[111] C. Wang, C. Chen, Q. Pei, N. Lv, and H. Song, "Popularity incentive caching for vehicular named data networking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 3640–3653, 2020.

[112] Z. Chen, Y. Liu, B. Zhou, and M. Tao, "Caching incentive design in wireless d2d networks: A stackelberg game approach," in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[113] K. Zhao, S. Zhang, N. Zhang, Y. Zhou, Y. Zhang, and X. Shen, "Incentive mechanism for cached-enabled small cell sharing: A stackelberg game approach," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.

[114] A. Ndikumana, N. H. Tran, T. M. Ho, D. Niyato, Z. Han, and C. S. Hong, "Joint incentive mechanism for paid content caching and price based cache replacement policy in named data networking," *IEEE Access*, vol. 6, pp. 33 702–33 717, 2018.

[115] A. Ndikumana, K. Thar, T. M. Ho, N. H. Tran, P. L. Vo, D. Niyato, and C. S. Hong, "In-network caching for paid contents in content centric networking," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.

[116] A. Alioua, S.-M. Senouci, H. Sedjelmaci, and S. Moussaoui, "Incentive edge caching in software-defined internet of vehicles: A stackelberg game approach," *International Journal of Communication Systems*, vol. 32, no. 17, p. e3787, 2019.

[117] P. K. Agyapong and M. Sirbu, "Economic incentives in information-centric networking: Implications for protocol design and public policy," *IEEE Communications Magazine*, vol. 50, no. 12, pp. 18–26, 2012.

[118] M. Hajimirsadeghi, N. B. Mandayam, and A. Reznik, "Joint caching and pricing strategies for popular content in information centric networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 3, pp. 654–667, 2017.

[119] N. Magaia, Z. Sheng, P. R. Pereira, and M. Correia, "Repsys: A robust and distributed incentive scheme for collaborative caching and dissemination in content-centric cellular-based vehicular delay-tolerant networks," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 65–71, 2018.

[120] J. Xu, K. Xue, C. Cao, and H. Yue, "Incentive cooperative caching for localized information-centric networks," pp. 1–6, 2017.

[121] C. Bernardini, T. Silverston, and O. Festor, "Mpc: Popularity-based caching strategy for content centric networks," in *2013 IEEE international conference on communications (ICC)*. IEEE, 2013, pp. 3619–3623.

[122] Z. Su, P. Ren, and X. Gan, "A novel algorithm to cache vehicular content with parked vehicles applications," in *2014 IEEE International Conference on Communications (ICC)*. IEEE, 2014, pp. 5665–5669.

[123] L. Yao, Y. Wang, Q. Xia, and R. Xu, "Popularity prediction caching using hidden markov model for vehicular content centric networks," in *2019 20th IEEE International Conference on Mobile Data Management (MDM)*. IEEE, 2019, pp. 533–538.

[124] W. Huang, T. Song, Y. Yang, and Y. Zhang, "Cluster-based cooperative caching with mobility prediction in vehicular named data networking," *IEEE Access*, vol. 7, pp. 23 442–23 458, 2019.

[125] Y. Wei, C. Xu, M. Wang, and J. Guan, "Cache management for adaptive scalable video streaming in vehicular content-centric network," in *2016 International Conference on Networking and Network Applications (NaNA)*. IEEE, 2016, pp. 410–414.

[126] F. de Moraes Modesto and A. Boukerche, "Utility-gradient implicit cache coordination policy for information-centric ad-hoc vehicular networks," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*. IEEE, 2017, pp. 10–17.

[127] W. Zhao, Y. Qin, D. Gao, C. H. Foh, and H.-C. Chao, "An efficient cache strategy in information centric networking vehicle-to-vehicle scenario," *IEEE Access*, vol. 5, pp. 12 657–12 667, 2017.

[128] M. D. Ong, M. Chen, T. Taleb, X. Wang, and V. C. Leung, "Fgpc: fine-grained popularity-based caching design for content centric networking," in *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, 2014, pp. 295–302.

[129] S. Kumar and R. Tiwari, "Dynamic popularity window and distance-based efficient caching for fast content delivery applications in ccn," *Engineering Science and Technology, an International Journal*, vol. 24, no. 3, pp. 829–837, 2021.

[130] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi, and S. Pack, "Wave: Popularity-based and collaborative in-network caching for content-oriented networks," in *2012 Proceedings IEEE INFOCOM Workshops*. IEEE, 2012, pp. 316–321.

[131] H. Wu, J. Li, T. Pan, and B. Liu, "A novel caching scheme for the backbone of named data networking," in *2013 IEEE International Conference on Communications (ICC)*. IEEE, 2013, pp. 3634–3638.

[132] I. Psaras, W. K. Chai, and G. Pavlou, "Probabilistic in-network caching for information-centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*, 2012, pp. 55–60.

[133] S. Tarnoi, K. Suksomboon, W. Kumwilaisak, and Y. Ji, "Performance of probabilistic caching and cache replacement policies for content-centric networks," in *39th Annual IEEE Conference on Local Computer Networks*. IEEE, 2014, pp. 99–106.

[134] R. Zhang, J. Liu, T. Huang, and R. Xie, "Popularity based probabilistic caching strategy design for named data networking," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2017, pp. 476–481.

[135] H. Wu, J. Li, J. Zhi, Y. Ren, and L. Li, "Design and evaluation of probabilistic caching in information-centric networking," *IEEE Access*, vol. 6, pp. 32 754–32 768, 2018.

[136] Y. Wang, M. Xu, and Z. Feng, "Hop-based probabilistic caching for information-centric networks," in *2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 2102–2107.

[137] B. Banerjee, A. Kulkarni, and A. Seetharam, "Greedy caching: An optimized content placement strategy for information-centric networks," *Computer Networks*, vol. 140, pp. 78–91, 2018.

[138] W. Sirichotedumrong, W. Kumwilaisak, S. Tarnoi, and N. Thatphithakkul, "Prioritized probabilistic caching algorithm in content centric networks," in *Recent Advances in Information and Communication Technology 2016: Proceedings of the 12th International Conference on Computing and Information Technology (IC2IT)*. Springer, 2016, pp. 255–265.

[139] Y. Gao and J. Zhou, "Probabilistic caching mechanism based on software defined content centric network," in *2019 IEEE 11th International Conference on Communication Software and Networks (ICCSN)*, 2019, pp. 210–214.

[140] G. Deng, L. Wang, F. Li, and R. Li, "Distributed probabilistic caching strategy in vanets through named data networking," in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2016, pp. 314–319.

[141] M. A. M. Hail, M. Amadeo, A. Molinaro, and S. Fischer, "On the performance of caching and forwarding in information-centric networking for the iot," in *Wired/Wireless Internet Communications: 13th International Conference, WWIC 2015, Malaga, Spain, May 25-27, 2015, Revised Selected Papers 13*. Springer, 2015, pp. 313–326.

[142] H. H. Nax and B. S. Pradelski, "Evolutionary dynamics and equitable core selection in assignment games," *International Journal of Game Theory*, vol. 44, no. 4, pp. 903–932, 2015.

[143] H. Zhang, W. Ding, F. Yang, J. Song, and Z. Han, "Resource allocation in heterogeneous network with visible light communication and d2d: A hierarchical game approach," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7616–7628, 2019.

[144] O. T. T. Kim, N. H. Tran, C. Pham, T. LeAnh, M. T. Thai, and C. S. Hong, "Parking assignment: Minimizing parking expenses and balancing parking demand among multiple parking lots," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1320–1331, 2019.

[145] D. Niyato, E. Hossain, and Z. Han, "Dynamics of multiple-seller and multiple-buyer spectrum trading in cognitive radio networks: A game-theoretic modeling approach," *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1009–1022, 2008.

[146] B. Wang, Y. Sun, S. Li, and Q. Cao, "Hierarchical matching with peer effect for low-latency and high-reliable caching in social iot," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 1193–1209, 2018.

[147] D. Wu, L. Zhou, Y. Cai, H.-C. Chao, and Y. Qian, "Physical–social-aware d2d content sharing networks: A provider–demander matching game," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 7538–7549, 2018.

[148] R. Mochaourab, B. Holfeld, and T. Wirth, "Distributed channel assignment in cognitive radio networks: Stable matching and walrasian equilibrium," *IEEE Transactions on Wireless Communications*, vol. 14, no. 7, pp. 3924–3936, 2015.

[149] J. Jiang, S. Zhang, B. Li, and B. Li, "Maximized cellular traffic offloading via device-to-device content sharing," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 1, pp. 82–91, 2015.

[150] L. Wang, H. Wu, Y. Ding, W. Chen, and H. V. Poor, "Hypergraph-based wireless distributed storage optimization for cellular d2d underlays," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 10, pp. 2650–2666, 2016.

[151] D. Hamza and J. S. Shamma, "Blma: A blind matching algorithm with application to cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 2, pp. 302–316, 2017.

[152] K. Shanmugam, N. Golrezaei, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers. arxiv 2011," *arXiv preprint arXiv:1109.4179*.

[153] S. Seng, C. Luo, X. Li, H. Zhang, and H. Ji, "User matching on blockchain for computation offloading in ultra-dense wireless networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1167–1177, 2020.

[154] C. Chen, C. Wang, T. Qiu, N. Lv, and Q. Pei, "A secure content sharing scheme based on blockchain in vehicular named data networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3278–3289, 2019.

[155] A. Benmoussa, A. el Karim Tahari, C. A. Kerrache, N. Lagraa, A. Lakas, R. Hussain, and F. Ahmad, "Msidn: Mitigation of sophisticated interest flooding-based ddos attacks in named data networking," *Future Generation Computer Systems*, vol. 107, pp. 293–306, 2020.

[156] Y. Xin, Y. Li, W. Wang, W. Li, and X. Chen, "A novel interest flooding attacks detection and countermeasure scheme in ndn," in *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–7.

[157] J. Dong, K. Wang, W. Quan, and H. Yin, "Interestfence: simple but efficient way to counter interest flooding attack," *Computers & Security*, vol. 88, p. 101628, 2020.

[158] W. Shang, A. Bannis, T. Liang, Z. Wang, Y. Yu, A. Afanasyev, J. Thompson, J. Burke, B. Zhang, and L. Zhang, "Named data networking of things," in *2016 IEEE first international conference on internet-of-things design and implementation (IoTDI)*. IEEE, 2016, pp. 117–128.

[159] D. Qu, G. Lv, S. Qu, H. Shen, Y. Yang, and Z. Heng, "An effective and lightweight countermeasure scheme to multiple network attacks in ndn," *IEEE/ACM Transactions On Networking*, vol. 30, no. 2, pp. 515–528, 2021.

[160] Z. Rezaeifar, J. Wang, and H. Oh, "A trust-based method for mitigating cache poisoning in name data networking," *Journal of Network and Computer Applications*, vol. 104, pp. 117–132, 2018.

[161] A. Miglani and N. Kumar, "Blockchain-based co-operative caching for secure content delivery in ccn-enabled v2g networks," *IEEE Transactions on Vehicular Technology*, pp. 1–16, 2022.

[162] C. Ghali, G. Tsudik, and E. Uzun, "Network-layer trust in named-data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 12–19, 2014.

[163] K. Lei, J. Fang, Q. Zhang, J. Lou, M. Du, J. Huang, J. Wang, and K. Xu, "Blockchain-based cache poisoning security protection and privacy-aware access control in ndn vehicular edge computing networks," *Journal of Grid Computing*, vol. 18, pp. 593–613, 2020.

[164] M. Xie, I. Widjaja, and H. Wang, "Enhancing cache robustness for content-centric networking," in *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 2426–2434.

[165] J. Zhou, J. Luo, J. Wang, and L. Deng, "Cache pollution prevention mechanism based on deep reinforcement learning in ndn," *Journal of Communications and Information Networks*, vol. 6, no. 1, pp. 91–100, 2021.

[166] P. Gasti and G. Tsudik, "Content-centric and named-data networking security: The good, the bad and the rest," in *2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*. IEEE, 2018, pp. 1–6.

[167] M. Labbi, Y. Chahid, N. Kannouf, and M. Benabdellah, "Blockchain-based trust and security in content-centric networking–based internet of things," in *Blockchain for Cybersecurity and Privacy*. CRC Press, 2020, pp. 141–158.

[168] M. Labbi, N. Kannouf, Y. Chahid, and M. Benabdellah, "Blockchain based trust and security in content-centric networking based internet of things (ccn-iot)."

[169] I. Petri, O. F. Rana, J. Bignell, S. Nepal, and N. Auluck, "Incentivising resource sharing in edge computing applications," in *Economics of Grids, Clouds, Systems, and Services: 14th International Conference, GECON 2017, Biarritz, France, September 19-21, 2017, Proceedings 14*. Springer, 2017, pp. 204–215.

[170] J. Guo, M. Wang, B. Chen, S. Yu, H. Zhang, and Y. Zhang, "Enabling blockchain applications over named data networking," IEEE, pp. 1–6, 2019.

[171] D. Kim, S. Nam, J. Bi, and I. Yeom, "Efficient content verification in named data networking," in *Proceedings of the 2nd ACM Conference on Information-Centric Networking*, 2015, pp. 109–116.

[172] K. Lei, Q. Zhang, J. Lou, B. Bai, and K. Xu, "Securing icn-based uav ad hoc networks with blockchain," *IEEE Communications Magazine*, vol. 57, no. 6, pp. 26–32, 2019.

[173] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE internet of things journal*, vol. 6, no. 3, pp. 4660–4670, 2018.

[174] H. Khelifi, S. Luo, B. Nour, H. Moungla, S. H. Ahmed, and M. Guizani, "A blockchain-based architecture for secure vehicular named data networks," *Computers & Electrical Engineering*, vol. 86, p. 106715, 2020.

[175] A. H. Magsi, L. V. Yovita, A. Ghulam, G. Muhammad, and Z. Ali, "A content poisoning attack detection and prevention system in vehicular named data networking," *Sustainability*, vol. 15, no. 14, p. 10931, 2023.

[176] A. Benmoussa, C. A. Kerrache, C. T. Calafate, and N. Lagraa, "Ndn-bda: A blockchain-based decentralized data authentication mechanism for vehicular named data networking," *Future Internet*, vol. 15, no. 5, p. 167, 2023.

[177] J. Shi, X. Zeng, and R. Han, "A blockchain-based decentralized public key infrastructure for information-centric networks," *Information*, vol. 13, no. 5, p. 264, 2022.

[178] M. Hassan, D. Pesavento, and L. Benmohamed, "Blockchain-based decentralized authentication for information-centric 5g networks," in *2022 IEEE 47th Conference on Local Computer Networks (LCN)*. IEEE, 2022, pp. 299–302.

[179] J. Lou, Q. Zhang, Z. Qi, and K. Lei, "A blockchain-based key management scheme for named data networking," in *2018 1st IEEE international conference on hot information-centric networking (HotICN)*. IEEE, 2018, pp. 141–146.

[180] B. Li, M. Ma, and R. Xia, "Hierarchical identity-based security mechanism using blockchain in named data networking," in *2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*. IEEE, 2020, pp. 148–153.

[181] B. Li and M. Ma, "An advanced hierarchical identity-based security mechanism by blockchain in named data networking," *Journal of Network and Systems Management*, vol. 31, no. 1, p. 13, 2023.

[182] N. Fotiou and G. C. Polyzos, "Decentralized name-based security for content distribution using blockchains," in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2016, pp. 415–420.

[183] M. Conti, M. Hassan, and C. Lal, "Blockauth: Blockchain based distributed producer authentication in icn," *Computer Networks*, vol. 164, p. 106888, 2019.

[184] S. Baranski and J. Konorski, "Mitigation of fake data content poisoning attacks in ndn via blockchain," in *2020 30th International telecommunication networks and applications conference (ITNAC)*. IEEE, 2020, pp. 1–6.

[185] Y. Ying, Z. Zhou, and Q. Zhang, "Blockchain-based collaborative caching mechanism for information center iot," *Journal of ICT Standardization*, vol. 11, no. 1, pp. 67–96, 2023.

[186] H. Liu, R. Zhu, J. Wang, and W. Xu, "Blockchain-based key management and green routing scheme for vehicular named data networking," *Security and Communication Networks*, vol. 2021, pp. 1–13, 2021.

[187] S. Tokunaga, S. Ohzahata, and R. Yamamoto, "A link state routing method for ccn with blockchain," in *2021 Ninth International Symposium on Computing and Networking Workshops (CANDARW)*. IEEE, 2021, pp. 49–55.

[188] S. Liao, J. Wu, J. Li, and K. Konstantin, "Information-centric massive iot-based ubiquitous connected vr/ar in 6g: A proposed caching consensus approach," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5172–5184, 2020.

[189] D. B. Rawat, R. Doku, A. Adebayo, C. Bajracharya, and C. Kamhoua, "Blockchain enabled named data networking for secure vehicle-to-everything communications," *IEEE Network*, vol. 34, no. 5, pp. 185–189, 2020.

[190] R. Doku, D. B. Rawat, M. Garuba, and L. Njilla, "Fusion of named data networking and blockchain for resilient internet-of-battlefield-things," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2020, pp. 1–6.

[191] Q. Lyu, Y. Qi, X. Zhang, H. Liu, Q. Wang, and N. Zheng, "Sbac: A secure blockchain-based access control framework for information-centric networking," *Journal of Network and Computer Applications*, vol. 149, p. 102444, 2020.

[192] S. Roy, F. J. A. Morais, M. Salimitari, and M. Chatterjee, "Cache attacks on blockchain based information centric networks: an experimental evaluation," in *Proceedings of the 20th International Conference on Distributed Computing and Networking*, 2019, pp. 134–142.

[193] F. Ahmad, C. A. Kerrache, F. Kurugollu, and R. Hussain, "Realization of blockchain in named data networking-based internet-of-vehicles," *IT Professional*, vol. 21, no. 4, pp. 41–47, 2019.

[194] R. Li and H. Asaeda, "A blockchain-based data life cycle protection framework for information-centric networks," *IEEE Communications Magazine*, vol. 57, no. 6, pp. 20–25, 2019.

[195] H. Li, K. Wang, T. Miyazaki, C. Xu, S. Guo, and Y. Sun, "Trust-enhanced content delivery in blockchain-based information-centric networking," *Ieee Network*, vol. 33, no. 5, pp. 183–189, 2019.

[196] X. Tan, C. Huang, and L. Ji, "Access control scheme based on combination of blockchain and xor-coding for icn," in *2018 5th IEEE international conference on cyber security and cloud computing (CSCloud)/2018 4th IEEE international conference on edge computing and scalable cloud (EdgeCom)*. IEEE, 2018, pp. 160–165.

[197] K. Fan, Y. Ren, Y. Wang, H. Li, and Y. Yang, "Blockchain-based efficient privacy preserving and data sharing scheme of content-centric network in 5g," *IET communications*, vol. 12, no. 5, pp. 527–532, 2018.

[198] K. Yang, J. J. Sunny, and L. Wang, "Blockchain-based decentralized public key management for named data networking," in *The international conference on computer communications and networks (ICCCN 2018)*, 2018.

[199] K. Zhu, Z. Chen, W. Yan, and L. Zhang, "Security attacks in named data networking of things and a blockchain solution," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4733–4741, 2018.

[200] J. A. Khan and K. Ozbay, "Affirm: Privacy-by-design blockchain for mobility data in web3 using information centric fog networks with collaborative learning," in *2023 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2023, pp. 456–462.

[201] K. Fizza, N. Auluck, and A. Azim, "Improving the schedulability of real-time tasks using fog computing," *IEEE Transactions on Services Computing*, vol. 15, no. 1, pp. 372–385, 2019.

[202] G. Edens and G. Scott, "A better way to organize the internet: Content-centric networking," *IEEE Spectrum Blogs*, vol. 23, 2017.

[203] S. Guo, H. Xie, and G. Shi, "Collaborative forwarding and caching in content centric networks," in *International Conference on Research in Networking*. Springer, 2012, pp. 41–55.

[204] IEEE Power and Energy Society. Database With a 24-Period Scenario of 1800 Realistic EVS., Available: https://site.ieee.org/pes-iss/data-sets/, [Accessed: April. 2022].

[205] D. D. Van, Q. Ai, Q. Liu, and D.-T. Huynh, "Efficient caching strategy in content-centric networking for vehicular ad-hoc network applications," *IET Intelligent Transport Systems*, vol. 12, no. 7, pp. 703–711, 2018.

[206] Z. Hou, H. Chen, Y. Li, and B. Vucetic, "Incentive mechanism design for wireless energy harvesting-based internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2620–2632, 2017.

[207] L. Song, D. Niyato, Z. Han, and E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication," *IEEE Wireless Communications*, vol. 21, no. 3, pp. 136–144, 2014.

[208] R. Zhang, X. Cheng, L. Yang, and B. Jiao, "Interference graph-based resource allocation (ingra) for d2d communications underlaying cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 8, pp. 3844–3850, 2014.

[209] L. A. Adamic and B. A. Huberman, "Zipf's law and the internet." *Glottometrics*, vol. 3, no. 1, pp. 143–150, 2002.

[210] M. U. Hassan, M. H. Rehmani, and J. Chen, "Differential privacy techniques for cyber physical systems: a survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 746–789, 2019.

[211] T. Alladi, V. Chamola, N. Sahu, V. Venkatesh, A. Goyal, and M. Guizani, "A comprehensive survey on the applications of blockchain for securing vehicular networks," *IEEE Communications Surveys & Tutorials*, 2022.

[212] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, J.-N. Liu, Y. Xiang, and R. H. Deng, "Crowdbc: A blockchain-based decentralized framework for crowdsourcing," *IEEE transactions on parallel and distributed systems*, vol. 30, no. 6, pp. 1251–1266, 2018.

[213] Q. Wang, R. Li, Q. Wang, S. Chen, and Y. Xiang, "Exploring unfairness on proof of authority: Order manipulation attacks and remedies," in *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security*, 2022, pp. 123–137.

[214] S. Micali, M. Rabin, and S. Vadhan, "Verifiable random functions," in *40th annual symposium on foundations of computer science (cat. No. 99CB37039)*. IEEE, 1999, pp. 120–130.

[215] M. Kumar and M. Nagar, "Big data analytics in agriculture and distribution channel," pp. 384–387, 2017.

[216] P. Biró, "The stable matching problem and its generalizations: an algorithmic and game theoretical approach," 2007.

[217] D. Hamza and J. S. Shamma, "Blma: A blind matching algorithm with application to cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 2, pp. 302–316, 2017.

[218] F. EDITION, "Probability, random variables, and stochastic processes," 2002.

[219] D. Dumont, "Mariages stables," *Pour la Science*, vol. 144, pp. 96–101, 1989.

[220] L. Gao, L. Duan, and J. Huang, "Two-sided matching based cooperative spectrum sharing," *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 538–551, 2016.

[221] B. S. Pradelski, "Decentralized dynamics and fast convergence in the assignment game," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, 2015, pp. 43–43.

[222] A. J. Perez and S. Zeadally, "Secure and privacy-preserving crowdsensing using smart contracts: Issues and solutions," *Computer Science Review*, vol. 43, p. 100450, 2022.

[223] O. Semiari, W. Saad, S. Valentin, M. Bennis, and H. V. Poor, "Context-aware small cell networks: How social metrics improve wireless resource allocation," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 5927–5940, 2015.

[224] Y. Wu, D. Wu, L. Ao, L. Yang, and Q. Fu, "Contention-based radio resource management for urllc-oriented d2d communications," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 9960–9971, 2020.

[225] G. Sagirlar, B. Carminati, E. Ferrari, J. D. Sheehan, and E. Ragnoli, "Hybrid-iot: Hybrid blockchain architecture for internet of things-pow sub-blockchains," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2018, pp. 1007–1016.

[226] T. P. Pedersen, "A threshold cryptosystem without a trusted party," in *Advances in Cryptology—EUROCRYPT'91: Workshop on the Theory and Application of Cryptographic Techniques Brighton, UK, April 8–11, 1991 Proceedings 10*. Springer, 1991, pp. 522–526.

[227] J. Li, K. Ren, and K. Kim, "A2be: Accountable attribute-based encryption for abuse free access control," *Cryptology ePrint Archive*, 2009.

[228] D. Sadhya and S. K. Singh, "Design of a cancelable biometric template protection scheme for fingerprints based on cryptographic hash functions," *Multimedia Tools and Applications*, vol. 77, pp. 15 113–15 137, 2018.

[229] L. Luo, D. Guo, R. T. Ma, O. Rottenstreich, and X. Luo, "Optimizing bloom filter: Challenges, solutions, and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1912–1949, 2018.

[230] D. Sadhya and S. K. Singh, "Providing robust security measures to bloom filter based biometric template protection schemes," *Computers & Security*, vol. 67, pp. 59–72, 2017.

[231] G.-G. Wang, S. Deb, A. H. Gandomi, Z. Zhang, and A. H. Alavi, "Chaotic cuckoo search," *Soft Computing*, vol. 20, pp. 3349–3362, 2016.

[232] K. Yu, L. Tan, C. Yang, K.-K. R. Choo, A. K. Bashir, J. J. Rodrigues, and T. Sato, "A blockchain-based shamir's threshold cryptography scheme for data protection in industrial internet of things settings," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8154–8167, 2021.

[233] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[234] W. Li, K. Xue, Y. Xue, and J. Hong, "Tmacs: A robust and verifiable threshold multi-authority access control system in public cloud storage," *IEEE Transactions on parallel and distributed systems*, vol. 27, no. 5, pp. 1484–1496, 2015.

[235] G. Nagasubramanian, R. K. Sakthivel, R. Patan, A. H. Gandomi, M. Sankayya, and B. Balusamy, "Securing e-health records using keyless signature infrastructure

blockchain technology in the cloud," *Neural Computing and Applications*, vol. 32, pp. 639–647, 2020.

[236] Y. He, H. Wang, Y. Li, K. Huang, V. C. Leung, F. R. Yu, and Z. Ming, "An efficient ciphertext-policy attribute-based encryption scheme supporting collaborative decryption with blockchain," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2722–2733, 2021.

[237] S. Xiong, Q. Ni, L. Wang, and Q. Wang, "Sem-acsit: secure and efficient multiauthority access control for iot cloud storage," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2914–2927, 2020.

[238] R. Guo, H. Shi, D. Zheng, C. Jing, C. Zhuang, and Z. Wang, "Flexible and efficient blockchain-based abe scheme with multi-authority for medical on demand in telemedicine system," *IEEE Access*, vol. 7, pp. 88 012–88 025, 2019.

[239] A. Miglani and N. Kumar, "Blockchain-based co-operative caching for secure content delivery in ccn-enabled v2g networks," *IEEE Transactions on Vehicular Technology*, 2022.